АЛЕКСЕЙ ПЕТЮШКИН

bhv

HTML в Web-дизайне

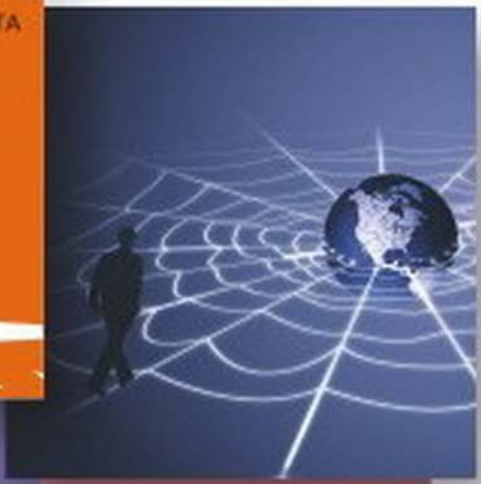
ОСНОВЫ HTML-ДОКУМЕНТА

РАБОТА С ГРАФИКОЙ И ТАБЛИЦАМИ

ИСПОЛЬЗОВАНИЕ СТИЛЕЙ CSS

ГРАФИЧЕСКИЙ МАКЕТ И ЕГО HTML-ВЕРСТКА

РЕКОМЕНДАЦИИ ПО СОЗДАНИЮ И ПРОДВИЖЕНИЮ WEB-САЙТА



PRO

ПРОФЕССИОНАЛЬНОЕ ПРОГРАММИРОВАНИЕ

Алексей Петюшкин

HTML в Web-дизайне

Санкт-Петербург
«БХВ-Петербург»
2004

УДК 681.3.068+800.92HTML
ББК 32.973.26-018.1
П29

Петюшкин А. В.

П29 HTML в Web-дизайне. — СПб.: БХВ-Петербург, 2004. — 400 с.: ил.

ISBN 5-94157-513-0

Кратко изложены необходимые основы HTML: спецификации, форматирование текста, списки, таблицы, графика, фреймы, мультимедиа, пользовательские формы, таблицы стилей и пр. Подробно и на примерах рассмотрены практические вопросы построения сайтов с использованием HTML: разработка концепции, макета и структуры сайта, организация стандартных сервисов, использование интерактивных технологий, создание графического макета и его верстка, продвижение, поддержка и сопровождение сайта. Книга содержит множество пояснительных иллюстраций, примеров с пошаговыми инструкциями, советов и рекомендаций.

Для программистов и Web-дизайнеров

УДК 681.3.068+800.92HTML
ББК 32.973.26-018.1

Группа подготовки издания:

Главный редактор	*Екатерина Кондукова*
Зам. главного редактора	*Евгений Рыбаков*
Зав. редакцией	*Григорий Добин*
Редактор	*Татьяна Лапина*
Компьютерная верстка	*Ольги Сергиенко*
Корректор	*Зинаида Дмитриева*
Дизайн серии	*Инны Тачиной*
Оформление обложки	*Игоря Цырульникова*
Зав. производством	*Николай Тверских*

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 07.06.04.
Формат 70×100 1/16. Печать офсетная. Усл. печ. л. 32,25.
Тираж 3000 экз. Заказ №
"БХВ-Петербург", 190005, Санкт-Петербург, Измайловский пр., 29.

Гигиеническое заключение на продукцию, товар № 77.99.02.953.Д.001537.03.02
от 13.03.2002 г. выдано Департаментом ГСЭН Минздрава России.

Отпечатано с готовых диапозитивов
в ГУП "Типография "Наука"
199034, Санкт-Петербург, 9 линия, 12.

ISBN 5-94157-513-0

© Петюшкин А. В., 2004
© Оформление, издательство "БХВ-Петербург", 2004

Содержание

Предисловие **7**

Содержание книги 7
Об авторе 8

ЧАСТЬ I. ОСНОВЫ HTML **11**

Глава 1. Язык разметки текста HTML **12**

Что такое HTML 12
Спецификация HTML 13
Какой браузер выбрать 14
Internet Explorer 14
Netscape 15
Opera 16
Структура HTML-документа 18
Раздел *HTML* 18
Раздел *HEAD* 19
Раздел *BODY* 24
Ваш первый HTML-документ 26

Глава 2. Форматирование текста и гиперссылки **28**

Форматирование текста 28
Теги логического форматирования 29
Теги физического форматирования 33
Структурное форматирование 38
Абзац 39
Текстовый блок 42
Центрирование 44
Отступы и перенос строки 44
Горизонтальный разделитель 48
Предварительно отформатированный текст 49
Комментарии 49
Специальные символы 51
Символы, отображающие элементы HTML-форматирования 51
Символы оформления документа 52
Буквы иностранных алфавитов 53
Гипертекстовые ссылки 55
Структура гиперссылок 56
Правила описания гиперссылок 59

Глава 3. Использование HTML-списков **67**

Зачем нужны HTML-списки 67
Нумерованные списки 68
Параметры тега <*OL*> 71
Маркированные списки 75
Параметры тега <*UL*> 76
Графические маркеры 78

Вложенные списки ... 80
Списки определений ... 81
Другие списки ... 83

Глава 4. Таблица как основа HTML-документа ... 85

Табличное представление данных ... 85
Создание простейших таблиц ... 85
Правила описания таблиц ... 87
Параметры тега *<TABLE>* ... 87
Параметры тегов *<TR>*, *<TD>* и *<TH>* ... 93
Нестандартное представление таблиц ... 98
Пустые ячейки ... 98
Группировка данных ... 102
Колонтитулы таблицы ... 104
Прорисовка структуры таблицы ... 105
Вложенные таблицы ... 107
Преимущества вложенных таблиц ... 107
Пример вложенных таблиц ... 108

Глава 5. Добавление графики средствами HTML ... 113

Роль графики ... 113
Характеристика графических стандартов ... 114
GIF (Graphics Interchange Format) ... 114
JPEG (Joint Photographic Experts Group) ... 115
PNG (Portable Network Graphics) ... 116
Вставка графики в HTML-документ ... 116
Описание графики в HTML-документе ... 116
Карты-изображения ... 125
Конфигурация карт-изображений ... 126

Глава 6. HTML-документы на основе фреймов ... 131

Понятие фрейма ... 131
Области применения фреймов ... 132
Панель навигации ... 132
Одновременное отображение информации ... 133
Постоянное визуальное присутствие объекта ... 134
Web-интерфейс для онлайн-игр ... 134
Преимущества и недостатки фреймов ... 134
Правила описания фреймов ... 135
Тег *<FRAMESET>* ... 136
Тег *<FRAME>* ... 138
Тег *<NOFRAMES>* ... 139
Дополнительные параметры фреймовых структур ... 140
Взаимодействие между фреймами ... 142
Плавающие фреймы ... 144

Глава 7. Добавление аудио ... 147

Звуковые форматы ... 147
Встраивание звуковых файлов в HTML-документ ... 148
Создание гиперссылки на музыкальный файл ... 148
Применение тега *<EMBED>* ... 148
RealAudio ... 151
Фоновый звук ... 154

Глава 8. Пользовательские формы ... 155

Что такое пользовательские формы ... 155
Организация обратной связи ... 156
Авторизация ... 159
Проведение исследований ... 160
Пользовательский профиль ... 160
Структура пользовательских форм ... 161
Параметр *ACTION* ... 161
Параметр *METHOD* ... 162
Параметр *NAME* ... 163
Параметр *ENCTYPE* ... 163
Тег *<INPUT>* ... 164
Тег *<TEXTAREA>* ... 173
Тег *<SELECT>* ... 174

Глава 9. Стилевые шаблоны CSS ... 181

Каскадные таблицы стилей ... 181
Общая информация ... 181
Уровни CSS ... 182
Способы определения таблиц стилей ... 183
Запись шаблона CSS ... 186
Области применения стилевых шаблонов CSS ... 189
Позиционирование объектов ... 201

ЧАСТЬ II. ПРАКТИКУМ МАСТЕРА ... 203

Глава 10. Dynamic HTML ... 204

Общая информация ... 204
Создание визуальных эффектов ... 205
Динамические блоки ... 205
Визуальные фильтры ... 210

Глава 11. Концепция Web-сайта ... 219

Классификация сайтов ... 219
Корпоративные Web-сайты ... 219
Промо/презентационные Web-сайты ... 223
Электронные магазины ... 226
Онлайн-сервисы ... 229
Контент-проекты ... 232
Порталы ... 236
Коммьюнити ... 238
Позиционирование Web-сайта ... 244
Выбор доменного имени для Web-сайта ... 244
Распределение обязанностей по работе над Web-сайтом ... 248
Работа по исследованию аудитории Web-сайта ... 251

Глава 12. Композиция Web-сайта ... 254

Понятие композиции Web-сайта ... 254
Логотип ... 254
Текст ... 255
Имиджевые элементы ... 256
Декоративные элементы ... 258

Пиктограммы ... 258
Требования ... 260
Функции ... 260
Концепция ... 260
Разработка пиктограмм ... 267

Глава 13. Навигация по Web-сайту ... 274

Меню навигации ... 274
Понятие системы навигации ... 274
Типы навигации ... 274
Требования к системе навигации ... 276
Карта сайта ... 284
Вопросы целесообразности ... 284
Многоуровневая навигация ... 285
Запутанная навигация ... 286

Глава 14. Шаблоны ошибок ... 288

Ошибка 404 ... 288
Техническая трактовка ... 288
Миф Ошибки 404 ... 289
Причины и последствия ... 289
Раздолье на ниве *HTML* ... 290
Рекомендации по созданию страницы Ошибки 404 ... 291
Ошибка 404 как хобби ... 292

Глава 15. Разработка графического макета будущего Web-сайта ... 297

Компоновка макета ... 297
Тип верстки ... 298
Работа над макетом ... 299

Глава 16. HTML-верстка графического макета: шаг за шагом ... 306

Глава 17. Продвижение Web-сайта в Интернете ... 336

Интернет-маркетинг ... 336
Размещение информации в каталогах интернет-ресурсов ... 337
Регистрация в поисковых системах ... 338
Организация рейтинга ... 339
Доски объявлений и форумы ... 339
Email-маркетинг ... 341
Контекстная реклама ... 342
Обмен ссылками ... 342
Партнерские программы ... 342
Спонсорство ... 343
Баннерная реклама ... 344

Заключение ... 379

Приложение 1. Обозначение цветов в HTML ... 382

Приложение 2. Список специальных символов HTML ... 387

Приложение 3. Полезные ссылки ... 394

Предметный указатель ... 395

Предисловие

Содержание книги

Книга "HTML в Web-дизайне" ориентирована на людей, уже знакомых со спецификацией языка разметки HTML 4.0 и основами создания Web-сайтов, но не имеющих необходимого опыта корректной и грамотной верстки электронных документов, работы с графическими макетами, выбором средств реализации элементов документа, оптимизации HTML-кода, а также поддержки готового Web-сайта и его продвижения в Интернете.

Материал, представленный в книге, изложен простым, нетехническим языком, содержит множество пояснительных иллюстраций, практических примеров с пошаговыми инструкциями, советов и рекомендаций и позволяет быстро научиться использовать возможности языка разметки HTML на все 100% и создавать привлекательные Web-сайты с высокой посещаемостью!

Структура книги состоит из двух частей. *Часть I* раскрывает основы работы с языком гипертекстовой разметки HTML, рассказывает о стилевых шаблонах CSS, затрагивает технологию Dynamic HTML:

- применение браузеров;
- структура HTML-документа;
- форматирование текста и работа со шрифтами;
- структурное форматирование и блоки информационных данных;
- нумерованные и маркированные списки, списки определений;
- создание простых и вложенных таблиц, оптимизация табличных данных, использование таблицы в качестве структурной основы HTML-документа;
- графические форматы и типы файлов, используемые в Интернете;
- внедрение графики на Web-странице;

- использование карт-изображений (Imagemap);
- разработка фреймов;
- внедрение звука на Web-странице.

Часть II книги посвящена проектированию и разработке вашего собственного Web-сайта, а также основам продвижения его в Интернете:

- концепция и композиция будущего Web-сайта;
- разработка системы навигации;
- пошаговая разработка графического макета Web-сайта (практический пример);
- пошаговая HTML-верстка готового графического макета (практический пример);
- основы продвижения Web-сайта в Интернете.

Книга "HTML в Web-дизайне" — ваш надежный помощник в мире интернет-проектирования. Только самый важный и полезный практический материал, по частицам собранный автором книги в процессе своей профессиональной деятельности.

Об авторе

Петюшкин Алексей Валерьевич имеет высшее рекламное образование, закончил Санкт-Петербургский гуманитарный университет профсоюзов по специальности "Рекламное дело".

Имеет опыт работы в интернет-индустрии свыше семи лет, является автором более 100 информационных, аналитических и научных статей, посвященных вопросам информационных технологий, психологии рекламы, электронной коммерции, интернет-маркетингу, usability, пользовательских интерфейсов и пр.

Статьи автора опубликованы во многих известных оффлайновых и сетевых изданиях, а также на интернет-серверах, посвященных интернет-технологиям. Среди них такие, как "CHIP-Россия", "ComputerPrice", "Мир Интернет", "Компьютерная Россия", "Магия ПК", "Новое Знание", "Компьютерная газета", "Санкт-Петербургские Ведомости", "Маркетер", "GetInfo.Ru", "eManual" и др.

Автор является сертифицированным специалистом по дисциплинам "Нейролингвистическое программирование (НЛП)", "Интернет-маркетинг", "Язык разметки HTML".

Принимал участие и проводил консультации по разработке и продвижению более сорока интернет-проектов и электронных презентаций, в том числе

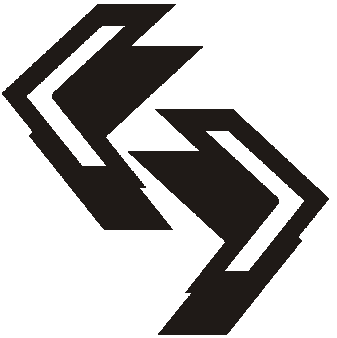
"GetInfo.Ru", "URSA-Россия", "Интернет-магазин программного обеспечения MSSoft.Ru", "Austrian Airlines", "Интернет-справочник Price.Ru", "TravellingRussia", "ТехноподиуМ", "Guides of St.-Petersburg", "Griffin Securities", "Система документооборота Комитета экономики и инвестиций Правительства Ленинградской области", "Информационно-справочная система TopPlan", "Интранет-система Ford Motor Russia" и пр.

Web-сайт автора (**http://www.alpet.spb.ru**) удостоин награды Golden Web Awards, а также номинировался на получение Национальной Интернет Премии-2003 в категории "Персональные страницы" (вход в итоговую десятку претендентов на соискание премии среди нескольких сотен кандидатов).

В апреле 2002 года издательство "БХВ-Петербург" выпустило книгу автора "Основы баннерной рекламы". Книга была рекомендована профессорско-преподавательским составом кафедры экономики и управления Санкт-Петербургского института международных экономических отношений (ИМЭО) в качестве учебного пособия по дисциплине "Менеджер по маркетингу в Интернете".

В мае 2003 года издательство "БХВ-Петербург" издало книгу автора "HTML. Экспресс-курс", которая стала одним из лидеров продаж в 2003—2004 гг. по тематике языка гипертекстовой разметки HTML.

Свои комментарии, замечания, а также предложения по книге "HTML в Web-дизайне" можно оставить на авторском сайте по адресу **http://www.alpet.spb.ru**.



ЧАСТЬ I

Основы HTML

Глава 1. Язык разметки текста HTML

Глава 2. Форматирование текста и гиперссылки

Глава 3. Использование HTML-списков

Глава 4. Таблица как основа HTML-документа

Глава 5. Добавление графики средствами HTML

Глава 6. HTML-документы на основе фреймов

Глава 7. Добавление аудио

Глава 8. Пользовательские формы

Глава 9. Стилевые шаблоны CSS

ГЛАВА 1

Язык разметки текста HTML

Что такое HTML

Всемирная паутина (World Wide Web) состоит из множества электронных документов, связанных между собой и представляющих огромный кладезь информационных данных, которые описываются специальными технологическими правилами. Эти правила составляются с помощью языка гипертекстовой разметки HTML — HyperText Markup Language.

Можно с уверенностью сказать, что сегодня язык разметки HTML является основой всех размещенных в Интернете электронных документов. Он выступает в роли фундамента, на базе которого реализуются прочие сетевые программные технологии, призванные повысить общую привлекательность, эффективность и интерактивность носителей информационных данных. Так же, как фундамент строящегося дома представляет собой несущую конструкцию, на которой впоследствии крепятся стены и производятся отделочные работы, HTML служит простейшим и, по большому счету, незаменимым средством создания гипертекстовых документов, которое в совокупности с более гибкими и функциональными интернет-технологиями позволяет получать поистине фантастические результаты.

HTML постоянно набирает популярность, причем не только в сфере интернет-технологий, но и в области предоставления презентационных услуг, рекламно-выставочной деятельности, внедрения в состав программного обеспечения и пр. Корпоративные клиенты все чаще разрабатывают CD-презентации и демонстрационные ролики, заказывают электронные визитки и рекламные обращения. Частный пользователь все больше склоняется к мысли о составлении интерактивных портфолио и резюме, позволяющих, в отличие от листа бумаги, ярче и привлекательнее преподнести свои знания и умения

потенциальному работодателю. Разработчики программного обеспечения делают информационные и рекламные вставки справочного характера внутри создаваемых ими программных пакетов. И все это реализуется с помощью простого и доступного, но вместе с тем эффективного языка разметки HTML.

Спецификация HTML

Многие называют HTML языком программирования. Это не совсем верно, т. к. в традиционном понимании HTML является языком разметки электронных документов, лишь указывающим программам просмотра HTML-страниц (браузерам) форму представления описанной в документе информации.

Начиная с середины 90-х годов XX века, HTML претерпел некоторые изменения в своей спецификации: варианты используемых инструкций, применяемых тегов и модулей горячо обсуждались и изменялись. С момента своего возникновения разработкой спецификации языка HTML стала заниматься организация под названием "Консорциум W3C" (World Wide Web Consortium). Ее основной задачей являлось составление и принятие технических рекомендаций единого стандарта разметки гипертекстовых документов. Практическая необходимость работы над стандартом была обусловлена постоянным ростом популярности Интернета, в рамках которого производители программ для просмотра Web-документов (браузеров) выдвигали свои предложения по улучшению правил описания гипертекстовых данных.

Версия HTML 4.0, выпущенная Консорциумом в середине 1997 года и ставшая признанной спецификацией в конце того же года, является сегодня последней номерной версией языка разметки HTML. Некоторые несущественные дополнения, внесенные в спецификацию в декабре 1999 года, мало повлияли на структуру языка, а сама версия получила небольшое окончание в виде цифры 1 — 4.01.

Хочется добавить, что деятельность Консорциума, по сути, призвана регулировать и контролировать развитие и совершенствование языка гипертекстовой разметки HTML, учитывая потребности сферы интернет-технологий и компаний-разработчиков, работающих на рынке браузеров. Однако в действительности ситуация не такая однозначная: производители программ для просмотра HTML-документов постоянно предлагают технологические нововведения в спецификацию языка, часть которых получает одобрение Консорциума. Остальная часть инноваций, не вошедшая в состав принятой W3C рекомендации, может, тем не менее, внедряться в программную платформу выпускаемых браузеров, что на практике вызывает проблемы несовместимости электронных документов при их просмотре браузерами разных моделей и версий.

Какой браузер выбрать

Специальные программы для просмотра электронных документов, созданных по правилам языка разметки HTML, называются *браузерами*. Основная функция браузера заключается в интерпретации кода HTML и выводе визуального результата на экран монитора пользователя. Сегодня существует большое количество самых разнообразных браузеров, из которых наибольшей популярностью пользуются три программы: Internet Explorer, Netscape и Opera. Рассмотрим немного подробнее каждую из них.

Internet Explorer

В августе 1995 года компания Microsoft выпустила очередное обновление своей операционной системы Windows 95, в состав которой впервые был включен интернет-браузер Internet Explorer 1.0. Однако первоначальный программный код браузера принадлежал не Microsoft, а компании Spyglass, которая впоследствии продала лицензию на изменение и коммерческое распространение кода разработчикам Microsoft. Последние поставили Internet Explorer на более высокий уровень развития, что позволило браузеру-дебютанту составить достойную конкуренцию Netscape — браузеру, которым к началу 1995 года пользовалось свыше 50% пользователей Интернета.

В конце того же года Microsoft выпускает окончательный и доработанный вариант Internet Explorer 2.0, а весной 1996-го появляется версия 3.0, содержащая целый ряд нововведений, таких как поддержка фреймов, подключение программных надстроек (plug-in) и пр.

Выход четвертой версии Internet Explorer (вторая половина 1997 года) положил начало стремительному увеличению доли рынка браузеров корпорации Microsoft: к концу года по разным данным она обладала 60% рынка, в то время как ее основной соперник — компания Netscape — всего около 30%.

Однако Microsoft не останавливается на достигнутом превосходстве. В 1999 году выходит пятая версия популярного во всем мире браузера, в котором были реализованы новые и усовершенствованы старые технологии. Примерно через год появляется версия 5.5 и, наконец, в октябре 2001 года свет увидела последняя (на момент написания книги) версия — 6.0.

Можно назвать несколько основных возможностей браузера, благодаря которым Internet Explorer получил такую популярность;

- быстрый запуск программы;
- поддержка многих технологий, не реализованных или реализованных недостаточно в других браузерах (ActiveX, CSS1/CSS2, "плавающие фреймы" и др.);

- демократичность интерпретации HTML-кода (при загрузке документа, код которого содержит незнакомые конструкции и/или ошибки, Internet Explorer, в большинстве случаев, просто не выведет на экран часть, вызывающую затруднения; в то время как, например, Netscape может отобразить структуру такого документа нарушенной или вообще ничего не вывести на экран монитора);
- полная интеграция с другими приложениями Microsoft, работающими под управлением ОС Windows;
- возможность масштабирования графических изображений, открытых в отдельном окне.

Среди недостатков можно выделить:

- нестабильность работы;
- невысокую скорость загрузки информации.

Netscape

Компания Netscape Communications Corporation практически с самого начала считалась основным противником корпорации Microsoft в продвижении своего браузера.

Современный браузер Netscape берет свое начало в марте 1993 года, когда один из основателей будущей компании — Марк Андрессен (Marc Andreessen) — анонсирует выход программы Mosaic — прототип будущего браузера Netscape. На следующий год Марк Андрессен и его коллега Джим Кларк (Jim Clark) создают компанию Mosaic Communications (будущая Netscape Communications Corporation), а через несколько месяцев на свет появляется первая версия интернет-браузера Netscape — 0.9.

Компания расширяется, Netscape получает распространение, и, в итоге, к лету 1995 года большинство пользователей Всемирной паутины (около 80%) применяют для путешествия по Интернету именно эту программу. Но вскоре появляется Internet Explorer, который начинает всерьез конкурировать с Netscape, и основатели компании выпускают новую версию своего браузера — 2.0 — снабдив его не только новым именем Netscape Navigator, но и новыми техническими возможностями, тем самым развязав знаменитую "войну браузеров".

В конце 1998 года крупнейший интернет-провайдер Соединенных Штатов Америки America-On-Line (AOL) покупает компанию Netscape и все права на развитие одноименного браузера. В 2000 году выходит Netscape 6.0, параллельно с выходом которого анонсируется проект Mozilla 0.6: оба приложения на тот момент использовали единое программное ядро Gecko, но Netscape как

торговая марка принадлежал AOL, а Mozilla разрабатывался в качестве независимого проекта.

Наконец, в августе 2002 года выходит последняя на момент написания книги версия Netscape — 7.0, а следом за ней появляется Mozilla 1.0.

Основными преимуществами браузеров семейства Netscape являются:

- сравнительно небольшой размер программы;
- расширенное управление пользователем содержанием электронных документов;
- организация внутренних данных;
- поддержка технологии смены *скинов* (изменение внешнего вида программы в соответствии с выбранной оформительской схемой).

Недостатков тоже хватает, учитывая трудный путь развития, который прошли браузеры Netscape:

- отсутствие поддержки некоторых интерактивных технологий, рекомендованных Консорциумом W3C;
- низкая скорость работы;
- слишком большое время запуска программы.

Opera

Компания Opera Software (г. Осло, Норвегия) разработала одноименный браузер в 1994 году для норвежской телекоммуникационной компании Telenor. Группа разработчиков, включавшая в себя двух основателей Opera Software — Йона Штефенсона фон Тежнера (Jon Stephenson fon Tetzchner) и Гера Иварсоя (Geir Ivarsoy), поставила перед собой цель создать интернет- и мультимедиа-приложение, которым можно было пользоваться независимо от системных требований компьютеров.

В первоначальную концепцию браузера были положены такие критерии, как быстрый запуск программы и загрузка информации, небольшой размер приложения, минимальные требования к ресурсам компьютера пользователя.

Программа, изначально задуманная как небольшой по размеру и быстрый браузер для компьютеров с маленькими ресурсами, какое-то время использовалась в пределах внутренней информационной сети компании Telenor, а к концу 1995 года авторы Opera покинули стены компании, чтобы продолжить самостоятельное развитие своего "детища". Наконец, во второй половине 1996 года браузер Opera 2.1 стал доступен для загрузки в Интернете в качестве 90-дневной условно-бесплатной (Shareware) версии.

К числу основных отличий Opera от других браузеров, которые с полной уверенностью можно считать преимуществами, можно отнести следующие:

- небольшой размер;
- минимальные системные требования;
- высокая скорость загрузки HTML-документов;
- расширенные настройки;
- высокая масштабируемость просматриваемого документа.

Однако и у Opera есть ряд недостатков, которые следует упомянуть:

- статус коммерческого программного продукта (регистрация — $39 для полной версии; Freeware-вариант будет постоянно вас "радовать" показом чужих рекламных баннеров);
- отсутствие поддержки некоторых русских кодировок в английских версиях программы;
- недостаточно высокий уровень надежности выполнения *скриптов* на стороне пользователя (JavaScript/VBScript).

На момент написания книги последней версией браузера Opera являлась версия 7.x (до конца 2003 года компанией-разработчиком было выпущено несколько обновлений браузера с незначительными функциональными дополнениями).

Как видите — каждый из описанных интернет-браузеров прошел долгую историю развития, имеет свои плюсы и минусы, получает одобрение и вызывает неудовольствие, обладает сторонниками и противниками своих функциональных возможностей. В конечном итоге выбор браузера, с которым вы будете работать в процессе освоения языка гипертекстовой разметки HTML, остается за вами.

В заключение приведем статистику использования браузеров. Согласно подсчетам известной исследовательской группы OneStat, на январь 2004 года около 94,4% пользователей во всем мире выходило в Интернет с помощью Internet Explorer (из них 68,1% — версия 6.0, 13,8% — версия 5.5, 11,8% — версия 5.0 и 0,7% — версия 4.0). На долю Netscape пришлось всего 1,8%. Что касается Opera 7.0, то этот браузер отстает безнадежно и насчитывает лишь 0,8%.

Статистика использования браузеров в российской части Интернета почти повторяет мировые показатели: Internet Explorer (версии: 4.0, 5.0, 5.2, 5.5 и 6.0) — 92%, Netscape — 2,8%, Opera 7.0 — 3,8% (источник: Система интернет-статистики HotLog, январь 2004 года).

Структура HTML-документа

Как уже было сказано выше, язык HTML представляет собой набор специальных правил. Каждому правилу соответствует свое название, свойство и значение. Например, чтобы задать правило жирного начертания обыкновенного текста, необходимо использовать следующую HTML-конструкцию:

```
<B>Обыкновенный текст</B>
```

Как видно из примера, текст, который должен отображаться жирным начертанием, обособлен символами B и /B, заключенными в угловые скобки < и >. Конструкцию <символ> принято называть *тегом*. Сразу следует сказать, что теги бывают одинарными и парными. В нашем случае тег <B> является парным, т. к. он закрывает HTML-конструкцию вместе с символом / (прямой слэш). Иногда теги, которые необходимо закрывать парным тегом, называют тегами-контейнерами.

Структура любого тега подразумевает указание самого тега, его параметра и значения этого параметра. При этом наименование параметра и его значения можно писать как строчными, так и прописными буквами. Значение параметра ставится в кавычки.

Кроме этого, параметры и их значения у определенного тега могут либо отсутствовать, либо считаться необязательными и устанавливаться только в случае изменения стандартных характеристик тега:

```
<TABLE WIDTH="100%" BORDER>
```

В приведенном примере для тега построения таблицы <TABLE> один параметр (WIDTH) указан со значением, другой (BORDER) такового не содержит.

Некоторые теги могут указываться вообще безо всяких параметров (например, принудительный перенос строки <BR>). Другие теги могут включать несколько значений одного параметра ("INDEX, FOLLOW" для метаопределения "robots").

Любой HTML-документ содержит три основных обязательных раздела: HTML, HEAD и BODY. Рассмотрим подробнее каждый из них.

Раздел *HTML*

Раздел HTML определяет специфику документа, содержание которого будет интерпретироваться браузером. Раздел описывается тегом-контейнером <HTML></HTML> и дает браузеру информацию о том, что документ разработан с помощью языка разметки HTML. В предыдущих версиях языка тег <HTML> использовался с некоторыми параметрами, однако спецификация 4.0 отмени-

ла это правило (в принципе, сегодня большинство браузеров способно распознать HTML-документ и без указания данного тега, тем не менее, пропускать раздел HTML разработчикам не рекомендуется).

Раздел *HEAD*

Раздел HEAD выполняет функцию рабочего заголовка HTML-документа и является, по сути, "бойцом невидимого фронта": теги, указываемые внутри раздела, чрезвычайно важны и могут сильно влиять на внешний вид документа, но сами остаются незаметными глазу пользователя. Данному разделу сопоставлен парный тег <HEAD></HEAD>.

Внутри раздела HEAD можно указывать следующие теги HTML.

Название документа *TITLE*

Парный тег <TITLE></TITLE> предназначен для указания имени созданного электронного документа. Следует помнить, что под именем документа в данном случае имеется в виду не файловое наименование, а визуальный заголовок HTML-страницы.

Указание конструкции <TITLE></TITLE> не является обязательным, однако рекомендуется по ряду причин:

- отсутствие тега названия документа заставит браузер при интерпретации HTML-кода вывести в заголовке окна фразу типа "Untitled Document" ("Документ без названия"), что не соответствует ни тематике вашего электронного документа, ни его наполнению;
- при попытке добавить созданный вами HTML-документ (без тегов <TITLE></TITLE>) в "закладки" браузера пользователю придется самостоятельно вписывать название добавляемой страницы;
- поисковые системы, столкнувшись с "безымянной" страницей, занесут ее в свои базы данных под заголовком Untitled, что сделает HTML-документ безликим и похожим на миллионы других электронных документов, размещенных в Интернете.

Связь между документами *LINK*

Часто бывает так, что несколько различных документов структурно или функционально взаимосвязаны между собой. В этом случае используется тег <LINK>, который не является парным, а значит — не требует указания закрывающего тега:

```
<LINK HREF="styles/main.css" TYPE="text/css" REL="stylesheet">
```

Приведенный пример показывает отношение между HTML-документом, в разделе HEAD которого указана данная кодовая конструкция, и внешним файлом шаблона стилей main.css.

Возможные параметры тега <LINK> приведены в табл. 1.1.

Таблица 1.1. *Возможные параметры тега <LINK>*

Параметр	Назначение
HREF	Указание ссылки на внешний документ
TYPE	Тип внешнего документа
REL	Отношение между текущим и внешним документом
REV	Отношение между внешним и текущим документом

Метаопределения *META*

Метаопределения электронного документа описываются целым рядом параметров, входящих в состав непарного тега <META>, и предназначены для описания внутренних свойств HTML-файла.

Все метаопределения имеют два основных типа данных: HTTP-EQUIV и NAME. Первый можно отождествить с заголовками протокола передачи гипертекстовых данных (HTTP — HyperText Transfer Protocol). Структура метатега этого типа выглядит так:

```
<META HTTP-EQUIV="имя" CONTENT="содержание">
```

Существуют следующие кодовые конструкции данного типа метаопределений (с использованием соответствующих параметров HTML).

META HTTP-EQUIV="expires"

Конструкция определяет возможность запрета *кэширования* HTML-страниц за счет указания даты устаревания документа (актуальна для интернет-ресурсов с динамически изменяющимся содержанием). По истечении срока, указанного в этом параметре, браузер — при повторном обращении к документу — будет перенаправлен к источнику для обновления информации. Значение параметра CONTENT описывается в следующей последовательности: день недели (Mon, Tue, Wed и т. д.), число (01, 02, 03 и т. д.), месяц (Jan, Feb, Mar и т. д.), год, время (часы, минуты, секунды) и часовой пояс (GMT):

```
<META HTTP-EQUIV="expires" CONTENT="Sat, 25 Dec 2004 15:30:00 GMT">
```

Приведенный пример говорит о том, что в субботу 25 декабря 2004 года, в 15 часов 30 минут по Гринвичу "срок годности" HTML-документа истечет,

что потребует от браузеров обновления данных при обращении к содержанию страницы.

META HTTP-EQUIV="refresh"

Такая конструкция предписывает браузеру перезагружать содержимое окна через заданный промежуток времени. При добавлении дополнительного параметра `URL` через указанное время (в секундах) будет произведена переадресация на внешний адрес. Следующий пример дает браузеру команду ровно через 10 секунд перенаправить посетителя по адресу **http://www.site.ru**:

```
<META HTTP-EQUIV="refresh" content="10; URL=http://www.site.ru">
```

Причин для использования такой конструкции может быть несколько: ваш Web-сайт переехал на другой сервер, вы хотите показать рекламный блок, а потом перенаправить посетителя на заглавную страницу и т. д. Простая перезагрузка содержимого страницы может пригодиться в случае динамического изменения информации, работы *скрипта* по случайному выводу данных, электронных презентаций и пр.

META HTTP-EQUIV="Content-Type"

Главная функция этой конструкции — определение типа и кодировки документа. Наиболее используемыми значениями кодировки русскоязычных HTML-документов являются Windows-1251 и KOI8-R. На некоторых серверах возможно автоматическое перекодирование документов, в связи с чем использовать эту конструкцию не рекомендуется. Также необходимо проследить, чтобы кодировка текста документа совпадала с кодировкой, обозначенной в значении `charset`.

```
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=Windows-1251">
```

Упомянутые и малоиспользуемые конструкции типа `META HTTP-EQUIV` перечислены в табл. 1.2.

Таблица 1.2. *Параметры типа метаопределений* `HTTP-EQUIV`

Параметр	Функция	Подпараметры
`expires`	Запрет кэширования HTML-документа по истечении указанного срока	—
`refresh`	Перезагрузка/переадресация через заданный промежуток времени	`URL`
`Content-Type`	Определение типа и кодировки документа	`charset`
`Content-Language`	Указание языка документа	—
`Cache-Control`	Контроль кэширования документа	—

Таблица 1.2 *(окончание)*

Параметр	Функция	Подпараметры
Window-target	Определение места загрузки документа (используется для фреймовых структур)	—
Pragma	Управление кэшированием документа по протоколу HTTP/1.0	—
Set-Cookie	Настройка чтения данных Cookies (данных пользовательского компьютера)	name expires domain path secure
Ext-cache	Управление альтернативным кэшем документа	name instructions
Location	Указание места расположения документа в Интернете (полный адрес)	—

Теперь рассмотрим основные конструкции метаопределений второго типа — NAME.

META NAME="Keywords"

Конструкция задает набор ключевых слов документа, предназначенных для индексирования поисковыми системами (добавления информации о документе в базы данных). Слова указываются через запятую:

```
<META NAME="keywords" CONTENT="HTML, web, WWW, Web-страница, Интернет">
```

META NAME="Description"

Предоставление небольшого описания текущего HTML-документа, также необходимого для поисковых систем (при поиске информации описание выводится рядом со ссылкой на найденный интернет-ресурс):

```
<META NAME="description" CONTENT="HTML. Быстрый старт: Быстрое освоение
популярного языка гипертекстовой разметки HTML">
```

META NAME="Author"

Конструкция предназначена для указания автора (авторов) текущего документа.

```
<META NAME="Author" CONTENT="Алексей Петюшкин">
```

META NAME="Robots"

Одна из наиболее важных конструкций, используемая для описания правил индексирования документа поисковыми системами (роботами).

Отдельно следует перечислить возможные значения параметра CONTENT:

- ALL — разрешение индексирования документа со всеми присутствующими в нем гиперсвязями (ссылками);
- NONE — запрет индексирования документа со всеми присутствующими в нем гиперсвязями;
- INDEX — разрешение индексирования документа;
- NOINDEX — запрет индексирования документа;
- FOLLOW — разрешение индексирования присутствующих в документе гиперсвязей;
- NOFOLLOW — запрет индексирования присутствующих в документе гиперсвязей.

Если конструкция META NAME="robots" не указывается, то поисковый робот индексирует документ со всеми присутствующими в нем гиперссылками (равнозначно значению ALL или INDEX, FOLLOW).

```
<META NAME="robots" CONTENT="ALL">
```

Эти и некоторые другие конструкции типа NAME перечислены в табл. 1.3.

Таблица 1.3. *Параметры типа метаопределений NAME*

Параметр	Функция	Подпараметры
keywords	Указание ключевых слов документа	—
description	Указание краткого описания документа	—
author	Информация об авторе (авторах) документа	—
robots	Составление правил для индексирования документа поисковыми системами (роботами)	ALL NONE INDEX NOINDEX FOLLOW NOFOLLOW
rating	Установление возрастной категории содержания документа	—
document-state	Определение частоты индексирования документа	—
revisit	Определение промежутка времени, через которое должно производиться повторное индексирование документа	—
distribution	Категория распространения документа (массовое или ограниченное)	—

Таблица 1.3 *(окончание)*

Параметр	Функция	Подпараметры
URL	Указание основного адреса документа для индексирования (в случае существования "зеркальных" копий)	—
copyright	Установление авторского права и перечисление условий распространения документа	—
generator	Информация о программном обеспечении, с помощью которого создавался документ	—

В целом, следует помнить, что использование всех метаопределений не обязательно. В зависимости от типа документа, его содержания и прочих факторов, следует выбрать только самые необходимые теги метаданных. Наиболее важными являются данные для поисковых систем и указание кодировки документа.

Элементы *STYLE* и *SCRIPT*

В структуру раздела документа HEAD также могут входить теги-контейнеры <STYLE></STYLE> и <SCRIPT></SCRIPT>. Первый парный тег описывает стилевые шаблоны документа, второй — содержит код исполняемых сценариев (скриптов).

Раздел *BODY*

Раздел BODY является одним из самых важных компонентов любого HTML-документа, т. к. в нем располагается содержательная часть, которая выводится браузером на экран монитора пользователя.

Раздел описывается парным тегом <BODY></BODY>, внутри которого размещается большинство существующих тегов HTML. Тег <BODY> имеет ряд параметров, которые условно можно разделить на четыре основные группы (см. табл. 1.4).

Параметры заднего фона

Параметрами заднего фона документа являются BGCOLOR, BACKGROUND и BGPROPERTIES. BGCOLOR устанавливает цвет заднего фона, значение которого может быть введено в символьном эквиваленте, в шестнадцатеричном коде или в формате цветовой модели RGB. Система указания цвета в HTML основана на трех основных цветах: красном, зеленом и синем (модель RGB — Red, Green, Blue). Любое значение RGB может быть преобразовано в *шестнадцатеричный формат* от 00 до FF с приставкой # (читается "диез"). Некоторым значениям упомянутых моделей соответствует символьное название цвета. Таким образом, один и тот же цвет можно указать тремя способами.

Ниже показано три варианта установки цвета заднего фона (белый):

```
<BODY BGCOLOR="white">
<BODY BGCOLOR="#FFFFFF">
<BODY BGCOLOR="255,255,255">
```

Параметр `BACKGROUND` позволяет накладывать на задний фон документа графическое изображение:

```
<BODY BACKGROUND="images/bg.gif">
```

Параметр `BGPROPERTIES` поддерживается только браузером Microsoft Internet Explorer и позволяет менять свойства заднего фона документа. Например, конструкция

```
<BODY BACKGROUND="images/bg.gif" BGPROPERTIES="fixed">
```

позволит прокручивать содержание документа, оставляя фоновое графическое изображение в зафиксированном виде.

Описанные параметры не являются обязательными, однако использование `BGCOLOR` рекомендуется по следующей причине: пользователь в настройках своего браузера может поставить любой цвет заднего фона, а разработчик, полагая, что белый цвет является основным по умолчанию, может не указать этот параметр. В результате, вместо подразумевающегося белого цвета может оказаться черный, зеленый и т. д., что может привести к нарушению оформления документа. Также часто вместе используются параметры цвета и графического изображения для заднего фона: если рисунок не загрузился браузером, подставляется наиболее близкий к нему цвет.

Параметры границ документа

Параметры границ HTML-документа создают отступы заданного размера от верхнего, нижнего, левого и правого края документа. Этим отступам соответствуют параметры `TOPMARGIN`, `BOTTOMMARGIN`, `LEFTMARGIN` и `RIGHTMARGIN`. Значения для них задаются в *пикселах*:

```
<BODY TOPMARGIN="5" BOTTOMMARGIN="5" LEFTMARGIN="10" RIGHTMARGIN="10">
```

Однако данные параметры неправильно распознаются некоторыми браузерами, в частности Netscape, который объединяет упомянутые параметры в две группы: горизонтальные отступы и вертикальные отступы:

```
<BODY MARGINWIDTH="10" MARGINHEIGHT="5">
```

Таким образом, если вы хотите учесть особенности всех браузеров, в HTML-конструкцию подставляются и те, и другие параметры:

```
<BODY TOPMARGIN="5" BOTTOMMARGIN="5" LEFTMARGIN="10" RIGHTMARGIN="10"
MARGINWIDTH="10" MARGINHEIGHT="5">
```

Параметры текста

Из параметров текста документа реально применяется только один — TEXT. Он задает цвет основного текста на странице (значение параметра может быть введено аналогично цвету заднего фона документа):

```
<BODY TEXT="black">
```

Параметры гиперссылок

Параметры *гиперссылок* (связей с внутренними или внешними документами) определяют цвет активных (ALINK), непосещенных (LINK) и посещенных (VLINK) ссылок:

```
<BODY LINK="#0000FF" ALINK="#0000FF" VLINK="blue">
```

Таблица 1.4. *Параметры тега* <BODY>

Параметр	Функция
BGCOLOR	Определение цвета заднего фона
BACKGROUND	Указание фонового рисунка
BGPROPERTIES	Изменение свойств заднего фона (например, фиксирование фонового рисунка)
TOPMARGIN, BOTTOMMARGIN, LEFTMARGIN, RIGHTMARGIN, MARGINWIDTH, MARGINHEIGHT	Определение размера отступов
TEXT	Определение цвета основного текста
LINK, ALINK, VLINK	Определение цвета гиперссылок

Ваш первый HTML-документ

Итак, мы рассмотрели основные разделы любого HTML-документа. В целом, код простейшей страницы выглядит так (рис. 1.1):

Листинг 1.1. Пример простейшего HTML-документа

```
<HTML>
  <HEAD>
    <TITLE>Мой первый HTML-документ</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=Windows-1251">
```

```
<META NAME="keywords" CONTENT="HTML, документ, первый">
<META NAME="description" CONTENT="Это мой первый HTML-документ">
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TOPMARGIN="30" TEXT="black" LINK="#00FF00"
ALINK="#00FF00" VLINK="blue">
Это мой первый HTML-документ!
</BODY>
</HTML>
```

Теперь сохраните код под любым именем, не забыв присвоить файлу расширение html или htm. Откройте файл в браузере. Это ваш первый HTML-документ!

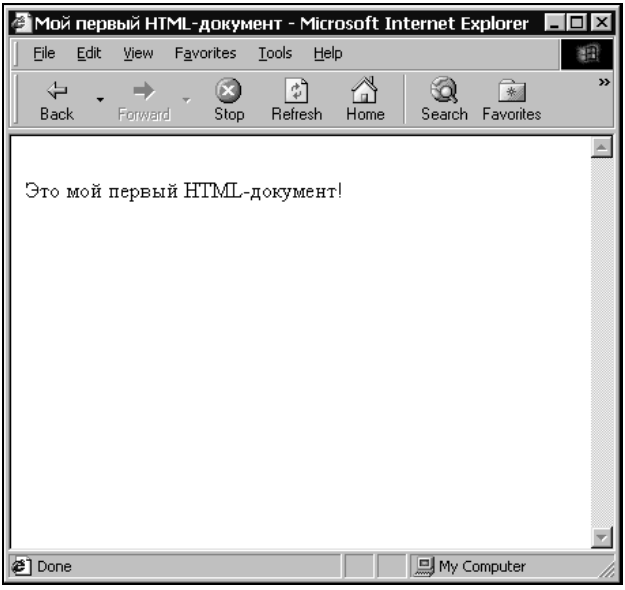


Рис. 1.1. Ваш первый HTML-документ

ГЛАВА 2

Форматирование текста и гиперссылки

Форматирование текста

Можно с полной уверенностью заявить, что текст занимает далеко не последнее место в структуре HTML-документа. В большинстве случаев мы посещаем интернет-сайты в поисках именно текстовой информации: статей и художественных произведений, новостей и технической документации и т. п. И от того, как оформлен текст, как он преподнесен посетителю, зависит многое в судьбе любого Web-сайта.

Область работы с текстом в HTML включает в себя большое количество всевозможных тегов, составляющих две основные группы: теги логического и физического форматирования.

В группу тегов *логического форматирования* входят теги, отображающие на экране монитора элементы документа таким образом, как установлено по умолчанию в спецификации языка разметки HTML. Переопределить их параметры или свойства нельзя, за исключением использования стилевых шаблонов CSS и обособления тегами физического форматирования. Результат действия разных тегов логического форматирования визуально может совпадать, ибо основное их предназначение заключается в логическом акценте отдельных элементов HTML.

Теги *физического форматирования* позволяют разработчику HTML-документа визуально изменять текст, варьируя его параметры и значения. Другими словами, теги физического форматирования предназначены для выделения отдельных текстовых фрагментов различными способами, установленными автором документа.

Рассмотрим более подробно наиболее используемые теги каждой группы.

Теги логического форматирования

Для логического форматирования служат следующие теги:

- <ACRONYM> — используется для расшифровки аббревиатур. Реализуется через параметр <TITLE> (листинг 2.1) и отображается в браузере при наведении курсора на слово-аббревиатуру (рис. 2.1).

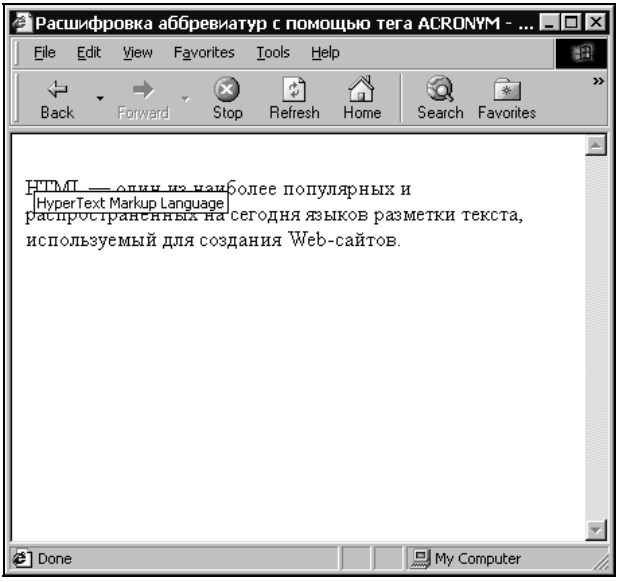


Рис. 2.1. Использование тега <ACRONYM>

Листинг 2.1. Использование тега <ACRONYM>

```
<HTML>
  <HEAD>
    <TITLE>Расшифровка аббревиатур с помощью тега ACRONYM</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<ACRONYM TITLE="HyperText Markup Language">HTML</ACRONYM> — один
из наиболее популярных и распространенных на сегодня языков разметки
текста, используемый для создания Web-сайтов.
</BODY>
</HTML>
```

Помимо <ACRONYM>, возможность отображения слова в сокращенном виде и подсказки к нему существует у тега <ABBR>.

- <CITE> — призван выделять различные цитаты и высказывания, названия библиографических источников и пр. Текст, помещенный между тегами <CITE> и </CITE>, браузером выделяется курсивом. Визуально аналогичен тегам <EM> и <I>;
- <CODE><CODE> — предназначен для визуального выделения небольших фрагментов программного кода. Код, размещенный в этом парном теге, отображается моноширинным шрифтом;
- <DEL> — используется для обозначения удаленного текста. Имеет необязательные для указания параметры DATETIME и CITE (первый показывает дату удаления, второй ссылается на источник причины удаления). Текст, заключенный между тегами <DEL>, отображается зачеркнутым и аналогичен действию тега <S> или <STRIKE>;
- <EM> — используется для интонационного выделения определенного текстового фрагмента (листинг 2.2). Браузерами отображается курсивом (рис. 2.2) (так же, как теги <CITE> и <I>);

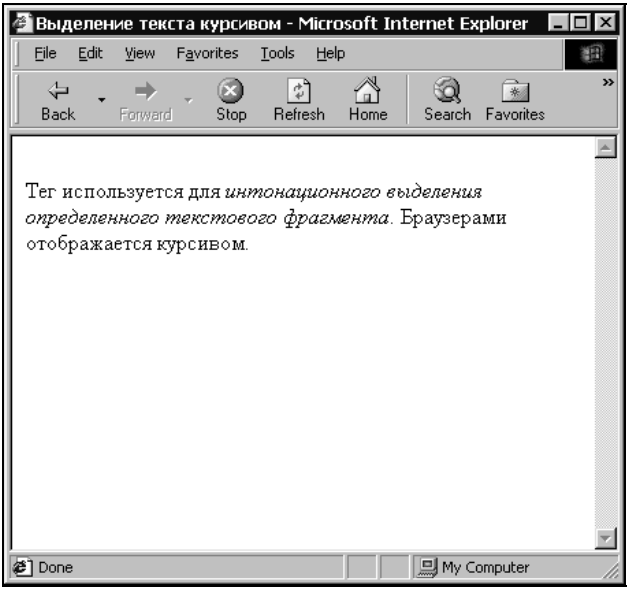


Рис. 2.2. Использование тега <EM>

Листинг 2.2. Использование тега <EM>

```
<HTML>
  <HEAD>
```

```
    <TITLE>Выделение текста курсивом</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
Тег используется для <EM>интонационного выделения определенного
текстового фрагмента</EM>. Браузерами отображается курсивом.
</BODY>
</HTML>
```

- `<H1>`, `<H2>`... `<H6>` — с указанием одной из шести цифр (от 1 до 6) обозначает определенного размера заголовок всего текста или его конкретного фрагмента. Соответственно, заголовок `<H6>` будет минимальным, а `<H1>` — самым большим (листинг 2.3 и рис. 2.3). Особенностью тегов `<H>` является то, что они уже подразумевают отступ от текстовой части, и разработчикам HTML-документов не приходится делать его самостоятельно.

  В общем понимании, слово "заголовок" подразумевает больший размер шрифта относительно прочего текста (названия глав в книге, анонсы статей в прессе и т. п.). Но в HTML заголовки `<H>` не будут являться самыми крупными текстовыми элементами, т. к. текст большего размера можно получить с помощью других конструкций, например `<FONT SIZE="7">Текст</FONT>` *(см. разд. "Теги физического форматирования" данной главы)*, или посредством стилевых шаблонов CSS *(см. главу 9)*;

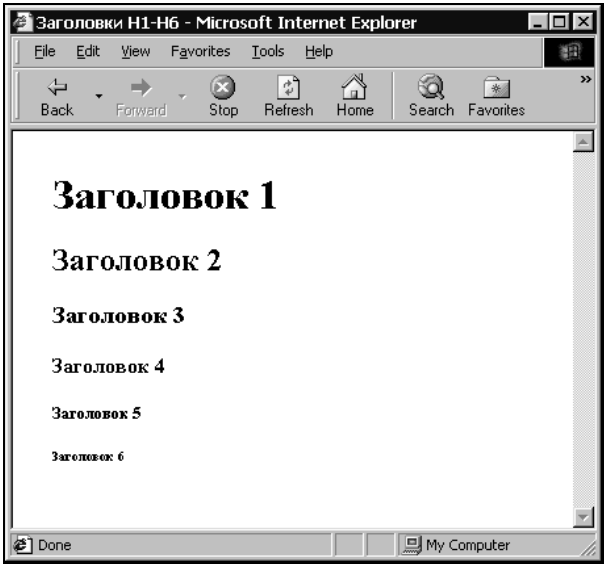


Рис. 2.3. Использование тегов `<H1>`-`<H6>`

Листинг 2.3. Использование тегов <H1>–<H6>

```
<HTML>
  <HEAD>
    <TITLE>Заголовки H1-H6</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<H1>Заголовок 1</H1>
<H2>Заголовок 2</H2>
<H3>Заголовок 3</H3>
<H4>Заголовок 4</H4>
<H5>Заголовок 5</H5>
<H6>Заголовок 6</H6>
</BODY>
</HTML>
```

- <STRONG> — используется для выделения фрагментов текста, на которых необходимо акцентировать внимание (листинг 2.4). Браузерами результат такого форматирования отображается в виде жирного начертания шрифта (рис. 2.4). В визуальном плане аналогичен тегу <B>.

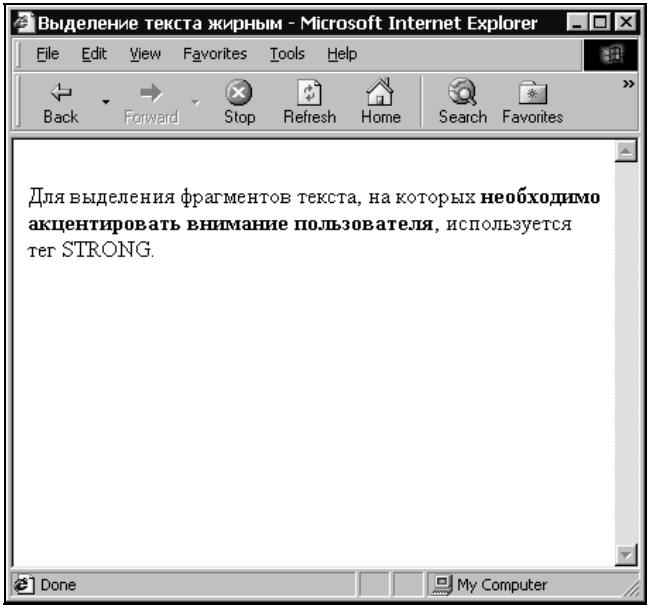


Рис. 2.4. Использование тега <STRONG>

Листинг 2.4. Использование тега <STRONG>

```
<HTML>
  <HEAD>
    <TITLE>Выделение текста жирным</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
Для выделения фрагментов текста, на которых <STRONG>необходимо
акцентировать внимание пользователя</STRONG>, используется тег STRONG.
</BODY>
</HTML>
```

Полный перечень рассмотренных тегов логического форматирования приведен в табл. 2.1.

Таблица 2.1. *Теги логического форматирования*

Тег	Функция
ACRONYM	Расшифровка аббревиатур
CITE	Указание цитаты
CODE	Указание фрагмента программного кода
DEL	Обозначение удаленной информации
EM	Выделение курсивом
H1-H6	Создание заголовков
STRONG	Выделение жирным начертанием

Теги физического форматирования

Для физического форматирования применяют следующие теги:

- <B> — его функция аналогична действию тега <STRONG>, а именно — визуальное выделение фрагмента текста, на котором следует сделать акцент (отображается жирным начертанием шрифта). В последней версии спецификации HTML вместо <B> рекомендуется использование тега логического форматирования <STRONG>;
- <BASEFONT> — предназначен для определения типа шрифта, а также его цвета и размера, которые будут считаться принятыми по умолчанию для всего HTML-документа. Параметры <BASEFONT> аналогичны тегу <FONT> и могут быть изменены по ходу текста дополнительным форматированием с помощью <FONT>.

Следует заметить, что тег <BASEFONT> используется в двух разделах документа: HEAD и BODY. Закрывающего тега <BASEFONT> не требует;

- <BIG> — если вам необходимо выделить часть текста небольшим увеличением размера шрифта относительно остальных слов, используйте тег <BIG>. Однако следует обратить внимание, что спецификация HTML не одобряет подобного подхода и рекомендует применять теги заголовков <H>;
- <BLINK> — поддерживается только браузером Netscape и реализует мигающий эффект выделенного фрагмента текста;
- <FONT> — один из основных тегов физического форматирования текста, отображающий свойства шрифтов. Для него могут использоваться следующие параметры:
  - FACE — посредством которого браузер отображает текст указанным в HTML-коде шрифтом. При этом значение параметра FACE должно соответствовать шрифту, установленному на компьютере пользователя. Если такого нет, текст будет показан стандартным (по умолчанию) шрифтом.

    Часто разработчики HTML-документов добавляют сразу несколько значений параметра FACE (листинг 2.5) для того, чтобы в случае отсутствия первого шрифта на компьютере пользователя браузер отобразил текст вторым. Не будет найдено второго шрифта, браузер приступит к поиску третьего и т. д. (рис. 2.5);

Листинг 2.5. Использование параметра FACE тега <FONT>

```
<HTML>
  <HEAD>
    <TITLE>Использование параметра FACE</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<H3>
<FONT FACE="Tahoma", "Arial", "Verdana">Этот текст будет показан
одним из 3-х указанных шрифтов.</FONT>
</H3>
<H3>
<FONT FACE="Allegro">Этот текст будет показан шрифтом по
умолчанию.</FONT>
</H3>
</BODY>
</HTML>
```

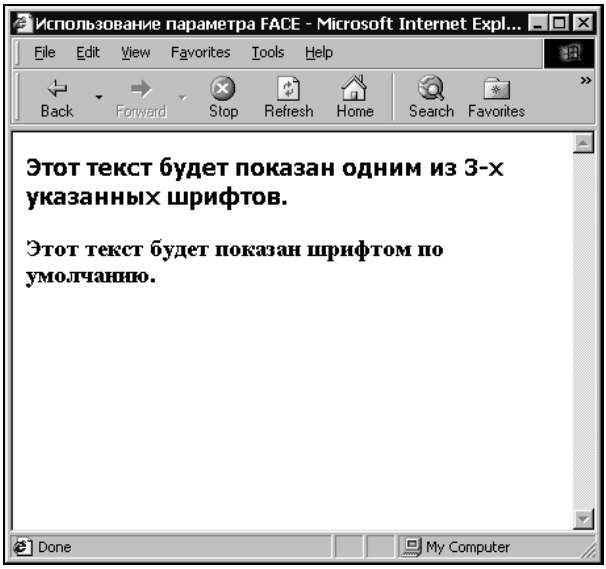


Рис. 2.5. Использование параметра FACE тега <FONT>

- COLOR — предназначен для выбора цвета, которым будет написан текст. HTML-конструкция имеет следующий вид:

```
<FONT COLOR="red">Этот текст будет написан красным цветом.</FONT>
```

- SIZE — его функцией является определение размера шрифта, которое происходит по условной шкале от 1 до 7. При этом указание размера может осуществляться как в виде относительного показателя (+1), так и абсолютного (1). Средним значением размера шрифта принято считать показатель 3. Однако окончательный вид шрифта с учетом свойств его размеров может различаться в зависимости от модели и версии браузера.

Наконец, возможно использование сразу всех параметров тега <FONT> (листинг 2.6, рис. 2.6) в пределах одной HTML-конструкции:

Листинг 2.6. Использование всех параметров тега <FONT>

```
<HTML>
  <HEAD>
    <TITLE>Использование всех параметров тега FONT</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
```

```
<FONT FACE="Tahoma", "Helvetica" COLOR="#000000" SIZE="2">Этот текст
будет показан одним из 3-х указанных шрифтов, черным цветом
и размером "1".</FONT>
</BODY>
</HTML>
```

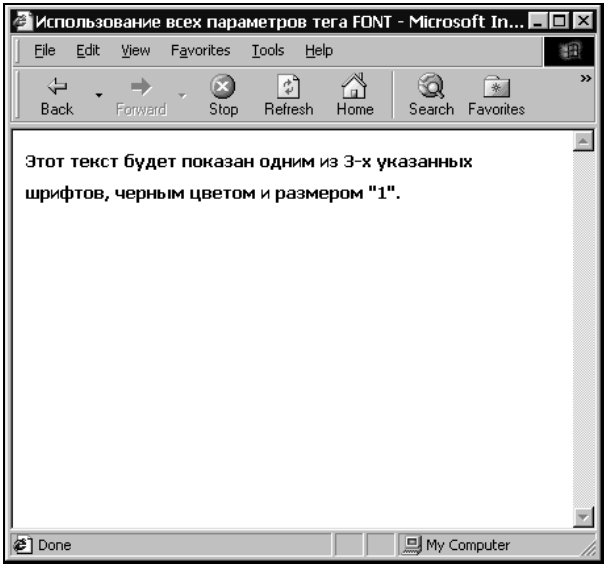


Рис. 2.6. Использование всех параметров тега <FONT>

- <I> — аналогичен тегам логического форматирования <CITE> и <EM>, выделяет нужную часть текста курсивом. В ряде случаев использование <I> рекомендуется заменять соответствующими тегами логического форматирования;
- <SMALL> — для установки размера шрифта, немного меньшего, чем остальные слова, следует заключить нужный фрагмент между парным тегом <SMALL>. Эффект, противоположный действию тега <BIG>;
- <SPAN> — применяется в случае, когда выделенному фрагменту текста необходимо присвоить определенные свойства, а к помощи логического тега прибегнуть нельзя;
- <S> и <STRIKE> — в последней версии спецификации HTML эти два тега были названы отмененными. Вместо них рекомендуется использовать тег <DEL>, также перечеркивающий выбранный фрагмент текста;
- <SUB> — позиционирует фрагмент текста относительно нижней линии строки (рис. 2.7). Весьма удобный инструмент для написания математических и химических формул;

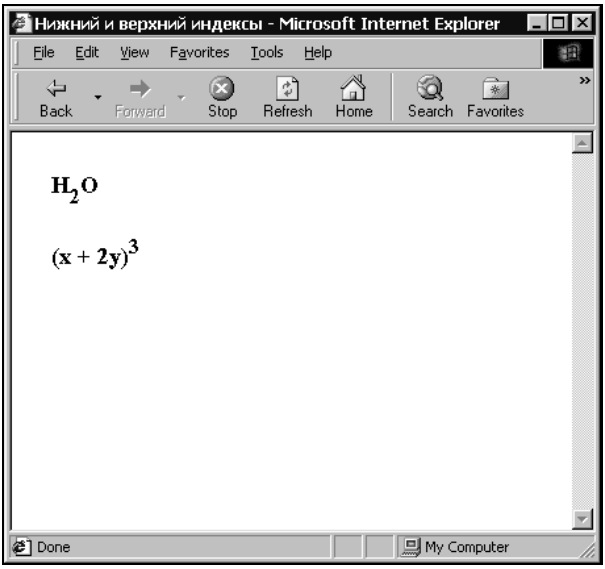


Рис. 2.7. Использование тегов нижнего и верхнего индексов <SUB> и <SUP>

- <SUP> — размещает текстовый фрагмент относительно верхней линии строки (рис. 2.7). Также подходит для включения в разнообразные формулы;
- <TT> — спецификация определяет <TT> как тег для отображения телетайпа или текста, набранного *моноширинным шрифтом*. В ряде случаев может быть заменен тегом <CODE>;
- <U> — делает текст подчеркнутым.

В завершение разговора о тегах физического форматирования необходимо добавить, что разные теги могут использоваться одновременно в отношении одного текстового фрагмента (листинг 2.7). Например, часть текста и даже отдельного слова может быть написана и курсивом, и жирным начертанием, и еще подчеркнута и т. д. Единственное, о чем следует помнить, — это сохранение последовательности указания закрывающих тегов. Другими словами, не рекомендуется, открыв фрагмент текста тегами <B> и <FONT>, закрывать HTML-конструкцию в том же порядке (т. е. порядок закрывающих тегов должен быть обратным).

Листинг 2.7. Одновременное использование нескольких тегов физического форматирования

```
<HTML>
  <HEAD>
```

```
    <TITLE>Одновременное использование нескольких тегов
     физического форматирования</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<FONT FACE="Verdana", "Arial" COLOR="gray" SIZE="3">В этом тексте
использовано <U>три различных тега</U> <B>физического
форматирования.</B></FONT>
</BODY>
</HTML>
```

Полный перечень рассмотренных тегов физического форматирования приведен в табл. 2.2.

Таблица 2.2. *Теги физического форматирования*

Тег	Функция
<B>	Выделение жирным начертанием
<BASEFONT>	Определение свойств шрифта для всего документа
<BIG>	Выделение фрагмента текста бóльшим размером, чем основной текст
<BLINK>	Создание эффекта мигания текста
<FONT>	Указание свойств шрифта (цвет, размер, название)
<I>	Выделение курсивом
<SMALL>	Выделение фрагмента текста меньшим размером, чем основной текст
<SPAN>	Присвоение свойств фрагменту текста при невозможности использования логических тегов
<S>, <STRIKE>	Обозначение удаленной информации
<SUB>	Создание нижнего индекса
<SUP>	Создание верхнего индекса
<TT>	Имитация телетайпа или моноширинного шрифта
<U>	Выделение подчеркиванием

Структурное форматирование

Как известно, любой текст имеет свою структуру. Книги разделены на части, главы и разделы, газеты и журналы имеют отдельные рубрики и подзаголовки, которые, в свою очередь, включают фрагменты текста, также имеющие собственную внутреннюю структуру: абзацы, отступы, параграфы и пр.

Текст, размещенный в HTML-документе, не исключение, он также должен иметь логичную, понятную каждому пользователю структуру. Ведь от того, насколько просто и удобно будет восприниматься текст на Web-странице, зависит многое: прежде всего, какое впечатление о HTML-документе сложится у посетителя.

Структурное форматирование в HTML подразумевает разбиение текстовых фрагментов электронного документа на логические блоки, которым соответствуют определенные элементы: абзац, текстовый блок, центрирование, отступы и перенос строки, горизонтальный разделитель, предварительно отформатированный текст и комментарии.

Рассмотрим каждый из названных элементов структурного форматирования.

Абзац

При наборе текста в текстовом редакторе (Microsoft Word, WordPerfect и др.) для обозначения *абзаца* мы дважды нажимаем клавишу <Enter>. Такое действие дает программе команду обособить один фрагмент текста от другого, задав "красную строку" (листинг 2.8). При создании HTML-документа для обозначения абзаца используется специальный тег `<P>`, который разделяет фрагменты текста вертикальным *отступом*. Двойной перевод строки в данном случае не поможет: браузер, интерпретируя код, не воспримет двойной отступ как команду создания абзаца и при выводе содержания на экран монитора объединит фрагменты текста вместе (рис. 2.8).

Листинг 2.8. Результат двойного перевода строки без использования тега <P>

```
<HTML>
  <HEAD>
    <TITLE>Результат двойного перевода строки без использования
    тега P</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<FONT FACE="Tahoma" COLOR="#000000" SIZE="3">При создании HTML-документа
для обозначения абзаца используется специальный тег <CODE>P</CODE>,
который разделяет фрагменты текста вертикальным отступом.

Двойной перевод строки в данном случае не поможет: браузер, интерпретируя
код, не воспримет двойной отступ как команду создания абзаца и при выводе
содержания на экран монитора объединит фрагменты текста вместе.</FONT>
</BODY>
</HTML>
```

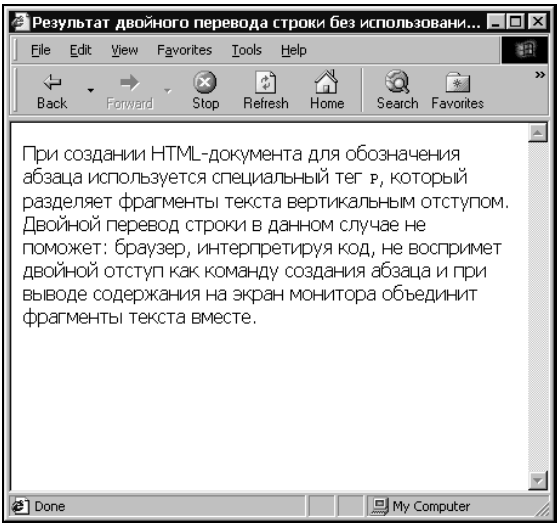


Рис. 2.8. Результат двойного перевода строки без использования тега `<P>`

Тег `<P>` является парным, однако наличие закрывающего тега, согласно последней спецификации HTML, необязательно. Так как `<P>` относится к разряду структурных тегов, включать другие элементы структурного форматирования он не может. Внутри него могут находиться только теги форматирования текста (логические и физические).

Тег `<P>` может содержать параметр `ALIGN` (листинг 2.9), отвечающий за тип горизонтального выравнивания текста в окне браузера (рис. 2.9):

- `ALIGN="left"` — значение параметра, принятое по умолчанию: текст выровнен по левому краю (при необходимости такого типа выравнивания указание параметра `ALIGN` необязательно);
- `ALIGN="center"` — текст располагается посередине окна браузера. Использование данного значения не рекомендуется при работе с большими фрагментами текста: из-за различной длины слов восприятие абзаца затрудняется;
- `ALIGN="right"` — выравнивание текста по правому краю. Идеально подходит для создания эпиграфов, подписей, заголовков и пр. Использование при работе с большими текстовыми фрагментами нежелательно;
- `ALIGN="justify"` — выравнивание по ширине окна браузера. Значение `JUSTIFY` стало поддерживаться браузерами сравнительно недавно, однако во многих случаях именно этот тип выравнивания подходит для работы с текстом.

Листинг 2.9. Использование различных типов выравнивания

```
<HTML>
  <HEAD>
    <TITLE>Использование различных типов выравнивания</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<P>При создании HTML-документа для обозначения абзаца используется
специальный тег <CODE>P</CODE>, который разделяет фрагменты текста
вертикальным отступом.</P>
<P ALIGN="center">Такой тип выравнивания подходит для заголовков
документов.</P>
<P ALIGN="right">Двойной перевод строки в данном случае не поможет:
браузер, интерпретируя код, не воспримет двойной отступ как команду
создания абзаца.</P>
<P ALIGN="justify">Значение JUSTIFY стало поддерживаться браузерами
сравнительно недавно, однако во многих случаях именно этот тип
выравнивания подходит для работы с текстом.</P>
</BODY>
</HTML>
```

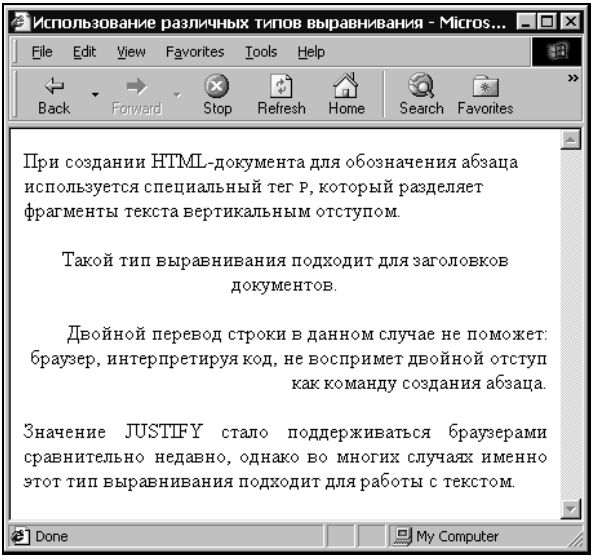


Рис. 2.9. Использование различных типов выравнивания

Как уже было сказано выше, тег создания абзаца `<P>` позволяет использовать все теги форматирования текста — как логического, так и физического. Следующий пример иллюстрирует эту возможность (листинг 2.10, рис. 2.10).

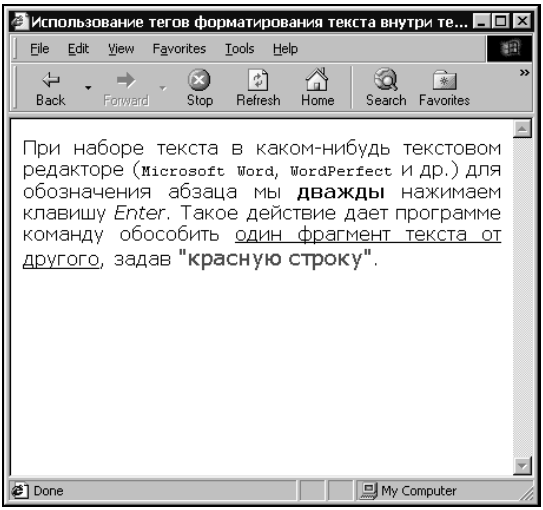


Рис. 2.10. Использование тегов форматирования текста внутри тега <P>

Листинг 2.10. Использование тегов форматирования текста внутри тега <P>

```
<HTML>
  <HEAD>
    <TITLE>Использование тегов форматирования текста внутри тега
P</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<P ALIGN="justify"><FONT FACE="Verdana" SIZE="3">
При наборе текста в каком-нибудь текстовом редакторе (<TT>Microsoft
Word</TT>, <TT>WordPerfect</TT> и др.) для обозначения абзаца мы
<STRONG>дважды</STRONG> нажимаем клавишу <EM>Enter</EM>. Такое действие
дает программе команду обособить <U>один фрагмент текста от другого</U>,
задав <FONT COLOR="red"><BIG>"красную строку"</BIG></FONT>.
</FONT></P>
</BODY>
</HTML>
```

В данном примере в конце кода стоят два закрывающих тега `</FONT>`: один завершает действие свойства `COLOR=red`, другой закрывает правила отображения свойств типа и размера шрифта (`FONT FACE="Verdana" SIZE="3"`).

Текстовый блок

В случае необходимости придания специальных свойств отдельному фрагменту текста используются теги текстовых блоков `<DIV>` и `<SPAN>`. Изменение

свойств осуществляется посредством назначения выбранному фрагменту текста стиля CSS, например:

```
<DIV STYLE="COLOR: GRAY;">
Фрагмент текста, набранный серым цветом
</DIV>
```

Подробную информацию о назначении стилей *см. в главе 9.*

Однако следует помнить, что между <DIV> и <SPAN> существуют важные отличия (рис 2.11). Во-первых, <DIV> является исключительно структурным тегом, а <SPAN> берет начало в области физического форматирования текста. Во-вторых, <DIV> создает принудительный перенос строки на одну позицию после своего закрывающего тега (в отличие от тега абзаца <P>, который осуществляет перенос на две позиции), поэтому задавать с его помощью отдельные свойства фрагмента внутри абзаца нельзя — это вызовет принудительный перенос строки. Если такая необходимость возникла, лучше использовать тег <SPAN>, который позволяет назначать новые правила отображения текстовых фрагментов без изменения структуры документа.

Для обоих элементов обязательно наличие закрывающих тегов, а также возможно добавление параметра типа выравнивания ALIGN, например:

```
<P ALIGN="justify">
Начало основного текста документа. <DIV ALIGN="right">Фрагмент
текста.</DIV> Окончание основного текста документа.
</P>
```

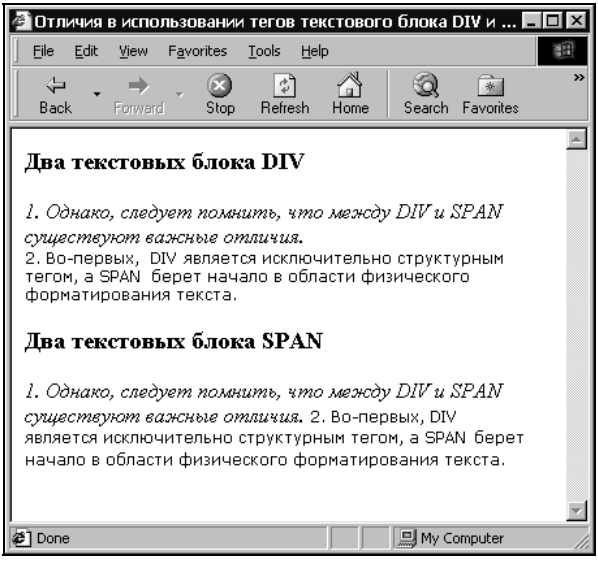


Рис. 2.11. Отличия в использовании тегов текстового блока <DIV> и <SPAN>

Центрирование

Центрирование любых элементов HTML-документа может быть осуществлено с помощью тега `<CENTER>`. Все данные, размещенные внутри тега-контейнера `<CENTER></CENTER>`, подлежат горизонтальному выравниванию посередине окна браузера. В сущности, тег `<CENTER>` представляет собой аналог значения `ALIGN="center"` того тега, результат отображения которого будет отцентрирован на экране монитора. К примеру, результат отображения фрагмента кода таблицы

```
<CENTER>
<TABLE><TR><TD></TD></TR></TABLE>
</CENTER>
```

и фрагмента кода

```
<TABLE ALIGN="center"><TR><TD></TD></TR></TABLE>
```

будет абсолютно одинаков. В первом случае использован структурный тег центрирования, во втором — указан соответствующий тип выравнивания всей таблицы относительно окна браузера.

Отступы и перенос строки

Часто во многих печатных и книжных изданиях мы можем наблюдать, что начало новой строки абзаца начинается после небольшого отступа. В текстовом редакторе эта возможность регулируется клавишей табуляции <Tab> или перемещением ползунка линейки настроек рабочей области документа (рис. 2.12).

В HTML размер отступа определяется кодовой конструкцией ` ` (листинг 2.11), которая визуально представляет собой обычный пробел. Создавая текст в документе, необязательно между каждым словом вставлять ` ` — браузер и так поймет, что нужно сделать отступ (рис. 2.13). Но если вам потребуется сделать отступ большего размера, и вы прямо в коде создадите несколько пробелов, браузер интерпретирует такие пропуски в виде единственного пробела.

Листинг 2.11. Создание отступов с помощью конструкции ` `

```
<HTML>
  <HEAD>
    <TITLE>Создание отступов</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
```

```
<P ALIGN="justify"><FONT FACE="Verdana" SIZE="3">
    При наборе текста в каком-нибудь текстовом
редакторе (Microsoft Word, WordPerfect и др.) для обозначения абзаца
мы дважды нажимаем клавишу Enter.
</FONT></P>
<P ALIGN="justify"><FONT FACE="Verdana" SIZE="3">
    Такое действие дает программе команду обособить
один фрагмент текста от другого, задав "красную строку".
</FONT></P>
</BODY>
</HTML>
```

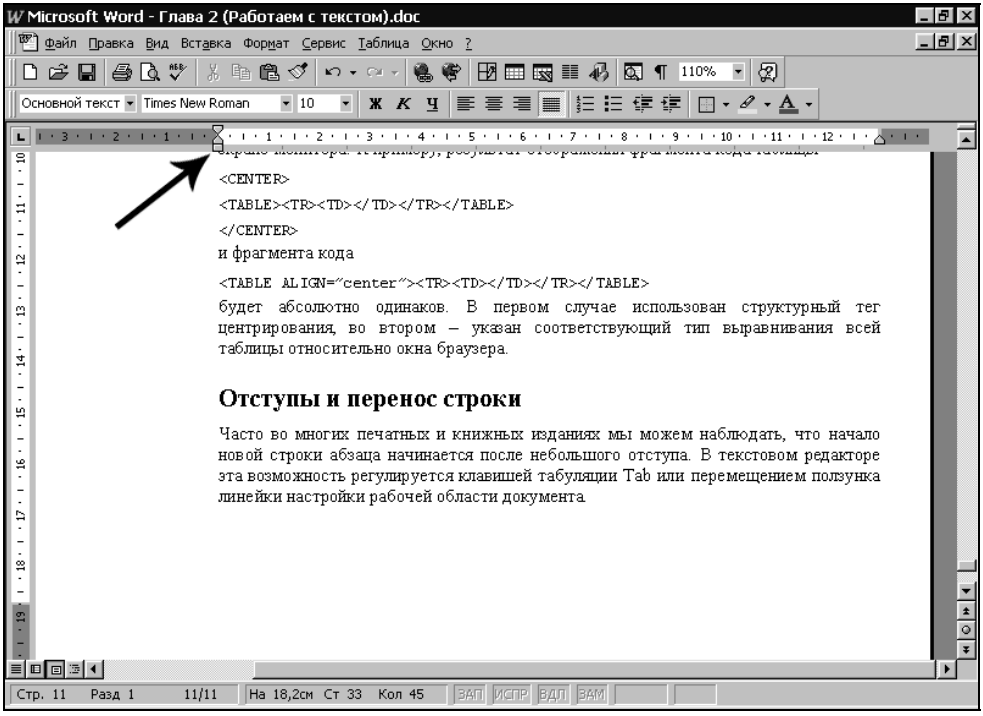


Рис. 2.12. Ползунок линейки рабочей области в текстовом редакторе Microsoft Word

Конструкция ` ` относится к группе *специальных символов* HTML, о которых пойдет речь в *разд. "Специальные символы" данной главы.*

Размер экрана монитора ограничен, и ширина рабочего окна браузера имеет свои рамки, вынуждающие переносить слова на следующую строку. Браузер переносит слова после пробела или некоторых символов (дефис, прямой слэш и пр.). То есть разделить целое слово, разместив начало на одной строке, а

окончание — на другой, браузер не может. Однако в ходе создания HTML-документа может возникнуть необходимость переноса строки: например, при размещении отрывка стихотворения каждая фраза должна начинаться с новой строки. Если мы переместим строки стихотворения посредством клавиши <Enter>, браузер отобразит отрывок в виде одного предложения. В таких случаях нужно использовать тег принудительного переноса строки `<BR>` (листинг 2.12, рис. 2.14).

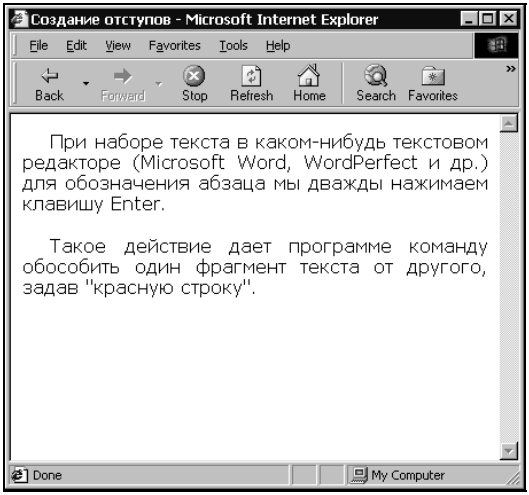


Рис. 2.13. Создание отступов с помощью конструкции ` `

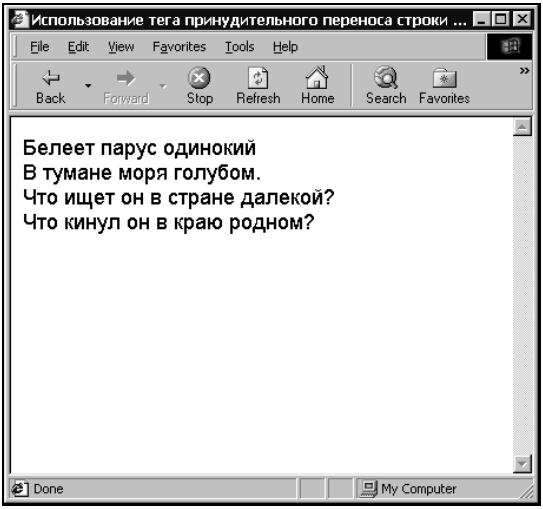


Рис. 2.14. Использование тега принудительного переноса строки `<BR>`

**Листинг 2.12. Использование тега принудительного переноса строки
**

```
<HTML>
  <HEAD>
    <TITLE>Использование тега принудительного переноса строки BR</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<P><FONT FACE="Arial" SIZE="4">
Белеет парус одинокий<BR>
В тумане моря голубом.<BR>
Что ищет он в стране далекой?<BR>
Что кинул он в краю родном?
</FONT></P>
</BODY>
</HTML>
```

Для переноса на несколько строк тег
 ставится соответствующее количество раз.

У разработчика HTML-документа может возникнуть ситуация, противоположная только что описанной: необходимость запрета переноса строки. Например, слово "интернет-каталог" при определенных размерах окна браузера может быть разделено на две части: "интернет" и "каталог", что, во-первых, неправильно с позиции правописания, а во-вторых, мешает восприятию слова целиком.

Чтобы этого не произошло, слово или словосочетание, для которого следует запретить перенос строки, необходимо поместить внутри тега-контейнера <NOBR></NOBR>.

```
<P><FONT COLOR="#003366"><STRONG>
Одним из наиболее эффективных способов рекламы Web-сайта является
регистрация в <NOBR>интернет-каталогах</NOBR> и поисковых системах.
</STRONG></FONT></P>
```

В этом случае переноса не произойдет ни при каких обстоятельствах, зато вероятны такие нежелательные последствия, как нарушение типа выравнивания текста, внезапный перенос смежных строк и возникновение горизонтальной прокрутки окна браузера.

Эти негативные последствия можно частично предотвратить с помощью тега <WBR>. Он размещается в том месте слова, для которого действует запрет переноса строки, которое является — по мнению автора документа — наименее опасным для переноса строки. Однако применение тега <WBR> сильно затруд-

нено тем, что многие браузеры (в частности, Netscape) его не поддерживают, а Internet Explorer распознает <WBR> только при наличии в коде тегов <NOBR>.

Горизонтальный разделитель

Ранее мы говорили об элементах структурного форматирования, которые разделяли фрагменты текста логически — посредством соответствующей HTML-разметки. В этом смысле использование *горизонтальных разделителей* (рис. 2.15) (или просто линий) можно назвать своеобразным исключением из правил, ибо они проводят визуальную границу между различными текстовыми фрагментами.

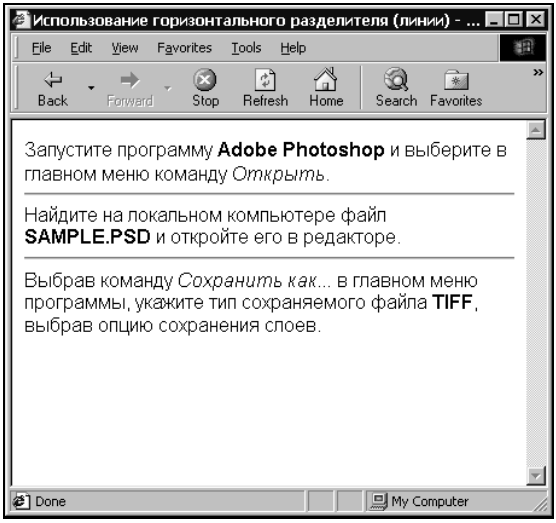


Рис. 2.15. Использование горизонтального разделителя (линии)

Горизонтальный разделитель описывается тегом <HR> (закрывающего тега нет) и имеет целый ряд параметров, позволяющих изменять визуальные свойства линии, добавлять рельефность и цвет, регулировать длину, задавать тип выравнивания и пр. Перечень параметров тега <HR> приведен в табл. 2.3.

Таблица 2.3. *Параметры тега <HR>*

Параметр	Функция
WIDTH	Указание длины линии (в процентах от ширины окна браузера или пикселах)
SIZE	Указание толщины линии (в пикселах)

Таблица 2.3 *(окончание)*

Параметр	Функция
ALIGN	Обозначение типа выравнивания линии (LEFT, CENTER, RIGHT)
COLOR	Выбор цвета линии (название, шестнадцатеричный код или формат RGB)
NOSHADE	Отмена эффекта рельефности линии

Пример HTML-конструкции для создания горизонтальной линии:

```
<HR WIDTH="90%" SIZE="1" ALIGN="CENTER" COLOR="#000000" NOSHADE>
```

В заключение следует добавить, что с точки зрения форматирования структуры документа, горизонтальная линия выполняет не только функцию визуального разделителя отдельных фрагментов текста, но и подразумевает перенос строки.

Предварительно отформатированный текст

Тег-контейнер *предварительно отформатированного текста* <PRE></PRE> позволяет выводить на экран монитора текст в том виде, как он выглядит в окне текстового редактора: со всеми переносами строк, отступами, абзацами и прочим форматированием. Такая возможность весьма удобна, например, при отображении фрагментов программного кода "как есть" — без его интерпретации браузером.

Визуально результат действия тега <PRE> выглядит как текст, набранный моноширинным шрифтом. В спецификации существует еще несколько похожих по функции тегов (<PLAINTEXT>, <XMP> и <LISTING>), однако они признаны устаревшими, а использовать Консорциум W3C рекомендует тег <PRE>.

Комментарии

Если горизонтальная линия является визуальным разделителем фрагментов текста, то *комментарии* невидимы на экране монитора, но в то же время могут нести большой объем информации.

Любой HTML-комментарий должен начинаться с конструкции <!-- и заканчиваться -->. Между ними может находиться любой текст, цифры, символы и прочее, за исключением тегов.

Для чего же нужны HTML-комментарии? Во-первых, они позволяют размечать структуру HTML-кода, давая заголовки различным логическим блокам элементов. Впоследствии такая разметка поможет быстро сориентироваться и найти необходимый фрагмент кода.

```
<!-- Начало блока новостей -->
Код блока новостей...
<!-- Окончание блока новостей -->
```

Во-вторых, в комментариях можно указать информацию об авторском праве и прочие персональные данные, которые будут проиндексированы поисковыми системами.

```
<!-- © 2004. Алексей Петюшкин. Все права защищены -->
```

При всем удобстве включения в HTML-код комментариев необходимо помнить, что просмотреть код *электронного документа* может практически любой пользователь Интернета, открыв нужную Web-страницу в любом текстовом редакторе. Поэтому размещать в комментариях информацию конфиденциального, коммерческого или личного характера не рекомендуется.

Перечень всех рассмотренных элементов и тегов структурного форматирования приведен в табл. 2.4.

Таблица 2.4. *Элементы и теги структурного форматирования*

Элемент	**Функция**	**Используемые теги**
Абзац	Создание текстовых абзацев	<P>
Текстовый блок	Визуальное выделение отдельных фрагментов текста	<DIV> <SPAN>
Центрирование	Горизонтальное выравнивание элементов по ширине окна браузера	<CENTER>
Отступы и перенос строки	Создание отступов в тексте и создание/запрет переносов строки	 <NOBR> <WBR>
Горизонтальный разделитель	Визуальное разделение фрагментов текста с помощью линии	<HR>
Предварительно отформатированный текст	Вывод отформатированного в текстовом редакторе на экран монитора в формате "как есть"	<PRE> <PLAINTEXT> <XMP> <LISTING>
Комментарии	Включение в HTML-код дополнительной информации о документе, невидимой при просмотре в браузере	<!-- -->

Специальные символы

К *специальным символам* HTML относятся символы, не входящие в состав стандартных ASCII-кодов. Их реализация в HTML-документах возможна при помощи отдельных *кодовых конструкций* или *числовых комбинаций*.

Специальные символы используются на многих интернет-ресурсах: это знак авторского права и зарегистрированной торговой марки, значки иностранных валют и математические символы, дробные числа и элементы HTML-форматирования, наконец, буквы иностранных алфавитов и многое другое. Например, значок доллара ($) можно легко ввести с помощью соответствующей клавиши на клавиатуре компьютера, не опасаясь за корректность отображения данного символа в любых кодировках, моделях и версиях браузеров. А вот отобразить таким же образом значок английской денежной единицы — фунта стерлингов — вам вряд ли удастся.

Условно все специальные символы HTML можно разделить на три большие категории:

- символы, отображающие элементы HTML-форматирования;
- символы оформления документа;
- буквы иностранных алфавитов.

Рассмотрим более подробно каждую категорию.

Символы, отображающие элементы HTML-форматирования

В ходе создания того или иного HTML-документа часто бывает необходимо отобразить в браузере символы, используемые в спецификации языка HTML: угловые скобки, обозначающие теги (< и >), знак *амперсанда* (&) и т. д. Чаще всего это случается при добавлении в исходный документ фрагмента HTML-кода, который впоследствии отображался бы в окне браузера как кодовый листинг, а не результат его исполнения.

Например, вы хотите обменяться с другими Web-сайтами кнопками, и для удобства обмена вам нужно предоставить посетителям вашего интернет-ресурса готовый HTML-код, который останется только выделить (прямо в окне браузера), скопировать в буфер обмена и вставить в соответствующее место на Web-странице:

```
<A HREF="http://www.site.ru/"><IMG SRC="http://www.site.ru/images
/button.gif" WIDTH="88" HEIGHT="31" BORDER="0"></A>
```

Если поместить этот код в данном виде на Web-страницу, то браузер загрузит указанный в коде графический файл со ссылкой на установленный в теге <A>

ресурс. Чтобы в браузере отображался сам код, нужно все левые угловые скобки в коде заменить на `<`, а правые — на `>`. После замены всех символов специальными кодовыми конструкциями первоначальный фрагмент кода примет следующий вид:

```
&lt;A HREF="http://www.site.ru/"&gt;&lt;IMG SRC="http://site.ru/images
/button.gif" WIDTH="88" HEIGHT="31" BORDER="0"&gt;&lt;/a&gt;
```

Символы оформления документа

К так называемым символам оформления HTML-документов относятся значки авторского права, зарегистрированной торговой марки, символы иностранных валют, математические обозначения ("плюс-минус", "умножить" и др.) и прочие символы. Приведенный на рис. 2.16 пример может быть получен за счет вставки в код документа следующих конструкций:

- `£` — фунт стерлингов;
- `®` — зарегистрированная торговая марка;
- `±` — знак "плюс-минус".

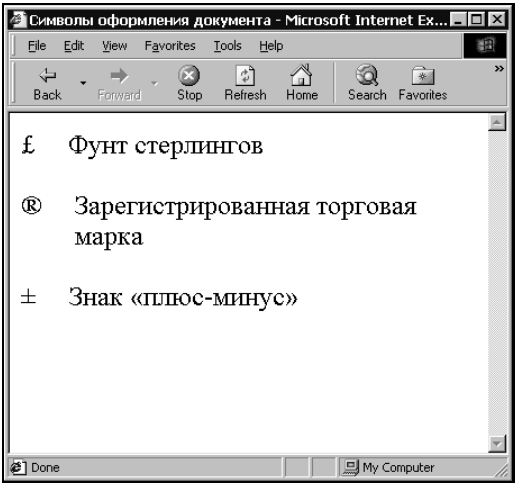


Рис. 2.16. Символы оформления документа

Как видно, первые два символа выводятся браузером с помощью специальной кодовой конструкции: знак амперсанда (&) плюс соответствующее наименование, а последний — путем указания числового кода (а также знак диеза — # — и амперсанда).

Данный пример не означает, что значок английской национальной валюты и зарегистрированной марки можно получить только посредством кодовой

конструкции, а знак "плюс-минус" — числового обозначения. Для каждого специального символа — будь это символ оформления документа, буква иностранного алфавита или символ отображения элементов HTML — существуют два способа записи: кодовый и числовой. К примеру, конструкции `§` и `§` совершенно одинаково будут выводиться в окне браузера в виде значка параграфа (§).

Зная конструкции символов оформления, можно сильно облегчить процесс создания HTML-документов. Например, если вы создаете HTML-код в Блокноте (NotePad, входящий в состав ОС Windows), вы обязательно должны использовать угловые кавычки («»), вам не нужно заходить в текстовый редактор типа Microsoft Word, вводить желаемый тип кавычек (очень часто Word по умолчанию отображает верхние кавычки), копировать их и, снова открыв окно Блокнота, вставлять в документ — просто укажите кодовую или числовую конструкцию для требуемого типа кавычек, и браузер отобразит символ так, как следует.

Если вы хотите разместить в HTML-документе некоторые математические данные, то уравнение, содержащее в качестве знака умножения букву "x", с визуальной точки зрения, сильно уступает уравнению, где применяется специальная конструкция знака "умножить" (`×`).

Говоря о математических уравнениях, следует сказать, что описываемые конструкции специальных символов не используются в отображении знака степени в математике и числа атомов в химическом соединении (соответственно, в химии). Для этого (а также для иных целей) предназначены теги-контейнеры `<SUP></SUP>` (верхний индекс) и `<SUB></SUB>` (нижний индекс).

Буквы иностранных алфавитов

Когда создается HTML-документ на русском или английском языке, никаких затруднений с отображением букв в браузере возникнуть не может. Все они есть на раскладке клавиатуры компьютера. Представим себе ситуацию, в которой нам необходимо разработать электронный документ, к примеру, на португальском языке, алфавит которого содержит буквы, начертание которых характерно только лишь для данного языка. Как можно выйти из такого положения?

Существует несколько вариантов. Первый заключается в том, чтобы добавить в раскладку клавиатуры новый язык, включить в используемом текстовом редакторе поддержку португальского и, выбрав на Панели задач нужный значок, спокойно набирать текст. Спокойно ли? Мы забыли о конечной цели ввода иностранного текста — быть интерпретированным (причем, корректно) браузером.

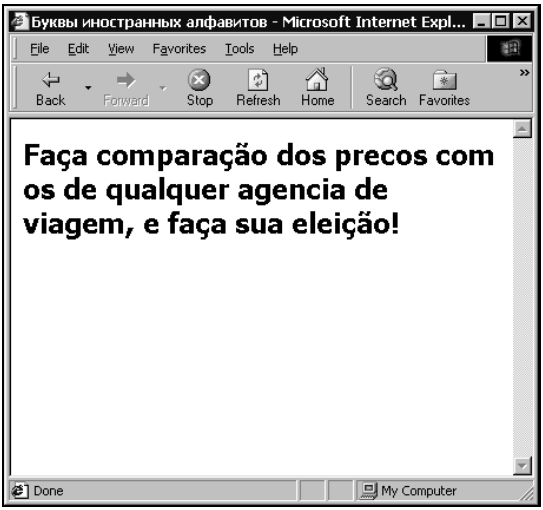


Рис. 2.17. Буквы иностранных алфавитов

Если же, набрав текст на португальском в текстовом редакторе, скопировать его и перенести фрагмент в Блокнот, то текст отобразится без специальных символов, свойственных данному языку. На рис. 2.17 показаны некоторые особенности португальского языка: часть букв "с" и "а" пишутся с использованием специальных элементов вверху и внизу буквы (они называются "седиль" и "тильда", соответственно), чего никак нельзя достигнуть путем переноса текста из текстового редактора в программу создания HTML-документов, подобную Блокноту.

Второй вариант — прибегнуть к услугам визуальных Web-редакторов (FrontPage, Word 2000 и др.), которые включают *мультиязыковую поддержку*. Однако последующая программная конвертация "чернового" текста в формат HTML может привести, во-первых, к некорректному отображению текста на Web-странице некоторыми моделями и версиями браузеров, а во-вторых, к генерации совершенно ненужного и бесполезного кода, только увеличивающего конечный размер файла.

Третий вариант подразумевает использование специальных кодовых или числовых конструкций, предназначенных для корректной визуализации нестандартных букв иностранного алфавита. Например, чтобы браузер правильно отобразил последнее слово в тексте на португальском языке, показанном на рис. 2.17, в код документа нужно вписать следующую конструкцию:

```
elei&ccedil;&atilde;o
```

В результате — никаких проблем с *кросс-браузерной совместимостью* и "утяжеленным" размером HTML-файла. К тому же жители Португалии или

люди, знающие португальский язык, не встретят в таком документе непонятных слов и выражений.

В завершение разговора о специальных символах, следует упомянуть один важный момент. Прежде всего, это касается случаев использования букв иностранных алфавитов: в метаопределениях таких HTML-документов необходимо указывать либо "родную", национальную кодировку, либо кодировку ISO8859, иначе некоторые браузеры (например, Netscape) вместо желанных специальных символов могут отображать смесь непонятных знаков. Internet Explorer, в отличие от своего конкурента на рынке браузеров, более неприхотлив и корректно отображает все специальные символы даже в русских кодировках (полный перечень специальных символов, а также их кодовые и числовые конструкции *см. в приложении 2*).

Гипертекстовые ссылки

Читая книги, мы часто видим, как автор, раскрывая ту или иную тему, ссылается на другой фрагмент книги или на другой книжный или печатный источник. Вот и мы, завершив рассказ о специальных символах HTML, упомянули *приложение 2*, в котором можно найти некую дополнительную информацию.

В Интернете размещены миллионы электронных документов, часто похожих по тематике и ориентированных на одну и ту же пользовательскую аудиторию. Переход на другие документы казался бы делом трудным и утомительным, если бы не возможность связывания одного HTML-документа с другим с помощью *гипертекстовых ссылок*.

По сути, любая гипертекстовая ссылка — это указатель адреса в глобальной сети, по которому можно моментально перейти из окна браузера. Успешный переход по ссылке возможен в двух случаях: если ресурс, на который ссылается документ, существует и если структура гиперссылки верна с точки зрения HTML.

Первый фактор является объективным, не зависящим от нас, т. к. разработчик электронного документа, однажды поставив в нем ссылку на внешний ресурс, может не узнать о том, что этот ресурс прекратил свое существование, переместился на другой адрес, временно закрыт и пр.

Второй фактор относится к разряду субъективных, ибо только от создателя HTML-документа может зависеть, сумеет ли посетитель перейти по ссылке или нет, если она составлена неверно. Чтобы последнего не произошло, рассмотрим структуру и правила описания гипертекстовых ссылок (гиперссылок).

Структура гиперссылок

Любая гиперссылка состоит из двух важных частей: указателя ссылки ("якоря" — от англ. anchor) и адреса ресурса, на который необходимо осуществить переход. Внешне отличить гиперссылку от обычного текста очень просто: при наведении курсора мыши на ссылку указатель принимает вид руки с указательным пальцем, как бы показывающим, что этот текст содержит гиперссылку (рис. 2.18). Сама ссылка подчеркивается (если указателем является текст).

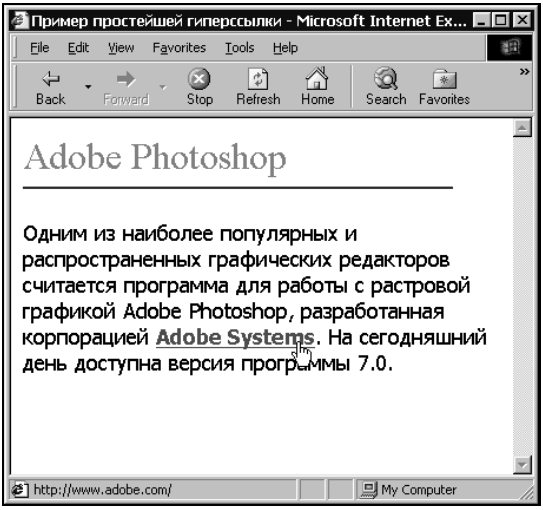


Рис. 2.18. Пример простейшей гиперссылки

В качестве указателя ссылки может выступать текст (отдельное слово, фразы и даже целые страницы текста) и графические изображения. В ряде случаев возможно объединение графики и текста в рамках единого указателя ссылки.

Указатель ссылки описывается тегом `<A>`, а адрес перехода реализуется с помощью параметра `HREF` (листинг 2.13), значением которого является путь к тому или иному интернет-ресурсу (закрывающий тег `</A>` является обязательным).

Листинг 2.13. Пример простейшей гиперссылки

```
<HTML>
  <HEAD>
    <TITLE>Пример простейшей гиперссылки</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#FF0000" ALINK="#FF0000"
VLINK="blue">
```

```
<FONT COLOR="gray" SIZE="6">Adobe Photoshop</FONT><HR COLOR="#003366"
WIDTH="90%" ALIGN="left">
<P><FONT FACE="Tahoma", "Verdana" SIZE="4">
Одним из наиболее популярных и распространенных графических редакторов
считается программа для работы с растровой графикой Adobe Photoshop,
разработанная корпорацией <A HREF="http://www.adobe.com/"><B>Adobe
Systems</B></A>. На сегодняшний день доступна версия программы 7.0.
</FONT></P>
</BODY>
</HTML>
```

Когда указателем ссылки является *графическое изображение*, внутри тега-контейнера `<A>` указывается путь к рисунку, выполняющему роль гипертекстовой ссылки (листинг 2.14).

Листинг 2.14. Пример гиперссылки с использованием графики в качестве указателя

```
<HTML>
  <HEAD>
    <TITLE>Пример гиперссылки с использованием графики в качестве
указателя</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#FF0000" ALINK="#FF0000"
VLINK="blue">
<P ALIGN="center">
<A HREF="http://www.adobe.com/"><IMG
SRC="http://www.adobe.com/images/adobe.gif" WIDTH="24" HEIGHT="31"
ALT="Adobe Systems" BORDER="0"></A>
</P>
</BODY>
</HTML>
```

Следует обратить внимание на то, что значение параметра границы рисунка `BORDER` равно нулю: в противном случае браузер, интерпретируя изображение как указатель ссылки, поставит вокруг рисунка рамку (более подробно о правилах описания графических изображений в HTML пойдет речь в *главе 5*).

Если необходимо объединить в структуре одной ссылки и текст, и графику, HTML-код принимает следующий вид (листинг 2.15, рис. 2.19).

Листинг 2.15. Пример гиперссылки с использованием в качестве указателя и текста, и графики

```
<HTML>
  <HEAD>
```

```
    <TITLE>Пример гиперссылки с использованием в качестве указателя
     и текста, и графики</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#FF0000" ALINK="#FF0000"
VLINK="blue">
<FONT COLOR="gray" SIZE="6">Adobe Photoshop</FONT><HR COLOR="#003366"
WIDTH="90%" ALIGN="left">
<P><FONT FACE="Tahoma", "Verdana" SIZE="4">
Одним из наиболее популярных и распространенных графических редакторов
считается программа для работы с растровой графикой Adobe Photoshop,
разработанная корпорацией <A HREF="http://www.adobe.com/"><IMG
SRC="http://www.adobe.com/images/adobe.gif" WIDTH="24" HEIGHT="31"
ALT="Adobe Systems" BORDER="0"><B>Adobe Systems</B></A>.
На сегодняшний день доступна версия программы 7.0.
</FONT></P>
</BODY>
</HTML>
```

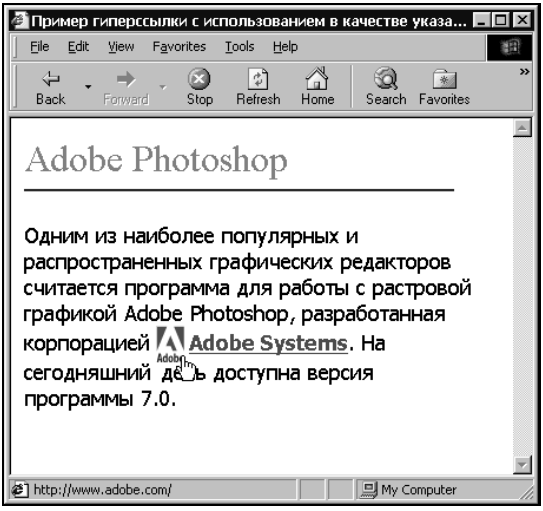


Рис. 2.19. Пример гиперссылки с использованием в качестве указателя и текста, и графики

Помимо HREF, тег <A> может содержать еще и другие параметры, например, STYLE, CLASS, NAME, TITLE и TARGET. Первые два параметра используются для создания стилевых шаблонов гиперссылки *(см. главу 9)*, третий предназначен для указания имени внутренней ссылки *(см. разд. "Правила описания гиперссылок" данной главы)*.

Параметр TITLE используется для создания всплывающей подсказки при наведении курсора мыши на гиперссылку.

Подсказка может выступать в качестве описания или комментария к ссылке, ведущей на электронный документ:

```
<A HREF="http://www.site.ru/" TITLE="Описание документа">
Ссылка на документ</A>
```

Необходимо добавить, что параметр `TITLE` тега `<A>` поддерживается только браузером Internet Explorer.

Что касается параметра `TARGET`, то он определяет назначение перехода по ссылке. Например, нажав на гиперссылку, документ может открыться в текущем или в отдельном (новом) окне. Это в ряде случаев является принципиальным моментом: к примеру, если вы не хотите уводить посетителя со своей Web-страницы, то стоит делать ссылки со следующей структурой:

```
<A HREF="http://www.site.ru/page.html" TARGET="_blank">
Ссылка на документ</A>
```

Наличие знака нижнего подчеркивания перед значением `blank` не является обязательным атрибутом для большинства популярных браузеров, однако Консорциум W3C настоятельно рекомендует указывать полную конструкцию параметра `TARGET`.

В завершение следует добавить, что параметр `TARGET` и некоторые другие его значения (помимо `blank`) используются в описании так называемой фреймовой структуры, рассказу о которой посвящена *глава 6*.

Правила описания гиперссылок

Все гиперссылки можно разделить на два типа: внешние и внутренние. Внешние ссылки ведут на другие ресурсы глобальной сети или другие документы одного Web-сайта, а внутренние — позволяют посетителю путешествовать в пределах одного HTML-документа.

Внешние ссылки

Структура *внешней ссылки* состоит из указателя ("якоря") и адреса.

Путь адреса ссылки

Существует два способа записи адреса перехода по внешней ссылке.

- абсолютный — в значении параметра `HREF` указывается полный путь к ресурсу, ссылка на который ставится в документе, например:

  ```
  <A HREF="http://www.site.ru/docs/page1.html">Ссылка на страницу 1</A>
  ```

  Независимо от того, с какого документа пользователь нажмет на такую ссылку, он перейдет по адресу **http://www.site.ru/docs/page1.html**;

- относительный — в значении параметра `HREF` указывается конечный документ, относительно которого размещена страница, содержащая ссылку, например:

  ```
  <A HREF="page2.html">Ссылка на страницу 2 со страницы 1</A>
  ```

  Такой формат записи внешней ссылки подразумевает расположение файла `page2.html` (на который существует гиперссылка) в том же каталоге, что и файл `page1.html` (с которого будет осуществляться переход). В нашем случае оба файла расположены в каталоге `docs`, находящимся на Web-сайте **www.site.ru**.

Теперь попытаемся усложнить ситуацию и представим, что файл `page2.html` размещен в корне сайта **www.site.ru**, а файл `page1.html` по-прежнему находится в каталоге `docs`. В этом случае запись внешней ссылки с использованием относительного пути к файлу примет следующий вид:

```
<A HREF="../page2.html">Ссылка на страницу 2 со страницы 1</A>
```

Таким образом, с помощью относительного пути перехода по ссылке мы указали браузеру подняться на два уровня вверх (минуя уровень файлов, размещенных в каталоге `docs`, и уровень размещения самой папки `docs`).

Следует помнить, что внешняя ссылка, путь к которой прописан относительно, может вести только на документы текущего сайта. Перенести посетителя на посторонние интернет-ресурсы, например, с сайта **www.site1.ru** на **www.site2.ru**, такая ссылка не может.

Указание протокола перехода по ссылке

Понятие Интернета как глобальной информационной сети подразумевает не только World Wide Web (WWW, Всемирная паутина), в состав которого входит большинство интернет-ресурсов. Интернет — это более емкая инфраструктура, включающая в себя различные информационные сервисы, работа которых реализуется с помощью так называемых протоколов — наборов технологических правил взаимодействия документов друг с другом.

Например, World Wide Web работает на основе протокола HTTP (HyperText Transfer Protocol, протокол передачи гипертекста). Помимо этого, существуют такие технологии, как FTP (File Transfer Protocol, протокол передачи файлов), E-mail (служба электронной почты), News (группы новостей), Gopher и др.

Вполне возможна ситуация, когда разработчику HTML-документа понадобится поставить ссылку на другие, отличные от HTTP сервисы. В этом случае структура гиперссылки остается прежней: указатель (текстовый и/или графический) и адресная часть. Однако последняя может меняться в зависимости от выбранной технологии передачи данных.

Например, ссылка на адрес электронной почты, которая сегодня получила гораздо большее распространение, чем интернет-ресурсы, может выглядеть так:

```
<A HREF="mailto:mail@site.ru">Ссылка на адрес электронной почты</A>
```

Если пользователь нажмет на такую ссылку, на его компьютере запустится установленная по умолчанию программа чтения и отправления электронной почты (например, Microsoft Outlook, The Bat!, Netscape Messenger, Eudora и др.). В этом случае можно несколько усложнить структуру ссылки, добавив в нее готовое поле заголовка электронного письма:

```
<A HREF="mailto:mail@site.ru?Subject=Здравствуйте!">
Ссылка на адрес электронной почты</A>
```

Такая гиперссылка не только запустит почтовую программу и откроет окно нового письма, но и добавит в поле заголовка электронного сообщения **Subject** обращение "Здравствуйте!".

Правила составления гиперссылок на различные информационные сервисы Интернета приведены в табл. 2.5.

Таблица 2.5. *Правила составления гиперссылок на различные информационные интернет-сервисы*

Название информационного сервиса	Пример гиперссылки
World Wide Web	`http://www.site.ru/`
E-mail	`mailto:mail@site.ru`
FTP	`ftp://ftp.site.ru/`
News	`news:news.users.information`
Gopher	`gopher://gopher.site.ru/`

Следует отметить, что отсутствие используемого протокола передачи данных или его неверное наименование вызовет ошибку браузера при попытке перехода по ссылке.

Например, в случае конструкции

```
<A HREF="www.site.ru">Ссылка на сайт</A>
```

браузер будет пытаться перейти на некий документ **www.site.ru** с отсутствующим расширением. Если же вы ненароком сделаете пробел в названии протокола при составлении гиперссылки на адрес электронной почты, браузер расценит эту команду как открытие документа в текущем каталоге под

названием "**mail to:inbox@mysupermail.ru**" и, скорее всего, опять ничего не найдет.

```
<A HREF="mail to:inbox@mysupermail.ru">Ссылка на e-mail</A>
```

Типы файлов для перехода по ссылке

Составляя гиперссылку, мы подразумеваем ссылку на HTML-документ (речь идет о службе WWW). Тем не менее, практика разработки электронных документов показывает актуальную необходимость в ссылках на файлы другого типа.

Например, на корпоративном Web-сайте можно создать ссылку на пример коммерческого договора, составленного в формате Microsoft Word (расширение doc), на сервере бесплатного программного обеспечения — на архив программы (расширения zip, rar и пр.), на ресурсе, содержащем коллекцию музыкальных файлов, — на аудиоролики в формате MP3 или RealPlayer.

Возникают два вопроса: какова структура ссылок для других типов файлов и каким образом такие документы будут открываться в браузере пользователя?

Что касается структуры организации гиперссылок, то она остается неизменной:

```
<A HREF="http://www.site.ru/docs/dogovor.doc">Коммерческий договор</A>
<A HREF="http://www.site.ru/download/program.zip">Архив программы</A>
<A HREF="http://www.site.ru/music/song.mp3">Музыкальный ролик</A>
```

В отношении загрузки запрашиваемого файла в формате, отличном от HTML, возможны три ситуации:

- браузер поддерживает запрашиваемый формат и откроет файл в своем рабочем окне (например, рисунки формата GIF, JPEG и пр.);
- браузер не поддерживает запрашиваемый формат и переадресовывает запрос операционной системе, которая запускает программу, предназначенную для просмотра файлов данного типа (например, видеоролики в формате MPEG4 вызовут программу Microsoft Windows Media);
- браузер не поддерживает запрашиваемый формат, который к тому же не соотносится ни с одной программой в настройках операционной системы. В этом случае браузер откроет стандартное диалоговое окно с возможными вариантами действий над открываемым файлом неизвестного формата: **Запустить** или **Сохранить на диске**. Аналогично браузер поступит и с некоторыми известными операционной системе форматами, к примеру, с ZIP-архивами (рис. 2.20).

Рис. 2.20. Диалоговое окно браузера с вариантами действий над открываемым файлом

Внутренние ссылки

Если HTML-документ слишком большого размера и нет возможности его разбить на несколько отдельных файлов, можно прибегнуть к помощи внутренних гиперссылок, перемещающих пользователя в пределах одной Web-страницы. Такой подход полезен, например, при составлении технической документации или юридических договоров, где быстрый переход от одного пункта документа к другому играет первостепенную роль.

Структура внутренней гиперссылки включает две части: сама ссылка и ее именной идентификатор (диез плюс имя элемента, аналогичное значению параметра `HREF` самой гиперссылки), позволяющий переместиться в нужное место электронного документа (листинг 2.16).

Листинг 2.16. Пример составления внутренних гиперссылок

```
<HTML>
  <HEAD>
    <TITLE>Пример составления внутренних гиперссылок</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#FF0000" ALINK="#FF0000"
VLINK="blue">
<FONT COLOR="gray" SIZE="6">Теги HTML</FONT><HR COLOR="#003366"
WIDTH="90%" ALIGN="left">
<P><A HREF="#font">FONT</A></P>
<P><A HREF="#span">SPAN</A></P>
<P><A HREF="#table">TABLE</A></P>
```

```
<HR ALIGN="center" WIDTH="95%" NOSHADE>
<P ALIGN="justify">
<A NAME="font"></A>
FONT - один из основных тегов физического форматирования текста,
отображающий свойства шрифтов.
</P>
</BODY>
</HTML>
```

Для обозначения места, в которое нужно перенести пользователя документа, используется пустая конструкция `<A NAME="font"></A>`. Дело в том, что при создании внутренних ссылок визуально выделять расположение именного идентификатора не имеет смысла: основная цель — переход в соответствующий раздел текущего документа.

Для лучшего усвоения материала основные параметры тега `<A>` приведены в табл. 2.6.

Итак, в этой главе мы рассмотрели основные параметры и свойства такого важнейшего компонента HTML-документа, как текст. Некоторые элементы собраны воедино в итоговом примере этой главы (рис. 2.21, листинг 2.17).

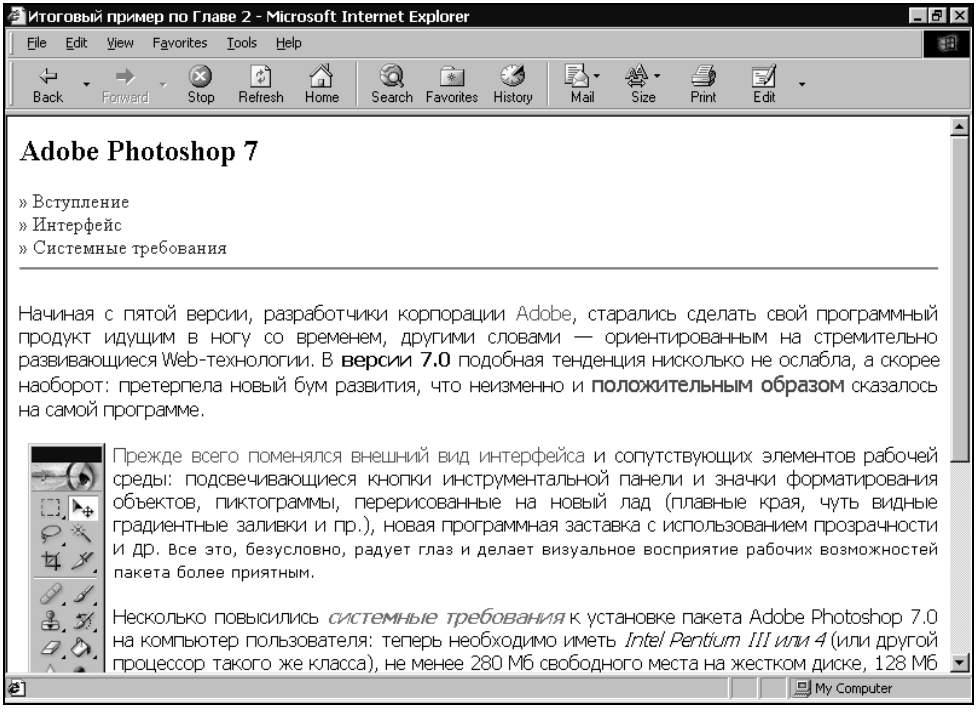


Рис. 2.21. Итоговый пример по главе 2

Таблица 2.6. *Параметры тега* <A>

Параметр	Функция
HREF	Указание адреса перехода по гиперссылке
TARGET	Определение места назначения перехода (текущее/новое окно)
STYLE	Задание стилевого шаблона
CLASS	Указание класса стилевого шаблона
NAME	Именной идентификатор внутренней гиперссылки

Листинг 2.17. Итоговый пример по главе 2

```
<HTML>
  <HEAD>
    <TITLE>Итоговый пример по Главе 2</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#FF0000" ALINK="#FF0000"
VLINK="blue">
<!-- Заголовок документа о программе Photoshop 7 -->
<H2>Adobe Photoshop 7</H2>
<DIV><A HREF="#intro">&raquo; Вступление</A></DIV>
<DIV><A HREF="#interface">&raquo; Интерфейс</A></DIV>
<DIV><A HREF="#system">&raquo; Системные требования</A></DIV>
<HR ALIGN="center" COLOR="gray">
<P ALIGN="justify"><FONT FACE="Tahoma" SIZE="3"><A NAME="intro"></A>
Начиная с пятой версии, разработчики корпорации <A
HREF="http://www.adobe.com/" TARGET="_blank">Adobe</A>, старались сделать
свой программный продукт идущим в ногу со временем, другими словами
&mdash; ориентированным на стремительно развивающиеся <NOBR>Web-
технологии</NOBR>. В <STRONG>версии 7.0</STRONG> подобная тенденция
нисколько не ослабла, а скорее наоборот: претерпела новый бум развития,
что неизменно и <FONT COLOR="green"><BIG>положительным
образом</BIG></FONT> сказалось на самой программе.
<BR><BR><A NAME="interface"></A>
<A HREF="http://www.adobe.com/"><IMG SRC="photoshop.gif" BORDER="0"
WIDTH="64" HEIGHT="441" ALIGN="left">Прежде всего поменялся внешний вид
интерфейса</A> и сопутствующих элементов рабочей среды: подсвечивающиеся
кнопки инструментальной панели и значки форматирования объектов,
пиктограммы, перерисованные на новый лад (плавные края, чуть видные
градиентные заливки и пр.), новая программная заставка с использованием
прозрачности и др. <FONT FACE="Verdana" SIZE="2">Все это, безусловно,
радует глаз и делает визуальное восприятие рабочих возможностей пакета
более приятным.</FONT>
<BR><BR><A NAME="system"></A>
```

```
Несколько повысились <EM><B><FONT COLOR="#098EFA">системные
требования</FONT></B></EM> к установке пакета Adobe Photoshop 7.0
на компьютер пользователя: теперь необходимо иметь <CITE>Intel
Pentium III или 4</CITE> (или другой процессор такого же класса), не
менее 280 Мб свободного места на жестком диске, 128 Мб оперативной памяти
(желательно 256 Мб), разрешение экрана не менее 800x600 точек с 16-битным
цветом.
</FONT></P>
</BODY>
</HTML>
```

ГЛАВА 3

Использование HTML-списков

Зачем нужны HTML-списки

Давайте задумаемся: как часто мы сталкиваемся в нашей повседневной жизни с перечнями информации? Грядет череда праздников, и надо совершить множество всевозможных покупок для праздничного стола — мы составляем список всех необходимых нам продуктов. Планируя свой день, мы указываем в своем ежедневнике перечень дел, мероприятий и встреч. Очутившись в кафе, мы знакомимся с предложенным нам меню, состоящим также из списка наименований тех блюд и напитков, которые мы можем заказать.

Таким образом, различные списки играют далеко не последнюю роль в нашей повседневной жизни. Возникает вопрос: а почему мы так часто прибегаем к этому средству представления информации?

Во-первых, информация в виде списка позволяет разбить большие массивы данных на отдельные, четкие фрагменты информации, которые человеку гораздо удобнее воспринимать, нежели весь поток информации целиком. Например:

Визуальные эффекты графического редактора Adobe Photoshop 7:

- Drop Shadow — на объект накладывается тень, размер, цвет, расположение и насыщенность которой можно изменять с помощью настроек эффекта;
- Bevel and Emboss — эффект добавляет объекту объемность посредством наложения параметров выпуклости, вдавленности и затенения;
- Color Overlay — заполнение объекта выбранным цветом с возможностью регулирования уровня насыщенности.

Во-вторых, списки позволяют создавать понятную и логичную внутреннюю структуру информационных данных, ориентироваться в которой — дело простое и удобное.

Например:

В повестку дня входит обсуждение следующих вопросов:

1. Практическая необходимость покупки служебного автотранспорта.
2. Выделение дополнительных финансовых средств на составление рекламного бюджета.
3. Избрание состава Совета директоров компании.

В-третьих, использование списков оптимально для отображения определенных пошаговых и прочих последовательных процессов. Например:

Для начала работы с компьютером вам необходимо:

- поехать в близлежащий компьютерный магазин;
- приобрести персональный компьютер;
- привезти компьютер домой и соединить все его составляющие компоненты между собой;
- включить питание компьютера и нажать кнопку **Power**.

При работе с текстом в различных программах типа Microsoft Word создание списков не представляется сложным делом: вам всего лишь понадобится указать область данных, которую текстовому редактору необходимо превратить в список.

В языке разметки HTML за организацию списков отвечает целый ряд тегов, конструкции которых должны соответствовать определенным правилам структуризации информационных данных с помощью HTML.

Спецификация HTML предусматривает три основных типа списков:

- нумерованные списки;
- маркированные списки;
- списки определений.

Рассмотрим каждый из названных типов более подробно.

Нумерованные списки

Нумерованный список часто называется упорядоченным списком (Ordered List) и представляет собой определенную последовательность элементов. То есть, каждый последующий пункт такого списка будет на единицу больше предыдущего.

Для создания нумерованного списка используется тег-контейнер `<OL>`, внутри которого располагаются пункты перечня информационных данных, за указа-

ние которых отвечает специальный тег <LI> (List Item — пункт списка или элемент списка) (листинг 3.1).

Листинг 3.1. Пример нумерованного списка

```
<HTML>
  <HEAD>
    <TITLE>Пример нумерованного списка</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<H3>Пример нумерованного списка</H3>
<HR ALIGN="center" NOSHADE WIDTH="98%">
<OL>
<FONT FACE="Tahoma"><B>Визуальные эффекты Adobe Photoshop 7:</B></FONT>
       <LI>Drop Shadow
       <LI>Bevel and Emboss
       <LI>Color Overlay
</OL>
</BODY>
</HTML>
```

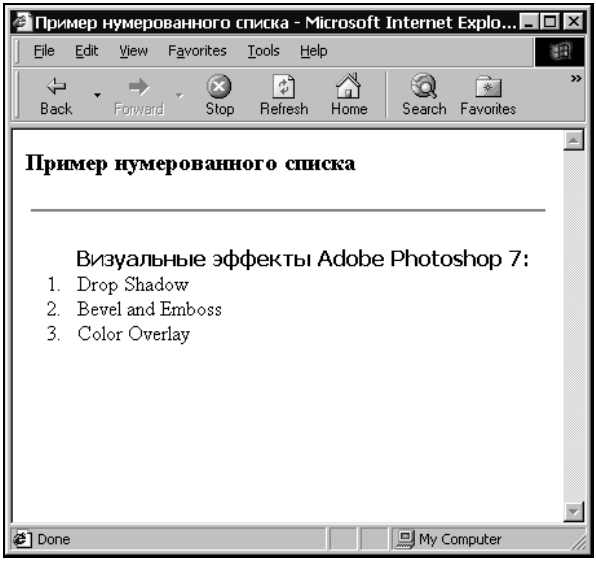


Рис. 3.1. Пример нумерованного списка

Как видно из примера (рис. 3.1), после каждого пункта списка осуществляется перенос строки, причем без указания каких-либо структурных тегов. Так-

же следует заметить, что закрывающий тег `</LI>` отсутствует, хотя и существует в спецификации HTML (Консорциум W3C признал этот закрывающий тег необязательным для указания).

Заголовок списка ("Визуальные эффекты Adobe Photoshop 7") позволяет применять любое форматирование текста и размещается между тегом `<OL>` и первым пунктом списка `<LI>` (рис. 3.2). В этом случае заголовок будет отделен от начала самого списка одним переносом строки и отступом от левого края документа (как и все пункты списка). При указании заголовка перед тегом `<OL>` браузер, интерпретируя HTML-код, создаст двойной перенос и разместит текст заголовка левее пунктов списка.

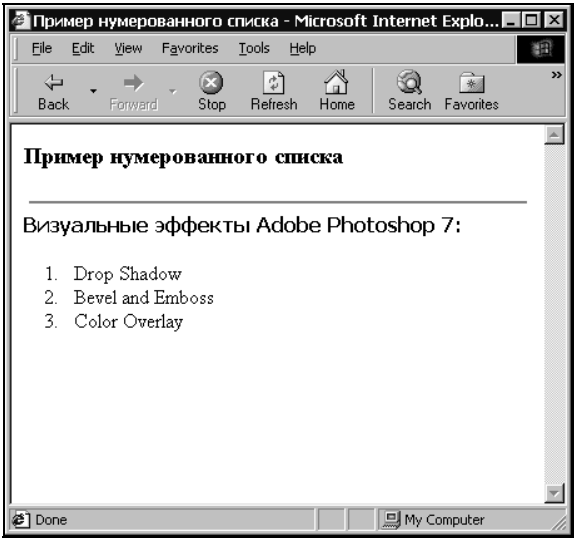


Рис. 3.2. Пример нумерованного списка с заголовком, размещенным перед тегом `<OL>`

Иногда может возникнуть ситуация, когда список содержит определенный разрыв в последовательности представления пунктов, например, после пункта 3 должен идти пункт 5. С этой целью используется специальный параметр тега `<LI>` — `VALUE` (листинг 3.2, рис. 3.3).

Листинг 3.2. Пример нумерованного списка с разрывом последовательности

```
<HTML>
  <HEAD>
    <TITLE>Пример нумерованного списка с разрывом
последовательности</TITLE>
  </HEAD>
```

```
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<H3>Пример нумерованного списка</H3>
<HR ALIGN="center" NOSHADE WIDTH="98%">
<FONT FACE="Tahoma"><B>Визуальные эффекты Adobe Photoshop 7:</B></FONT>
<OL>
       <LI>Drop Shadow
       <LI>Bevel and Emboss
       <LI>Color Overlay
       <BR>...
       <LI VALUE="5">Stroke
</OL>
</BODY>
</HTML>
```

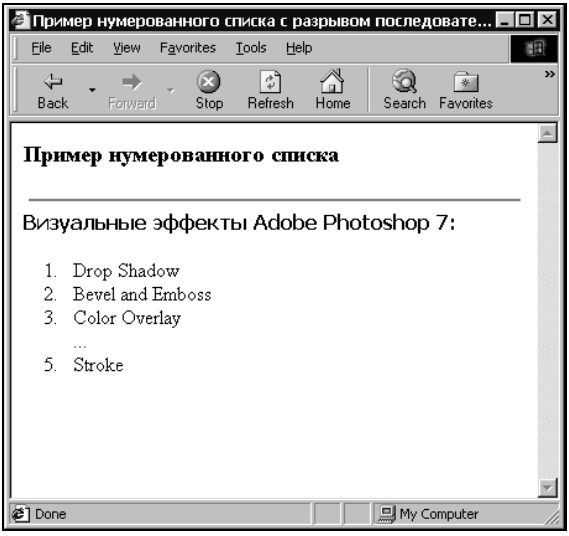


Рис. 3.3. Пример нумерованного списка с разрывом последовательности

Параметры тега <*OL*>

Тег составления нумерованных списков `<OL>` может включать параметры `TYPE`, `START` и `COMPACT`.

Параметр `TYPE` используется для указания типа нумерованного списка. Помимо обычных арабских цифр могут применяться большие и маленькие римские цифры, а также строчные и прописные латинские буквы (листинг 3.3, рис. 3.4).

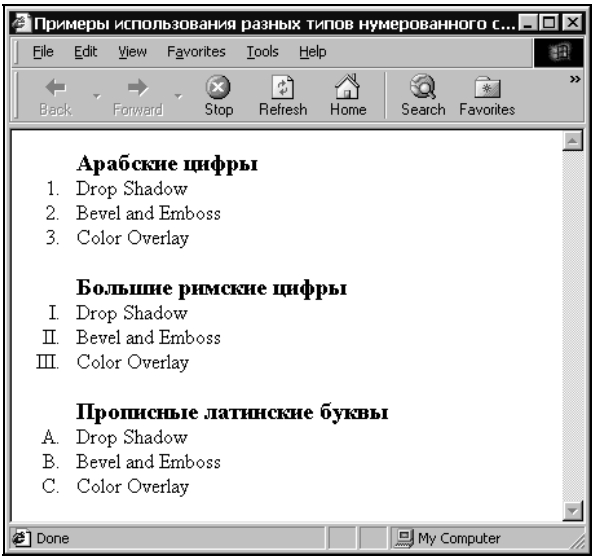


Рис. 3.4. Примеры использования разных типов нумерованного списка

Листинг 3.3. Примеры использования разных типов нумерованного списка

```
<HTML>
  <HEAD>
    <TITLE>Примеры использования разных типов нумерованного
списка</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<OL TYPE="1">
<FONT SIZE="4"><B>Арабские цифры</B></FONT>
       <LI>Drop Shadow
       <LI>Bevel and Emboss
       <LI>Color Overlay
</OL>
<OL TYPE="I">
<FONT SIZE="4"><B>Большие римские цифры</B></FONT>
       <LI>Drop Shadow
       <LI>Bevel and Emboss
       <LI>Color Overlay
</OL>
<OL TYPE="A">
<FONT SIZE="4"><B>Прописные латинские буквы</B></FONT>
       <LI>Drop Shadow
```

```
        <LI>Bevel and Emboss
        <LI>Color Overlay
</OL>
</BODY>
</HTML>
```

Все значения параметра TYPE приведены в табл. 3.1.

Таблица 3.1. *Значения параметра TYPE тега <OL>*

Значение	Описание
TYPE="1"	Нумерация арабскими цифрами
TYPE="I"	Нумерация большими римскими цифрами
TYPE="i"	Нумерация маленькими римскими цифрами
TYPE="A"	Нумерация прописными латинскими буквами
TYPE="a"	Нумерация строчными латинскими буквами

Типом нумерации, используемым по умолчанию, является TYPE="1", поэтому при создании нумерованных списков такого типа необязательно задавать значение параметра TYPE (сам параметр также можно пропустить).

Параметр START позволяет начинать нумерованный список не с первой позиции. При этом значением этого параметра может быть только числовой эквивалент конкретного типа списка. Например, чтобы начать список, созданный на основе нумерации большими латинскими буквами, с буквы F, надо указать следующую HTML-конструкцию:

```
<OL TYPE="A" START="6">
```

В данном случае значением параметра START будет являться цифра 6, т. к. буква F — шестая по счету в латинском алфавите.

Параметр COMPACT изначально был предусмотрен для списков, отображение которых должно было производиться в компактном виде, например, с использованием меньшего размера шрифта, отступа от края страницы или расстояния между пунктами списка. Однако на практике параметр почти не используется, наиболее распространенные браузеры никак не реагируют на его применение, к тому же Консорциум W3C в последней спецификации HTML вообще не рекомендует использовать параметр COMPACT.

В завершение разговора о создании нумерованных списков следует добавить, что возможно смешанное использование различных типов нумерации в пределах одного списка. Например, первый пункт списка можно сделать типом

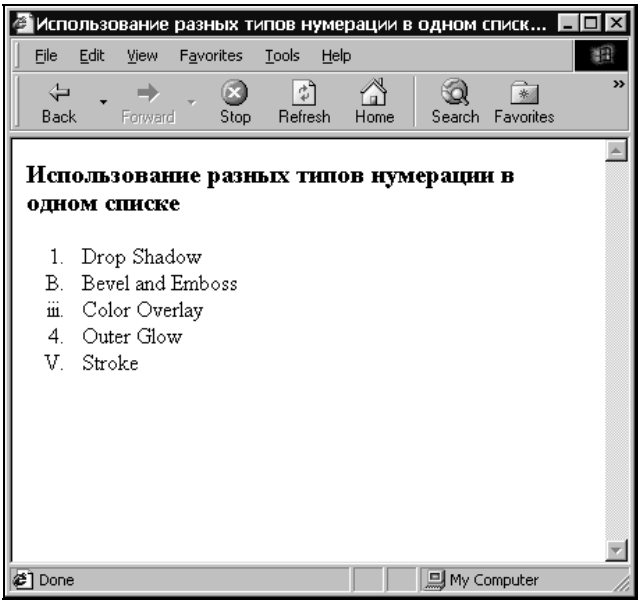


Рис. 3.5. Отображение списка со смешанной нумерацией в Internet Explorer

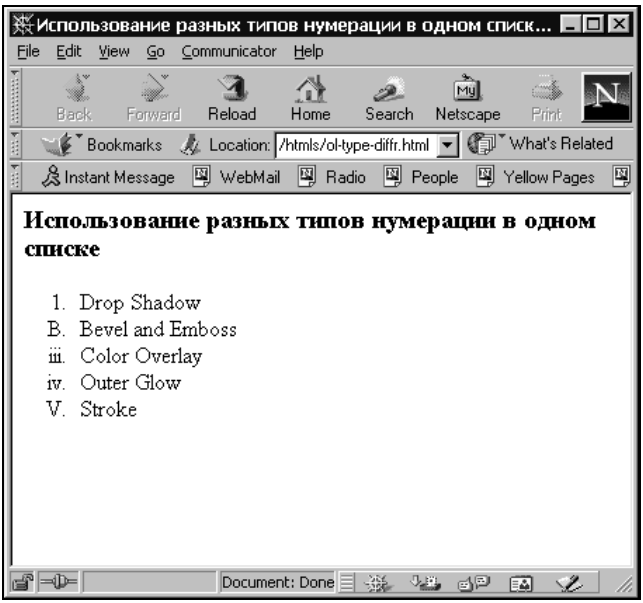


Рис. 3.6. Отображение списка со смешанной нумерацией в Netscape Navigator

арабских цифр, второй — буквенным способом, третий — с помощью римских цифр и т. д.

Однако браузеры по-разному реагируют на такое смешение стилей отображения списков. К примеру, HTML-конструкция:

```
<OL TYPE="1">
       <LI>Drop Shadow
       <LI TYPE="A">Bevel and Emboss
       <LI TYPE="i">Color Overlay
       <LI>Outer Glow
       <LI TYPE="I">Stroke
</OL>
```

будет интерпретирована браузером Internet Explorer так, как показано на рис. 3.5: пункт Outer Glow озаглавлен арабской цифрой 4 (четвертый пункт в списке по счету).

А вот другой браузер — Netscape Navigator — расценит пропуск типа нумерации в пункте Outer Glow как команду отображать данные по предыдущему типу (маленькие римские цифры) (рис. 3.6).

Маркированные списки

Другим распространенным способом структуризации информационных данных является создание *маркированных списков*. Маркированный список представляет собой ненумерованный или неупорядоченный (Unordered List) перечень элементов, для заголовка которых используются специальные маркеры. В качестве маркеров выступают специальные символы, называемые *буллетами* (bullets).

Для создания маркированных списков применяется тег-контейнер `<UL>` (листинг 3.4), внутри которого располагаются элементы самого списка (как и в случае с нумерованными списками, используется тег `<LI>`, не требующий закрывающего тега).

Пример маркированного списка показан на рис. 3.7.

Листинг 3.4. Пример маркированного списка

```
<HTML>
  <HEAD>
    <TITLE>Пример маркированного списка</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
```

```
<H3>Пример маркированного списка</H3>
<HR ALIGN="center" NOSHADE WIDTH="98%">
<FONT FACE="Tahoma"><B>Популярные редакторы векторной графики:</B></FONT>
<UL>
       <LI>CorelDRAW
       <LI>Adobe Illustrator
       <LI>Macromedia FreeHand
</UL>
</BODY>
</HTML>
```

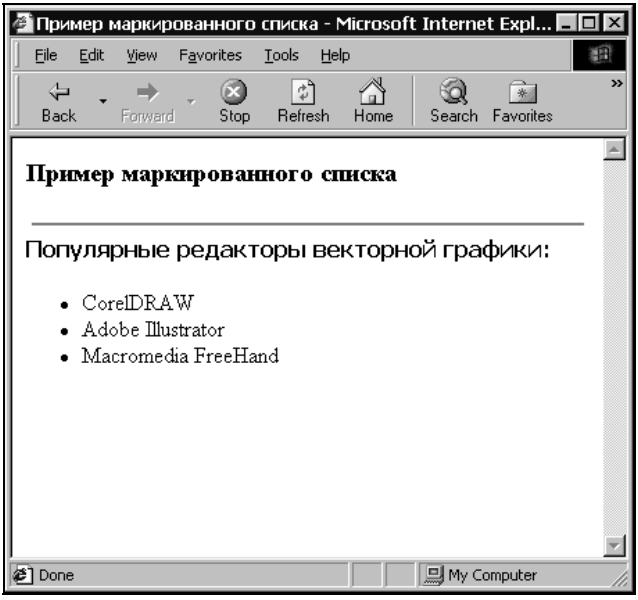


Рис. 3.7. Пример маркированного списка

Параметры тега <*UL*>

Тег маркированного списка `<UL>` может содержать два основных параметра: `TYPE` и `COMPACT`. Параметр `TYPE` отвечает за форму отображаемого браузером маркера. Существуют три возможных значения данного параметра:

- `TYPE=disc` — значение параметра `TYPE`, используемое по умолчанию (указание необязательно), которое отображает маркер в виде закрашенного круга;
- `TYPE=circle` — маркер отображается в виде незакрашенного круга (кольца);
- `TYPE=square` — отображение маркера в виде закрашенного квадрата.

Следует отметить, что размер того или иного маркера может изменяться в зависимости от размера используемого в электронном документе шрифта. Кроме этого, как и в случае с нумерованными списками, при смешении правил маркировки различные браузеры могут отображать маркеры по-разному (листинг 3.5, рис. 3.8)

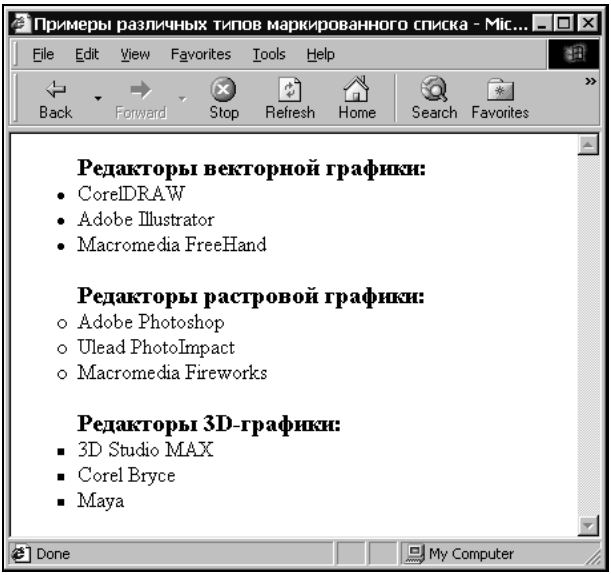


Рис. 3.8. Примеры различных типов маркированного списка

Листинг 3.5. Примеры различных типов маркированного списка

```
<HTML>
  <HEAD>
    <TITLE>Примеры различных типов маркированного списка</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<UL>
<FONT SIZE="4"><B>Редакторы векторной графики:</B></FONT>
       <LI>CorelDRAW
       <LI>Adobe Illustrator
       <LI>Macromedia FreeHand
</UL>
<UL TYPE="circle">
<FONT SIZE="4"><B>Редакторы растровой графики:</B></FONT>
       <LI>Adobe Photoshop
```

```
        <LI>Ulead PhotoImpact
        <LI>Macromedia Fireworks
</UL>
<UL TYPE="square">
<FONT SIZE="4"><B>Редакторы 3D-графики:</B></FONT>
        <LI>3D Studio MAX
        <LI>Corel Bryce
        <LI>Maya
</UL>
</BODY>
</HTML>
```

Параметр тега `<UL> COMPACT` имеет те же характеристики, что и для нумерованных списков.

Графические маркеры

Отличительной особенностью маркированных списков является возможность использования вместо стандартных HTML-маркеров свои собственные графические изображения. Такой подход позволяет создавать более привлекательные списки в электронных документах.

Для указания браузеру, что в качестве маркера будет использовано графическое изображение, внутри тега-контейнера `<UL>` вместо указателя элемента `<LI>` размещается обыкновенная HTML-конструкция описания графических объектов (листинг 3.6, рис. 3.9).

Листинг 3.6. Пример использования графических маркеров

```
<HTML>
  <HEAD>
    <TITLE>Пример использования графических маркеров</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<UL>
<FONT SIZE="4"><B>Редакторы компьютерной графики:</B></FONT>
        <IMG SRC="marker.gif">CorelDRAW
        <BR>
        <IMG SRC="marker.gif">Adobe Illustrator
        <BR>
        <IMG SRC="marker.gif">Macromedia FreeHand
        <BR>
        <IMG SRC="marker.gif">Adobe Photoshop
```

```
        <BR>
        <IMG SRC="marker.gif">Macromedia Fireworks
        <BR>
        <IMG SRC="marker.gif">3D Studio MAX
        <BR>
        <IMG SRC="marker.gif">Corel Bryce
</UL>
</BODY>
</HTML>
```

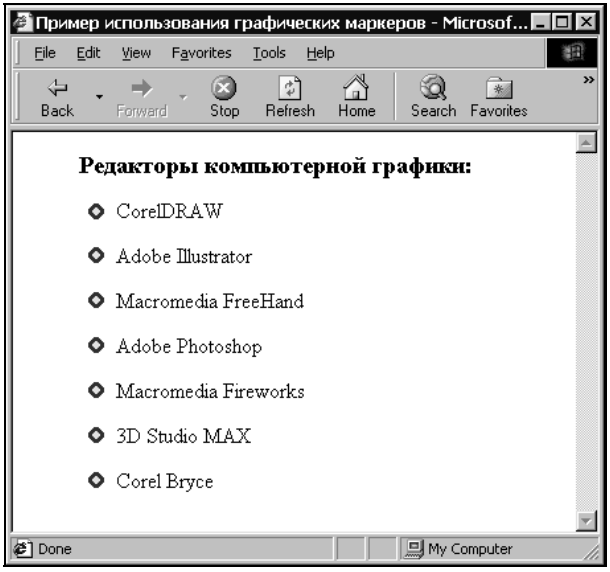


Рис. 3.9. Пример использования графических маркеров

Браузер поймет такую конструкцию как команду задать отступ от левого края документа до начала элементов маркированного списка (где повторяющимся маркером является файл рисунка "`marker.gif`").

В данном случае после каждого пункта маркированного списка стоит тег переноса строки `<BR>`. Если его не указывать, все элементы списка будут выстроены в один ряд (это связано с тем, что создание графических маркеров не относится к каким-то особенным HTML-конструкциям, а реализовано за счет обособления рисунков-маркеров тегом `<UL>`; т. к. тег элемента списка `<LI>` отсутствует, браузер не делает переноса строки).

Вложенные списки

Часто информация имеет сложную разветвленную структуру, которую, с одной стороны, трудно отобразить в виде одного списка, с другой — нельзя разбивать на несколько отдельных списков. В этом случае используются так называемые *вложенные списки* — списки, имеющие иерархическое строение и содержащие в себе другие списки (как нумерованные, так и маркированные) (листинг 3.7).

Листинг 3.7. Пример использования вложенных списков

```
<HTML>
  <HEAD>
    <TITLE>Пример использования вложенных списков</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<UL TYPE="square">
<FONT SIZE="4"><B>Редакторы компьютерной графики:</B></FONT>
       <LI>Векторные редакторы
              <UL TYPE="disc">
              <LI>CorelDRAW
                     <OL>
                     <LI>CorelDRAW
                     <LI>Corel PhotoPaint
                     <LI>Corel R.A.V.E.
                     </OL>
              </UL>
       <LI>Растровые редакторы
              <UL TYPE="disc">
              <LI>Adobe Photoshop
                     <OL>
                     <LI>Adobe Photoshop
                     <LI>Adobe ImageReady
                     </OL>
              </UL>
</UL>
</BODY>
</HTML>
```

В данном примере (рис. 3.10) присутствуют и нумерованные, и маркированные списки. Графические маркеры также могут применяться внутри вложенных списков наравне со списками остальных типов.

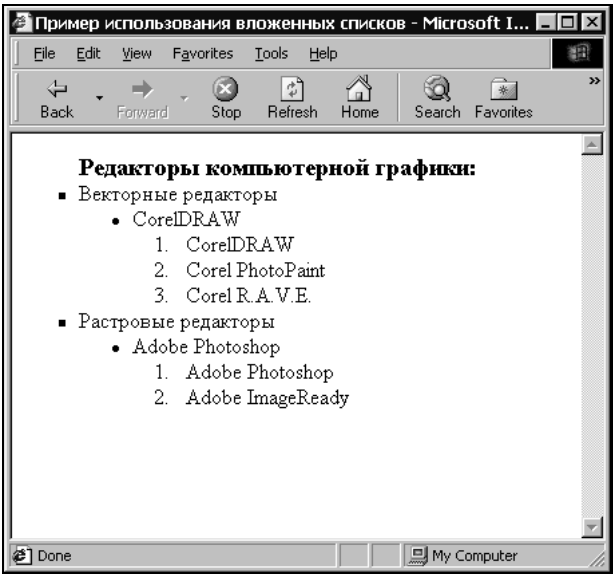


Рис. 3.10. Пример использования вложенных списков

В заключение необходимо добавить, что при составлении сложных вложенных списков особое внимание следует уделять корректному указанию закрывающих тегов `</UL>` и `</OL>`. Для того чтобы не запутаться в разветвленной структуре HTML-кода вложенных списков, каждый последующий уровень вложенности лучше всего отделять от предыдущего небольшим отступом от левого края документа.

Списки определений

Списки определений — это особый тип структуризации информационных данных, идеально подходящий для отображения терминологических и толковых словарей, а также справочников средствами HTML.

В отличие от нумерованных и маркированных списков, которые могут содержать не только по несколько элементов, но и другие списки, список определений состоит всего из двух частей: самого определения (термина) и его содержательной части (описания).

Структура списка определений описывается тегом-контейнером `<DL>` (Definition List), внутри которого указывается тег `<DT>` (Definition Term), выделяющий заголовок определения, и тег `<DD>` (Definition Description), описывающий содержательную часть определения. Для двух последних тегов наличие закрывающих тегов необязательно (листинг 3.8, рис. 3.11).

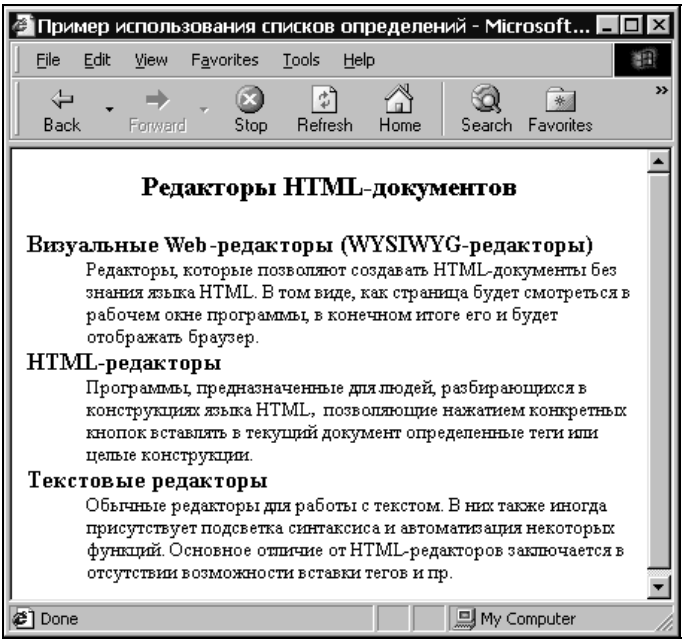


Рис. 3.11. Пример использования списков определений

Листинг 3.8. Пример использования списков определений

```
<HTML>
  <HEAD>
    <TITLE>Пример использования списков определений</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<DL>
<H3 ALIGN="center">Редакторы HTML-документов</H3>
<DT><B>Визуальные Web-редакторы (WYSIWYG-редакторы)</B>
<DD><SMALL>Редакторы, которые позволяют создавать HTML-документы без
знания языка HTML. В том виде, как страница будет смотреться в рабочем
окне программы, в конечном итоге его и будет отображать
браузер.</SMALL><BR>
<DT><B>HTML-редакторы</B>
<DD><SMALL>Программы, предназначенные для людей, разбирающихся в
конструкциях языка HTML, позволяющие нажатием конкретных кнопок вставлять
в текущий документ определенные теги или целые конструкции.</SMALL><BR>
<DT><B>Текстовые редакторы</B>
<DD><SMALL>Обычные редакторы для работы с текстом. В них также иногда
присутствует подсветка синтаксиса и автоматизация некоторых функций.
```

```
Основное отличие от HTML-редакторов заключается в отсутствии возможности
вставки тегов и пр.</SMALL><BR>
</DL>
</BODY>
</HTML>
```

Как видно из листинга 3.8, заголовок определения и содержательная часть списка определений позволяют применять форматирование текста, а также некоторые теги структурного форматирования. Видно, что каждое определение в коде разделено тегом переноса строки
 (рис. 3.11). Если этого не сделать, большинство браузеров создаст между последней и предпоследней строкой содержательной части определения промежуток, больший по размеру, чем между остальными строками описания.

Помимо этого, в структуру списков определений можно включать другие типы списков — нумерованные и маркированные (с графическими маркерами, в том числе).

Тег структуры списка определений <DL> может содержать параметр COMPACT, суть которого была описана в разделах данной главы, посвященных нумерованным и маркированным спискам.

Другие списки

К спискам других типов можно отнести списки, описываемые тегами <DIR> и <MENU>. Изначально такие списки планировалось делать более компактными и строгими по отношению к какому-либо форматированию. Также предполагалось наложить ряд ограничений: на длину строки элементов списка, на количество символов в тексте и пр.

Тем не менее, в последней спецификации HTML списки этих типов были названы отмененными, несмотря на их поддержку большинством современных браузеров. Вместо <DIR> и <MENU> Консорциум W3C рекомендует использовать стандартные маркированные списки.

Для закрепления полученной в этой главе информации все описанные типы HTML-списков приведены в табл. 3.2.

Таблица 3.2. *Типы HTML-списков*

Тип списка	Описание	Используемые теги/параметры
Нумерованный список	Упорядоченный список с возможностью нумерации по буквам и цифрам	<OL>, <LI> TYPE, START, COMPACT

Таблица 3.2 *(окончание)*

Тип списка	Описание	Используемые теги/параметры
Маркированный список	Неупорядоченный список с возможностью маркировки элементов (в том числе — графическими изображениями)	`<UL>`, `<LI>` `TYPE`, `COMPACT`
Список определений	Список терминов с содержательной частью (описанием термина)	`<DL>`, `<DT>`, `<DD>` `COMPACT`

ГЛАВА 4

Таблица как основа HTML-документа

Табличное представление данных

Одним из наиболее мощных и гибких средств представления информационных данных в HTML по праву являются *таблицы*. В повседневной жизни мы часто сталкиваемся с таблицами, решая те или иные задачи. В HTML таблицы являются не только удобным средством структурирования информации. Сегодня таблица становится основой большинства электронных документов, структура которых может включать самые разнообразные элементы HTML. Таблицы, изначально избранные в качестве визуального способа представления данных, сегодня выполняют гораздо более важную функцию — управление структурой HTML-документа в целом. Удобство размещения данных в таблице неоспоримо, а преимущества перед другими средствами представления информации (например, списки) позволяют считать таблицы основополагающим структурным элементом любого HTML-документа.

И все же первостепенной задачей таблицы является представление информационных данных. Поэтому далее пойдет речь о том, как правильно создавать таблицы (подробнее о таблицах, используемых в качестве структурной основы HTML-документов, будет рассказано в *разд. "Вложенные таблицы" данной главы*).

Создание простейших таблиц

HTML-конструкция таблицы может содержать множество различных тегов и параметров, часть которых распространена, а часть используется лишь в редких случаях.

Основным тегом-контейнером таблицы является тег `<TABLE>`, который требует обязательного присутствия закрывающего тега. Любая таблица состоит из

ряда (тег-контейнер <TR>, Table Row), содержащего определенное количество ячеек (тег-контейнер <TD>, Table Data). В принципе, тег <TD> предназначен для указания данных в таблице, а для размещения заголовков в ячейке используется тег <TH> (Table Header). Различие между этими двумя тегами заключается в визуальном представлении данных внутри таблицы: по умолчанию, текст, размещенный в теге <TH>, отображается жирным начертанием с типом выравнивания по центру, а текст тега <TD> — обычным начертанием и с выравниванием по левому краю.

Теги <TR>, <TD> и <TH> могут не иметь своих закрывающих тегов. И все-таки рекомендуется их указывать для предотвращения ошибок, которые могут возникнуть при создании сложных вложенных таблиц. Тег <TABLE>, как уже было сказано выше, обязательно должен иметь закрывающий тег.

Таблица может состоять из любого количества рядов (<TR>), каждый из которых может включать любое количество ячеек (<TD>, <TH>). При этом один ряд таблицы не может содержать ячейки другого ряда.

Таблице можно присвоить заголовок с помощью тега-контейнера <CAPTION> (листинг 4.1, рис. 4.1), который указывается сразу же после тега <TABLE> (в принципе, расположение тега <CAPTION> возможно в любом месте основного тега таблицы, за исключением областей описания тегов рядов и ячеек, однако в спецификации такой подход не приветствуется).

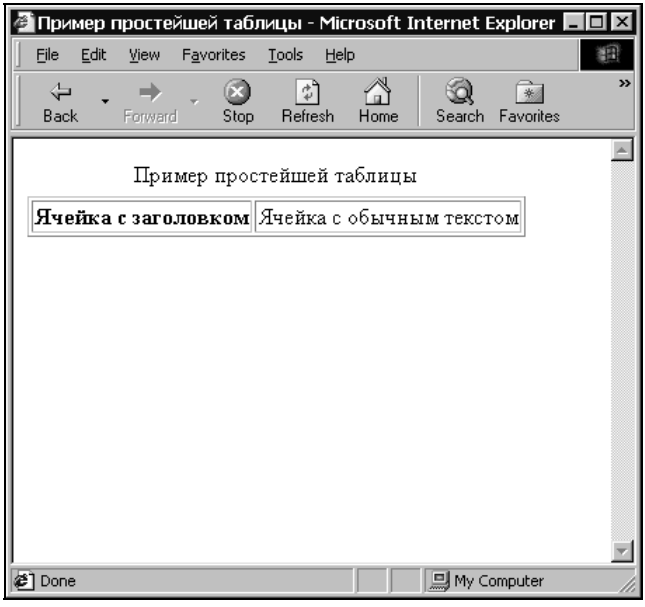


Рис. 4.1. Пример простейшей таблицы

Листинг 4.1. Пример простейшей таблицы

```
<HTML>
  <HEAD>
    <TITLE>Пример простейшей таблицы</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<TABLE BORDER>
<CAPTION>Пример простейшей таблицы</CAPTION>
<TR>
<TH>Ячейка с заголовком</TH>
<TD>Ячейка с обычным текстом</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

Правила описания таблиц

В этом разделе более подробно будет рассказано об основных параметрах (а также их возможных значениях) вышеупомянутых тегов таблицы.

Параметры тега <*TABLE*>

Тег `<TABLE>` является основным тегом таблицы и может содержать ряд параметров, каждый из которых не обязателен для указания. Следует заметить, что некоторые популярные браузеры (Internet Explorer, Netscape и др.) поддерживают часть дополнительных параметров таблицы. Далее речь пойдет о наиболее распространенных параметрах тега `<TABLE>`, поддерживаемых большинством современных браузеров.

Параметр *BORDER*

С помощью параметра `BORDER` можно задавать рамку, показывающую внутреннюю структуру таблицы, которая состоит из определенного количества рядов и ячеек. Отсутствие данного параметра означает, что рамка не будет отображаться браузером. Необходимость в таком варианте может возникнуть, например, при составлении больших списков или создании таблицы в качестве основы HTML-документа.

Значением параметра `BORDER` может быть любое целое число, указываемое в пикселах:

```
<TABLE BORDER="2">
<TR>
<TD>Ячейка 1</TD>
<TD>Ячейка 2</TD>
</TR>
</TABLE>
```

При отсутствии установленного значения параметр BORDER принимает вид BORDER="1".

Параметр *BORDERCOLOR*

Параметр BORDERCOLOR предназначен для определения цвета рамки вокруг таблицы (поддерживается только Internet Explorer). Значение указывается в шестнадцатеричном формате (например, #FFCC00) или в виде наименования (например, green).

Параметр *CELLSPACING*

Параметр CELLSPACING задает расстояние между соседними ячейками таблицы по вертикали и горизонтали. Значение параметра указывается в пикселах и не может быть пропущено. При конструкции CELLSPACING="0" ячейки таблицы сольются, т. к. параметр устанавливает нулевое значение расстояния между соседними ячейками.

```
<TABLE BORDER="2" CELLSPACING="2">
<TR>
<TD>Ячейка 1</TD>
<TD>Ячейка 2</TD>
</TR>
</TABLE>
```

Расстояние между соседними ячейками (как по вертикали, так и по горизонтали) всегда будет одинаково, т. к. HTML не позволяет назначать разные значения в пределах данного параметра.

Параметр *CELLPADDING*

Для создания и регулирования отступа между рамкой ячейки и ее содержимым (по вертикали и горизонтали) используется параметр CELLPADDING. Форма указания значения аналогична параметру CELLSPACING и также не может быть пропущена.

```
<TABLE BORDER="2" CELLSPACING="2" CELLPADDING="7">
<TR>
<TD>Ячейка 1</TD>
```

```
<TD>Ячейка 2</TD>
</TR>
</TABLE>
```

При значении параметра CELLPADDING, равном нулю, содержимое ячейки будет вплотную прижато к рамке таблицы, что ухудшит восприятие информации (это утверждение еще более актуально для таблиц с прозрачной рамкой, т. е. значением BORDER="0": в этом случае текст соседних ячеек сольется).

Отступ между рамкой ячейки и ее содержимым (как по вертикали, так и по горизонтали) всегда будет одинакового размера (как и в случае с параметром CELLSPACING), т. к. HTML не позволяет назначать разные значения в пределах данного параметра.

Параметр *ALIGN*

Данный параметр устанавливает тип выравнивания всей таблицы относительно ширины HTML-документа. Существует три возможных значения параметра ALIGN:

- ALIGN="left" — выравнивание таблицы по левому краю документа. Значение принято по умолчанию, и в случае отсутствия параметра ALIGN таблице будет присвоено именно это значение;
- ALIGN="right" — выравнивание таблицы по правой границе документа;
- ALIGN="center" — центрирование таблицы относительно ширины документа.

В некоторых учебниках и руководствах по языку разметки HTML утверждается, что выравнивание таблицы по центру не поддерживается популярными браузерами. На самом деле, это неверно: все три наиболее распространенных браузера (Internet Explorer, Netscape и Opera), описание которых можно найти в *главе 1*, поддерживают центрирование таблицы по горизонтали.

Листинг 4.2. Пример выравнивания таблицы по центру

```
<HTML>
  <HEAD>
    <TITLE>Пример выравнивания таблицы по центру</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue" LEFTMARGIN="40" RIGHTMARGIN="40" MARGINWIDTH="40">
<TABLE ALIGN="center" BORDER="2" CELLSPACING="2" CELLPADDING="7">
<TR>
<TH>Состав персонального компьютера</TH>
```

```
<TD>
<FONT FACE="Tahoma">В состав персонального компьютера входят следующие
компоненты:</FONT>
<UL TYPE="disc">
       <LI>Системный блок
       <LI>Монитор
       <LI>Клавиатура
       <LI>Мышь
</UL>
</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

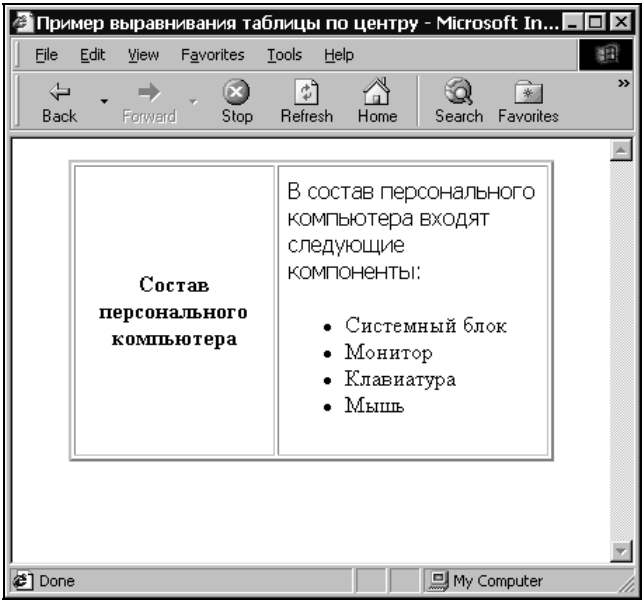


Рис. 4.2. Пример выравнивания таблицы по центру (Internet Explorer)

Как видно из рис. 4.2—4.4, отступы между таблицей и левой и правой границами документа абсолютно равны, отсюда можно сделать вывод о том, что выравнивание таблицы по центру поддерживается наиболее распространенными браузерами (при сравнении отступов не следует принимать во внимание отступ, оставленный Internet Explorer для возможной полосы прокрутки).

Помимо функции выравнивания таблицы по горизонтали документа, параметр `ALIGN` позволяет получать эффект "обтекания" таблицы текстом (рис. 4.5).

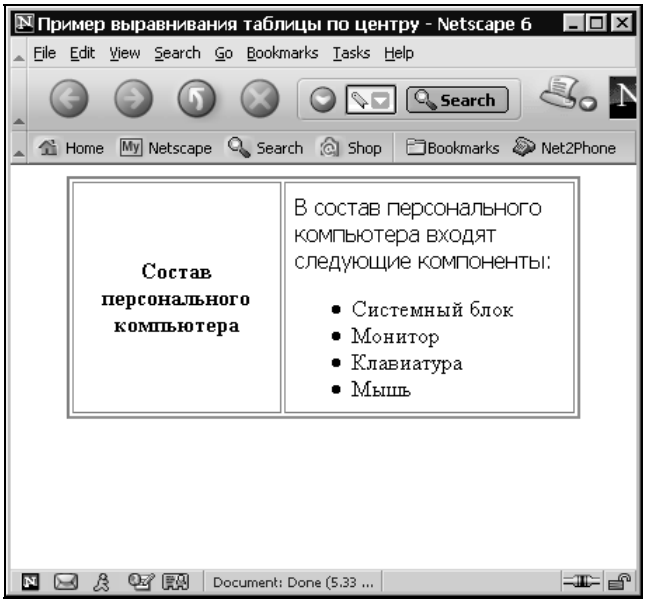


Рис. 4.3. Пример выравнивания таблицы по центру (Netscape)

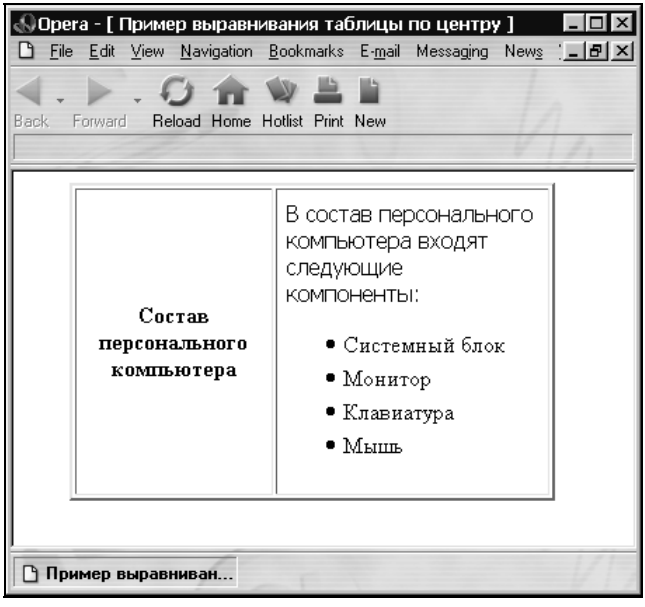


Рис. 4.4. Пример выравнивания таблицы по центру (Opera)

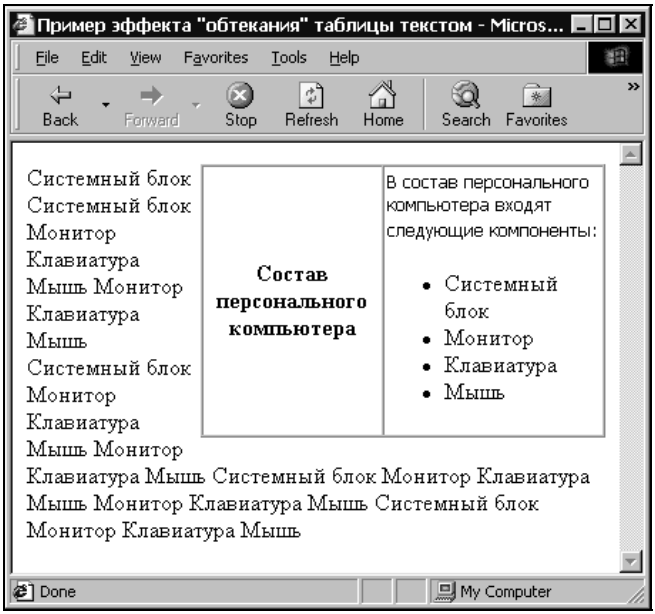


Рис. 4.5. Пример эффекта "обтекания" таблицы текстом

Для этого нужно выбрать тип выравнивания таблицы (для "обтекания" текстом центрирование не может быть выбрано) и назначить ширину таблицы меньше 100% или зафиксировать ее (подробнее об изменении параметра ширины таблицы рассказано в следующем разделе). Текст, который должен огибать таблицу, размещается сразу же после закрывающего тега `</TABLE>`.

Параметры *WIDTH* и *HEIGHT*

Параметры `WIDTH` и `HEIGHT` отвечают за ширину и высоту таблицы, соответственно. Значение этих параметров может быть указано в процентах или пикселах:

```
<TABLE WIDTH="500" HEIGHT="50%">
```

В случае указания ширины или высоты в пикселах таблица будет иметь фиксированные размеры. Такая таблица при изменении размеров окна браузера останется неизменной. При использовании процентов в качестве единицы размеров таблицы, ширина/высота ее будет варьироваться в зависимости от размеров окна браузера. При этом размер ячеек таблицы будет уменьшаться/увеличиваться пропорционально заявленному значению в процентах. Содержимое таблицы будет перемещаться в окне браузера соответственно изменению его размеров. Например, при уменьшении окна браузера текст, раз-

мещенный в ячейке, может быть перенесен на другую строку, растянут или, наоборот, сжат (при выравнивании текста по ширине: `P ALIGN="justify"`).

Описанные параметры тега `<TABLE>` перечислены в табл. 4.1.

Таблица 4.1. *Параметры тега* `<TABLE>`

Параметр	Функция
BORDER	Создание рамки вокруг таблицы
BORDERCOLOR	Определение цвета рамки вокруг таблицы
CELLSPACING	Указание расстояния между соседними ячейками
CELLPADDING	Создание отступа от границы ячейки до ее содержимого
ALIGN	Указание типа выравнивания таблицы по ширине окна браузера
WIDTH	Определение ширины таблицы
HEIGHT	Определение высоты таблицы

Параметры тегов *<TR>*, *<TD>* и *<TH>*

Теги `<TR>`, `<TD>` и `<TH>` составляют внутреннюю структуру таблицы и могут содержать набор определенных параметров. Прежде чем перейти к их рассмотрению, заметим, что внутри тега указания ряда таблицы `<TR>` не могут располагаться текст, списки, графические изображения и прочие HTML-элементы и теги. Форматирование любых типов и указание других тегов может быть только в пределах тегов `<TD>` и `<TH>`, определяющих содержимое табличных ячеек.

Параметры *ALIGN* и *VALIGN*

Параметр `ALIGN` предназначен для указания типа выравнивания содержимого ячеек. Если при использовании этого параметра в теге `<TABLE>` вся таблица выравнивается определенным образом, то здесь назначается тип выравнивания для конкретной ячейки или ряда таблицы. Если необходимо, чтобы каждая ячейка содержала разные типы выравнивания, придется указывать соответствующее значение параметра `ALIGN` в каждом теге `<TD>` или `<TH>`. Если требуется задать один тип выравнивания для всего ряда (включающего все ячейки), значение параметра `ALIGN` прописывается в теге `<TR>` (листинг 4.3, рис. 4.6).

Возможными значениями параметра `ALIGN` являются `ALIGN="left"`, `ALIGN="right"` и `ALIGN="center"`. При этом для `<TD>` значением по умолчанию считается выравнивание по левому краю ячейки, для `<TH>` — выравнивание по центру.

Параметр `VALIGN` также определяет тип выравнивания содержимого ячеек таблицы, но по вертикали. Может принимать следующие значения:

- `VALIGN="middle"` — выравнивание по середине ячейки (значение по умолчанию);
- `VALIGN="top"` — выравнивание по верхнему краю ячейки;
- `VALIGN="bottom"` — выравнивание по нижней границе ячейки;
- `VALIGN="baseline"` — выравнивание по базовой линии (привязка текста одной строки к единому уровню).

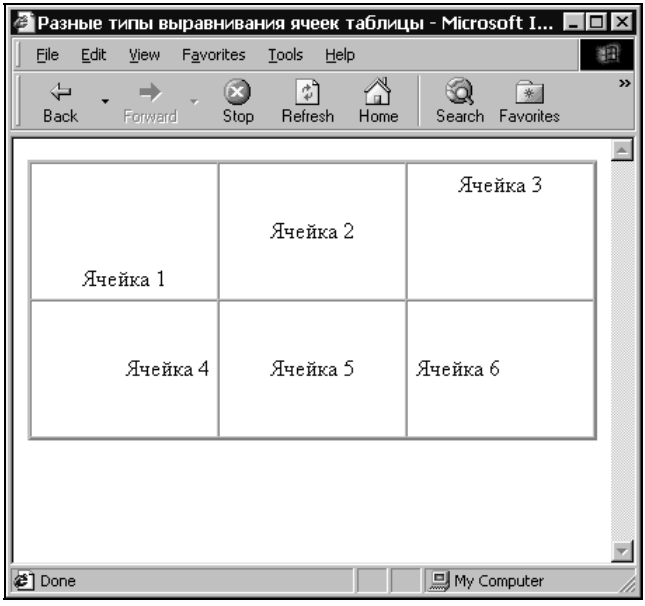


Рис. 4.6. Разные типы выравнивания ячеек таблицы

Листинг 4.3. Разные типы выравнивания ячеек таблицы

```
<HTML>
  <HEAD>
    <TITLE>Разные типы выравнивания ячеек таблицы</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue" LEFTMARGIN="40" RIGHTMARGIN="40" MARGINWIDTH="40">
<TABLE ALIGN="center" BORDER="2" CELLSPACING="0" CELLPADDING="5"
WIDTH="100%" HEIGHT="200">
<TR ALIGN="center">
```

```
<TD VALIGN="bottom">Ячейка 1</TD>
<TD>Ячейка 2</TD>
<TD VALIGN="top">Ячейка 3</TD>
</TR>
<TR>
<TD align="right">Ячейка 4</TD>
<TD align="center">Ячейка 5</TD>
<TD>Ячейка 6</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

Параметры ALIGN и VALIGN используются в тегах <TR> (для всего ряда таблицы), <TD> и <TH>.

Параметры *BGCOLOR* и *BACKGROUND*

Параметры BGCOLOR и BACKGROUND используются для работы с задним фоном ячеек или целых рядов таблицы. Параметр BGCOLOR позволяет задать определенный цвет ячеек, а BACKGROUND — указать путь к графическому изображению, которое будет служить задним фоном (рис. 4.7).

Рис. 4.7. Использование цвета и графического изображения для заднего фона ячеек

Действие данных параметров в отношении тега ряда <TR> и тегов ячейки <TD> и <TH> аналогично типу выравнивания ALIGN и VALIGN.

Параметр BGCOLOR используется для <TR>, <TD> и <TH>. Параметр BACKGROUND применим только к тегам ячеек <TD> и <TH>.

Параметры *WIDTH* и *HEIGHT*

Параметры ширины и высоты либо отдельных ячеек, либо целых рядов таблицы. Правила указания аналогичны этим же параметрам тега <TABLE>. Используются для <TR>, <TD> и <TH>.

Параметр *NOWRAP*

Параметр NOWRAP запрещает принудительный перенос строки в ячейке или табличном ряде. Не рекомендуется использовать данный параметр во всех ячейках, т. к. это может сильно понизить уровень масштабируемости таблицы (при условии, что значение ширины и/или высоты таблицы указано в процентах). Параметр NOWRAP применим в <TR>, <TD> и <TH>.

Параметры *COLSPAN* и *ROWSPAN*

Иногда при построении таблиц возникает необходимость в объединении нескольких соседних ячеек в одну. Для этого используются параметры COLSPAN и ROWSPAN (листинг 4.4, рис. 4.8).

Параметр COLSPAN показывает, на сколько ячеек по горизонтали следует расширить <TD> или <TH>, в которых указан данный параметр. Параметр ROWSPAN делает то же самое, но увеличивает область <TD> или <TH> по вертикали.

Листинг 4.4. Пример объединения ячеек с помощью COLSPAN и ROWSPAN

```
<HTML>
  <HEAD>
    <TITLE>Пример объединения ячеек с помощью COLSPAN и ROWSPAN</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue" LEFTMARGIN="40" RIGHTMARGIN="40" MARGINWIDTH="40">
<TABLE ALIGN="center" BORDER="2" CELLSPACING="0" CELLPADDING="5"
WIDTH="100%" HEIGHT="200">
<TR ALIGN="center" BGCOLOR="#CECECE">
<TH COLSPAN="2">Ячейка 1</TH>
</TR>
<TR>
<TD ALIGN="center" ROWSPAN="3">Ячейка 2</TD>
```

```
<TD ALIGN="center">Ячейка 3</TD>
</TR>
<TR>
<TD ALIGN="center">Ячейка 4</TD>
</TR>
<TR>
<TD ALIGN="center">Ячейка 5</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

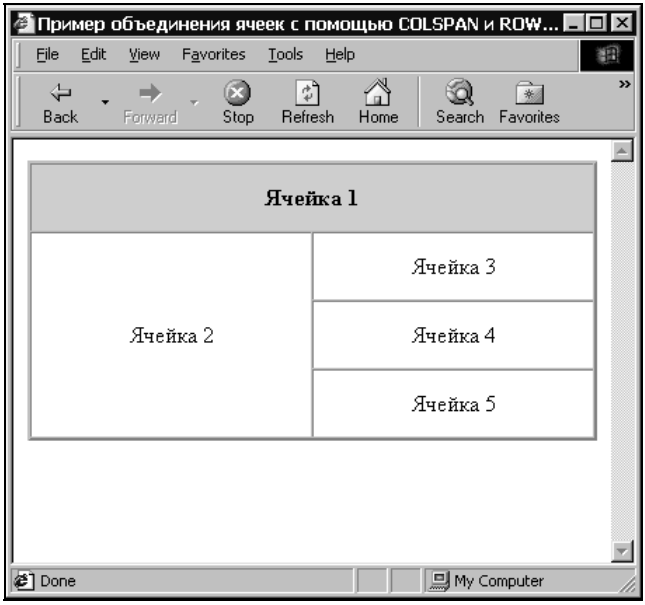


Рис. 4.8. Пример объединения ячеек с помощью COLSPAN и ROWSPAN

При использовании параметров COLSPAN и ROWSPAN особое внимание следует уделять корректному объединению соседних ячеек таблицы, а также своевременному указанию закрывающих тегов ячеек и рядов. Возникновение ошибки может привести к нарушению структуры таблицы, "заползанию" одних ячеек на другие, перекрытию текста и даже невозможности отобразить таблицу в браузере.

Параметры COLSPAN и ROWSPAN используются только в тегах ячейки <TD> и <TH>.

Перечень параметров тегов <TR>, <TD> и <TH> приведен в табл. 4.2.

Таблица 4.2. *Параметры тегов* `<TR>`, `<TD>` *и* `<TH>`

Параметр	Функция	Применение
ALIGN	Выравнивание содержимого ячейки или ряда по горизонтали	<TR>, <TD>, <TH>
VALIGN	Выравнивание содержимого ячейки или ряда по вертикали	<TR>, <TD>, <TH>
WIDTH	Определение ширины ячейки или ряда	<TR>, <TD>, <TH>
HEIGHT	Определение высоты ячейки или ряда	<TR>, <TD>, <TH>
BGCOLOR	Указание цвета для заднего фона ячейки или ряда	<TR>, <TD>, <TH>
BACKGROUND	Указание рисунка для заднего фона ячейки	<TD>, <TH>
NOWRAP	Запрет принудительного переноса строки в ячейке или ряду	<TR>, <TD>, <TH>
COLSPAN	Объединение соседних ячеек по горизонтали	<TD>, <TH>
ROWSPAN	Объединение соседних ячеек по вертикали	<TD>, <TH>

Нестандартное представление таблиц

Этот раздел будет посвящен некоторым специфическим особенностям построения таблиц, поддержка которых, по большому счету, присутствует только в браузере Internet Explorer, и различным нестандартным способам работы с таблицами.

Пустые ячейки

В процессе построения таблиц может возникнуть необходимость оставить одну или более ячеек незаполненными. Однако банальное отсутствие символов внутри тегов `<TD>` и `<TH>`, в зависимости от браузера, может привести к самым разнообразным вариантам визуального отображения таблицы на экране (листинг 4.5).

Листинг 4.5. Пример отображения пустых ячеек таблицы различными браузерами

```
<HTML>
  <HEAD>
    <TITLE>Пример отображения пустых ячеек таблицы различными
     браузерами</TITLE>
  </HEAD>
```

```
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue" LEFTMARGIN="40" RIGHTMARGIN="40" MARGINWIDTH="40">
<TABLE ALIGN="center" BORDER="1" CELLSPACING="0" CELLPADDING="5"
WIDTH="100%" HEIGHT="200">
<TR ALIGN="center">
<TD>Заполненная ячейка</TD>
<TD></TD>
<TD>Заполненная ячейка</TD>
</TR>
<TR ALIGN="center">
<TD>Заполненная ячейка</TD>
<TD></TD>
<TD>Заполненная ячейка</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

Из рис. 4.9—4.11 видно, что Internet Explorer и Netscape не прорисовывают внутреннюю структуру пустых ячеек, а Opera, к тому же, оттеняет правую границу ячеек с информацией, размещенных слева от пустых табличных ячеек.

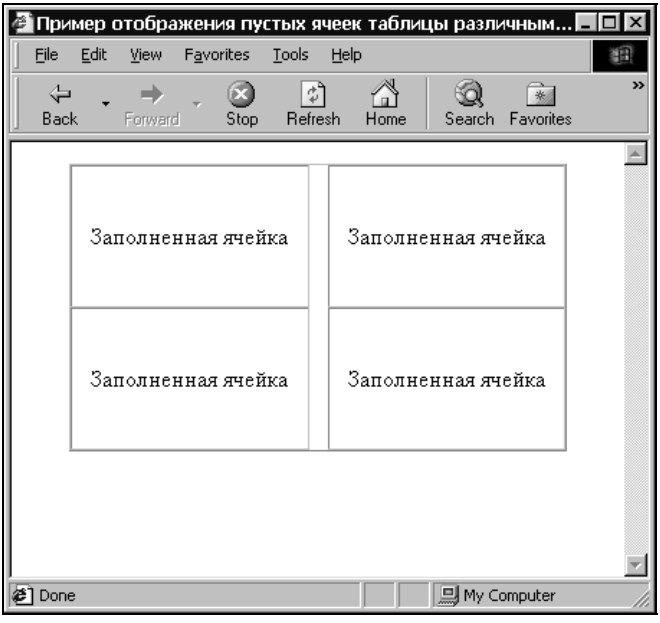


Рис. 4.9. Пример отображения пустых ячеек таблицы браузером Internet Explorer

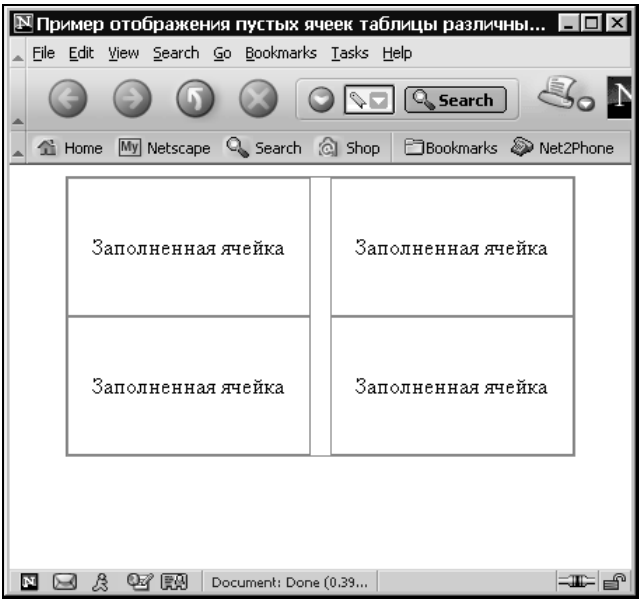


Рис. 4.10. Пример отображения пустых ячеек таблицы браузером Netscape

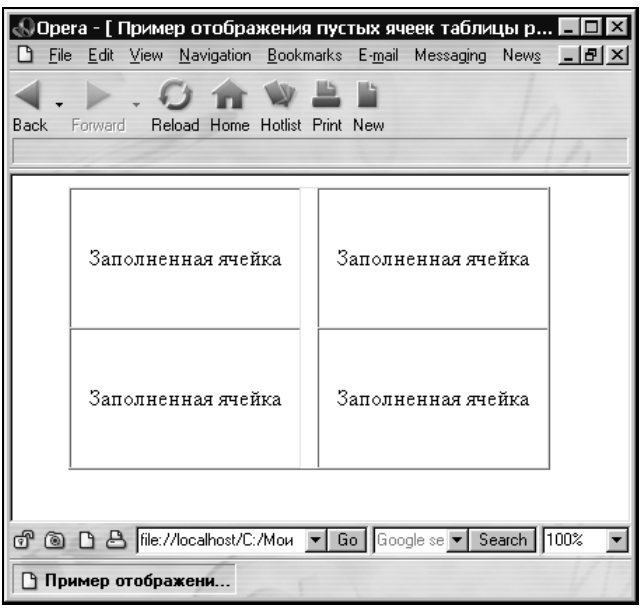


Рис. 4.11. Пример отображения пустых ячеек таблицы браузером Opera

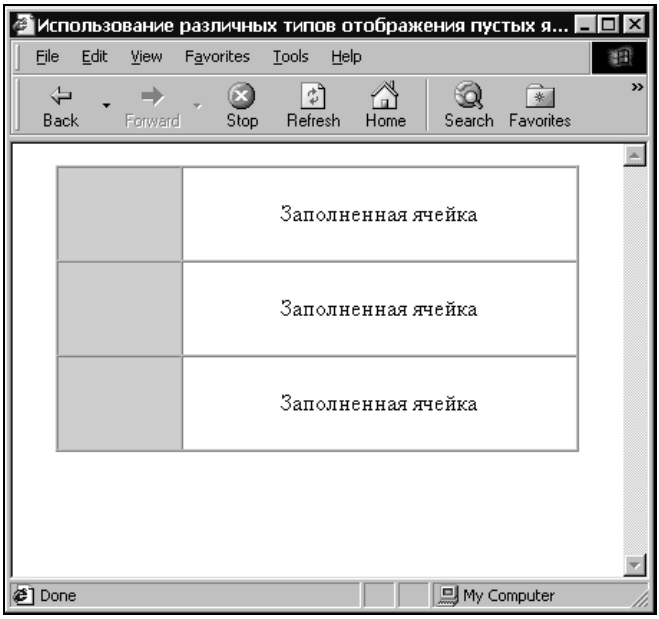


Рис. 4.12. Использование различных типов отображения пустых ячеек таблицы

Чтобы избежать таких различий в отображении таблиц, внутри тегов ячеек `<TD>` и `<TH>` следует размещать невидимые на экране данные, позволяющие браузерам не отображать такие ячейки, как пустые (листинг 4.6, рис. 4.12). Подобные данные могут быть следующих типов:

- теги структурного форматирования — в данном случае под тегами структурного форматирования подразумеваются тег принудительного переноса строки `<BR>` и конструкция запрета такого переноса ` `. При использовании этих элементов следует помнить, что размер пустой (в визуальном плане) ячейки будет зависеть от свойств шрифта, заданных для HTML-документа в целом или определенных для конкретного блока (`<SPAN>`, `<DIV>`) или ячейки (`<TD>`, `<TH>`);
- цветные текстовые блоки — разместив в ячейке, которая должна визуально казаться пустой, текст небольшого объема цветом, аналогичным значению параметра `BGCOLOR` тега `<TD>` или `<TH>`, при просмотре такого документа в браузере мы получим "пустую" табличную ячейку;
- прозрачные графические распорки — способ отображения пустых ячеек, практически не зависящий от модели браузера и свойств шрифта. Заключается в размещении прозрачного графического изображения (распорки, спейсера) нужных размеров в пределах табличной ячейки. Прозрачная распорка (в формате GIF) невидима на странице и очень гибка в масшта-

бировании. К примеру, невидимый спейсер исходного размера 10×10 пикселов при необходимости можно искусственно "увеличить" или "уменьшить" путем переназначения его параметров WIDTH и HEIGHT.

Листинг 4.6. Использование различных типов отображения пустых ячеек таблицы

```
<HTML>
  <HEAD>
    <TITLE>Использование различных типов отображения пустых ячеек
     таблицы</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<TABLE ALIGN="center" BORDER="1" CELLSPACING="0" CELLPADDING="5"
WIDTH="90%" HEIGHT="200">
<TR ALIGN="center">
<TD BGCOLOR="#CECECE"> </TD>
<TD>Заполненная ячейка</TD>
</TR>
<TR ALIGN="center">
<TD BGCOLOR="#CECECE"><FONT COLOR="#CECECE">текст</FONT></TD>
<TD>Заполненная ячейка</TD>
</TR>
<TR ALIGN="center">
<TD BGCOLOR="#CECECE"><IMG SRC="spacer.gif" WIDTH="10" HEIGHT="10"
ALT=""></TD>
<TD>Заполненная ячейка</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

Группировка данных

При построении таблиц мы можем легко задать единый тип выравнивания для отдельной ячейки и даже целого табличного ряда. Гораздо чаще может возникнуть необходимость в едином выравнивании информационных данных конкретного столбца. Столбец в таблице — это последовательность ячеек, располагающихся в разных рядах.

Стандартными средствами HTML нам пришлось бы задавать один и тот же тип выравнивания для отдельно взятой ячейки, формирующей нужный столбец:

```
<TABLE>
<TR>
<TD ALIGN="right">Ячейка 1 с выравниванием вправо</TD>
<TD ALIGN="center">Ячейка 1 с выравниванием по центру</TD>
</TR>
<TR>
<TD ALIGN="right">Ячейка 2 с выравниванием вправо</TD>
<TD ALIGN="center">Ячейка 2 с выравниванием по центру</TD>
</TR>
</TABLE>
```

Работая с браузером Internet Explorer, труд разработчика электронного документа может быть сильно облегчен за счет таких тегов, как `<COL>` и `<COLGROUP>`.

Теги `<COL>` и `<COLGROUP>` предназначены для определения свойств отображения табличных данных, сгруппированных по конкретному признаку. Оба могут иметь параметр `SPAN`, задающий количество соседних столбцов, и `ALIGN`, устанавливающий единый тип выравнивания для выбранного столбца (возможные значения: по левому краю, по правому краю, по центру; формат записи аналогичен тегам `<TD>` и `<TH>`) (листинг 4.7, рис. 4.13).

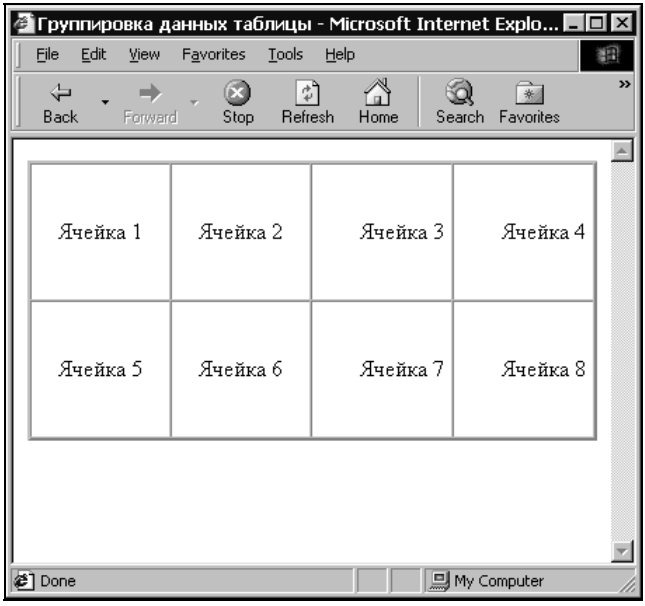


Рис. 4.13. Группировка данных таблицы

Помимо этого, тег `<COLGROUP>` может содержать дополнительный параметр вертикального выравнивания данных — `VALIGN` (возможные значения: по

верхнему краю, по нижнему краю, по середине; формат записи аналогичен тегам <TD> и <TH>).

Разница между тегами <COL> и <COLGROUP> заключается в условном объединении (группировке) взаимосвязанных данных отдельно взятого столбца таблицы.

Листинг 4.7. Группировка данных таблицы

```
<HTML>
  <HEAD>
    <TITLE>Группировка данных таблицы</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<TABLE ALIGN="center" BORDER="2" CELLSPACING="0" CELLPADDING="5"
WIDTH="100%" HEIGHT="200">
<COLGROUP ALIGN="center" SPAN="2">
<COLGROUP ALIGN="right" SPAN="2">
<TR>
<TD>Ячейка 1</TD><TD>Ячейка 2</TD><TD>Ячейка 3</TD><TD>Ячейка 4</TD>
</TR>
<TR>
<TD>Ячейка 5</TD><TD>Ячейка 6</TD><TD>Ячейка 7</TD><TD>Ячейка 8</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

Из рис. 4.13 видно, что таблица состоит из четырех столбцов, данные в которых сгруппированы по заданному признаку. Два правых столбца имеют тип выравнивания по центру, два левых — по правому краю.

Колонтитулы таблицы

Браузер Internet Explorer позволяет использовать дополнительные теги структурирования табличных данных, а именно — <THEAD>, <TBODY> и <TFOOT>. Данные теги предназначены для создания колонтитулов таблицы различных уровней (соответственно, для верхнего, основного (содержательного) и нижнего уровней таблицы).

Теги верхнего и нижнего колонтитулов <THEAD> и <TFOOT> могут быть использованы в структуре таблицы лишь единожды, причем для них необязательно наличие закрывающих тегов.

Тег основного колонтитула <TBODY> может встречаться неоднократно в пределах одной таблицы, однако требует своего закрывающего тега.

Верхний и нижний колонтитулы функционально очерчивают логические заголовки соответствующего уровня и применимы, в основном, в больших таблицах, не помещающихся в пределах одной страницы электронного документа.

Основные колонтитулы выполняют функцию, аналогичную тегам группировки <COL> и <COLGROUP>, расставляя логические метки по ходу изложения основной содержательной части таблицы.

Прорисовка структуры таблицы

Еще одной замечательной возможностью нестандартного представления таблиц, работающей только в браузере Internet Explorer, является прорисовка внутренней структуры таблицы, а именно рамок и линеек. Другими словами, можно оформить таблицу таким образом, что от рамки останутся только верхняя и нижняя границы, а между ячейками останется только вертикальная линейка (рис. 4.14).

За изменение свойств рамки отвечает параметр FRAME, а с помощью параметра RULES варьируется внешний вид линеек таблицы (оба параметра применимы внутри тега <TABLE>) (листинг 4.8).

Рис. 4.14. Нестандартные возможности прорисовки структуры таблицы

Листинг 4.8. Нестандартные возможности прорисовки структуры таблицы

```
<HTML>
  <HEAD>
    <TITLE>Нестандартные возможности прорисовки структуры таблицы</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<TABLE ALIGN="center" BORDER="1" CELLSPACING="0" CELLPADDING="4"
WIDTH="100%" FRAME="hsides" RULES="rows">
<CAPTION><B>Таблица 1</B></CAPTION>
<TR>
<TD>Ячейка 1</TD><TD>Ячейка 2</TD><TD>Ячейка 3</TD><TD>Ячейка 4</TD>
</TR>
<TR>
<TD>Ячейка 5</TD><TD>Ячейка 6</TD><TD>Ячейка 7</TD><TD>Ячейка 8</TD>
</TR>
</TABLE>
<TABLE ALIGN="center" BORDER="1" CELLSPACING="0" CELLPADDING="4"
WIDTH="100%" FRAME="box" RULES="groups">
<CAPTION><B>Таблица 2</B></CAPTION>
<TR>
<TD>Ячейка 1</TD><TD>Ячейка 2</TD><TD>Ячейка 3</TD><TD>Ячейка 4</TD>
</TR>
<TR>
<TD>Ячейка 5</TD><TD>Ячейка 6</TD><TD>Ячейка 7</TD><TD>Ячейка 8</TD>
</TR>
</TABLE>
<TABLE ALIGN="center" BORDER="1" CELLSPACING="0" CELLPADDING="4"
WIDTH="100%" FRAME="above" RULES="all">
<CAPTION><B>Таблица 3</B></CAPTION>
<TR>
<TD>Ячейка 1</TD><TD>Ячейка 2</TD><TD>Ячейка 3</TD><TD>Ячейка 4</TD>
</TR>
<TR>
<TD>Ячейка 5</TD><TD>Ячейка 6</TD><TD>Ячейка 7</TD><TD>Ячейка 8</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

Возможные значения параметров `FRAME` и `RULES` приведены в табл. 4.3 и 4.4, соответственно.

Таблица 4.3. *Возможные значения параметра* `FRAME`

Значение	Функция
BOX	Рамка с четырех сторон
BORDER	Рамка с четырех сторон
ABOVE	Рамка только сверху
BELOW	Рамка только снизу
HSIDES	Верхняя и нижняя части рамки
VSIDES	Левая и правая части рамки
LHS	Только левая часть рамки
RHS	Только правая часть рамки
VOID	Нет рамок

Таблица 4.4. *Возможные значения параметра* `RULES`

Значение	Функция
ALL	Отображение линейки целиком
GROUPS	Часть линейки, разделяющая сгруппированные данные
COLS	Часть линейки, разделяющая столбцы
ROWS	Часть линейки, разделяющая строки
NONE	Отсутствие линейки

Вложенные таблицы

Одной из замечательных особенностей HTML-таблиц по праву считается поддержка многоуровневой вложенности. Другими словами, одна таблица может включать другую, та, в свою очередь, еще одну и т. д.

Преимущества вложенных таблиц

Эта особенность, в отличие от других способов представления данных в электронном документе, позволяет более точно размещать отдельные элементы страницы относительно друг друга и границ самого документа, отображаемого браузером.

Например, два разнородных блока текста и нумерованный список, размещенные внутри тега `<BODY>`, невозможно разместить на одном уровне и тем бо-

лее — на одном уровне со смещением в какую-либо сторону. Использование таблиц с легкостью решает эту проблему, позволяя располагать различные элементы и их комбинации в разных местах документа посредством видимых и невидимых ячеек и рядов таблицы.

Вот почему в последнее время преобладающее большинство HTML-документов создается на основе таблиц, где в качестве несущей основы берется таблица с невидимыми краями, содержащая вложенные таблицы с разным оформлением и отличающимися значениями параметров.

Подводя итог сказанному выше, выделим следующие преимущества вложенных таблиц:

- гибкая масштабируемость структуры электронного документа в целом;
- широкие возможности позиционирования отдельных элементов страницы;
- многоуровневое представление разнородных информационных данных;
- расширенные оформительские возможности;
- поддержка популярными браузерами.

Пример вложенных таблиц

Правила построения вложенных таблиц ничем не отличаются от создания таблиц одного уровня: используются те же теги и параметры, задаются те же свойства и значения.

Но в ходе создания сложных вложенных таблиц нужно помнить следущее:

- каждая таблица последующего уровня размещается внутри тега-контейнера `<TD>` или `<TH>` таблицы предыдущего уровня;
- вложенная таблица не может быть создана за пределами вышеназванных тегов ячейки таблицы;
- таблица одного уровня может содержать любое количество вложенных таблиц другого уровня, идущих друг за другом в пределах тега ячейки таблицы верхнего уровня;
- количество тегов таблиц всех уровней должно соответствовать количеству закрывающих тегов этих же таблиц.

Рассмотрим один из вариантов использования многоуровневых вложенных таблиц на примере.

Листинг 4.9. Пример использования многоуровневых вложенных таблиц

```
<HTML>
  <HEAD>
```

```
    <TITLE>Пример использования многоуровневых вложенных таблиц</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<H2 ALIGN="center">ПОСТРОЕНИЕ ВЛОЖЕННЫХ ТАБЛИЦ</H2>
<!-- Основная несущая таблица. Начало -->
<TABLE ALIGN="center" BORDER="0" CELLSPACING="0" CELLPADDING="2"
WIDTH="100%">
<TR>
<TD VALIGN="top" WIDTH="50%">
<!-- Левая таблица с текстом. Начало -->
<TABLE ALIGN="center" BORDER="1" CELLSPACING="3" CELLPADDING="5"
WIDTH="100%">
<TR>
<TH BGCOLOR="gray"><FONT COLOR="white">Вложенные таблицы</FONT></TH>
</TR>
<TR>
<TD VALIGN="top">
<P ALIGN="justify">
Одной из замечательных особенностей HTML-таблиц по праву считается
поддержка многоуровневой вложенности.
<BR><BR>
Другими словами, одна таблица может включать другую, та, в свою очередь,
еще одну и т. д.
<BR><BR>
Эта особенность, в отличие от других способов представления данных
в электронном документе, позволяет более точно размещать отдельные
элементы страницы относительно друг друга и границ самого документа,
отображаемого браузером.
</P>
</TD>
</TR>
</TABLE>
<!-- Левая таблица с текстом. Окончание -->
</TD>
<TD VALIGN="top" WIDTH="50%">
<!-- Правая таблица со списком. Начало -->
<TABLE ALIGN="center" BORDER="1" CELLSPACING="3" CELLPADDING="5"
WIDTH="100%">
<TR>
<TH BGCOLOR="#000000"><FONT COLOR="white">Преимущества вложенных
таблиц</FONT></TH>
</TR>
<TR>
<TD VALIGN="top">
```

```
<OL TYPE="1">
    <LI>Гибкая масштабируемость структуры электронного документа
    в целом<BR><BR>
    <LI>Широкие возможности позиционирования отдельных элементов
    страницы<BR><BR>
    <LI>Многоуровневое представление разнородных информационных
     данных<BR><BR>
    <LI>Расширенные оформительские возможности<BR><BR>
    <LI>Поддержка популярными браузерами
</OL>
</TD>
</TR>
</TABLE>
<!-- Правая таблица со списком. Окончание -->
</TD>
</TR>
</TABLE>
<!-- Основная несущая таблица. Окончание -->
</BODY>
</HTML>
```

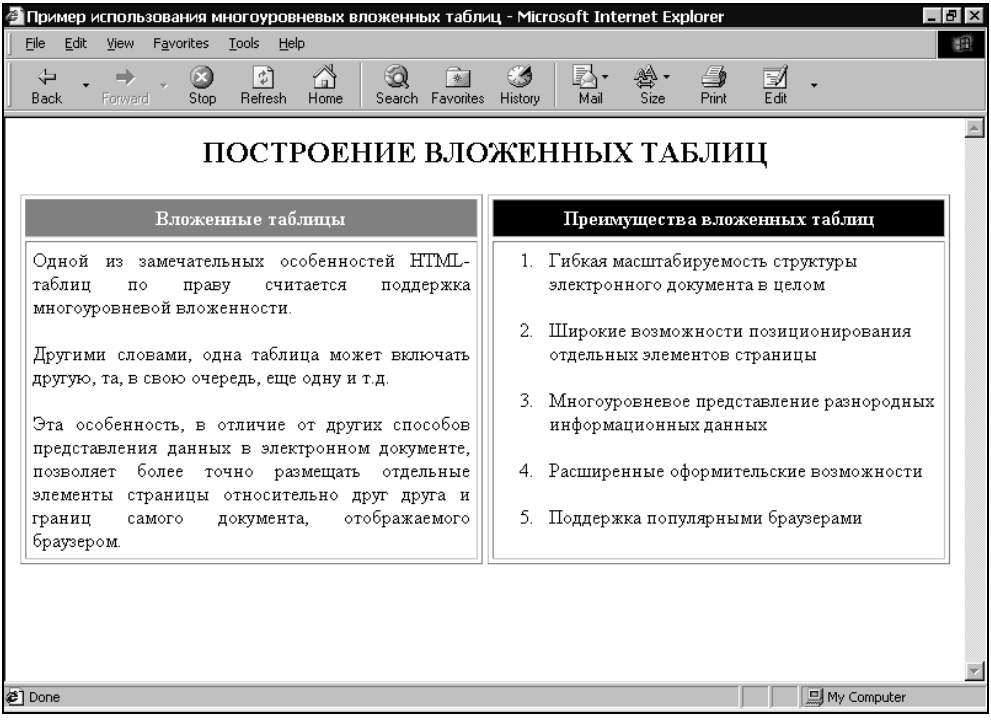


Рис. 4.15. Пример использования многоуровневых вложенных таблиц

Как видно из листинга 4.9, электронный документ состоит из трех таблиц: первая является основной (несущей) и имеет невидимые рамки; две других расположены в ячейках основной таблицы и содержат блок текста и нумерованный список, соответственно (рис. 4.15).

Для удобства восприятия HTML-кода были использованы комментарии, обособляющие структуру каждой из трех таблиц. Ширина ячеек несущей таблицы — по 50%, что позволяет даже при изменении размеров окна браузера разделять информацию на две равные части.

В завершение данной главы для закрепления пройденного материала все описанные теги и параметры, используемые при построении HTML-таблиц, приведены в табл. 4.5.

Таблица 4.5. *Теги и параметры, используемые для построения таблиц*

Тег	Описание	Параметры
<TABLE>	Основной тег построения структуры таблицы. Определяет наличие и цвет рамки, отступы между границами ячеек и соседними ячейками, тип выравнивания, ширину и высоту таблицы. Задает свойства прорисовки рамок и линеек таблицы	BORDER BORDERCOLOR CELLSPACING CELLPADDING ALIGN WIDTH HEIGHT FRAME RULES
<CAPTION>	Тег заголовка таблицы. Задает тип горизонтального и вертикального выравнивания заголовка таблицы	ALIGN VALIGN
<TR>	Тег ряда таблицы. Устанавливает следующие свойства для ряда таблицы: тип вертикального и горизонтального выравнивания, ширину и высоту, цвет заднего фона, запрет переноса строки	ALIGN VALIGN WIDTH HEIGHT BGCOLOR NOWRAP
<TD>, <TH>	Теги ячейки таблицы. Устанавливают следующие свойства для ячейки таблицы: тип вертикального и горизонтального выравнивания, ширину и высоту, цвет и графическое изображение заднего фона, запрет переноса строки, объединение соседних ячеек в одну (по горизонтали и вертикали)	ALIGN VALIGN WIDTH HEIGHT BGCOLOR BACKGROUND NOWRAP ROWSPAN COLSPAN

Таблица 4.5 *(окончание)*

Тег	Описание	Параметры
<COL>	Тег группировки табличных данных. Задает количество соседних столбцов и единый тип горизонтального выравнивания для всех столбцов	SPAN ALIGN
<COLGROUP>	Тег группировки табличных данных с логическим объединением столбцов с информацией. Задает количество соседних столбцов и единый тип горизонтального и вертикального выравнивания для всех столбцов	SPAN ALIGN VALIGN
<THEAD>	Тег верхнего колонтитула таблицы	—
<TBODY>	Тег основного (содержательного) колонтитула таблицы	—
<TFOOT>	Тег нижнего колонтитула таблицы	—

ГЛАВА 5

Добавление графики средствами HTML

Роль графики

Как интересный журнал или проспект теряет в своей привлекательности без цветных иллюстраций, так любой HTML-документ кажется сухим и невзрачным без использования *графики*. Значение графических изображений в аспекте создания электронных документов нельзя переоценить: реклама и коммерческие предложения компаний и юридических лиц становятся более выразительными и яркими; иллюстрации и схемы способны превратить скучный перечень услуг или расценок в интересный информативный материал; любой художественный рассказ или произведение будет восприниматься легче и естественнее при наличии картинок или фотографий.

Однако всегда и во всем следует помнить о чувстве меры. HTML-документ, перенасыщенный иллюстративным материалом, будет неоправданно отвлекать внимание пользователя от истинного содержания страницы — информации. К тому же, чрезмерное увлечение графикой влечет за собой увеличение времени загрузки электронного документа, что может сказаться на общей оценке посетителем вашего интернет-ресурса.

Также необходимо четко и внимательно отбирать графический материал для последующего размещения в HTML-документах, следить за его соответствием тематике вашего интернет-ресурса в целом и содержанию отдельной страницы, в частности.

Следует помнить о том, что графика призвана привлечь внимание посетителя, заострить его интерес на конкретных моментах или формулировках, но ни в коем случае не для отвлечения его от основного содержания Web-сайта. Разумеется, данное утверждение имеет ряд оговорок и исключений из правил (например, в отношении компьютерных галерей и прочих ресурсов, где став-

ка сделана именно на графические изображения), тем не менее, в большинстве случаев необходимо строго следить за количеством графики на ваших HTML-документах.

Характеристика графических стандартов

Любая графическая информация может храниться в двух основных форматах: *векторном* и *растровом*.

Графический файл векторного формата состоит из отдельных математических данных, которые с помощью отрезков прямых, называемых векторами, отображает графический объект на экране. Преимуществом *векторной графики* является независимость качества изображения от масштаба рисунка. Минус — малые возможности для работы с фотографическими изображениями. Примером файлов векторного формата могут послужить CDR (CorelDRAW), AI (Adobe Illustrator), FH (FreeHand), SVG (Scalable Vector Graphics) и др. Векторный формат распространен, в основном, в полиграфии, предпечатной подготовке высококачественных документов и т. д.

Отображение файла растрового формата основано на обработке минимальной единицы рабочей области экрана — точки (пиксела). Изменение размеров растровых изображений может существенно повлиять на их качество, т. к. масштабирование осуществляется без вмешательства каких-либо сложных математических операций. Наиболее распространенными растровыми форматами являются PSD (Photoshop Document), BMP (Bitmap Image).

Исходя из маломасштабности среды, в которой демонстрируется графика в Интернете (средний размер мониторов пользователей не позволяет выводить на экран файлы больших размеров), и некоторых других факторов, форматом, избранным для демонстрации Web-графики, стал растровый формат.

На сегодняшний день для создания графических изображений, предназначенных для размещения в электронных документах, используются три основных стандарта: GIF, JPEG и PNG.

Остановимся на каждом из них немного подробнее.

GIF (Graphics Interchange Format)

Стандарт GIF был разработан компанией Compuserve Inc. для передачи графической информации в пределах определенных компьютерных сетей (разработка велась еще до появления Интернета).

Надо заметить, что в изображении этого стандарта количество цветов ограничивается всего лишь 256-ю, однако, нельзя сказать, чтобы это являлось

большим недостатком. Среди других черт стандарта GIF можно назвать возможность построчного чередования, внесение комментариев, реакцию на пользовательские действия, создание прозрачного фона и поддержку режима анимации кадров.

Две последние возможности стандарта GIF сегодня являются основополагающими и повсеместно используемыми: от создания навигационных меню до интернет-рекламы. К сожалению, такое интересное средство, как реакция на манипуляции пользователя, не реализовано в интернет-технологиях (не поддерживается браузерами).

На сегодняшний день стандарт GIF по-прежнему является самым используемым и популярным в ходе разработки электронных документов благодаря следующим преимуществам:

1. Наличие специального алгоритма сжатия данных позволяет подвергать компрессии файлы GIF без изменения качества изображения.
2. Возможность чередования кадров анимированного GIF-файла приводит к получению большого объема информации, который может быть помещен на один рекламный носитель.
3. По сравнению со статичными изображениями, анимированный GIF-объект привлекает больше внимания со стороны пользователей.
4. Поддержка прозрачности позволяет сэкономить на исходном размере файла.

JPEG (Joint Photographic Experts Group)

Как видно из расшифровки аббревиатуры стандарта JPEG, он был создан специальной группой экспертов в области фотографии и предназначался для хранения графических изображений с большой глубиной цвета.

Профиль деятельности разработчиков JPEG внес свой вклад в позиционирование стандарта: в основном, он использовался (и по сей день используется) для передачи фотографических изображений. В стандарте применен специальный алгоритм компрессии данных: при повышении степени сжатия качество изображения ухудшается за счет вычета "ненужной" информации (в отличие от алгоритма сжатия GIF, который позволяет производить подобную процедуру практически без потерь).

На данный момент стандарт JPEG идет на втором месте по популярности после GIF и используется для создания изображений, в композицию которых входят фотографии, сложные коллажи, объекты, подвергнутые действию различных графических эффектов и фильтров.

PNG (Portable Network Graphics)

Судя по названию, стандарт разрабатывался с учетом особенностей Интернета. И это на самом деле так. PNG вобрал в себя наиболее сильные стороны двух предыдущих стандартов и исключил их недостатки.

В стандарте реализованы следующие средства:

- прозрачный фон;
- построчное чередование;
- сжатие без потерь и др.

Однако PNG, несмотря на свои преимущества, не получил такого признания пользователями Интернета, как стандарты GIF и JPEG (одна из возможных причин — отсутствие поддержки анимации). В Сети можно встретить графические файлы с расширением png, но достаточно редко и в специфических направлениях интернет-отрасли (например, работа программного модуля `GD::Graph`, автоматически генерирующего графики и диаграммы для систем статистики, рейтинга и пр.).

Вставка графики в HTML-документ

Прежде чем перейти к рассмотрению правил встраивания графики в HTML-документы, следует напомнить, что графические изображения также могут быть включены в основной раздел документа, описываемый тегом-контейнером `<BODY>`, а также выступать в качестве заднего фона ячеек таблицы:

```
<BODY BACKGROUND="bg.gif">
<TABLE>
<TR>
<TD BACKGROUND="bg2.jpg">Текст</TD>
</TR>
</TABLE>
```

Описание графики в HTML-документе

Для вставки графических изображений в HTML-документы используется специальный тег `<IMG>`, который не требует наличия закрывающего тега.

Далее рассмотрим существующие параметры этого тега.

Параметры тега *<IMG>*

Параметр *SRC*. Единственный параметр, который является обязательным для указания. Он выполняет важную роль в отображении графического изо-

бражения на странице — задает путь (относительный или абсолютный) к рисунку. Формат указания следующий:

```
<IMG SRC="picture.gif">
```

Следует заметить, что в данном случае браузер станет искать файл "picture.gif" в том же каталоге, что и HTML-документ, в котором данный рисунок будет расположен. Обычно для графических изображений выделяется отдельная папка:

```
<IMG SRC="images/picture.gif">
```

Параметр *BORDER*. Предназначен для отображения рамки определенной толщины вокруг рисунка (рис. 5.1). Толщина указывается в пикселах, по умолчанию (если параметр пропущен), рамка браузером не отображается:

```
<IMG SRC="images/clock.gif" BORDER="5">
```

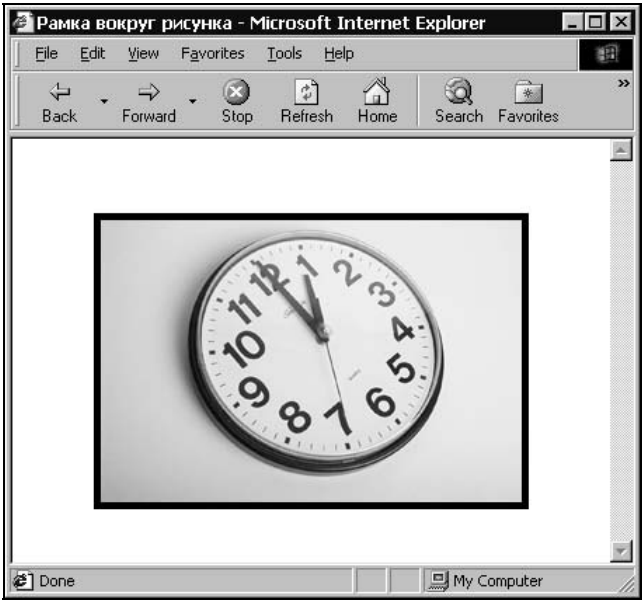


Рис. 5.1. Пример рамки вокруг рисунка

Если графическое изображение является гиперссылкой, браузер автоматически отобразит вокруг рисунка рамку толщиной в 1 пиксел (кроме этого, некоторые браузеры делают рамку определенного цвета, обычно — синего).

Поэтому, если никакой необходимости в рамке вокруг графического указателя ссылки нет, следует дать браузеру соответствующую инструкцию:

```
<A HREF="clock.html"><IMG SRC="images/clock.gif" BORDER="0"></A>
```

Параметры *WIDTH* и *HEIGHT*. По аналогии с другими HTML-элементами (табличные ячейки, горизонтальные разделители), параметры `WIDTH` и `HEIGHT` используются для определения размеров графического изображения, встраиваемого в электронный документ (ширина и высота, в пикселах):

```
<IMG SRC="images/clock.gif" BORDER="1" WIDTH="300" HEIGHT="201">
```

Использование этих параметров необязательно, однако, рекомендуется по двум основным причинам:

1. Во время загрузки изображения браузер сразу зарезервирует на странице столько места, сколько необходимо для отображения рисунка. Многие разработчики пренебрегают этим правилом, в итоге при загрузке изначально под рисунок занимается слишком мало места, а потом страница начинает "скакать" (т. к. браузер одновременно подставляет под маленькое пространство реальные размеры файла).
2. Иногда размеры изображения слишком велики, чтобы рисунок поместился в окне браузера без полос прокруток. Для этого в HTML-коде указываются уменьшенные пропорционально размеры рисунка (например, нажав на уменьшенный рисунок по ссылке, изображение откроется, что называется, "в полный рост").

Размещая на странице уменьшенный вариант изображения, не забывайте обязательно пропорционально уменьшать рисунок. В противном случае, при нарушении соотношения "ширина — высота" рисунок получится искаженным и неприглядным (рис. 5.2).

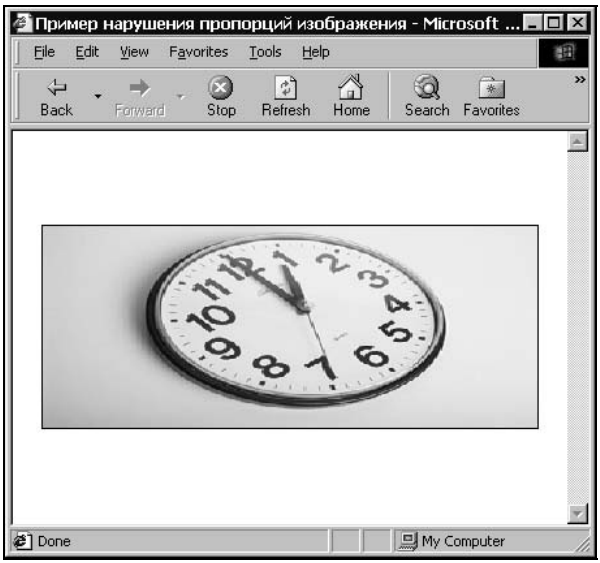


Рис. 5.2. Пример нарушения пропорций изображения

Корректно изменить пропорции рисунка можно следующими способами:

- указать в теге <IMG> измененным только один из параметров (либо WIDTH, либо HEIGHT). Браузер самостоятельно изменит второй параметр, сохраняя пропорцию;
- открыть рисунок в любом растровом редакторе (например, Adobe Photoshop) и изменить пропорции программным способом.

Что касается пропорционального увеличения графических изображений в HTML-документах, то такой подход не рекомендуется по причине особенностей самого растрового формата (ухудшение качества изображения при увеличении его размеров). Однако, если степень увеличения незначительна (в среднем, на 5—10 пикселов), скорее всего, визуально качество останется неизменным.

В завершение разговора о параметрах WIDTH и HEIGHT следует сказать, что возможно указание размеров графического изображения в процентах. Такой формат записи предписывает браузеру растянуть или сузить рисунок в соответствии с размерами окна страницы. Такой подход не рекомендован, и может применяться только в исключительных случаях и только на примере однородных рисунков (некоторые браузеры либо не поддерживают такое обозначение размеров, либо выводят их некорректно).

Параметр *ALIGN*. Параметр ALIGN предназначен для определения типа выравнивания рисунка относительно текста, а также прочих элементов электронного документа.

Существующие значения параметра ALIGN представлены в табл. 5.1.

Таблица 5.1. *Значения параметра ALIGN*

Значение	Функция
LEFT	Текст обтекает рисунок по правому краю (изображение слева)
RIGHT	Текст обтекает рисунок по левому краю (изображение справа)
TOP	Верхний край рисунка выравнивается по самому высокому элементу строки
TEXTTOP	Верхний край рисунка выравнивается по самому высокому текстовому элементу строки
MIDDLE	Выравнивание середины рисунка по базовой линии строки
ABSMIDDLE	Выравнивание середины рисунка по середине строки
BOTTOM	Выравнивание нижнего края рисунка по базовой линии строки

Таблица 5.1 *(окончание)*

Значение	Функция
BASELINE	Выравнивание нижнего края рисунка по базовой линии строки
ABSBOTTOM	Выравнивание нижнего края рисунка по нижнему краю строки

Отметим, что некоторые популярные браузеры (например, Netscape) интерпретируют значения ABSMIDDLE и ABSBOTTOM как, соответственно, MIDDLE и BOTTOM. Internet Explorer корректно отображает все приведенные в табл. 5.1 значения.

Основное различие между значениями нижнего края и базовой линии заключается в особенностях некоторых букв алфавита и прочих символов: нижний край — это крайняя точка текстового блока, а базовая линия проходит по нижней части строки текста без учета фрагментов текста, выходящих за строку. Другими словами, при наличии в тексте букв "y, "p", "q", "g" и т. п. нижняя часть этих символов будет определяться значением ABSBOTTOM, а часть букв на уровне запятой в предложении — значением BOTTOM или BASELINE.

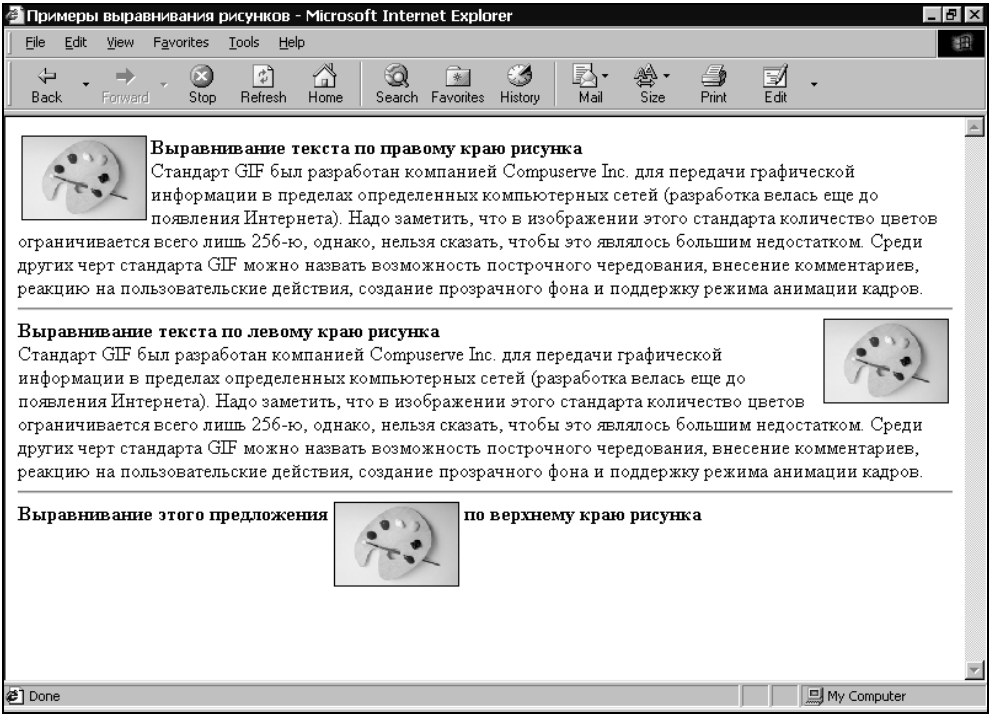


Рис. 5.3. Примеры выравнивания рисунков

Листинг 5.1. Примеры выравнивания рисунков

```
<HTML>
  <HEAD>
    <TITLE>Примеры выравнивания рисунков</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<IMG SRC="brush.jpg" ALIGN="left" BORDER="1" WIDTH="100"
HEIGHT="67"><B>Выравнивание текста по правому краю
рисунка</B><BR>Стандарт GIF был разработан компанией Compuserve Inc. для
передачи графической информации в пределах определенных компьютерных
сетей (разработка велась еще до появления Интернета). Надо заметить, что
в изображении этого стандарта количество цветов ограничивается всего лишь
256-ю, но нельзя сказать, чтобы это являлось большим недостатком.
Среди других черт стандарта GIF можно назвать возможность построчного
чередования, внесение комментариев, реакцию на пользовательские действия,
создание прозрачного фона и поддержку режима анимации кадров.
<HR>
<IMG SRC="brush.jpg" ALIGN="right" BORDER="1" WIDTH="100"
HEIGHT="67"><B>Выравнивание текста по левому краю рисунка</B><BR>Стандарт
GIF был разработан компанией Compuserve Inc. для передачи графической
информации в пределах определенных компьютерных сетей (разработка велась
еще до появления Интернета). Надо заметить, что в изображении этого
стандарта количество цветов ограничивается всего лишь 256-ю, однако,
нельзя сказать, чтобы это являлось большим недостатком. Среди других черт
стандарта GIF можно назвать возможность построчного чередования, внесение
комментариев, реакцию на пользовательские действия, создание прозрачного
фона и поддержку режима анимации кадров.
<HR>
<B>Выравнивание этого предложения <IMG SRC="brush.jpg" ALIGN="top"
BORDER="1" WIDTH="100" HEIGHT="67"> по верхнему краю рисунка</B>
</BODY>
</HTML>
```

Как видно из примеров, все типы выравнивания условно можно разделить на две группы:

- выравнивание относительно края окна документа (значения `LEFT` и `RIGHT`);
- выравнивание относительно текущей строки текста (все остальные значения).

Параметры *HSPACE* и *VSPACE* — определяют размер горизонтального и вертикального отступов от встраиваемого изображения до текста (листинг 5.2, рис. 5.4). Формат записи значений — в пикселах. По умолчанию, значения обоих типов отступа равно нулю.

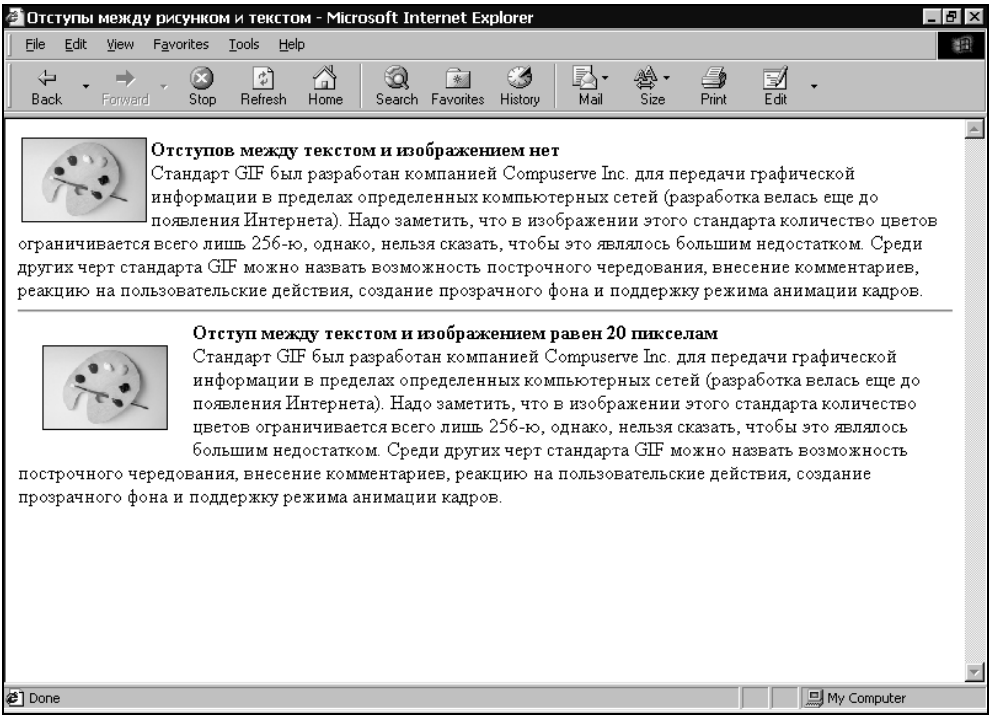


Рис. 5.4. Отступы между рисунком и текстом

Листинг 5.2. Отступы между рисунком и текстом

```
<HTML>
  <HEAD>
    <TITLE>Отступы между рисунком и текстом</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<IMG SRC="brush.jpg" ALIGN="left" BORDER="1" WIDTH="100"
HEIGHT="67"><B>Отступов между текстом и изображением нет</B><BR>Стандарт
GIF был разработан компанией Compuserve Inc. для передачи графической
информации в пределах определенных компьютерных сетей (разработка велась
еще до появления Интернета). Надо заметить, что в изображении этого
стандарта количество цветов ограничивается всего лишь 256-ю, однако,
нельзя сказать, чтобы это являлось большим недостатком. Среди других черт
стандарта GIF можно назвать возможность построчного чередования, внесение
комментариев, реакцию на пользовательские действия, создание прозрачного
фона и поддержку режима анимации кадров.
<HR>
```

```
<IMG SRC="brush.jpg" ALIGN="left" BORDER="1" WIDTH="100" HEIGHT="67"
HSPACE="20" VSPACE="20"><B>Отступ между текстом и изображением равен
20 пикселам</B><BR>Стандарт GIF был разработан компанией Compuserve Inc.
для передачи графической информации в пределах определенных компьютерных
сетей (разработка велась еще до появления Интернета). Надо заметить, что
в изображении этого стандарта количество цветов ограничивается всего лишь
256-ю, однако, нельзя сказать, чтобы это являлось большим недостатком.
Среди других черт стандарта GIF можно назвать возможность построчного
чередования, внесение комментариев, реакцию на пользовательские действия,
создание прозрачного фона и поддержку режима анимации кадров.
</BODY>
</HTML>
```

Необходимо обратить внимание на то, что значение, указанное для параметров `HSPACE` и `VSPACE`, устанавливается с обеих сторон графического изображения.

Параметр *ALT*. Параметр альтернативного текста `ALT` играет существенную роль в процессе встраивания графики в электронные документы.

Изначально его предназначение заключалось в возможности предоставить пользователям неграфических браузеров информацию о рисунке, которая появляется при наведении на графический объект курсора мыши. Однако сегодня понятие "неграфического браузера" стало неактуально, поэтому функция параметра `ALT` сильно расширилась и повсеместно используется для вывода смысловых подсказок для значимых графических элементов. Для не имеющих какой-либо смысловой нагрузки рисунков альтернативный текст указывать не следует.

Например, для рисунка "`services.gif`", демонстрирующего образ услуг коммерческой компании, будет уместным добавить альтернативный текст — "Наши услуги". Для прозрачной распорки, размещенной в пустой табличной ячейке, совершенно бессмысленно писать что-то вроде "Прозрачная графическая распорка".

Для незначимых графических изображений обычно задается следующая конструкция:

```
<IMG SRC="images/spacer.gif" WIDTH="5" HEIGHT="5" ALT="">
```

Таким образом, даже если ваш HTML-документ будут просматривать в текстовом браузере (к примеру, Lynx), композиция страницы сохранится, а пользователю будет выведена информация о наличии рисунка.

Указывать альтернативный текст `ALT` рекомендуется, если изображение по каким-либо причинам не загрузилось (рис. 5.5). Подсказка, отображенная на месте рисунка, даст пользователю хотя бы минимальную информацию о незагруженном объекте.

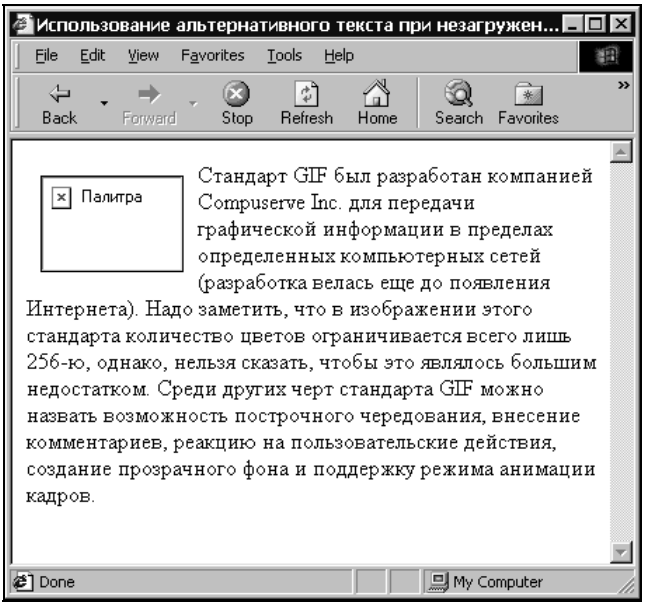


Рис. 5.5. Использование альтернативного текста при незагруженном изображении

Параметр *TITLE*. Данный параметр, по сути, выполняет функцию, аналогичную `ALT`, и предназначен для отображения подсказок к рисункам. Различие в том, что при незагруженном изображении параметр `TITLE` не предоставит пользователю никакой информации — для этого существует параметр альтернативного текста `ALT`.

Следует отметить, что последние версии некоторых браузеров (Opera, Netscape) не отображают значение поля `ALT` при наведении на изображение курсора мыши.

Для закрепления материала все рассмотренные параметры тега `<IMG>` приведены в табл. 5.2.

Таблица 5.2. *Параметры тега* `<IMG>`

Параметр	Функция
`SRC`	Указание пути к встраиваемому изображению
`BORDER`	Указание рамки рисунка
`WIDTH`	Определение ширины изображения
`HEIGHT`	Определение высоты изображения
`ALIGN`	Указание типа выравнивания рисунка

Таблица 5.2 *(окончание)*

Параметр	Функция
HSPACE	Определение горизонтального отступа между рисунком и текстом
VSPACE	Определение вертикального отступа между рисунком и текстом
ALT	Указание альтернативного текста
TITLE	Создание информационной подсказки к изображению

Карты-изображения

На многих HTML-документах сегодня с успехом используются так называемые *карты-изображения* (Imagemaps), которые представляют собой обычные графические файлы (как правило, стандарта GIF или JPEG) с привязанными к различным областям этого изображения гиперссылками.

Такие области описываются специальными координатами, в соответствии с которыми браузер переносит пользователя на нужную страницу.

Правила конфигурации карт-изображений мы затронем чуть позже, а сейчас рассмотрим основные преимущества и недостатки карт-изображений как средства навигации (перемещения) по HTML-документам.

Преимущества и недостатки карт-изображений

К основным преимуществам карт-изображений можно отнести следующие:

1. Более удобного средства, чем карта-изображение, для создания сложных навигационных меню (в особенности географических, топографических и прочих карт) не найти. Процесс создания и пространственного размещения на странице нескольких десятков кнопок для обозначения, например, всех областей Российской Федерации, чрезвычайно сложен и потребует больших временных затрат.
2. Использование карты-изображения в качестве навигационных меню на каждой странице интернет-проекта может существенно сократить время загрузки электронных документов и сэкономить место на Web-сервере.
3. Для использования карты-изображения потребуется изготовить всего один рисунок.
4. Использование карт-изображений позволит разработчику HTML-документов реализовать самые смелые дизайнерские задумки: можно создавать графические объекты абсолютно любой сложности и формы, не волнуясь об их пространственном размещении на странице — это придаст интер-

нет-ресурсу оригинальность и сделает его более запоминающимся для посетителей.

Однако без некоторых недостатков также не обошлось:

1. Графические заготовки для карт-изображений, имеющие большой размер файла, могут заметно увеличить время загрузки электронных документов по сравнению со страницами с обычными текстовыми ссылками.
2. Для посетителей, которые экономят свое время пребывания в Интернете, переход к HTML-документам по ссылкам, указанным в конфигурации карты-изображения, не позволяет отслеживать страницы, на которых они уже были, т. к. гиперссылки карт-изображений не могут менять цвета после посещения их пользователем.
3. При наведении курсора мыши на определенную активную область карты-изображения в серверном варианте, в статусной строке браузера отображается не адрес электронного документа, а координаты области, которые хранятся в конфигурационном файле на самом Web-сервере.
4. Если параллельно с картой-изображением не предусмотрено дублирующее текстовое меню, то посетители, которые по каким-либо причинам не могут загрузить графику или отключили ее в своем браузере, останутся не у дел.

Область применения

Карты-изображения можно применять в самых разнообразных областях компьютерных технологий. Наиболее распространенными из них являются:

- геоинформационные и картографические системы;
- баннерные рекламные сети и системы электронной коммерции;
- электронный и сотовый банкинг, платежные системы;
- игровые трехмерные и двумерные интернет-ресурсы;
- корпоративные серверы;
- интернет-ресурсы широкого профиля.

Как видно из списка, диапазон применения карт-изображений может охватывать практически все отрасли современных технологий.

Конфигурация карт-изображений

Прежде всего следует сказать, что карта-изображение не является принципиально новой технологией. Однако раньше их применение было осложнено рядом причин.

1. Ранние версии некоторых браузеров (например, Netscape Navigator 1.x) не поддерживали технологию карт-изображений.
2. Часть Web-серверов не позволяли использовать графику для создания карт-изображений вообще или предоставляли такую возможность только лишь профессиональным разработчикам или крупным компаниям.
3. Раньше был возможен единственный вариант реализации карт-изображений — серверный, который подразумевал управление переходом на соответствующий документ после нажатия на конкретной области изображения самим сервером.
4. Сегодня существует клиентский вариант карт-изображений, что во многом облегчает применение данной технологии, которую поддерживает большинство современных браузеров.

Сама карта-изображение представляет собой обыкновенное графическое изображение, а ее конфигурация определяется в виде значений координатных кривых, которые прописываются в HTML-коде. Эти значения описывают активные области изображения и содержат информацию о том, куда следует перейти браузеру после нажатия на одну или несколько из этих областей.

Активные области могут иметь форму прямоугольника, круга и многоугольника. Также позволительно комбинировать эти три варианта или определять область, расположенную вне активных областей карты-изображения.

Типы карт-изображений

Как уже было сказано выше, карты-изображения могут быть реализованы в двух вариантах: серверном и клиентском. Далее подробно рассмотрим каждый из них.

Серверный вариант. Этот вариант реализации карты-изображения (Server-Side Imagemap) подразумевает, что документ, в котором прописаны координаты областей, находится на самом Web-сервере, который обязательно должен поддерживать выполнение CGI-сценариев.

Процесс перехода по гиперссылкам на карте-изображении осуществляется следующим образом. При нажатии на одной из активных областей браузером передаются координаты на сервер, который обращается к специальному конфигурационному файлу. Результат обработки данных возвращается назад браузеру в виде адреса, соответствующего данной активной области, и он загружается в окне обозревателя.

Если конфигурационный файл не содержит данных об обозначенной активной области, выводится сообщение об отсутствии данного документа. Для того чтобы браузер определил графический объект как карту-изображение,

необходимо в теге `<IMG>`, описывающем конфигурацию карты-изображения, указать параметр `ISMAP`, а файл конфигурации сохранить в формате MAP.

Серверный вариант поддерживает два формата реализации карт-изображений — CERN и NCSA.

Первый разработан научным центром European Organization for Nuclear Research и предлагает следующую запись координат активных областей:

```
тип_области координаты адрес
```

Значения пар координат разделяются запятой и заключаются в круглые скобки, например:

```
rect (54, 127)(45, 344) http://www.site.ru/
```

В начале конфигурации стоит значение `rect` (прямоугольное выделение для задания активной области). Другими значениями могут быть: `circle` (круг), `poly` (многоугольник) и `default` (значение по умолчанию).

Формат NCSA разработан центром приложений для суперкомпьютеров National Center for Supercomputing Applications и предлагает несколько иной формат записи конфигурации карт-изображений:

```
тип_области адрес координаты
```

Координаты X, Y тоже разделяются запятыми, но в скобки не заключаются, например:

```
rect http://www.site.ru/ 54, 127  45, 344
```

Кроме типов областей, предложенных CERN, данный формат разрешает использование типа `point` (после нажатия активизируется ссылка, обозначенная ближе всего к точке соприкосновения).

Клиентский вариант. Позволяет разместить все данные об активных областях карты-изображения в самом HTML-документе (рис. 5.6). В этом случае количество обращений к серверу сильно сокращается, а конфигурировать саму карту-изображение можно параллельно с изменением HTML-кода.

Размещать файл конфигурации на Web-сервере не нужно: координаты активных областей указываются в самом документе, куда встроен графический объект для карты-изображения. Используя клиентский вариант (Client-Side Imagemap), в тег `<IMG>` добавляется параметр `USEMAP`.

В связи с тем, что клиентский вариант на сегодня распространен более широко и к тому же более доступен, подробно рассмотрим процесс конфигурации карты-изображения на примере Client-Side Imagemap.

Для определения конфигурации активных областей карты-изображения используется специальный тег-контейнер `<MAP>` (он требует наличие закрываю-

щего тега) с параметром NAME (листинг 5.3), который должен соответствовать свойству параметра USEMAP в теге <IMG>.

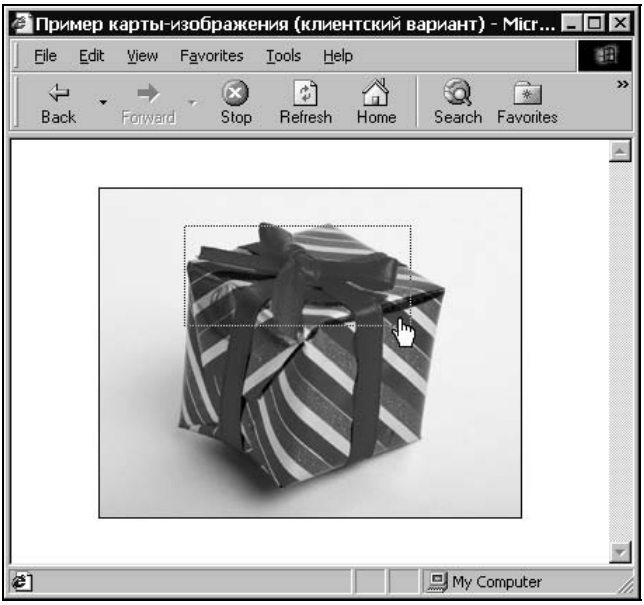


Рис. 5.6. Пример карты-изображения (клиентский вариант)

Необходимо отметить, что описание карты-изображения следует сразу же после указания тега рисунка <IMG>:

```
<IMG SRC="picture.gif" USEMAP="#mymap">
<MAP NAME="mymap">
Координаты активных областей...
</MAP>
```

Описание активных областей карты-изображения осуществляется с помощью тега <AREA>, не требующего за собой закрывающего тега.

Тег <AREA> может включать следующие параметры:

- SHAPE — определяет форму активной области (значения такие же, как в формате CERN серверного варианта);
- COORDS — осуществляет выбор конкретной активной области и содержит значения пар координат. Начало координат размещается в верхнем левом углу графического изображения, которому соответствует начальное значение 0, 0;
- TARGET — используется при работе с фреймами (рассмотрению фреймов посвящена *глава 6*);

- ALT — действие параметра аналогично использованию в обычных гиперссылках на основе графических указателей (альтернативный текст);
- HREF — наличие гиперссылки для данной области;
- NOHREF — отсутствие гиперссылки для данной области.

Листинг 5.3. Пример карты-изображения (клиентский вариант)

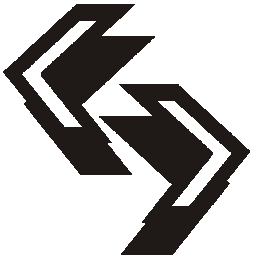
```
<HTML>
  <HEAD>
    <TITLE>Пример карты-изображения (клиентский вариант)</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<IMG SRC="box.jpg" BORDER="1" WIDTH="300" HEIGHT="234" ALT=""
USEMAP="#gift">
<MAP NAME="gift">
<AREA SHAPE="rect" COORDS="60, 26, 222, 98" HREF="bantik.html"
ALT="Бантик">
<AREA SHAPE="rect" COORDS="63, 88, 135, 188" HREF="left.html" ALT="Левая
сторона">
<AREA SHAPE="rect" COORDS="151, 102, 225, 200" HREF="right.html"
ALT="Правая сторона">
</MAP>
</BODY>
</HTML>
```

На рис. 5.6 видна выделенная по нажатию активная область карты-изображения, ведущая по гиперссылке на документ "bantik.html".

В завершение следует добавить, что возможно совмещение серверного и клиентского вариантов карты-изображения. В этом случае в теге рисунка <IMG> необходимо указывать оба параметра: ISMAP и USEMAP. При этом следует помнить, что параметр USEMAP будет доминирующим.

Применять такой комбинированный вариант можно во избежание трудностей с функционированием карт-изображений из-за отсутствия поддержки данной технологии некоторыми браузерами (хотя таких можно насчитать единицы). Если браузер не поддерживает Client-Side Imagemap, то параметр USEMAP будет проигнорирован, и реализуется серверный вариант при наличии конфигурационного файла на сервере и поддержке последним CGI.

ГЛАВА 6



HTML-документы на основе фреймов

Понятие фрейма

Фрейм (от англ. *frame* — рамка, каркас, кадр) представляет собой отдельное рабочее окно браузера, в ряде случаев разделенное еще на несколько различных по параметрам и размеру фреймам. Совокупность таких окон принято называть фреймовой структурой.

Чаще всего такие структуры называют просто "окнами" (или же в соответствии с "родным" названием — фреймы). Фреймовая структура позволяет разбивать основную область на сколько угодно составляющих подфреймов (подобластей), причем по необходимости определяя внутреннее поведение подфреймов.

Каждый фрейм — это не что иное, как отдельный HTML-документ, загружающийся браузером в предназначенном для него окне. Таким образом, HTML-документ, созданный на фреймовой основе, является набором взаимосвязанных электронных документов, параметры и свойства которых определяются настройками всей фреймовой структуры.

Впервые возможность отображения фреймов была реализована компанией Netscape в одноименном браузере версии 2.0. Но ничего особенного первоначальный вариант поддержки подобной структуры своим пользователям не предлагал. Однако уже с выходом версии 3.0 в браузере Netscape появился ряд нововведений, расширивших диапазон параметров основных тегов описания фреймов.

Компания Microsoft тоже не тратила время впустую и, решив потеснить Netscape на перспективном рынке браузеров, выпустила летом 1996 года свой новый продукт — Microsoft Internet Explorer 3.0, который поддерживал практически все основные расширения Netscape и, к тому же, предполагал реализацию так называемых "плавающих" фреймов.

Далее история развивалась таким образом: и та и другая компания старалась покорить пользователя посредством расширения возможностей фреймовых структур, которые поддерживались только собственными браузерами. К сожалению, на сегодняшний день ситуация не изменилась настолько, чтобы можно на 100% быть уверенным в том, что электронный документ, созданный с применением многооконной (фреймовой) структуры, одинаково корректно отобразится в браузерах обеих компаний.

Тем не менее, сегодня фреймы получили достаточное распространение и являются еще одним вариантом представления системы навигации интернет-ресурса.

Области применения фреймов

Диапазон применения фреймов не настолько широк, как, например, у карт-изображений или меню навигации в виде обыкновенных текстовых гиперссылок. Чаще всего прибегают к фреймовой структуре в следующих случаях:

- создание неподвижной или прокручиваемой навигационной панели управления;
- одновременное отображение информации в нескольких местах;
- постоянное визуальное присутствие определенного текстового, графического или иного объекта;
- разработка Web-интерфейса для онлайн-игр.

Далее рассмотрим каждую из названных областей применения.

Панель навигации

Независимо от типа интернет-ресурса (корпоративный сервер, персональная страница, общественная организация и т. д.), в последнее время стало весьма популярным применение фреймов (рис. 6.1) для создания привлекательных и удобных навигационных панелей (меню), которые могут быть статичными, т. е. не зависящими от действий посетителя, и динамическими, когда какое-либо действие со стороны пользователя — нажатие кнопкой мыши, комбинации клавиш и пр. — приводит к различным по виду и масштабу изменениям (начиная с замены внешнего вида навигационных кнопок — эффект RollOver/MouseOver — и заканчивая исполнением сложных программных сценариев).

Такая реализация фреймов удобна прежде всего тем, что она позволяет посетителю перемещаться по сайту и независимо от его месторасположения (двадцать страниц он пролистал или всего одну) постоянно иметь перед глазами панель управления с перечнем всех или основных разделов сайта.

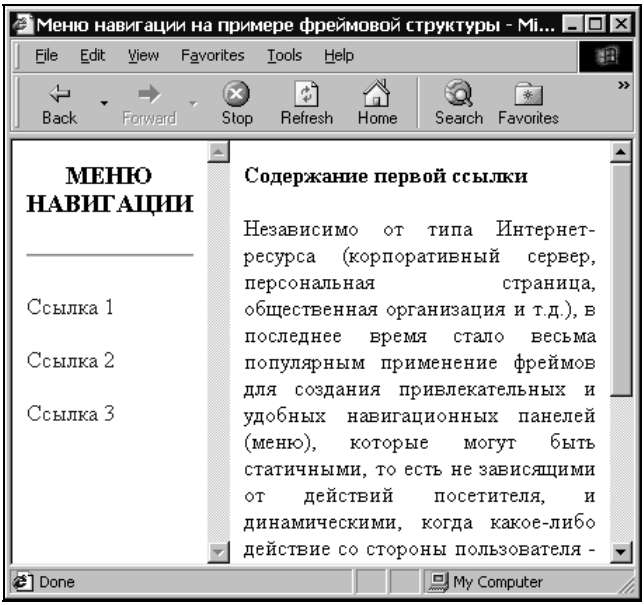


Рис. 6.1. Меню навигации на примере фреймовой структуры

Реализовать такую панель управления можно разными способами:

1. Обозначить только основные разделы Web-сайта в главном фрейме, а подразделы открывать в другом окне.
2. Разместить в главном фрейме ссылки на документы подразделов: по нажатию на ссылку конкретного подраздела главный фрейм перегружается и открывается расширенный вариант меню.
3. Расположить в главном меню динамическое меню в виде раскрывающегося списка, по нажатию на пункты которого во втором фрейме будет открываться нужный документ.

Одновременное отображение информации

Иногда бывает необходимо иметь перед глазами одновременно несколько текстовых или иных информационных блоков: сравнительные характеристики котировок акций (покупка-продажа), структура книги (главы, разделы, подразделы, части и соответствующее им содержание), описание товара в электронном магазине (характеристики товара, цена, достоинства, сферы и правила применения, порядок эксплуатации в одном окне, вид товара во всех ракурсах — в другом окне) и др.

Согласитесь, что было бы весьма неудобно и просто непрактично размещать раздельно документы с данными о покупке и документы с информацией

о продаже акций; публиковать электронную версию книги без подробного описания ее иерархии (где вы видели обычные бумажные книги без оглавления?); растягивать знакомство с товаром на продолжительное время, подразумевающее переход с одной страницы (название и фирма-производитель) к другой (изображение товара), тем более, что подобный переход может включать еще несколько стадий, что чревато ослаблением внимания посетителя как потенциального покупателя и потерей интереса к товару.

Постоянное визуальное присутствие объекта

Иногда разработчики HTML-документов ставят перед собой задачу размещения конкретного объекта на странице таким образом, чтобы он был хорошо виден посетителям независимо от их действий (конечно, в пределах данного Web-сайта).

То есть, что бы ни делал посетитель: переходил из одного раздела в другой, оставлял сообщение в гостевой книге, копировал интересный материал, просматривал документы и т. д., объект всегда будет доступен в визуальном плане.

В качестве такого объекта могут выступать фирменные логотипы, эмблемы, фотографии, рекламные баннеры, навигационные меню, надписи, рубрикаторы и многое другое.

Что касается навигационных меню и баннеров, то возможно создание таковых при помощи технологий Dynamic HTML и JavaScript, что позволяет разработчику таким образом определить характеристики того же меню, что оно всегда будет видно пользователю: например, пользователь немного прокрутил окно вниз — меню автоматически тоже опускается вниз, документ прокручивается вверх — движение меню полностью повторяет поведение страницы.

Однако не следует забывать, что многих такое "дрейфующее" положение рекламы и меню сильно раздражает, а порой мешает рассмотреть некоторые элементы на самой Web-странице. Аналогом, хотя не и динамическим, являются фреймы.

Web-интерфейс для онлайн-игр

Наконец, фреймы применяются не только на познавательно-информационных сайтах и коммерческих серверах. С их помощью можно создавать Web-интерфейсы для самых разнообразных игр в режиме онлайн.

Преимущества и недостатки фреймов

Само собой разумеется, фреймовая структура, как и любое другое технологическое решение, имеет своих противников и сторонников. Кому-то фреймы

не нравятся исключительно из визуальных соображений (нажимаешь в одном месте, а меняется все в другом), другим — по ряду технических причин.

Ниже приведены основные преимущества и недостатки применения фреймовых структур.

Начнем, как водится, с положительных сторон:

1. Фреймы позволяют экономить на объеме пересылаемых пользователю файлов, т. к. после активизации ссылки изменяется только один фрейм.
2. Возможность перехода по другим ссылкам в пределах интернет-ресурса при неизменном доступе к панели навигации заметно облегчает навигацию по электронным документам.
3. Возможность работы сразу с несколькими информационными блоками в пределах одного окна позволяет сэкономить время.
4. Использование правил описания фреймовых структур позволяет разработчику HTML-документов как угодно варьировать размеры полей фреймов, что дает более широкий спектр возможностей пространственного размещения объектов.

А теперь — недостатки фреймов:

1. Некоторые поисковые механизмы не в состоянии индексировать документы с фреймовой структурой или делают это не совсем корректно, что приводит к индексированию одного из составляющих фреймов в качестве начального, а не "родительского" фрейма, как требуется.
2. Компоновка страниц с фреймами часто выглядит неуклюже и грубо, многие не понимают, куда можно перейти по ссылке, чтобы загрузилась нужная информация.
3. На Web-сайтах с использованием фреймовых структур при перемещении "назад-вперед" (с помощью соответствующих клавиш клавиатуры или кнопок браузера) в адресной строке ничего не меняется — там постоянно указан начальный адрес сайта.
4. Некоторые малоизвестные браузеры (а также ранние версии популярных) при попытке перейти к предыдущему документу, который вы просматривали, возвращаются в начало Web-сайта. То же происходит, если попробовать обновить страницу с фреймовой структурой.

Правила описания фреймов

Для описания фреймовых структур в HTML предусмотрен ряд специальных тегов и параметров, которые будут рассмотрены ниже.

Тег <FRAMESET>

Тег <FRAMESET> является основным тегом-контейнером фреймовой структуры HTML-документа (закрывающий тег обязателен). Особенностью любого электронного документа с фреймами является то, что в HTML-коде отсутствует тег-контейнер <BODY>: <FRAMESET> вставляется сразу же после раздела HEAD.

Структурный HTML-документ (тот, который определяет структуру фреймов) не может содержать никаких тегов форматирования или любых HTML-элементов.

Тег <FRAMESET> может содержать два основных параметра — ROWS и COLS.

Параметры ROWS и COLS определяют количество горизонтальных (ROWS) и/или вертикальных (COLS) фреймов. Формат записи значения может быть в пикселах, процентах или относительных единицах. Причем число значений соответствует количеству фреймов.

Значения записываются через запятую, например:

```
<FRAMESET ROWS="30%, 70%">
```

В данном примере мы имеем два горизонтальных фрейма, один из которых (верхний) занимает 30% рабочей области окна браузера, второй (нижний) — 70% (общая сумма должна всегда составлять 100%).

Запись значений в пикселах не очень удобна ввиду того, что разработчик фреймовой структуры (да и вообще всего HTML-документа) не знает, на каком мониторе и при каком экранном разрешении будет просматриваться его документ. В этом смысле оптимально подходит указание значений в процентном соотношении: при изменении размеров окна браузера размеры фреймов будут пропорционально меняться.

Формат записи значений в относительных единицах выглядит так:

```
<FRAMESET COLS="*, 2*, 3*">
```

Символ "звездочки" (*) является одной частью целого числа и осуществляет пропорциональное разделение окна браузера на указанное количество фреймов (в нашем случае — это три вертикальных фрейма). Чтобы определить дробное число, соответствующее размеру конкретного фрейма, нужно сложить все цифры, стоящие перед символом * и подставлять получившуюся сумму в качестве знаменателя от целого числа. Таким образом: 1 + 2 + 3 = 6, т. е. первый фрейм занимает 1/6 окна, второй — 2/6 (1/3) окна и третий — 3/6 (1/2) окна браузера (рис. 6.2). Отсутствие цифры перед символом "звездочки" подразумевает 1.

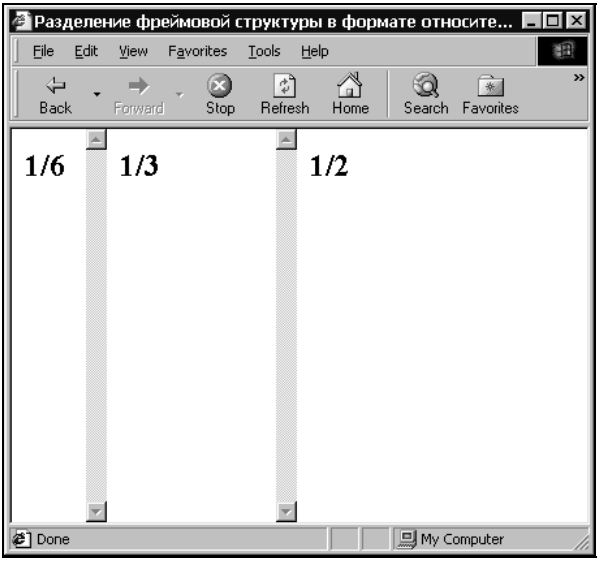


Рис. 6.2. Разделение фреймовой структуры в формате относительных единиц

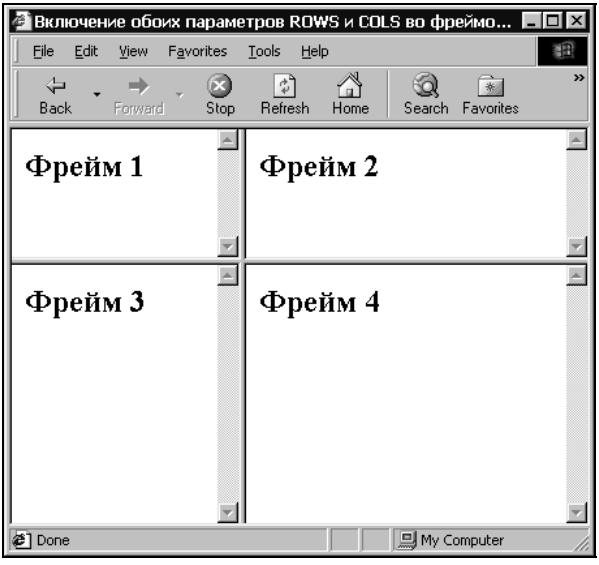


Рис. 6.3. Включение параметров `ROWS` и `COLS` во фреймовую структуру

Язык разметки HTML позволяет для параметров `ROWS` и `COLS` задавать смешанные значения, например:

```
<FRAMESET ROWS="50, 50%, *, 3*">
```

Такая структура включает в себя четыре горизонтальных фрейма: первый строго зафиксирован — 50 пикселов, второй занимает 50% от всего размера окна браузера, два последних фрейма поделят оставшееся пространство в соотношении 1/4 и 3/4.

Обязательного порядка записи смешанных значений не существует, однако рекомендуется в первую очередь указывать фиксированные значения (пикселы), далее желательно должны идти проценты, напоследок — относительные единицы.

Также в тег `<FRAMESET>` можно включать оба параметра: `ROWS` и `COLS` (рис. 6.3):

```
<FRAMESET COLS="40%, 60%" ROWS="*, 2*">
```

Тег *<FRAME>*

Тег `<FRAME>` описывает свойства отдельного фрейма, входящего в состав фреймовой структуры. Соответственно, сколько фреймов — столько и должно быть тегов `<FRAME>`, заключенных внутри тега фреймовой структуры `<FRAMESET>`. `<FRAME>` не требует закрывающего тега и может содержать следующие параметры.

Параметр *SRC*

Параметр `SRC` указывает путь к HTML-документу, который занимает текущий фрейм:

```
<FRAME SRC="frames/menu.html">
```

В данном случае во фрейм браузером будет загружен документ `"menu.html"`. В качестве источника фрейма может быть указан любой допустимый адрес.

Также следует помнить, что свойства документа, который является составным компонентом фреймовой структуры, определяются в HTML-коде этого документа, а не в пределах конструкций `<FRAMESET>` или `<FRAME>`.

Параметр *NAME*

Данный параметр присваивает текущему фрейму внутреннее имя, которое используется для взаимодействия между различными фреймами (подробную информацию *см. в разд. "Взаимодействие между фреймами" этой главы*).

Параметр *SCROLLING*

С помощью параметра `SCROLLING` можно управлять полосой прокрутки в отдельном фрейме. Существует три возможных значения данного параметра:

- `YES` — полоса прокрутки появляется в любом случае (не рекомендуется присваивать это значение, ибо постоянное наличие прокрутки может негативно сказаться на дизайне всего электронного документа);

- NO — запрет полосы прокрутки (прокрутка не появится даже если содержание документа превышает размер видимого пространства окна браузера; в итоге, оставшаяся часть документа будет недоступна пользователю);
- AUTO — автоматическое появление полосы прокрутки по мере необходимости (рекомендуемое значение, регулирующее прокрутку в зависимости от объема информации в текущем фрейме).

Параметр *NORESIZE*

Данный параметр запрещает ручное изменение пользователем размеров текущего фрейма. Такое изменение возможно при наведении курсора мыши на границу фрейма, однако, в ряде случаев разработчику может потребоваться запретить изменение размеров окон фреймовой структуры.

Параметр NORESIZE не требует указания никаких значений. Но следует помнить, что при наличии данного параметра изменить размеры соседних фреймов тоже будет нельзя.

Параметры *MARGINWIDTH* и *MARGINHEIGHT*

Параметры MARGINWIDTH и MARGINHEIGHT предназначены для определения горизонтальных и вертикальных отступов внутри фрейма. Значение указывается в пикселах и является равнозначным для обеих сторон, например:

```
<FRAME MARGINWIDTH="5" MARGINHEIGHT="3">
```

В данном случае горизонтальный отступ будет равен 5 пикселам, а вертикальный — 3.

При указании этих двух параметров необходимо помнить о том, что такие же отступы могут быть определены не только в файле фреймовой структуры, но и непосредственно в HTML-коде документа выбранного фрейма (параметры LEFTMARGIN, RIGHTMARGIN, TOPMARGIN, BOTTOMMARGIN, MARGINWIDTH и MARGINHEIGHT в теге <BODY>).

Тег *<NOFRAMES>*

Возможность работы с фреймами не была предусмотрена спецификацией HTML вплоть до версии 3.2. И это несмотря на то, что многие браузеры поддерживали данную технологию. С принятием HTML 4.0 была введена поддержка фреймовых структур, а Консорциум W3C стал рекомендовать использование тега-контейнера <NOFRAMES> (закрывающий тег обязателен), предоставляющего информацию пользователям браузеров без поддержки фреймов.

```
<NOFRAMES>
Этот документ содержит фреймы.
</NOFRAMES>
```

Текст, заключенный между двумя тегами <NOFRAMES>, будет показан владельцам браузеров без поддержки фреймов — современные браузеры просто проигнорируют эту информацию.

Дополнительные параметры фреймовых структур

К дополнительным параметрам фреймовых структур можно отнести параметры, которые поддерживаются не всеми браузерами.

Это параметры BORDER (толщина границы между фреймами), FRAMEBORDER (наличие границы между фреймами), BORDERCOLOR (цвет границы фрейма) и FRAMESPACING (отступ между соседними фреймами).

В завершение разговора о правилах построения фреймовых структур рассмотрим итоговый пример (листинг 6.1, рис. 6.4).

Листинг 6.1. Пример разветвленной фреймовой структуры

```
<HTML>
  <HEAD>
    <TITLE>Пример разветвленной фреймовой структуры</TITLE>
  </HEAD>
<FRAMESET COLS="30%, 70%" FRAMEBORDER="0" FRAMESPACING="0" BORDER="0">
<FRAME NAME="left" SRC="left.html" SCROLLING="yes" NORESIZE
MARGINWIDTH="10" MARGINHEIGHT="10">
<FRAMESET ROWS="*, 2*" FRAMEBORDER="0" FRAMESPACING="0" BORDER="0">
<FRAME NAME="top" SRC="top.html">
<FRAME NAME="bottom" SRC="bottom.html">
</FRAMESET>
<NOFRAMES>
Ваш браузер не поддерживает фреймы.
</NOFRAMES>
</HTML>
```

Пример описывает фреймовую структуру, состоящую из двух вертикальных и двух горизонтальных фреймов.

Конструкция

```
<FRAMESET COLS="30%, 70%" FRAMEBORDER="0" FRAMESPACING="0" BORDER="0">
```

показывает, что окно браузера поделено на две части: 30% и 70%, соответственно. Следующий за ней тег <FRAME> задает свойства левого фрейма.

Правый фрейм разбит еще на два фрейма, на этот раз горизонтальных, для чего дается отдельная конструкция:

```
<FRAMESET ROWS="*, 2*" FRAMEBORDER="0" FRAMESPACING="0" BORDER="0">
```

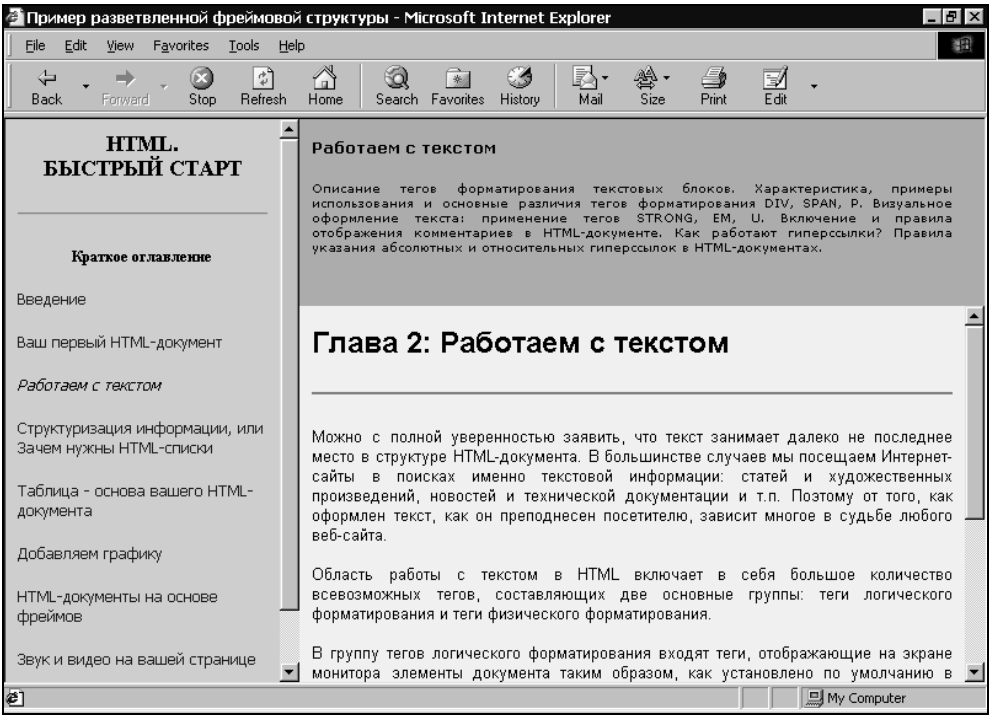


Рис. 6.4. Пример разветвленной фреймовой структуры

После этого идет описание двух упомянутых выше горизонтальных фреймов — посредством тега `<FRAME>`. Завершает код файла фреймовой структуры тег-контейнер `<NOFRAMES>`, предназначенный для браузеров без поддержки фреймов.

Для закрепления пройденного материала основные теги фреймовых структур и их параметры приведены в табл. 6.1.

Таблица 6.1. *Основные теги и параметры фреймовых структур*

Тег	Параметры	Функция
`<FRAMESET>`	COLS ROWS	Основной тег фреймовой структуры
`<FRAME>`	SRC NAME SCROLLING NORESIZE MARGINWIDTH MAGINHEIGHT	Тег описания свойств отдельного фрейма

Таблица 6.1 *(окончание)*

Тег	Параметры	Функция
<NOFRAMES>	—	Тег информации для браузеров без поддержки фреймов

Взаимодействие между фреймами

Работа с фреймовыми структурами имеет специфические особенности, которые следует учитывать при использовании данного средства навигации для HTML-документов.

Взаимодействие между отдельными фреймами происходит посредством загрузки документа по соответствующей ссылке в установленное окно. Реализуется это действие с помощью параметра TARGET тега <A>.

По умолчанию вызванный по гиперссылке документ будет загружаться в текущий фрейм. Но чаще всего этого нужно избежать и сделать так, чтобы страница появлялась в соседнем окне. Существуют специальные зарезервированные имена действий, по которым происходит загрузка документов на узлах с фреймовой структурой: "_blank", "_self", "_parent" и "_top".

Действие

```
<A HREF="file1.html" TARGET="_blank">
```

осуществляет загрузку документа в новое окно без имени, определяемого параметром NAME тега <FRAME>, поэтому этот вариант исключает изменение содержимого созданного окна.

Действие

```
<A HREF="file2.html" TARGET="_self">
```

открывает документ в текущем окне.

Действие

```
<A HREF="file3.html" TARGET="_parent">
```

производит загрузку документа в область, занимаемую "родительским" фреймом текущего фрейма.

Действие

```
<A HREF="file4.html" TARGET="_top">
```

вызывает загрузку документа в полном окне.

В данном случае не был задействован параметр NAME в теге описания текущего фрейма <FRAME>.

Рассмотрим несколько примеров по взаимодействию между фреймами и открытию отдельных окон браузера с помощью параметра TARGET.

Создадим файл со следующей фреймовой структурой:

```
<FRAMESET ROWS="2*, *">
<FRAME NAME="menu" SRC="menu.html" NORESIZE FRAMEBORDER="1">
<FRAMESET COLS="50%, 50%">
<FRAME NAME="left" SRC="left.html">
<FRAME NAME="right" SRC="right.html">
</FRAMESET>
```

Верхний фрейм будет содержать перечень гиперссылок (листинг 6.2), а два нижних фрейма предназначены для открытия в них содержимого конкретной ссылки.

Листинг 6.2. Файл верхнего фрейма "menu.html"

```
<HTML>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#FF0000" ALINK="#00FF00"
VLINK="blue">
<H3>Ссылки верхнего фрейма</H3>
<HR>
<FONT FACE="Tahoma" SIZE="2">
<UL TYPE="square">
<LI><A HREF="text.html" TARGET="left">Файл с текстом в левом нижнем
фрейме</A>
<LI><A HREF="text.html" TARGET="right">Файл с текстом в правом нижнем
фрейме</A>
<LI><A HREF="text.html" TARGET="menu">Файл с текстом в верхнем фрейме</A>
<LI><A HREF="text.html" TARGET="_top">Файл с текстом в полном окне</A>
<LI><A HREF="text.html" TARGET="_blank">Файл с текстом в новом окне</A>
<LI><A HREF="text.html" TARGET="_self">Файл с текстом в текущем
фрейме</A>
</UL>
</FONT>
</BODY>
</HTML>
```

Документ "menu.html" содержит шесть гиперссылок на один и тот же файл — "text.html" (рис. 6.5).

Первая ссылка откроется в левом нижнем фрейме по причине указания конструкции TARGET="left" ("left" в нашем случае — это внутреннее имя данного фрейма).

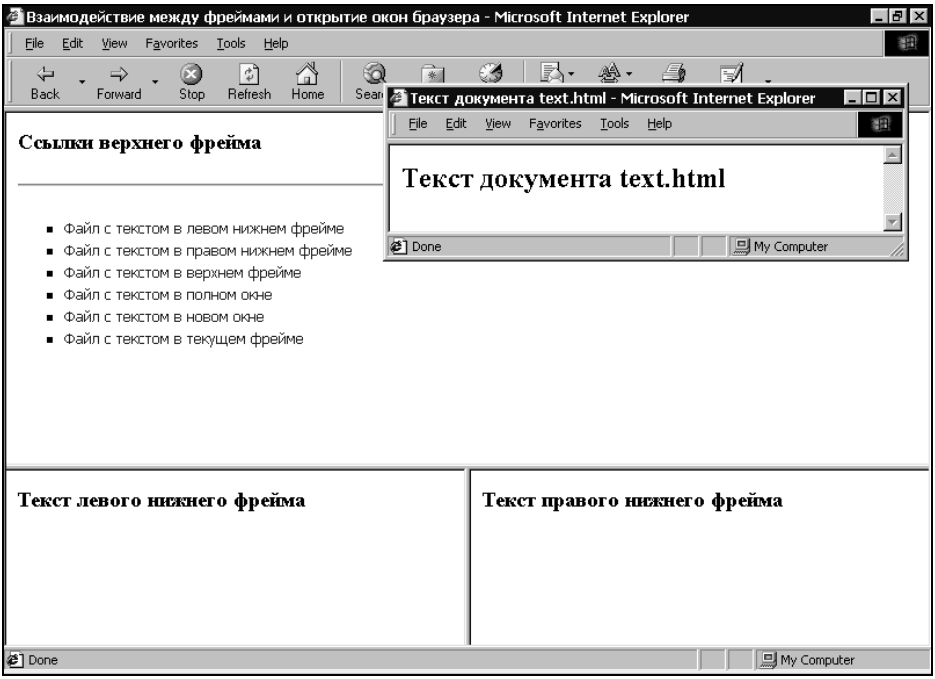


Рис. 6.5. Взаимодействие между фреймами и открытие окон браузера

Вторая ссылка откроется в правом нижнем фрейме из-за указания внутреннего имени фрейма "`right`".

Третья ссылка будет открыта в этом же окне, т. к. "`menu`" — это имя текущего фрейма, из которого открываются гиперссылки.

Четвертая ссылка откроется в полном окне браузера (конструкция "`_top`").

Пятая ссылка будет открыта в отдельном новом окне поверх фреймовой структуры (действие "`_blank`").

Наконец, последняя ссылка откроется в текущем фрейме (аналогично действию `TARGET="menu"`).

Плавающие фреймы

Плавающие фреймы являются стандартом, который поддерживается только браузером Internet Explorer. Netscape подобные фреймы просто-напросто не отображает.

Суть плавающих фреймов заключается в возможности встраивать обыкновенные фреймы (с указанием источника), являющиеся, в сущности, HTML-документами, в другие электронные документы.

Плавающие фреймы (листинг 6.3, рис. 6.6) описываются тегом-контейнером `<IFRAME>` и могут иметь параметры, присущие обыкновенным фреймам, а также параметры, аналогичные описанию графических изображений (а именно: `ALIGN`, `WIDTH`, `HEIGHT`, `HSPACE` и `VSPACE`).

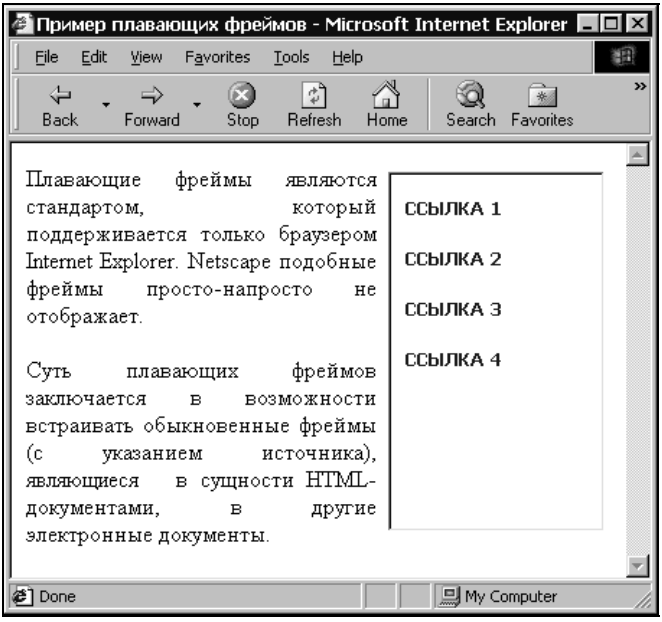


Рис. 6.6. Пример плавающих фреймов

Листинг 6.3. Пример плавающих фреймов

```
<HTML>
<HEAD>
    <TITLE>Пример плавающих фреймов</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#FF0000" ALINK="#00FF00"
VLINK="blue">
<IFRAME SRC="iframe.html" NAME="iframe" WIDTH="150" HEIGHT="250"
HSPACE="5" VSPACE="5" SCROLLING="auto" ALIGN="right">
Ваш браузер не отображает плавающие фреймы.
</IFRAME>
<P ALIGN="justify">
Плавающие фреймы являются стандартом, который поддерживается только
браузером Internet Explorer. Netscape подобные фреймы просто-напросто
не отображает.
<BR><BR>
```

```
Суть плавающих фреймов заключается в возможности встраивать обыкновенные
фреймы (с указанием источника), являющиеся, в сущности, HTML-документами,
в другие электронные документы.
</P>
</BODY>
</HTML>
```

Браузеры, которые не поддерживают плавающие фреймы, в указанном месте документа выведут сообщение "Ваш браузер не отображает плавающие фреймы".

ГЛАВА 7

Добавление аудио

В нашей повседневной жизни звук играет далеко не последнюю роль. Мы слушаем радио и музыку, записанную на CD, пользуемся мелодиями для мобильных телефонов и пейджеров, ежедневно слышим звук из динамиков телевизоров, устанавливаем затейливые мелодии на будильниках, даже поздравительные открытки — и те стали музыкальными.

Развитие компьютерных технологий, в частности Интернета, на сегодняшний день позволяет применять в электронных документах компоненты звука. Существует несколько возможных вариантов воспроизведения звуковых файлов в Интернете, мы же в рамках знакомства с HTML остановимся именно на возможностях этого языка разметки.

Звуковые форматы

Существует огромное количество *звуковых форматов*: распространенных и не особо встречающихся, применяемых в различных операционных системах и для различных нужд.

Нас интересуют те форматы, файлы которых позволительно встраивать в HTML-документы и воспроизводить непосредственно в браузере пользователя или с его помощью. К ним можно отнести:

- WAV — формат звуковых файлов операционной системы Microsoft Windows;
- MIDI — Musical Instrument Digital Interface — формат электронных инструментов;
- AU — формат Sun/NeXT;
- AIFF — формат платформы Macintosh;

- RealAudio — формат передачи звуковых файлов в режиме реального времени.

В последнее время большую популярность получил формат MP3 (формат сжатия аудиосигнала с потерей качества, лежащий за пределами возможностей человеческого слуха). Однако с помощью стандартных средств HTML файлы MP3 встроить в структуру электронного документа нельзя.

Встраивание звуковых файлов в HTML-документ

Для встраивания практически любых звуковых файлов в HTML-документ можно использовать два пути:

1. Создание гиперссылки на музыкальный файл и размещение ее на странице.
2. Применение специального тега вставки музыкальных файлов `<EMBED>`.

Создание гиперссылки на музыкальный файл

Правила записи гиперссылки на файлы музыкальных форматов абсолютно идентичны записи ссылок на HTML-документы или графику:

```
<A HREF="music.wav">Музыкальный файл (30 Кб)</A>
```

При нажатии такой ссылки на компьютере пользователя запускается установленный по умолчанию проигрыватель звуковых файлов.

Следует обратить особое внимание на указание размера музыкального файла, стоящее в ссылке сразу после названия. Дело в том, что некоторые звуковые данные (песни, фрагменты мелодий, фонограммы и пр.) могут иметь весьма большой размер. Пользователь, прежде чем нажать на подобную ссылку, исходя из возможностей своего компьютера и скорости соединения к Интернету, должен иметь хотя бы примерное представление о том, сколько времени займет открытие файла.

Применение тега *<EMBED>*

Тег `<EMBED>` имеет ряд параметров, которые позволяют управлять воспроизведением музыкального файла непосредственно в окне браузера (листинг 7.1, рис. 7.1).

В этом случае браузер отобразит встроенную панель воспроизведения указанного музыкального файла (`"melody.wav"`).

Листинг 7.1. Пример встраивания звукового файла с помощью тега <EMBED>

```
<HTML>
  <HEAD>
    <TITLE>Пример встраивания звукового файла</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#FF0000" ALINK="#00FF00"
VLINK="blue">
<H2 ALIGN="center">Пример встраивания звукового файла</H2>
<CENTER>
<EMBED SRC="melody.wav" WIDTH="287" HEIGHT="213" AUTOSTART="true">
</CENTER>
</BODY>
</HTML>
```

Рис. 7.1. Пример встраивания звукового файла с помощью тега <EMBED>

Параметры тега <EMBED> приведены в табл. 7.1.

Таблица 7.1. *Параметры тега <EMBED>*

Параметр	Описание
SRC	Указание пути к воспроизводимому звуковому файлу форматов AU, AIFF, WAV и MIDI. Является обязательным параметром

Таблица 7.1 *(окончание)*

Параметр	Описание
WIDTH	Ширина панели воспроизведения (обязательный параметр)
HEIGHT	Высота панели воспроизведения (обязательный параметр)
AUTOSTART	Значение по умолчанию — FALSE. Для автоматического начала воспроизведения используется значение TRUE
AUTOLOAD	Значение по умолчанию — TRUE. Для запрета автоматической загрузки файла используется значение FALSE
ALIGN	Тип выравнивания панели воспроизведения относительно текста на странице. Возможные значения: LEFT, RIGHT, CENTER, TOP и BASELINE. Значение по умолчанию — BASELINE
VOLUME	Установка громкости воспроизведения (определяется в процентах)
STARTTIME	Время начала воспроизведения в минутах и секундах (формат записи: "mm:ss"; по умолчанию — 00.00)
ENDTIME	Время окончания воспроизведения в минутах и секундах (формат записи: "mm:ss"; по умолчанию — конец звукового файла)
CONTROLS	Указание элементов управления на панели воспроизведения (значения CONSOLE, SMALLCONSOLE, PLAYBUTTON, PAUSEBUTTON, STOPBUTTON, VOLUMELEVER). Значение по умолчанию — CONSOLE
CONSOLE	Отображение полного набора элементов управления на панели воспроизведения
SMALLCONSOLE	Компактный набор элементов управления на панели воспроизведения
PLAYBUTTON	Отображение только кнопки воспроизведения
PAUSEBUTTON	Отображение только кнопки паузы
STOPBUTTON	Отображение только кнопки остановки воспроизведения (звуковой файл при этом будет выгружен)
VOLUMELEVER	Отображение только регулятора громкости

Особое внимание при встраивании музыкальных файлов в HTML-документы следует обратить на совместимость перечисленных параметров тега <EMBED> с различными браузерами.

Дело в том, что браузер Internet Explorer использует для воспроизведения встроенных звуковых файлов модуль ActiveMovie (проигрыватель Windows Media), а Netscape — LiveAudio. Каждый из этих модулей имеет ряд своих преимуществ, однако, результат встраивания музыкального файла визуально может сильно различаться при просмотре в этих двух браузерах.

RealAudio

Формат RealAudio был разработан компанией RealNetworks в период развития аудио- и видеоконференций в режиме реального времени. Новый формат положил начало возможности передавать звуковые файлы в Интернете в реальном времени.

Технология RealAudio состоит из трех программных компонентов.

1. Кодировщик (конвертирование звуковых файлов в формат RealAudio).
2. Сервер (передача файлов формата RealAudio по сети).
3. Проигрыватель (воспроизведение музыкальных файлов в формате RealAudio).

Для прослушивания RealAudio-файлов достаточно иметь установленный проигрыватель. Для распространения собственных файлов придется обзавестись программой-кодировщиком и сервером.

Для встраивания звукового файла в формате RealAudio на HTML-страницу используются два варианта: посредством ссылки на источник RealAudio-данных и с помощью уже известного нам тега `<EMBED>`.

Ссылка на файл RealAudio выглядит так:

```
pnm://audio.real.com/melody.ra
```

где:

- `pnm://` — указание сервера RealAudio;
- `audio.real.com` — адрес сервера RealAudio;
- `melody.ra` — название звукового файла RealAudio (расширение ra).

Совокупность ссылок такого вида обычно размещают в обыкновенном текстовом файле, содержащем информацию о звуковых данных RealAudio, которому присваивают расширение ram (RealAudio Metafile).

Таким образом, позволительно указание обоих расширений (и ra, и ram), но вариант с ram предоставляет дополнительные возможности по определению точки проигрывания файла и даты его воспроизведения.

```
pnm://audio.real.com/melody.ra$0:30
```

В этом примере файл формата RealAudio будет воспроизводиться с тридцатой секунды (запись типа `$мин:сек`). Дата начала проигрывания определяется конструкцией `$dd:hh:mm:ss:t` (`день:час:минута:секунда:десятая доля секунды`).

Также можно вставить в HTML-документ обыкновенную гиперссылку, при нажатии которой на компьютере пользователя запустится проигрыватель файлов RealAudio (рис. 7.2):

```
<A HREF="audio/melody.ra">Файл в формате RealAudio</A>
```

Рис. 7.2. Окно проигрывателя звуковых файлов RealAudio

При встраивании файлов RealAudio с помощью тега <EMBED> (рис. 7.3 и листинг 7.2) необходимо помнить, что расширение файла должно быть не ram, а rpm (для активизации подключаемого к браузеру модуля воспроизведения звуковых файлов RealAudio).

Рис. 7.3. Встраивание файлов RealAudio в HTML-документ

Листинг 7.2. Встраивание файлов RealAudio в HTML-документ

```
<HTML>
  <HEAD>
    <TITLE>Встраивание файлов RealAudio в HTML-документ</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#FF0000" ALINK="#00FF00"
VLINK="blue">
<H2 ALIGN="center">Встраивание файлов RealAudio</H2>
<CENTER>
<EMBED SRC="audio/melody.rpm" WIDTH="300" HEIGHT="128" AUTOSTART="true"
CONTROLS="All">
</CENTER>
</BODY>
</HTML>
```

Параметры тега `<EMBED>` для воспроизведения звуковых файлов в формате RealAudio перечислены в табл. 7.2.

Таблица 7.2. *Параметры тега* `<EMBED>` *(RealAudio)*

Параметр	Описание
`SRC`	Указание пути к воспроизводимому звуковому файлу формата RealAudio. Является обязательным параметром
`WIDTH`	Ширина панели воспроизведения (обязательный параметр)
`HEIGHT`	Высота панели воспроизведения (обязательный параметр)
`CONTROLS`	Набор элементов управления на панели воспроизведения (полный перечень возможных значений приведен в табл. 7.3)
`AUTOSTART`	Если значение `TRUE`, проигрывание файла начинается автоматически (порядок загрузки и последовательность воспроизведения нескольких файлов в формате RealAudio зависит от настроек браузера и программы-сервера)
`CONSOLE`	Создание нескольких панелей воспроизведения
`NOLABELS`	При значении `TRUE` вывод информации о проигрываемом файле RealAudio (автор, название и пр.) будет запрещен

Перечень возможных значений параметра `CONTROLS` (выбор элементов панели воспроизведения) можно найти в табл. 7.3.

Таблица 7.3. *Значения параметра* `CONTROLS`

Значение	Описание
`All`	Значение по умолчанию: отображаются все элементы управления
`ControlPanel`	Кнопки воспроизведения и остановки, индикатор воспроизведения файла и регулятор громкости
`InfoPanel`	Вывод информации о файле
`InfoVolumePanel`	Вывод информации о файле и регулятор громкости
`StatusBar`	Окно состояния воспроизведения файла, указание времени и продолжительности файла
`PlayButton`	Только кнопка воспроизведения/паузы
`StopButton`	Только кнопка остановки
`VolumeSlider`	Только регулятор громкости
`PositionField`	Время воспроизведения и продолжительность файла в строке состояния
`PositionSlider`	Только индикатор текущего положения файла
`StatusField`	Текстовые сообщения в строке состояния

Фоновый звук

В завершение разговора о музыкальных форматах, используемых в современном Интернете, следует упомянуть о такой возможности языка разметки HTML, как указание фонового звука для электронного документа:

```
<BODY BGSOUND="audio/intro.wav" LOOP="infinite">
```

Данный пример задает для страницы фоновый звук `"intro.wav"` с помощью параметра `BGSOUND`. Конструкция `LOOP="infinite"` дает браузеру команду проигрывать указанный файл бесконечное количество раз. Для ограниченного количества воспроизведения фонового звука в качестве значения необходимо использовать целые числа (1, 2, 3 и т. д.).

ГЛАВА 8

Пользовательские формы

Что такое пользовательские формы

Пользовательская форма — это совокупность стандартных HTML-конструкций ввода текстовой и прочей информации и программы-обработчика этой информации, работающей на Web-сервере. Иными словами, пользовательская форма (или HTML-форма) служит для передачи информационных данных серверу.

Результат конструкций языка разметки HTML интерпретируется браузером, с помощью которого пользователь электронного документа получает информацию. Таким образом, объединив все эти формулировки, можно заключить, что HTML-форма выступает в роли посредника между пользователем и сервером.

Посетитель Web-страницы вводит в HTML-форму определенные данные, которые обрабатываются программой и отсылаются на Web-сервер. Все эти действия укладываются в три стадии.

- Ввод пользователем информации.
- Обработка введенной информации программой, установленной на сервере.
- Результат отправления введенной информации на Web-сервер (открытие нового HTML-документа, переадресация на предыдущую страницу и пр.).

В качестве программы-обработчика чаще всего выступает CGI-сценарий (скрипт, взаимодействующий со специальным компонентом Web-сервера — Common Gateway Interface; обычно разрабатывается на языке Perl или C/C++), а также программы, написанные на основе таких серверных языков программирования, как PHP, ASP, JSP и др.

Значение пользовательских форм трудно переоценить: они являются особым средством HTML, дающим посетителю возможность не только пассивно про-

сматривать информацию, но и быть задействованным в содержании Web-сайта. Такое свойство принято называть интерактивностью, которая на сегодняшний день встречается практически во всех электронных документах.

Что же дает присутствие пользовательских форм на странице? Диапазон функций, выполняемых HTML-формами, чрезвычайно широк, поэтому остановимся лишь на некоторых, наиболее распространенных направлениях.

Организация обратной связи

Размещая электронный документ во всемирной сети, мы уже подразумеваем реакцию пользовательской аудитории: похвалу и одобрение, несогласие и критику.

В качестве средства обратной связи между пользователем и автором электронного документа могут выступать:

- почтовая форма — посетителя просят указать свои персональные данные и изложить суть обращения (рис. 8.1). Текст обращения обрабатывается программой и высылается автору сайта по электронной почте. При этом другие посетители сайта не могут получить доступ к содержанию отправленного сообщения;

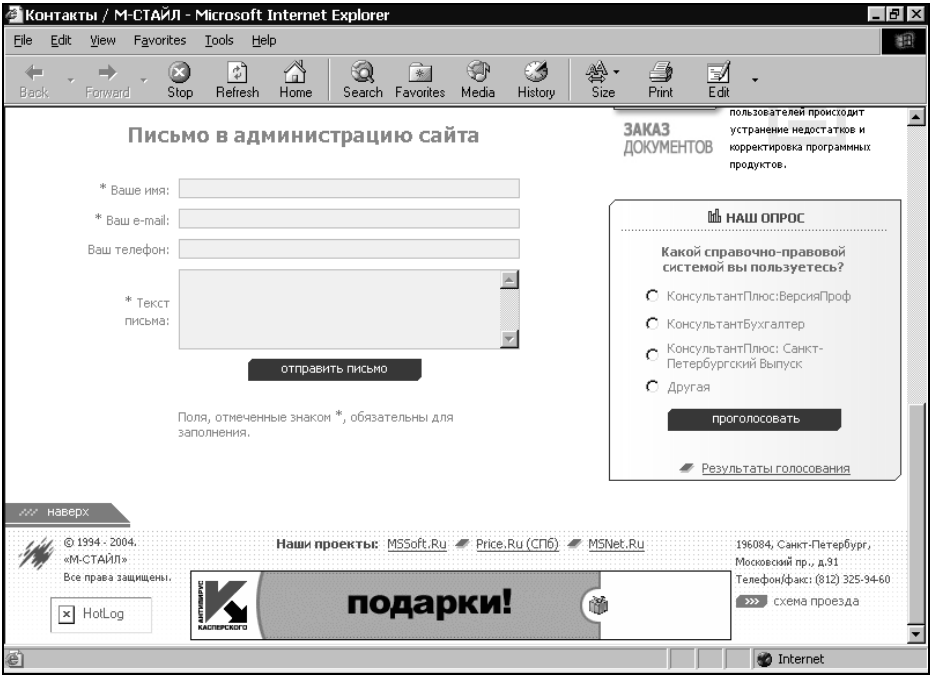


Рис. 8.1. Пример почтовой формы

- гостевая книга — электронная книга жалоб и предложений, в которой любой желающий может оставить свой отзыв, поделиться впечатлениями о сайте, изложить критические замечания и указать на ошибки автора. В этом случае сообщения одного посетителя видимы для остальных пользователей гостевой книги (рис. 8.2);

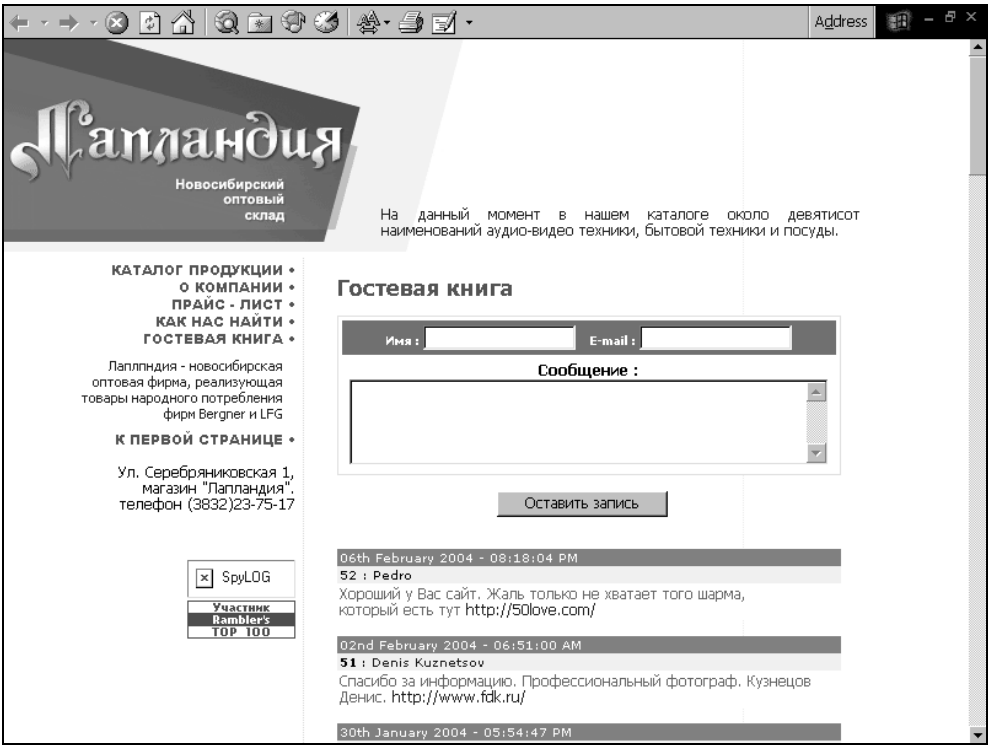


Рис. 8.2. Пример гостевой книги

- интернет-форум — более мощный инструмент стимулирования оценки пользовательской аудитории, чем почтовая форма или гостевая книга. В интернет-форумах (рис. 8.3) происходит многосторонний диалог — причем не только между человеком, задающим вопрос, и автором Web-сайта, а между всеми пользователями данного интернет-ресурса. На вопрос одного пользователя может ответить другой, третий подскажет то, чего не упомянул второй, наконец, автор сайта подытожит высказывания всех пользователей и даст свой развернутый ответ.

  Часто в рамках интернет-форума работают дополнительные функциональные средства общения: личные сообщения, скрытые разделы форума для привилегированных пользователей, опросы общественного мнения и пр.;

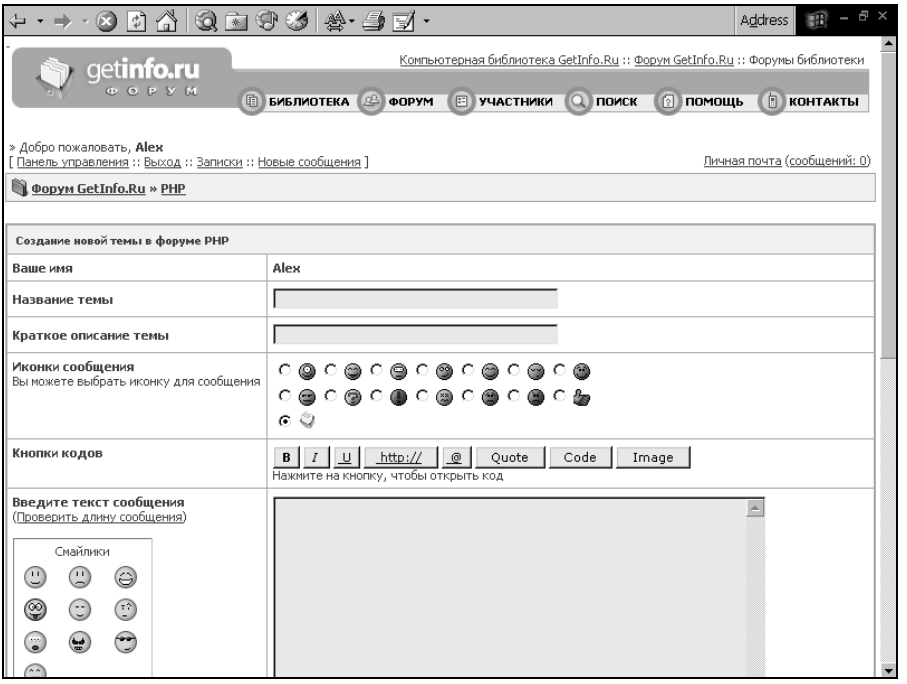


Рис. 8.3. Пример интернет-форума

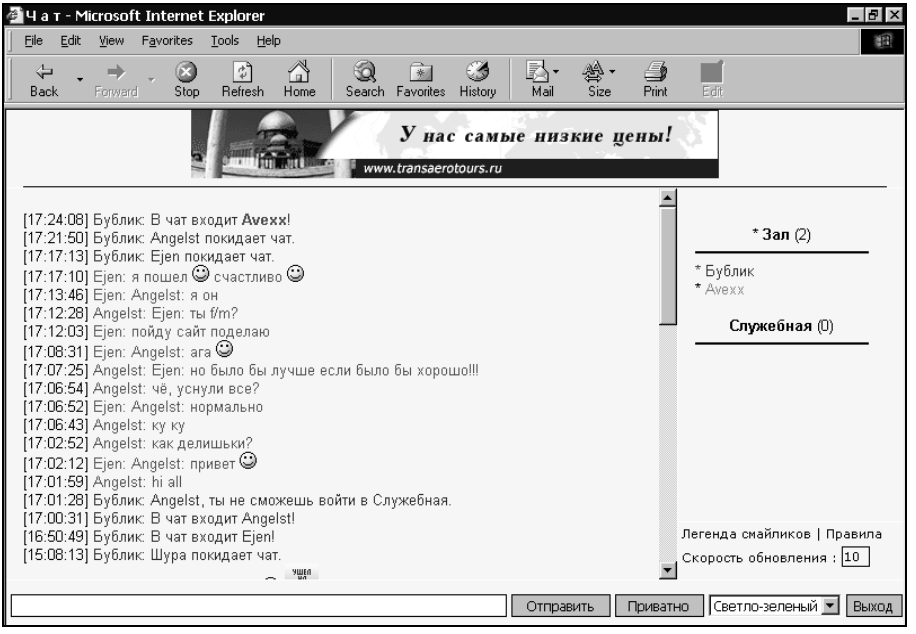


Рис. 8.4. Пример чата

- чат — свойство чата (от англ. *to chat* — разговаривать, болтать) подразумевает единовременное общение многих пользователей в режиме реального времени (рис. 8.4). В отличие от гостевой книги или форума, где ответ на поставленный вопрос может быть выслан и через час, и через день, общение в чате происходит как в реальной жизни между двумя (или более) собеседниками, находящимися в одной комнате.

  Собеседники могут находиться в разных городах, государствах и временных зонах, однако, присутствуя в чате, все пользователи способны общаться одновременно.

  В чатах могут присутствовать некоторые дополнительные функции: приваты (отдельный разговор между двумя пользователями), система почтовых сообщений, голосования и т. д.

Авторизация

Для получения доступа к информации на многих интернет-ресурсах существует необходимость регистрации в сервисе. После регистрации, которая подразумевает указание уникального пользовательского имени и пароля, пользователь должен пройти процедуру авторизации — проверки заявленного имени и пароля на наличие в базе данных сайта (рис. 8.5).

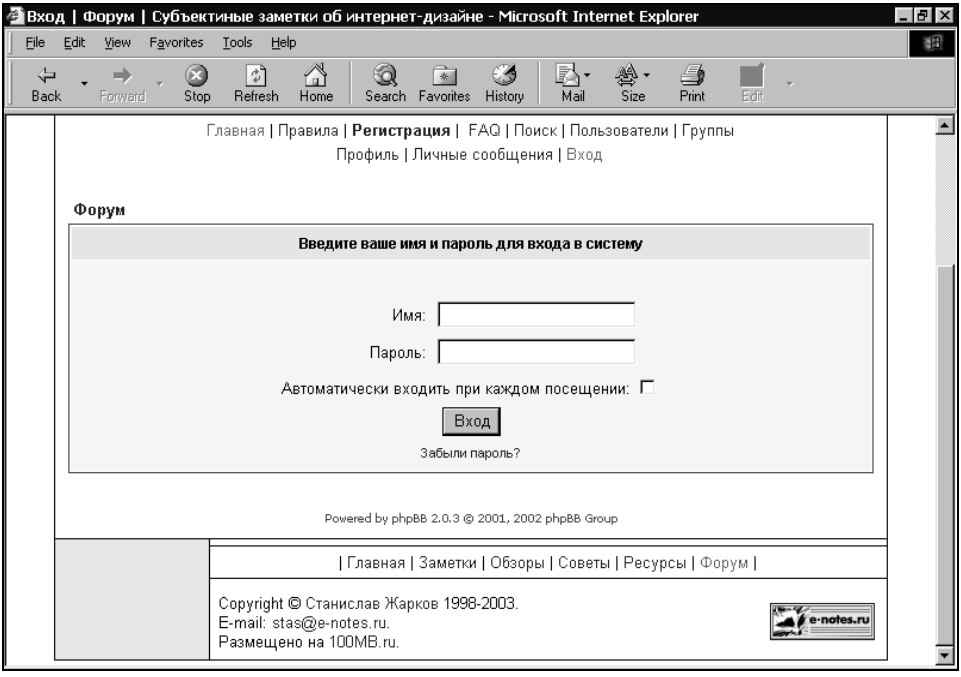


Рис. 8.5. Пример формы авторизации

Система авторизации распространена на информационных ресурсах (новостные ленты, афиши событий), корпоративных серверах (закрытые разделы для клиентов, партнеров, дистрибьюторов), сайтах электронной почты и т. д.

Проведение исследований

Для изучения пользовательского спроса на тот или иной вид услуг, выявления и последующего анализа реакции посетителей на многих сайтах устраиваются всевозможные опросы, анкетирования (рис. 8.6), голосования, тесты и пр.

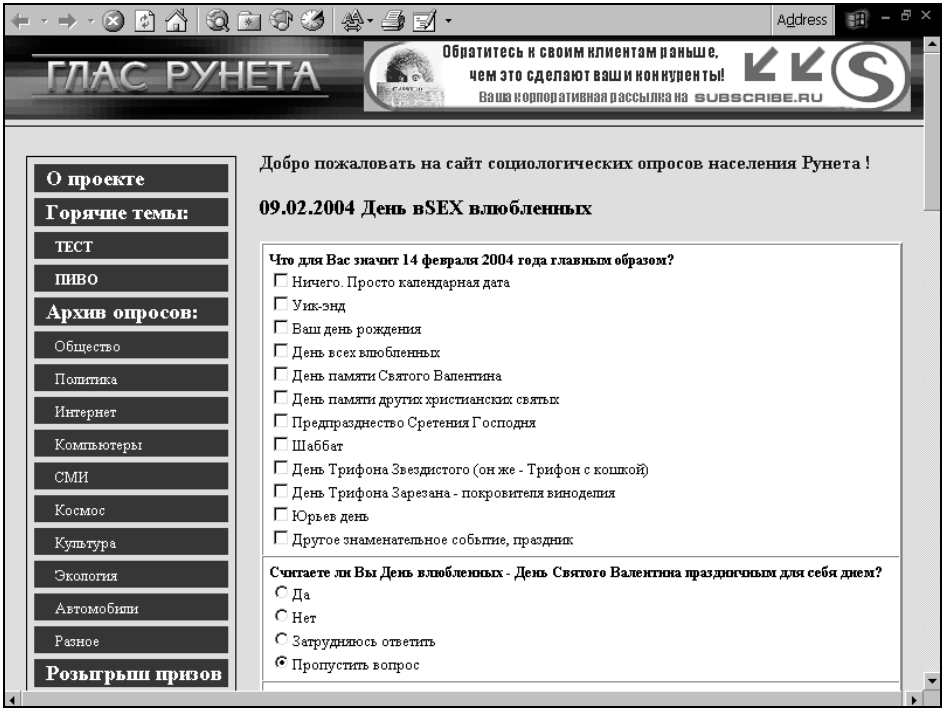


Рис. 8.6. Пример анкеты

Набор стандартных HTML-конструкций облегчает участие пользователя в подобных исследованиях: все, что ему потребуется, — это отметить подходящие варианты ответа и подтвердить участие в опросе.

Пользовательский профиль

Регистрируясь на каком-либо интернет-ресурсе, человек указывает всякого рода информацию: имя, город и страну проживания, увлечения, место рабо-

ты, контактные данные и пр. Совокупность таких данных составляет *пользовательский профиль* (рис. 8.7), который впоследствии можно будет изменить путем редактирования соответствующих полей HTML-формы.

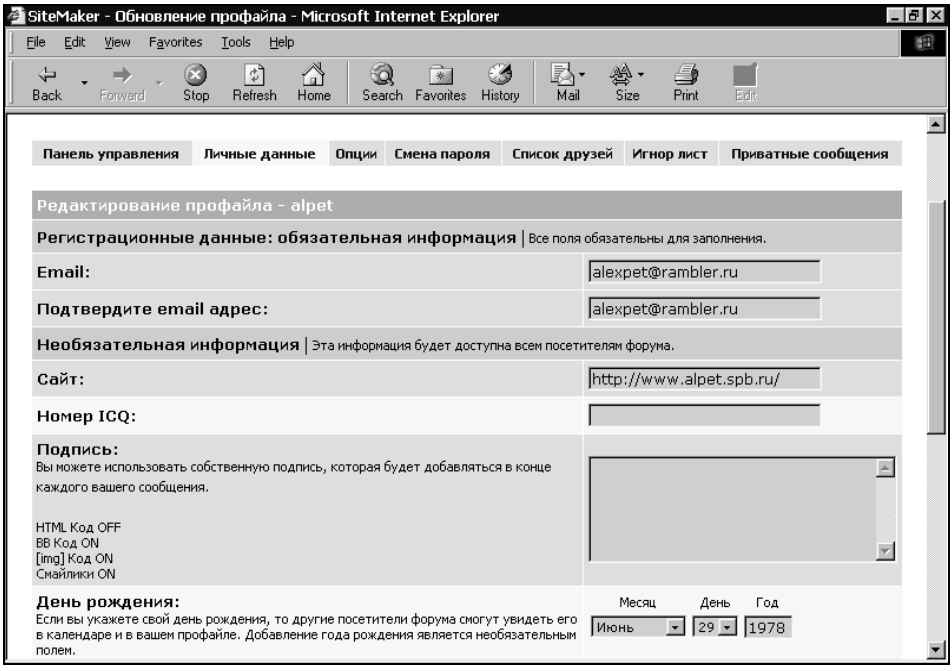


Рис. 8.7. Пример пользовательского профиля

Структура пользовательских форм

Структура любой *пользовательской формы* описывается тегом-контейнером `<FORM>`, внутри которого могут располагаться теги и параметры составляющих элементов HTML-формы. В пределах тега `<FORM>` не может содержаться другая пользовательская форма. Закрывающий тег обязателен для указания.

Тег `<FORM>` может содержать четыре основных параметра: `ACTION`, `METHOD`, `NAME` и `ENCTYPE`. Рассмотрим каждый из них.

Параметр *ACTION*

Параметр `ACTION` является единственным обязательным параметром тега `<FORM>` и предназначен для указания пути на Web-сервере к программе-обработчику данных пользовательской формы:

```
<FORM ACTION="/cgi-bin/form.cgi">
```

Путь к программе-обработчику может быть как относительный, так и абсолютный.

Этот параметр имеет важное значение с точки зрения работоспособности HTML-формы: если путь или название программы указаны неверно, или указанный файл не является исполняемым на стороне сервера, обработка введенных пользователем данных может быть нарушена, произведена некорректно или вовсе не осуществлена.

Наиболее распространенными типами исполняемых на стороне сервера файлами являются CGI-программы (pl, cgi, fcgi), PHP-скрипты (php, php3, phtm, phtml), ASP-приложения (asp), JSP-обработчики (jsp) и др.

Параметр *METHOD*

Параметр `METHOD` используется для определения типа передачи данных на Web-сервер. Его возможными значениями являются `GET` (по умолчанию) и `POST`.

При использовании типа `GET` данные пользовательской формы пересылаются в составе адреса запроса браузера. После имени программы-обработчика ставится знак вопроса "?", обозначающий вывод запроса браузера к переменным HTML-формы, и последовательность переменных и их свойств самой формы. Последовательность переменных формы разделяется символом амперсанда "&".

Пример адреса запроса браузера с использованием типа передачи данных `GET`:

```
http://www.site.ru/cgi-bin/form.cgi?Name=Vasya&Email=vasya@vasya.ru
```

Из структуры ссылки, образовавшейся в ходе обработки данных формы, понятно, что пользователь ввел свое имя ("`Vasya`") и адрес электронной почты ("`vasya@vasya.ru`").

При использовании типа `POST` данные формы не отображаются в адресной строке браузера, а передаются в составе самого запроса программы-обработчика. Таким образом, используя этот тип, мы получим следующую гиперссылку (учитывая данные формы, аналогичные типу `GET`):

```
http://www.site.ru/cgi-bin/form.cgi
```

Следует отметить, что пользовательская форма может включать информацию о пользователе, носящую конфиденциальный характер (например, пароли доступа), которая будет отображена в запросе браузера при использовании типа `GET`. В этом случае вся секретная информация, введенная одним пользователем, будет доступна для просмотра любому пользователю данного компьютера (просмотр истории перехода по Web-сайтам в браузере легко обнаружит ссылку с конфиденциальными данными).

Однако, в конечном итоге, выбор типа передачи данных на Web-сервер определяется конкретной задачей, стоящей перед применением HTML-формы, особенностями сервера и самой программы-обработчика.

Параметр *NAME*

Параметр `NAME` присваивает HTML-форме уникальное имя, которое используется в программе-обработчике для идентификации пользовательских данных, например:

```
<FORM NAME="mail">
```

Параметр *ENCTYPE*

Параметр `ENCTYPE` предназначен для определения типа данных при кодировании информации, введенной пользователем, и последующей ее передаче на Web-сервер. Кодирование осуществляется браузером и используется для предотвращения разного рода искажений в процессе передачи на сервер.

Возможными значениями параметра могут быть: `application/x-www-form-urlencoded` (по умолчанию) и `multipart/form-data`.

Первое значение используется, если помимо текста необходимо передать на сервер данные иного типа, к примеру, графику или запакованные файлы. Формат записи состоит из указания типа и его подтипа, где тип данных — это определение общего типа данных (текст, графика, архив, программа и т. д.), например, `text`, `image`, `application`, а подтип — это вид данных внутри определенного общего типа (`image/gif`, `text/html`).

Значение `multipart/form-data` используется в специфических случаях, например, при необходимости предоставить пользователю возможность загрузки на сервер любого файла со своего локального компьютера.

Основные параметры тега `<FORM>` приведены в табл. 8.1.

Таблица 8.1. *Основные параметры тега* `<FORM>`

Параметр	Функция
`ACTION`	Указание пути на Web-сервере к программе-обработчику данных пользовательской формы
`METHOD`	Определение типа передачи пользовательских данных на Web-сервер
`NAME`	Указание уникального имени пользовательской формы для идентификации программой-обработчиком
`ENCTYPE`	Определение типа кодирования пользовательских данных при передаче от браузера к Web-серверу

Внутри тега <FORM> могут размещаться дополнительные теги и параметры, используемые для включения в состав пользовательской формы различных элементов управления. Они и определяют визуальное наполнение HTML-формы (тег <FORM> и его параметры невидимы пользователю и предназначены для указания необходимой информации браузеру и программе-обработчику на сервере).

Тег *<INPUT>*

Тег <INPUT> является наиболее распространенным и способен отобразить широкий набор элементов управления пользовательской формой:

- текстовую строку;
- поле ввода пароля;
- поле выбора локального файла для загрузки на Web-сервер;
- опцию выбора;
- опцию переключения;
- кнопку отправления пользовательских данных;
- графический вариант кнопки отправления пользовательских данных;
- кнопку сброса введенных пользовательских данных;
- скрытые поля.

Тег не является парным и может содержать следующие параметры.

Параметр *TYPE*

Параметр TYPE позволяет указывать один из девяти приведенных далее элементов формы, каждому из которых соответствует отдельное значение.

TYPE="TEXT". Элемент текстовой строки. Может содержать ряд дополнительных параметров, описанных в табл. 8.2.

Таблица 8.2. *Параметры текстовой строки* *<INPUT TYPE="text">*

Параметр	Описание
SIZE	Определение максимального количества символов, видимых в текстовой строке без перемещения курсора
MAXLENGTH	Определение максимального количества символов, допустимых для ввода в текстовой строке. По умолчанию количество вводимых символов не ограничено
NAME	Присвоение идентификационного имени для программы-обработчика

Таблица 8.2 *(окончание)*

Параметр	Описание
VALUE	Указание значения текстовой строки (при просмотре в браузере выводится в виде обычного текста в самой строке)

Рассмотрим несколько вариантов использования элемента текстовой строки в HTML-форме (листинг 8.1, рис. 8.8).

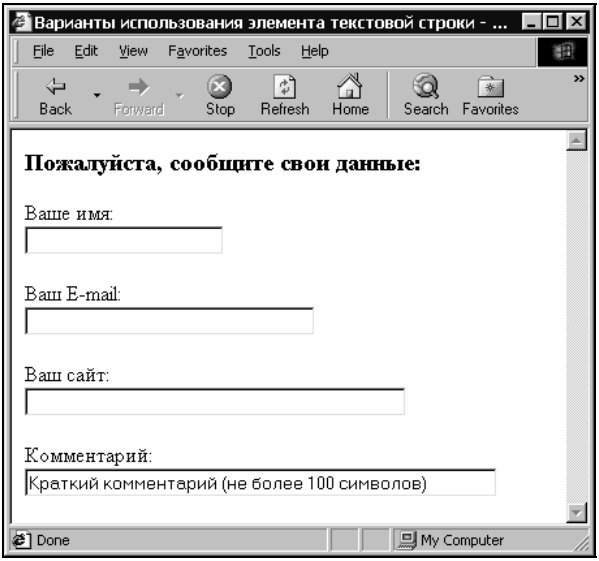


Рис. 8.8. Варианты использования элемента текстовой строки

Листинг 8.1. Варианты использования элемента текстовой строки

```
<HTML>
  <HEAD>
    <TITLE>Варианты использования элемента текстовой строки</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<H3>Пожалуйста, сообщите свои данные:</H3>
<FORM NAME="mail" ACTION="/cgi-bin/mail.cgi" METHOD="post">
Ваше имя:<BR>
<INPUT TYPE="text" SIZE="20" MAXLENGTH="50" NAME="name">
<BR><BR>
```

```
Ваш E-mail:<BR>
<INPUT TYPE="text" SIZE="30" MAXLENGTH="35" NAME="email">
<BR><BR>
Ваш сайт:<BR>
<INPUT TYPE="text" SIZE="40" MAXLENGTH="60" NAME="www">
<BR><BR>
Комментарий:<BR>
<INPUT TYPE="text" SIZE="50" MAXLENGTH="100" NAME="comment"
VALUE="Краткий комментарий (не более 100 символов)">
</FORM>
</BODY>
</HTML>
```

TYPE="PASSWORD". Элемент ввода пользовательского пароля (рис. 8.9). Ничем не отличается от обыкновенной текстовой строки за исключением того, что набранный текст отображается в виде "звездочек":

```
<INPUT TYPE="password" SIZE="30" NAME="password">
```

Такая мера связана с сохранением конфиденциальности пользовательских данных (однако не стоит забывать, что данные, вводимые в это поле, при использовании типа передачи `GET` будут отображаться в ссылке запроса браузера).

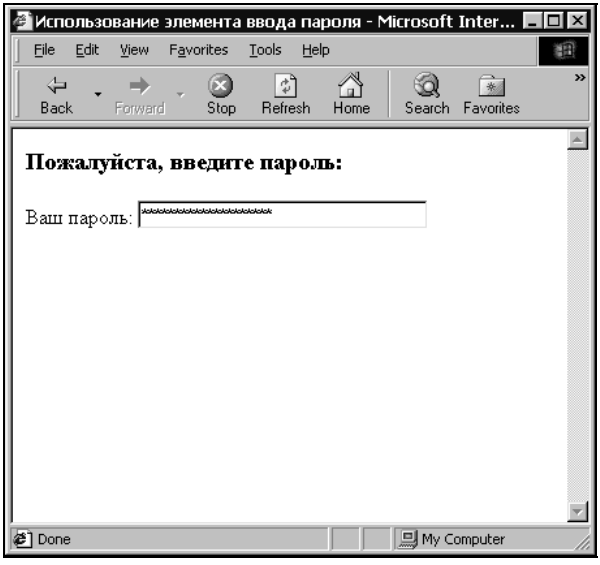


Рис. 8.9. Использование элемента ввода пароля

TYPE="FILE". Элемент выбора файла, расположенного на локальном компьютере пользователя, для загрузки на Web-сервер (рис. 8.10). Из дополнительных параметров действуют только `SIZE` и `MAXLENGTH`.

Для корректной передачи файла необходима конструкция `ENCTYPE="multipart/form-data"` и указание типа передачи данных `POST` в теге `<FORM>`. В противном случае браузер передаст не выбранный файл, а путь к нему на компьютере пользователя.

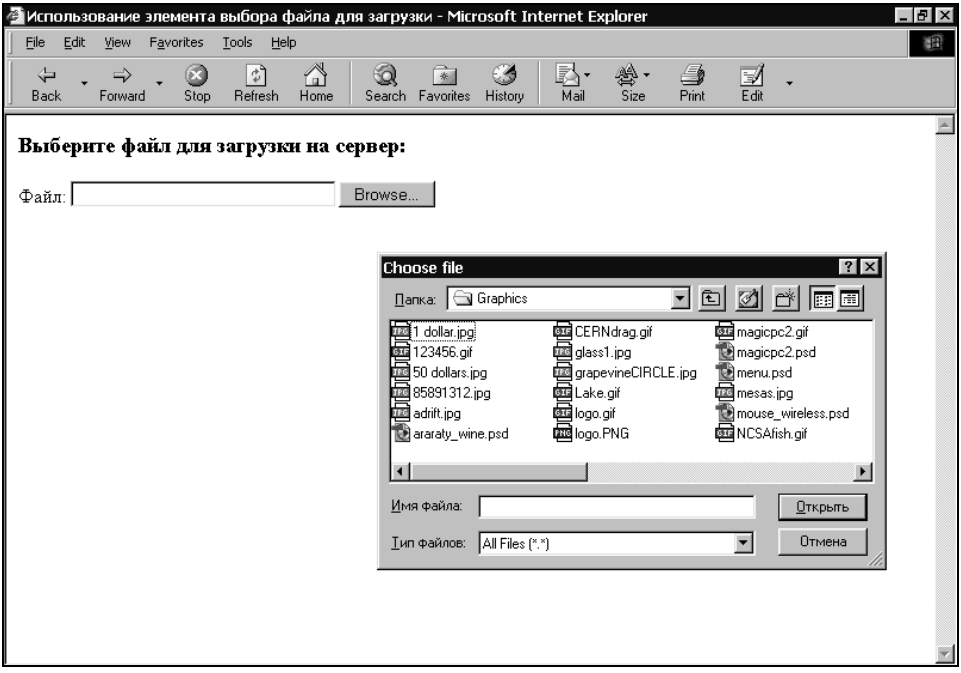


Рис. 8.10. Использование элемента выбора файла для загрузки

Следует добавить, что средств HTML, способных изменять название кнопки просмотра содержимого локального компьютера, не существует. Значение может быть только одно и зависит от модели браузера: **Browse** или **Обзор**.

Элемент выбора локального файла ранними версиями популярных браузеров поддерживается не полностью: например, Internet Explorer 3.x отображает только строку пути для файла, не давая возможности поиска файла на компьютере прямо из браузера (в этом случае приходится вручную вписывать полный путь к нужному файлу на локальном компьютере пользователя).

TYPE="CHECKBOX". Элемент опции выбора, устанавливающий/снимающий "галочку" для конкретного поля пользовательской формы. Может содержать ряд дополнительных параметров, описанных в табл. 8.3.

Таблица 8.3. *Параметры опции выбора* `<INPUT TYPE="checkbox">`

Параметр	Описание
NAME	Указание общего для всех вариантов выбора идентификационного имени
VALUE	Определение значения для конкретного варианта выбора (обязательный параметр). Не должен повторяться, т. к. при установлении "галочки" передается на Web-сервер
CHECKED	Данный вариант является выбранным по умолчанию

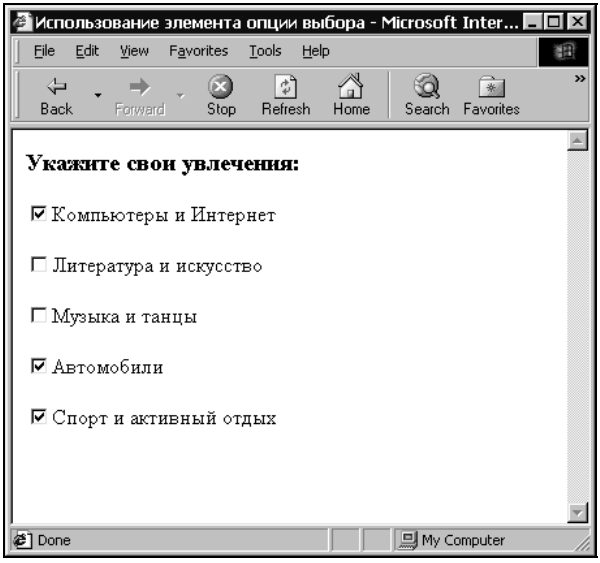


Рис. 8.11. Использование элемента опции выбора

Как видно из рис. 8.11, пользователь может выбирать несколько вариантов поля CHECKBOX, значение каждого из которых будет передано программой-обработчиком на Web-сервер (листинг 8.2).

Листинг 8.2. Использование элемента опции выбора

```
<HTML>
  <HEAD>
    <TITLE>Использование элемента опции выбора</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
```

```
<H3>Укажите свои увлечения:</H3>
<FORM NAME="select_hobby" ACTION="/cgi-bin/hobby.cgi" METHOD="get">
<INPUT TYPE="checkbox" NAME="hobby" VALUE="computers" CHECKED>Компьютеры
и Интернет
<BR><BR>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="art">Литература и искусство
<BR><BR>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="music">Музыка и танцы
<BR><BR>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="auto">Автомобили
<BR><BR>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="sport">Спорт и активный отдых
</FORM>
</BODY>
</HTML>
```

TYPE="RADIO". Элемент опции переключения между различными вариантами выбора (рис. 8.12). В отличие от CHECKBOX, вариант выбора может быть только один. Также имеет дополнительные параметры NAME, VALUE и CHECKED, действие которых аналогично опции выбора CHECKBOX (листинг 8.3).

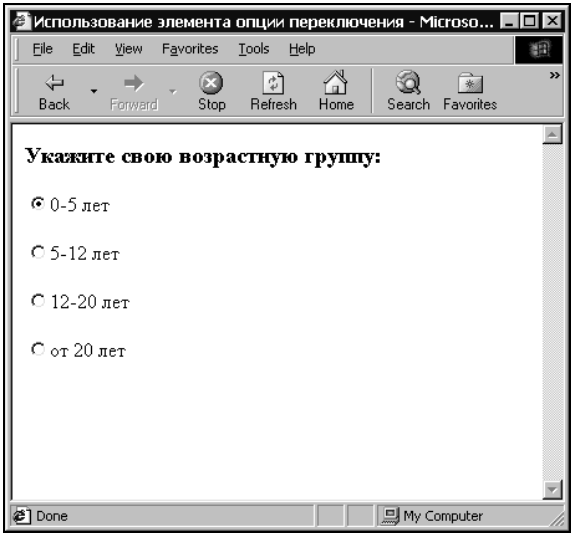


Рис. 8.12. Использование элемента опции переключения

Листинг 8.3. Использование элемента опции переключения

```
<HTML>
  <HEAD>
```

```
    <TITLE>Использование элемента опции переключения</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<H3>Укажите свою возрастную группу:</H3>
<FORM NAME="select_age" ACTION="/cgi-bin/age.cgi" METHOD="get">
<INPUT TYPE="radio" NAME="age" VALUE="baby" CHECKED>0-5 лет
<BR><BR>
<INPUT TYPE="radio" NAME="age" VALUE="child">5-12 лет
<BR><BR>
<INPUT TYPE="radio" NAME="age" VALUE="junior">12-20 лет
<BR><BR>
<INPUT TYPE="radio" NAME="age" VALUE="adult">от 20 лет
</FORM>
</BODY>
</HTML>
```

TYPE="SUBMIT". Кнопка отправления пользовательских данных на Web-сервер. При нажатии на нее запускается программа-обработчик, которая анализирует введенные пользователем данные и отправляет результат на сервер.

В принципе, необязательный элемент. В этом случае отправление данных может быть выполнено нажатием клавиши <Enter> на клавиатуре пользователя при размещении курсора мыши на любом поле формы. Однако подобный подход не приветствуется, т. к. отсутствие кнопки подтверждения может запутать неопытного пользователя.

По умолчанию отображается в виде прямоугольной кнопки с надписью **Submit** (**Submit Query** и пр. — в зависимости от браузера).

Для элемента отправления данных на сервер также могут использоваться дополнительные (необязательные) параметры: `NAME` (имя кнопки отправления, значение которого также будет передано на сервер) и `VALUE` (присвоение собственного имени кнопки).

TYPE="IMAGE". Графический аналог стандартной кнопки отправления данных формы на Web-сервер. В этом случае автор электронного документа не ограничен средствами HTML и может использовать свою фантазию на создание оригинальной, привлекательной кнопки передачи данных.

Для данного параметра используются некоторые дополнительные параметры:

- `SRC` — указание относительного или абсолютного пути на сервере к файлу графического изображения, служащего кнопкой передачи данных формы;
- `ALIGN` — указание типа выравнивания текста относительно графической кнопки отправления данных формы;

- `BORDER` — определение толщины рамки кнопки (как правило, значение равно нулю);
- `ALT` — указание альтернативного текста/подсказки для кнопки отправления данных.

TYPE="RESET". Кнопка сброса введенных пользователем данных HTML-формы. При нажатии на нее восстанавливаются все установленные по умолчанию значения полей формы.

Элемент необязателен для указания, однако, в ряде случаев, весьма полезен (при работе с многочисленными текстовыми строками и опциями выбора пользователь может допустить ошибку или захочет заново заполнить форму: в этом случае ему придется либо перезагружать страницу, либо вручную удалять текст из каждого поля формы; кнопка сброса позволит одним нажатием добиться желаемого).

По умолчанию, названием кнопки сброса является **Reset**. Изменить наименование элемента можно с помощью дополнительного параметра `VALUE` (рис. 8.13, листинг 8.4).

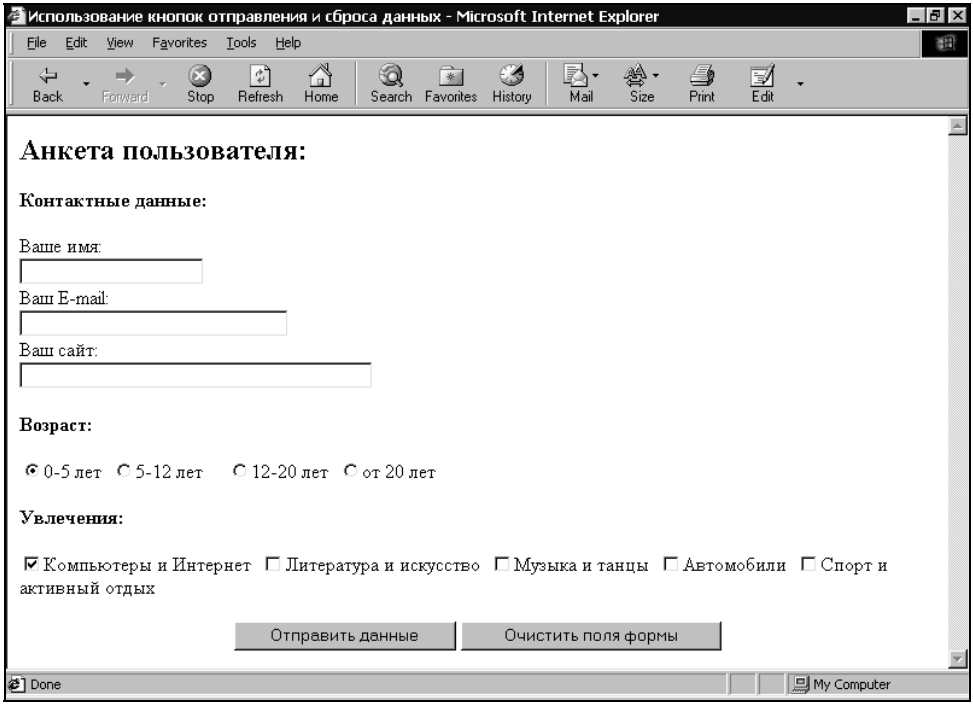


Рис. 8.13. Использование кнопок отправления и сброса данных

Листинг 8.4. Использование кнопок отправления и сброса данных

```
<HTML>
  <HEAD>
    <TITLE>Использование кнопок отправления и сброса данных</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<FORM NAME="user_data" ACTION="/cgi-bin/anketa.cgi" METHOD="post">
<H2>Анкета пользователя:</H2>
<P><B>Контактные данные:</B></P>
Ваше имя:<BR>
<INPUT TYPE="text" SIZE="20" MAXLENGTH="50" NAME="name">
<BR>
Ваш E-mail:<BR>
<INPUT TYPE="text" SIZE="30" MAXLENGTH="35" NAME="email">
<BR>
Ваш сайт:<BR>
<INPUT TYPE="text" SIZE="40" MAXLENGTH="60" NAME="www">
<P><B>Возраст:</B></P>
<INPUT TYPE="radio" NAME="age" VALUE="baby" CHECKED>0-5
лет  <INPUT TYPE="radio" NAME="age" VALUE="child">5-12
лет    <INPUT TYPE="radio" NAME="age"
VALUE="junior">12-20 лет  <INPUT TYPE="radio" NAME="age"
VALUE="adult">от 20 лет
<P><B>Увлечения:</B></P>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="computers" CHECKED>Компьютеры
и Интернет  <INPUT TYPE="checkbox" NAME="hobby"
VALUE="art">Литература и искусство  <INPUT TYPE="checkbox"
NAME="hobby" VALUE="music">Музыка и танцы  <INPUT
TYPE="checkbox" NAME="hobby" VALUE="auto">Автомобили  <INPUT
TYPE="checkbox" NAME="hobby" VALUE="sport">Спорт и активный отдых
<P ALIGN ="center">
<INPUT TYPE="submit" VALUE="Отправить данные"> <INPUT TYPE="reset"
VALUE="Очистить поля формы">
</P>
</FORM>
</BODY>
</HTML>
```

TYPE="HIDDEN". Служебное поле, невидимое пользователю. Не может быть изменено ни браузером, ни самим пользователем. Анализируется программой-обработчиком и отсылается на Web-сервер (отсылаемое значение варьируется специальным параметром `VALUE`).

Дополнительные параметры тега *<INPUT>*

Кроме основных параметров, в теге `<INPUT>` могут присутствовать дополнительные параметры, поддерживаемые не всеми браузерами и используемые с помощью вспомогательных по отношению к HTML средств. К ним можно отнести следующие параметры:

- `DISABLED` — отменяет активность выбранного поля формы. При указании параметра `DISABLED` редактирование элемента формы будет недоступно. Параметр поддерживается только браузером Internet Explorer. Используется в сложных, динамически изменяющихся HTML-формах в совокупности с такими технологическими средствами, как DHTML, CSS, JavaScript и др.;
- `TITLE` — отображает всплывающую подсказку для элемента формы. Поддерживается только браузером Internet Explorer;
- `STYLE` — определяет стиль CSS для выбранного элемента формы;
- `CLASS` — устанавливает класс стилевого шаблона CSS (ссылается на внешний или внутренний набор стилей).

Тег *<TEXTAREA>*

Тег-контейнер `<TEXTAREA>` создает элемент текстового поля заданной ширины и высоты (листинг 8.5). Указание закрывающего тега обязательно. Внутри тега `<TEXTAREA>` не может быть никаких других тегов или параметров.

Главное отличие элемента текстового поля от элемента текстовой строки заключается в визуальном преимуществе объема вводимой информации (рис. 8.14). Основные параметры тега `<TEXTAREA>` приведены в табл. 8.4.

Таблица 8.4. *Параметры текстового поля* `<TEXTAREA>`

Параметр	Описание
`COLS`	Определение количества столбцов текстового поля
`ROWS`	Определение количества рядов текстового поля
`NAME`	Присвоение уникального имени, необходимого для идентификации программой-обработчиком

Листинг 8.5. Использование текстового поля

```
<HTML>
  <HEAD>
    <TITLE>Использование текстового поля</TITLE>
  </HEAD>
```

```
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<H3>Пожалуйста, сообщите свои данные:</H3>
<FORM NAME="mail" ACTION="/cgi-bin/mail.cgi" METHOD="get">
Ваше имя:<BR>
<INPUT TYPE="text" SIZE="30" MAXLENGTH="35" NAME="name">
<BR>
Ваш E-mail:<BR>
<INPUT TYPE="text" SIZE="30" MAXLENGTH="35" NAME="email">
<BR>
Комментарий:<BR>
<TEXTAREA COLS="25" ROWS="5" NAME="comment">Ваш комментарий...</TEXTAREA>
<BR><BR>
<INPUT TYPE="submit" VALUE="Отправить"> <INPUT TYPE="reset"
VALUE="Очистить">
</FORM>
</BODY>
</HTML>
```

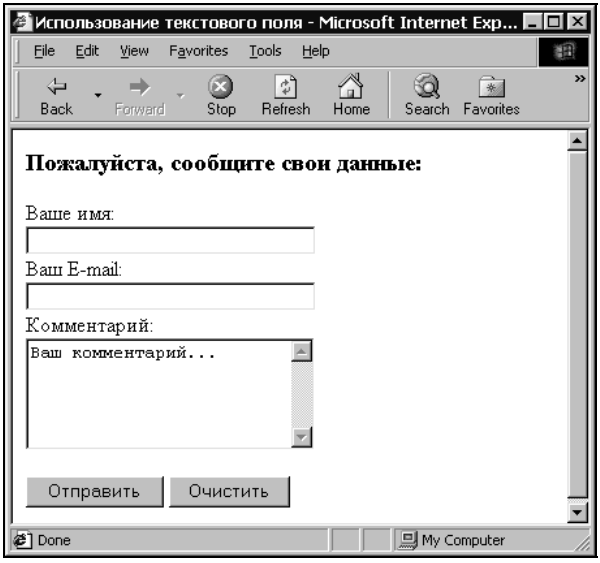


Рис. 8.14. Использование текстового поля

Тег *<SELECT>*

Тег `<SELECT>` предназначен для компактной группировки большого количества элементов пользовательской формы. К примеру, размещение нескольких десятков элементов `CHECKBOX` или `RADIO` займет слишком много места на стра-

нице, когда как группировка данных с помощью тега <SELECT> позволяет заметно сократить размеры, занятые под отображение HTML-формы.

Такой вид формы может быть представлен ниспадающим меню или списком наименований. Ниже рассмотрим теги и параметры для каждого из этих видов.

Тег-контейнер <SELECT> определяет структуру и вид группировки элементов формы, требует закрывающего тега и может включать в себя следующие параметры:

- NAME — уникальное имя, предназначенное для идентификации программой-обработчиком. Является обязательным параметром, значение которого передается на Web-сервер;
- SIZE — при значении, большем 1, ниспадающее меню превращается в список наименований, состоящий из заданного количества позиций (число позиций соответствует значению данного параметра);
- MULTIPLE — параметр, разрешающий выбор нескольких позиций из списка наименований (выбор осуществляется с помощью курсора мыши, удерживая клавишу <Ctrl>).

Внутри тега-контейнера <SELECT> не может находиться никакой информации, а также других тегов и их параметров, за исключением тега <OPTION>, который задает свойства для каждой позиции ниспадающего меню или списка наименований.

В теге-контейнере <OPTION> могут быть размещены два основных параметра:

- VALUE — параметр, значение которого передается программой-обработчиком на Web-сервер. В данном случае будет отправлено значение конкретной выбранной позиции, а не общего значения меню или списка.

  Следует отметить, что при отсутствии значения параметра VALUE на сервер будет передано содержимое выбранной позиции, т. е. содержимого тега-контейнера <OPTION>;
- SELECTED — параметр отмечает текущую позицию ниспадающего меню или списка наименований как выбранную. Таким образом, возможно сделать визуальный акцент на любой по счету позиции меню или списка (листинг 8.6, рис. 8.15).

Листинг 8.6. Использование ниспадающего меню и списка наименований

```
<HTML>
  <HEAD>
    <TITLE>Использование ниспадающего меню и списка наименований</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
```

```
<FORM NAME="anketa" ACTION="/cgi-bin/anketa.cgi" METHOD="get">
<H2>В каком городе вы живете?</H2>
<SELECT NAME="city" SIZE="5">
 <OPTION VALUE="Moscow">Москва
 <OPTION VALUE="Spb" SELECTED>Санкт-Петербург
 <OPTION VALUE="Samara">Самара
 <OPTION VALUE="Ryazan">Рязань
 <OPTION VALUE="Krasnoyarsk">Красноярск
 <OPTION VALUE="Komsomolsk">Комсомольск-на-Амуре
 <OPTION VALUE="Vladivostok">Владивосток
</SELECT>
<H2>Ваша профессия?</H2>
<SELECT NAME="profession">
 <OPTION VALUE="Teacher">Учитель
 <OPTION VALUE="Technic">Слесарь
 <OPTION VALUE="Designer">Дизайнер
 <OPTION VALUE="Driver" SELECTED>Водитель
 <OPTION VALUE="Fireman">Пожарник
 <OPTION VALUE="Actor">Актер
</SELECT>
<BR><BR>
<INPUT TYPE="submit" VALUE="Отправить"> <INPUT TYPE="reset"
VALUE="Очистить">
</FORM>
</BODY>
</HTML>
```

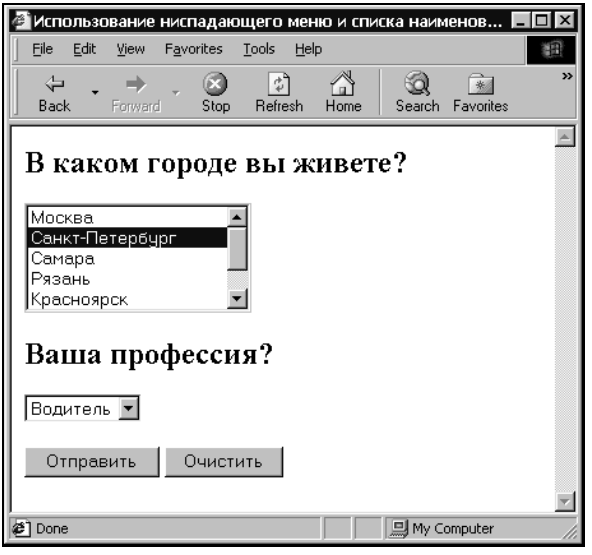


Рис. 8.15. Использование ниспадающего меню и списка наименований

Следует отметить, что тег <OPTION> не требует обязательного присутствия закрывающего тега.

В табл. 8.5 приведены все описанные выше теги и параметры, используемые для организации пользовательских форм.

Таблица 8.5. *Теги и параметры пользовательских форм*

Тег	Параметры	Описание
<FORM>	ACTION METHOD NAME ENCTYPE	Основной тег структуры пользовательской формы. Определяет путь на Web-сервере к программе-обработчику, тип передачи данных и кодирования пересылаемой информации, а также присваивает уникальное имя формы для программы-обработчика
<INPUT>	TYPE NAME VALUE SIZE CHECKED MAXLENGTH	Элемент текстовой строки, поле ввода пароля, опции выбора/переключения, подтверждения/отмены отправления данных формы, загрузки локального файла на Web-сервер, определение скрытых полей формы
<TEXTAREA>	COLS ROWS NAME	Элемент текстового поля
<SELECT>	NAME SIZE MULTIPLE	Группировка данных пользовательской формы путем составления ниспадающих меню и списков наименований
<OPTION>	VALUE SELECTED	Элемент (позиция) ниспадающего меню или списка наименований

В завершение этой главы рассмотрим итоговый пример организации пользовательских форм с применением таблиц, текстового и структурного форматирования (листинг 8.7, рис. 8.16).

Листинг 8.7. Итоговый пример по организации пользовательских форм

```
<HTML>
  <HEAD>
    <TITLE>Итоговый пример по организации пользовательских форм</TITLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
```

```
<H2 ALIGN="center">ПОЛЬЗОВАТЕЛЬСКАЯ АНКЕТА</H2>
<FORM NAME="user_anketa" ACTION="/cgi-bin/anketa.cgi" METHOD="post"
ENCTYPE="multipart/form-data">
<TABLE BORDER="0" CELLPADDING="5" CELLSPACING="3" ALIGN="center"
WIDTH="98%">
<TR>
<TH BGCOLOR="#999999" WIDTH="50%">Личные данные</TH>
<TH BGCOLOR="#999999" WIDTH="50%">Увлечения</TH>
</TR>
<TR>
<TD VALIGN="top" BGCOLOR="#EEEEEE" WIDTH="50%">
Ваше имя:<BR>
<INPUT TYPE="text" SIZE="30" MAXLENGTH="35" NAME="name">
<BR>
Ваш пароль:<BR>
<INPUT TYPE="password" SIZE="30" MAXLENGTH="35" NAME="password">
<BR>
Ваша фотография:<BR>
<INPUT TYPE="file" SIZE="18" NAME="photo">
</TD>
<TD VALIGN="top" BGCOLOR="#EEEEEE" WIDTH="50%">
<INPUT TYPE="checkbox" NAME="hobby" VALUE="computers" CHECKED>Компьютеры
и Интернет
<BR>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="art">Литература и искусство
<BR>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="music">Музыка и танцы
<BR>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="auto">Автомобили
<BR>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="sport">Спорт и активный отдых
</TD>
</TR>
<TR>
<TH BGCOLOR="#999999" WIDTH="50%">Место жительства</TH>
<TH BGCOLOR="#999999" WIDTH="50%">Комментарии</TH>
</TR>
<TR>
<TD VALIGN="top" BGCOLOR="#EEEEEE" WIDTH="50%">
<SELECT NAME="city" SIZE="5">
 <OPTION VALUE="">--- Выберите город ---
 <OPTION VALUE="Moscow">Москва
 <OPTION VALUE="Spb">Санкт-Петербург
 <OPTION VALUE="Samara">Самара
```

```
 <OPTION VALUE="Ryazan">Рязань
 <OPTION VALUE="Krasnoyarsk">Красноярск
 <OPTION VALUE="Komsomolsk">Комсомольск-на-Амуре
 <OPTION VALUE="Vladivostok">Владивосток
</SELECT>
<BR><BR>
<SELECT NAME="country">
 <OPTION VALUE="">--- Выберите страну ---
 <OPTION VALUE="Russia">Россия
 <OPTION VALUE="England">Англия
 <OPTION VALUE="USA">США
 <OPTION VALUE="Germany">Германия
 <OPTION VALUE="France">Франция
</SELECT>
</TD>
<TD VALIGN="top" BGCOLOR="#EEEEEE" WIDTH="50%">
<TEXTAREA COLS="25" ROWS="5" NAME="comments"></TEXTAREA>
<BR>
<INPUT TYPE="radio" NAME="site" VALUE="first" CHECKED>Я впервые
на этом сайте
<BR>
<INPUT TYPE="radio" NAME="site" VALUE="not_first">Я тут не в первый раз
</TD>
</TR>
<TR>
<TD COLSPAN="2" ALIGN="center" BGCOLOR="#999900">
<INPUT TYPE="submit" VALUE="Отправить"> <INPUT TYPE="reset"
VALUE="Очистить">
</TD>
</TR>
</TABLE>
</FORM>
</BODY>
</HTML>
```

В данном примере были использованы следующие элементы пользовательских форм:

- текстовая строка;
- поле для ввода пароля;
- поле выбора локального файла для загрузки на Web-сервер;
- опция выбора нескольких вариантов данных;
- опция переключения между несколькими вариантами данных;

- текстовое поле;
- ниспадающее меню;
- список наименований;
- кнопка отправления данных;
- кнопка сброса данных.

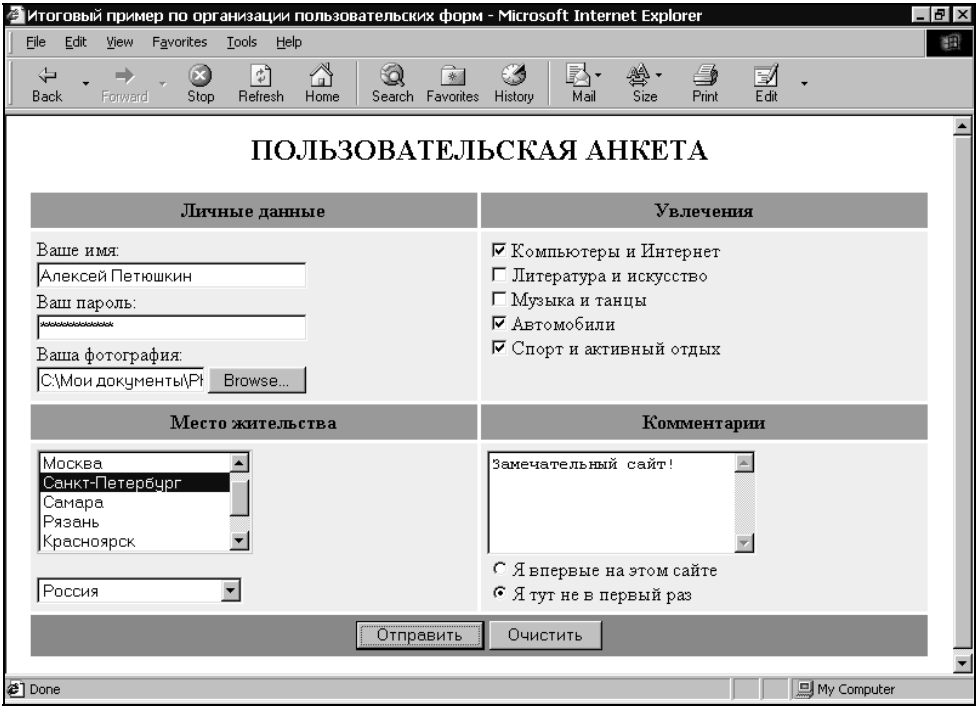


Рис. 8.16. Итоговый пример по организации пользовательских форм

ГЛАВА 9

Стилевые шаблоны CSS

Каскадные таблицы стилей

Спецификация языка разметки HTML позволяет разработчику электронных документов изменять внешний вид некоторых элементов страниц. Для этого составляются специальные правила отображения конкретного элемента в HTML-документе, называемые *каскадными таблицами стилей* (Cascading Style Sheets, CSS) или *стилевыми шаблонами.*

Общая информация

Разберем по составу понятие "Каскадная таблица стилей":

- каскадная — спецификация HTML разрешает использовать для одного и того же элемента несколько стилевых правил, интерпретируемых браузером последовательно, другими словами, каскадом;
- таблица — формат записи стилевых правил CSS напоминает табличное представление данных: заголовок таблицы соответствует наименованию элемента, класса или идентификатора стиля; в качестве ячеек и рядов таблицы выступают стилевые свойства и их значения. Судите сами:

```
DIV
{
font-family: Tahoma;
color: black;
font-size: 12px;
}
```

- стилей — под стилем принято понимать приведение какого-либо явления к общему набору правил и определений. CSS в этом понимании — способ дополнительного форматирования стандартных тегов HTML.

Версия HTML 4.01 включает следующие характеристики CSS:

- гибкое использование стилевых шаблонов (Flexible placement of style information) — правила задания стилевых таблиц позволяют использовать несколько возможных способов определения стиля: создание внешнего CSS-файла, использование внутреннего шаблона и импортирования таблицы стилей и пр. (более подробно о способах определения таблиц стилей рассказано в *разд. "Способы определения таблиц стилей" данной главы*);
- независимость от других языков определения стиля (Independence from specific style sheet languages) — спецификация HTML позволяет, помимо CSS, использовать другие языки определения стилевых шаблонов, что делает создаваемый электронный документ стиленезависимым, а значит, более адаптируемым к изменению внешнего вида определенных HTML-элементов;
- каскадность (Cascading) — возможность определения нескольких стилевых правил (указания нескольких таблиц стилей) для одного элемента HTML (не все языки таблиц стилей поддерживают каскадность). Свойство каскадности реализуется за счет последовательного "чтения" стилевых данных браузером;
- привязанность к носителю информации (Media dependencies) — с помощью стилевых шаблонов CSS возможна ориентация создаваемого электронного документа на различные носители информационных данных: операционные системы Windows и Macintosh, телевизионные и игровые приставки, мобильные телефоны и карманные персональные компьютеры (КПК), устройства распознавания человеческой речи и пр.;
- альтернативные стили (Alternate styles) — разработчики электронных документов могут создавать несколько вариантов отображения элементов HTML с помощью различных таблиц стилей CSS.

Уровни CSS

Впервые каскадные таблицы стилей CSS были реализованы в браузере Internet Explorer 3.0. Однако в то время развитие CSS находилось в зачаточном состоянии, поэтому правила составления стилевых шаблонов были весьма разрозненными и неопределенными.

С момента своего возникновения структура CSS была несколько раз пересмотрена: были добавлены новые элементы и убраны (видоизменены) старые. Существует три уровня CSS, определяемые наличием завершенной редакции структуры CSS:

- CSS 1 — первый уровень структуры стилевых шаблонов, окончательно утвержденный 11 января 1999 года;

- CSS 2 — второй уровень стилевых конструкций, начало обсуждения которого датируется маем 1998 года;
- CSS 3 — третий уровень стилевого оформления электронных документов; принят к обсуждению 23 мая 2001 года, на момент написания книги находился в стадии проработки.

В завершение разговора об уровнях CSS следует добавить, что переход от одного уровня к другому, в основном, сопровождался некоторыми видоизменениями в структуре и функциональности правил стилевого оформления, технологическими дополнениями, а также попытками систематизировать применение CSS (именно третий уровень CSS — CSS 3 — призван разработать некую единую систему представления стилей на электронном документе, основанную на использовании специальных модулей).

Способы определения таблиц стилей

Как уже было сказано выше, любая таблица стилей CSS должна быть интерпретирована браузером для того, чтобы правила CSS, обозначенные для конкретных элементов электронного документа, вступили в силу.

Определение таблицы стилей (стилевого шаблона) возможно четырьмя способами:

1. Ссылка на внешний файл — если все стилевые шаблоны для конкретного HTML-документа разместить в одном текстовом файле (с расширением css), то с помощью специального тега `<LINK>` из текущего документа можно сделать ссылку на внешний CSS-файл стилевых шаблонов, например:

   ```
   <LINK REL="stylesheet" TYPE="text/css" HREF="style.css">
   ```

   Браузер, анализируя HTML-код, обратится по указанному пути и, обнаружив указанный файл стилевого оформления, отобразит элементы страницы в соответствии с определенными правилами CSS.

   Следует помнить, что конструкция указания пути к внешнему CSS-файлу должна находиться в пределах раздела `HEAD` HTML-документа.

2. Внедрение в документ — под внедрением в документ подразумевается задание стилевой конструкции внутри самой HTML-страницы, например:

   ```
   <STYLE TYPE="text/css">
   <!--
   BODY { font-family: Arial, Helvetica; }
   INPUT { background-color: #CECECE; }
   -->
   </STYLE>
   ```

Данная конструкция также должна присутствовать в разделе `HEAD`. Для браузеров, не поддерживающих CSS вообще или отдельные правила стилевого оформления, описание шаблонов заключается между символами комментариев (при отсутствии поддержки CSS браузер пропустит содержание стилевых шаблонов; если поддержка есть — браузер интерпретирует правила CSS).

3. Включение в теговые конструкции — любой отдельный HTML-элемент может быть подвергнут форматированию средствами CSS. Для этого необходимо задать определенное правило реализации того или иного тега, например:

```
<P ALIGN="justify" STYLE="color: #000000; font-family: Verdana;">
Текст параграфа...
</P>
```

В этом случае задано отдельное правило для конкретного параграфа. Также можно присваивать отдельному HTML-элементу определенный класс стилевого шаблона:

```
<TABLE>
<TR>
<TD CLASS="header"></TD>
<TD CLASS="text"></TD>
</TR>
</TABLE>
```

Описание классов должно строиться следующим способом (на примере внедрения стилевого шаблона в документ):

```
<STYLE TYPE="text/css">
<!--
.header { font-weight: bold; color: gray; }
.text { color: black; font-size: 11px; }
-->
</STYLE>
```

В данном случае текст табличной ячейки класса `.header` будет отображаться жирным начертанием и серым цветом, а ячейки класса `.text` — обычным начертанием, черным цветом и размером шрифта 11 пикселов.

4. Импортирование (стилевого шаблона CSS) — аналогично указанию ссылки на внешний файл:

```
<STYLE TYPE="text/css">
<!--
@import: url(style.css);
-->
</STYLE>
```

Все четыре способа определения стилевого шаблона CSS можно использовать одновременно в пределах одного HTML-документа (листинг 9.1). Такая возможность позволяет задавать основное правило CSS, к примеру, в виде внешнего файла шаблонов, а для исключительных или редких HTML-элементов — отдельные конструкции либо в теге `<STYLE>`, либо в кодовых конструкциях самих тегов.

Листинг 9.1. Совмещение различных способов определения стилевого шаблона CSS

```
<HTML>
  <HEAD>
    <TITLE>Совмещение различных способов определения CSS</TITLE>
    <LINK REL="stylesheet" TYPE="text/css" HREF="style.css">
<STYLE TYPE="text/css">
<!--
P { text-align: justify; color: green; }
.title { color: blue; font-weight: bold; font-size: 16px; }
-->
</STYLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<FONT CLASS="title">Способы определения шаблонов CSS</FONT>
<P>
Ниже перечислены существующие способы определения стилевых шаблонов CSS,
даны характеристики каждого способа, приведены примеры их использования.
</P>
<UL>
<LI>Ссылка на внешний файл
<LI>Внедрение в документ
<LI>Включение в теговые конструкции
<LI>Импортирование
</UL>
</BODY>
</HTML>
```

Следует сказать, что использование каждого способа определения стилевых шаблонов CSS имеет свои недостатки:

- в случае ошибки интерпретации HTML-кода браузером, плохой связи с сервером и пр. внешний файл CSS может не загрузиться, вследствие чего стиль для нужных элементов HTML не будет переопределен; если внешний файл CSS включает слишком большое количество стилевых шаблонов

(что отражается на конечном размере файла), то существует вероятность того, что браузер не сумеет полностью интерпретировать файл CSS или вообще исчерпает лимит времени по загрузке данных. В первом случае, стили для части элементов не будут переопределены (браузер успеет "обнаружить" только те правила, которые размещены в верхней части CSS-файла). Во втором случае, все элементы страницы останутся без изменения, загрузившись по умолчанию;

- при использовании способа включения стиля в сам документ наличие слишком большого количества шаблонов CSS заметно увеличит конечный размер HTML-страницы, что скажется на времени загрузки документа в браузере;
- при сочетании различных способов определения стилевых шаблонов следует учитывать особенности браузеров. К примеру, Netscape "плохо относится" к использованию символа нижнего подчеркивания "_" в указании классов для элементов HTML (`.news_title`, `._about` и т. д.). Также ряд браузеров (Netscape в том числе) не подключает стилевой шаблон класса, присвоенного ячейке таблицы `<TD>`/`<TH>`: конструкция `<TD CLASS="text">Текст</TD>` выведет текст по умолчанию, а `<TD><FONT CLASS="text">Текст</FONT></TD>` отобразит текст заданным стилем.

Запись шаблона CSS

Рассмотрим правила составления стилевых шаблонов CSS.

Группировка и наследование

Любое правило таблицы стилей CSS состоит из *селектора* и *определения шаблона*.

Селектор — это условие отображения стиля для заданного HTML-элемента или наименование внутреннего класса (идентификатора) для него.

Определение шаблона — это описание стилевых правил для обозначенных элементов HTML, которые чередуются через точку с запятой и заключаются в фигурные скобки.

```
H3 { color: blue; font-family: Tahoma, Verdana, Arial; }
```

В данном примере селектором является элемент заголовка `H3`, для шаблона которого следует такое определение: цвет — синий, шрифт — Tahoma (либо Verdana, либо Arial).

Как видно из примера, для одного селектора приведено описание, которое содержит два правила — по цвету заголовка и наименованию гарнитуры. Это

говорит о том, что CSS позволяет группировать несколько стилевых правил для одного селектора в рамках единого описания шаблона.

Сравнив запись вида:

```
H3 { color: blue; }
H3 { font-family: Tahoma, Verdana, Arial; }
```

и

```
H3 { color: blue; font-family: Tahoma, Verdana, Arial; }
```

можно сделать вывод о том, что группировка правил по селектору позволяет, во-первых, экономить размер CSS-файла, во-вторых, систематизировать структуру описания шаблона.

Другой особенностью таблиц стилей CSS является свойство наследования стилевых правил для нескольких селекторов одновременно, например:

```
TD, TH, P, DIV { text-align: justify; color: gray: font-size: 10px; }
```

Такая запись назначает единый стиль отображения текстовой информации для элементов ячейки таблицы (`<TD>`, `<TH>`), а также параграфов (`<P>`) и блоков (`<DIV>`), а именно: тип выравнивания — по ширине, цвет — серый, размер шрифта — 10 пикселов.

Селекторы

В качестве селектора CSS могут выступать:

- элементы HTML — переопределение стиля для конкретного *элемента* страницы:

  ```
  BODY { color: orange; }
  ```

  В этом случае весь текст в пределах раздела `BODY` будет оранжевым. При добавлении, например, таблицы — назначение стилевого шаблона пропадет для текста внутри ячеек;

- классы — использование *классов* позволяет переопределять стиль как для конкретного элемента, так и для любого другого элемента, которому присвоен данный класс. Наименование класса начинается с точки и обычно пишется строчными буквами (допускается использование латинских букв и цифр; наличие специальных символов, нижних подчеркиваний и прочих нестандартных элементов не рекомендуется).

  ```
  .red { color: red; }
  ```

  В этом случае любой, позволяющий менять цвет элемент HTML будет отображаться красным цветом, если ему присвоить класс `.red`:

  ```
  <FONT CLASS="red">Текст красным цветом</FONT>
  ```

или

```
<HR CLASS="red">
```

Если мы дополним селектор класса наименованием конкретного HTML-элемента, то действие стилевого правила будет распространяться только на данный элемент:

```
HR.red { color: red; }
```

При указании классов стилевого шаблона следует внимательно следить за тем, поддерживает ли HTML-элемент присвоенный тип переопределения стиля. Например, запись вида:

```
HR { text-align: justify; }
```

будет бессмысленной, т. к. горизонтальный разделитель относится к области структурного форматирования и не может содержать текст, который, согласно стилевому правилу, следует растянуть по ширине;

- идентификаторы — запись *идентификатора* начинается с символа "#" (диез) и заканчивается наименованием:

```
#black { background-color: black; }
```

Например, присвоив данный идентификатор тегу `<TD>`, мы получим ячейку таблицы, залитую черным цветом:

```
<TD ID="black">Ячейка черного цвета</TD>
```

Сравнив функции селектора класса и идентификатора, можно задаться вполне закономерным вопросом: "Чем же различаются эти селекторы?" Действительно, формат определения селектора обоих типов аналогичен по структуре и присвоению HTML-элементам. Однако селектор идентификатора часто применяется для задания уникального имени элементу, который задействован в программном сценарии (скрипте). В отличие от него, селектор класса ограничивается, в основном, применением в стилевых шаблонах.

В заключение необходимо обратить особое внимание на невозможность сочетания селекторов различных типов. Нельзя переопределить стиль одновременно и для стандартного элемента HTML, и для него же, но по конкретному классу/идентификатору.

Псевдоклассы

Псевдоклассами называют определенные условия форматирования HTML-элемента, в соответствии с которыми браузер подставляет необходимые стилевые правила отображения данных. При этом, в исходной структуре электронного документа такие классы не присутствуют: они создаются в процессе интерпретации HTML-кода браузером.

В основном, псевдоклассы предназначены для задания различных типов форматирования по нескольким разновидностям элементов. Рассмотрим функциональность псевдоклассов на примере гиперссылок.

Согласно спецификации HTML и стандарту CSS, гиперссылка может принимать четыре состояния: непосещенная (`link`), посещенная (`visited`), активная (`active`) и ссылка при наведении курсора мыши (`hover`). Первые три состояния обычно прописываются в теге `<BODY>` HTML-документа (уровень CSS 1), четвертое относится к уровню CSS 2 и подразумевает изменение цвета ссылки при наведении на нее курсора мыши пользователя (событие работает в браузерах Internet Explorer 4 и выше, Opera 5 и выше, Netscape 6 и выше, Mozilla 1.0; браузеры более ранних версий, а также некоторые, не особенно распространенные браузеры не поддерживают состояние `hover`).

Эти состояния и будут являться псевдоклассами при записи правил отображения гиперссылок в стилевом шаблоне:

```
A:link { color: blue; }
A:active { text-decoration: underline; }
A:visited { color: gray; }
A:hover { color: orange; }
```

В данном случае все присутствующие в электронном документе гиперссылки будут отображаться в соответствии с заданным стилевым правилом. Однако часто возникает необходимость визуально выделить одни ссылки относительно других. Для этого, наряду с псевдоклассами, используются обычные селекторы классов:

```
A:active.red { color: red; }
A:hover.red { color: blue; }
A:active.white { color: white; }
A:hover.white { color: black; }
```

Области применения стилевых шаблонов CSS

При определении стилевых таблиц далеко не всегда свойства стандартного HTML-элемента соответствуют описанию шаблона стиля. Например, в HTML для жирного начертания используется тег-контейнер `<B>` (`<STRONG>`), а в CSS — конструкция `font-weight: bold;` (для элемента или селектора). Для выделения текста подчеркиванием в HTML предусмотрен тег `<U>`, в CSS же используется запись вида `text-decoration: underline;` и т. д.

В рамках данной книги, посвященной применению языка разметки HTML, сложно рассказать обо всех свойствах переопределения стиля с помощью CSS. Поэтому мы остановимся лишь на некоторых аспектах использования

каскадных таблиц стилей, а именно: на *форматировании текста*, *структурном форматировании* и организации *пользовательских форм*.

CSS в форматировании текста

CSS предоставляет разработчику электронных документов гораздо более широкий набор возможностей работы с текстовой информацией, нежели стандартный HTML. Помимо способов выделения текста (подчеркивание, курсив, жирное начертание, выбор гарнитуры и размер шрифта), с помощью средств CSS возможно изменять такие параметры, как межбуквенный и межстрочный интервалы, тип регистра (строчные и прописные буквы) и многое другое.

Единицы измерения в таблицах стилей. Все единицы измерения свойств элементов в CSS можно разделить на абсолютные и относительные (табл. 9.1).

Абсолютные единицы измерения используются в случае, когда известны характеристики устройства, которое отображает информацию.

Относительные единицы измерения определяют масштаб форматируемого элемента относительно других элементов, что позволяет сохранить первозданность документа при выводе на передающее устройство, характеристики которого заранее не известны.

Таблица 9.1. *Единицы измерения CSS*

Абсолютные	Относительные
`in` (дюйм ~ 2,5 см)	`em` (высота шрифта элемента)
`mm` (миллиметр)	`ex` (высота буквы x)
`cm` (сантиметр)	`px` (пиксел)
`pt` (пункт ~ 1/7 дюйма)	`%` (процентное соотношение)
`pc` (пика = 12 пунктов)	

В табл. 9.2 приведены наиболее распространенные свойства форматирования текста в CSS.

Таблица 9.2. *Свойства форматирования текста в CSS*

Свойство	Формат записи	Функция
`font-family`	`font-family: Tahoma, Arial;`	Выбор гарнитуры для отображения (допускается перечисление нескольких названий через запятую)

Таблица 9.2 *(окончание)*

Свойство	Формат записи	Функция
font-size	font-size: 11px;	Размер шрифта
font-style	font-style: italic;	Выбор наклона текста (курсив)
font-weight	font-weight: bold;	Наличие/отсутствие жирного начертания
font-variant	font-variant: small-caps;	Преобразование строчных букв в прописные с изменением пропорций
text-decoration	text-decoration: underline;	Подчеркивание текста
text-align	text-align: right;	Определение типа выравнивания текста
text-transform	text-transform: uppercase;	Выбор регистра букв
letter-spacing	letter-spacing: 1em;	Межбуквенный интервал
line-height	line-height: 5mm;	Межстрочный интервал
color	color: #FFFFFF;	Цвет текста
background-color	background-color: white;	Цвет заднего фона текста

Рассмотрим на примере (листинг 9.2, рис. 9.1) некоторые свойства форматирования текста в CSS.

Листинг 9.2. Форматирование текста в CSS

```
<HTML>
  <HEAD>
    <TITLE>Форматирование текста в CSS</TITLE>
<STYLE TYPE="text/css">
<!--
.text { font-family: Tahoma; color: #003366;
        line-height: 7mm; font-size: 12pt; }
#kursiv { font-style: italic; }
SPAN.font { font-size: 11px; }
.color { background-color: #CECECE; }
-->
</STYLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
```

```
<FONT CLASS="text">
CSS предоставляет разработчику электронных документов гораздо более
широкий набор возможностей работы с текстовой информацией, нежели
стандартный HTML. Помимо способов выделения текста
(<FONT STYLE="text-decoration: underline;">подчеркивание</FONT>,
<FONT ID="kursiv">курсив</FONT>, <FONT STYLE="font-weight: bold;">жирное
начертание</FONT>, <FONT STYLE="font-family: Times New Roman;">выбор
гарнитуры</FONT> и <SPAN CLASS="font">размер шрифта</SPAN>), с помощью
средств CSS возможно <FONT CLASS="color">изменять такие параметры</FONT>,
как <FONT STYLE="letter-spacing: 3px;">межбуквенный</FONT> и межстрочный
интервал, <FONT STYLE="text-transform: uppercase;">тип регистра</FONT>
(строчные и прописные буквы) и многое другое.
</FONT>
</BODY>
</HTML>
```

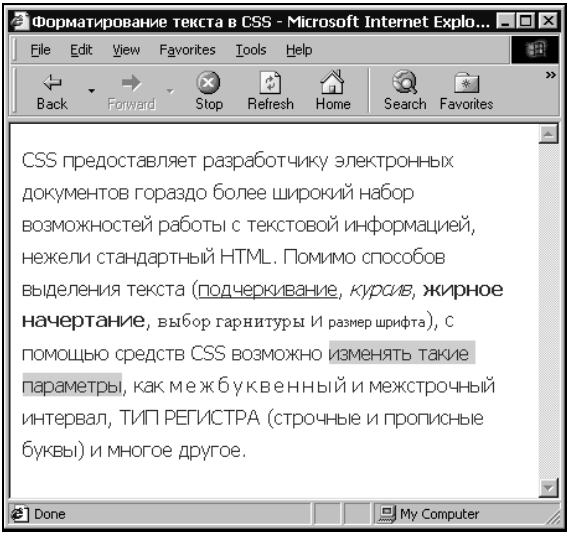


Рис. 9.1. Форматирование текста в CSS

Структурное форматирование

Каскадные таблицы стилей CSS привнесли много нового и заметно расширили функциональность форматирования структурных элементов электронного документа: `P`, `DIV` и пр.

В табл. 9.3 приведены наиболее распространенные свойства структурного форматирования в CSS.

Рассмотрим на примере (листинг 9.3, рис. 9.2) некоторые свойства структурного форматирования в CSS.

Таблица 9.3. *Свойства структурного форматирования в CSS*

Свойство	Формат записи	Функция
border-width	border-width: 20px;	Ширина границы структурного элемента
border-style	border-style: solid;	Тип декоративного отображения границы элемента
border-color	border-color: gray;	Цвет границы структурного элемента
list-style-type	list-style-type: square;	Тип нумерованного или маркированного списка
list-style-image	list-style-image: url("bullet.gif");	Указание пути к графическому маркеру
margin	margin: 1px 2px 3px 4px;	Определение размера поля относительно верхнего, правого, нижнего и левого края структурного элемента
padding	padding-top: 10em; padding-right: 25px;	Определение отступа от верхнего, правого, нижнего и левого края структурного элемента
width	width: 300px;	Ширина структурного элемента
height	height: 120px;	Высота структурного элемента
background-color	background-color: #CECECE;	Цвет заднего фона структурного элемента
float	float: left;	"Плавающее" расположение структурного элемента относительно других элементов

Листинг 9.3. Структурное форматирование в CSS

```
<HTML>
  <HEAD>
    <TITLE>Структурное форматирование в CSS</TITLE>
<STYLE TYPE="text/css">
<!--
.text { font-family: Tahoma; color: #003366;
        line-height: 7mm; font-size: 12pt;
        margin: 10px 10px 10px 10px; padding-left: 15px;
        padding-right: 15px; border-color: black;
        border-style: dotted; background-color: #C0C0C0;
        width: 500px; text-align: justify; }
```

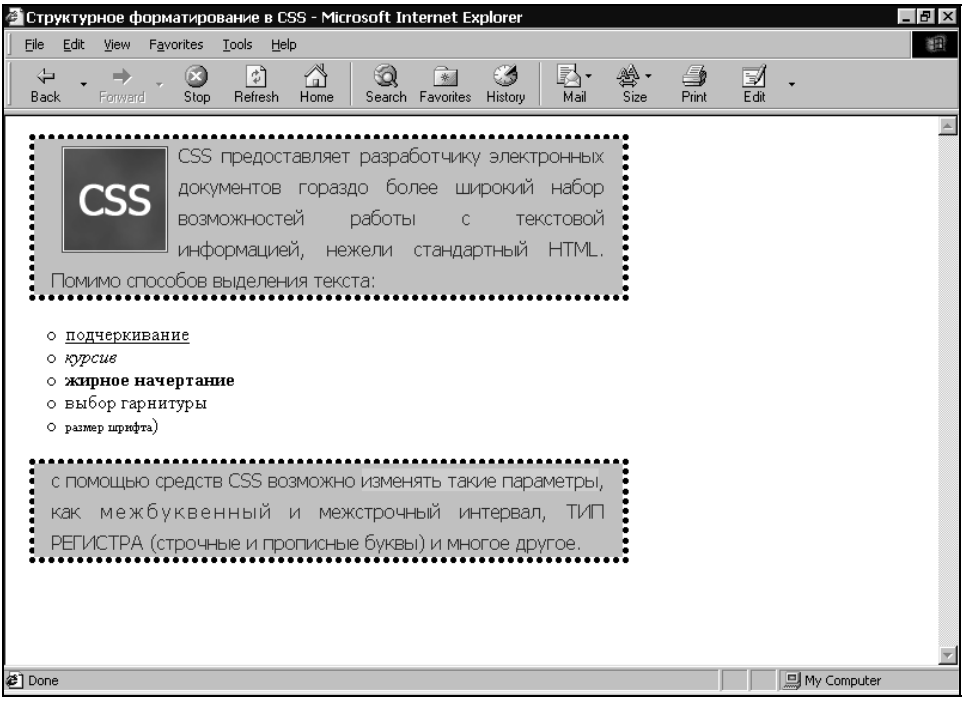


Рис. 9.2. Структурное форматирование в CSS

```
#kursiv { font-style: italic; }
SPAN.font { font-size: 11px; }
.color { background-color: #CECECE; }
IMG { float: left; width: 100px; height: 100px; }
LI {list-style-type: circle; }
-->
</STYLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<P CLASS="text">
<IMG SRC="picture.gif">CSS предоставляет разработчику электронных
документов гораздо более широкий набор возможностей работы с текстовой
информацией, нежели стандартный HTML. Помимо способов выделения текста:
</P>
<UL>
<LI><FONT STYLE="text-decoration: underline;">подчеркивание</FONT>
<LI><FONT ID="kursiv">курсив</FONT>
<LI><FONT STYLE="font-weight: bold;">жирное начертание</FONT>
<LI><FONT STYLE="font-family: Times New Roman;">выбор гарнитуры</FONT>
```

```
<LI><SPAN CLASS="font">размер шрифта</SPAN>)
</UL>
<P CLASS="text">
с помощью средств CSS возможно <FONT CLASS="color">изменять такие
параметры</FONT>, как <FONT STYLE="letter-spacing:
3px;">межбуквенный</FONT> и межстрочный интервал,
<FONT STYLE="text-transform: uppercase;">тип регистра</FONT> (строчные
и прописные буквы) и многое другое.
</P>
</BODY>
</HTML>
```

Как видно из примера, некоторые структурные элементы позволяют применять свойства CSS, относящиеся к форматированию текста: `text-align`, `background-color` и др. Таким образом, при формировании таблиц стилей допускается сочетание свойств CSS как для текстового, так и для структурного форматирования.

Пользовательские формы в CSS

Язык разметки HTML предлагает разработчику электронных документов широкий набор возможностей по организации пользовательских форм. Однако при своей развитой функциональности внешний вид всевозможных текстовых полей, переключателей, ниспадающих меню зачастую оставляет желать лучшего. Это досадное упущение компенсируется посредством таблиц стилей CSS.

В табл. 9.4 приведены наиболее распространенные свойства пользовательских форм в CSS.

Таблица 9.4. *Свойства пользовательских форм в CSS*

Свойство	Формат записи	Функция
background-color	background-color: white;	Цвет заднего фона элемента формы
border-width	border-width: 1px;	Толщина рамки вокруг элемента формы
border-style	border-style: solid;	Тип декоративного отображения рамки вокруг элемента формы
border-color	border-color: #003366;	Цвет рамки вокруг элемента формы
border-top border-right border-bottom border-left	border-top: 1px; border-right: 2px; border-bottom: 3px; border-left: 4px;	Толщина рамки для верхней, правой, нижней и левой части элемента формы

Таблица 9.4 *(окончание)*

Свойство	Формат записи	Функция
font-family font-size font-weight	font-family: Tahoma; font-size: 12em; font-weight: bold;	Свойства текста внутри элемента формы (аналогичны свойствам форматирования обычного текста)
margin	margin: 2px 2px 2px 2px	Размер поля вокруг элемента формы (верх, право, низ, лево)
padding	padding-top: 1px; padding-right: 2px;	Размер внутреннего отступа от края элемента формы
width	width: 200px;	Ширина элемента формы
height	height: 50px;	Высота элемента формы

Рассмотрим на примере (листинг 9.4, рис. 9.3) некоторые свойства пользовательских форм в CSS.

Листинг 9.4. Пользовательские формы в CSS

```
<HTML>
  <HEAD>
    <TITLE>Пользовательские формы в CSS</TITLE>
<STYLE TYPE="text/css">
<!--
INPUT.text { font-family: Tahoma; color: #003366;
             font-size: 12px; padding-left: 5px;
             padding-right: 5px; border-color: black;
             border-style: solid; border-width: 1px;
             background-color: #F9F9F9; width: 250px; }

TEXTAREA. field { font-family: Verdana, Arial;
                  color: #003366; font-size: 12px;
                  padding-left: 5px; padding-right: 5px;
                  border-color: black; border-style: solid;
                  border-width: 1px; background-color: #F9F9F9; }

INPUT.button { font-family: Tahoma; color: #FFCC00;
               font-size: 11px; font-weight: bold;
               border-color: white; border-style: solid;
               border-width: 1px; background-color: gray;
               width: 200px; }
```

```
BODY, TD { font-family: Tahoma; color: #003366;
           font-size: 11px; }

TH { color: #FFFFFF; }
-->
</STYLE>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<H2 ALIGN="center">ПОЛЬЗОВАТЕЛЬСКАЯ АНКЕТА</H2>
<FORM NAME="user_anketa" ACTION="/cgi-bin/anketa.cgi" METHOD="post"
ENCTYPE="multipart/form-data">
<TABLE BORDER="0" CELLPADDING="5" CELLSPACING="3" ALIGN="center"
WIDTH="98%">
<TR>
<TH BGCOLOR="#999999" WIDTH="50%">Личные данные</TH>
<TH BGCOLOR="#999999" WIDTH="50%">Увлечения</TH>
</TR>
<TR>
<TD VALIGN="top" BGCOLOR="#EEEEEE" WIDTH="50%">
Ваше имя:<BR>
<INPUT TYPE="text" SIZE="30" MAXLENGTH="35" NAME="name" CLASS="text">
<BR>
Ваш пароль:<BR>
<INPUT TYPE="password" SIZE="30" MAXLENGTH="35" NAME="password"
CLASS="text">
<BR>
Ваша фотография:<BR>
<INPUT TYPE="file" SIZE="18" NAME="photo" CLASS="text">
</TD>
<TD VALIGN="top" BGCOLOR="#EEEEEE" WIDTH="50%">
<INPUT TYPE="checkbox" NAME="hobby" VALUE="computers" CHECKED>Компьютеры
и Интернет
<BR>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="art">Литература и искусство
<BR>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="music">Музыка и танцы
<BR>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="auto">Автомобили
<BR>
<INPUT TYPE="checkbox" NAME="hobby" VALUE="sport">Спорт и активный отдых
</TD>
</TR>
<TR>
```

```
<TH BGCOLOR="#999999" WIDTH="50%">Место жительства</TH>
<TH BGCOLOR="#999999" WIDTH="50%">Комментарии</TH>
</TR>
<TR>
<TD VALIGN="top" BGCOLOR="#EEEEEE" WIDTH="50%">
<SELECT NAME="city" SIZE="5" STYLE="font-family: Verdana; font-size:
11px;">
 <OPTION VALUE="" STYLE="background-color: #CECECE;">--- Выберите город -
--
 <OPTION VALUE="Moscow">Москва
 <OPTION VALUE="Spb" STYLE="background-color: #CECECE;">Санкт-Петербург
 <OPTION VALUE="Samara">Самара
 <OPTION VALUE="Ryazan" STYLE="background-color: #CECECE;">Рязань
 <OPTION VALUE="Krasnoyarsk">Красноярск
 <OPTION VALUE="Komsomolsk" STYLE="background-color:
#CECECE;">Комсомольск-на-Амуре
 <OPTION VALUE="Vladivostok">Владивосток
</SELECT>
<BR><BR>
<SELECT STYLE="font-family: Verdana; font-size: 11px;">
 <OPTION VALUE="">--- Выберите страну ---
 <OPTION VALUE="Russia">Россия
 <OPTION VALUE="England">Англия
 <OPTION VALUE="USA">США
 <OPTION VALUE="Germany">Германия
 <OPTION VALUE="France">Франция
</SELECT>
</TD>
<TD VALIGN="top" BGCOLOR="#EEEEEE" WIDTH="50%">
<TEXTAREA COLS="25" ROWS="5" NAME="comments" CLASS="field"></TEXTAREA>
<BR>
<INPUT TYPE="radio" NAME="site" VALUE="first" CHECKED>Я впервые
на этом сайте
<BR>
<INPUT TYPE="radio" NAME="site" VALUE="not_first">Я тут не в первый раз
</TD>
</TR>
<TR>
<TD COLSPAN="2" ALIGN="center" BGCOLOR="#999900">
<INPUT TYPE="submit" VALUE="Отправить" CLASS="button"> 
<INPUT TYPE="reset" VALUE="Очистить" CLASS="button">
</TD>
</TR>
</TABLE>
```

```
</FORM>
</BODY>
</HTML>
```

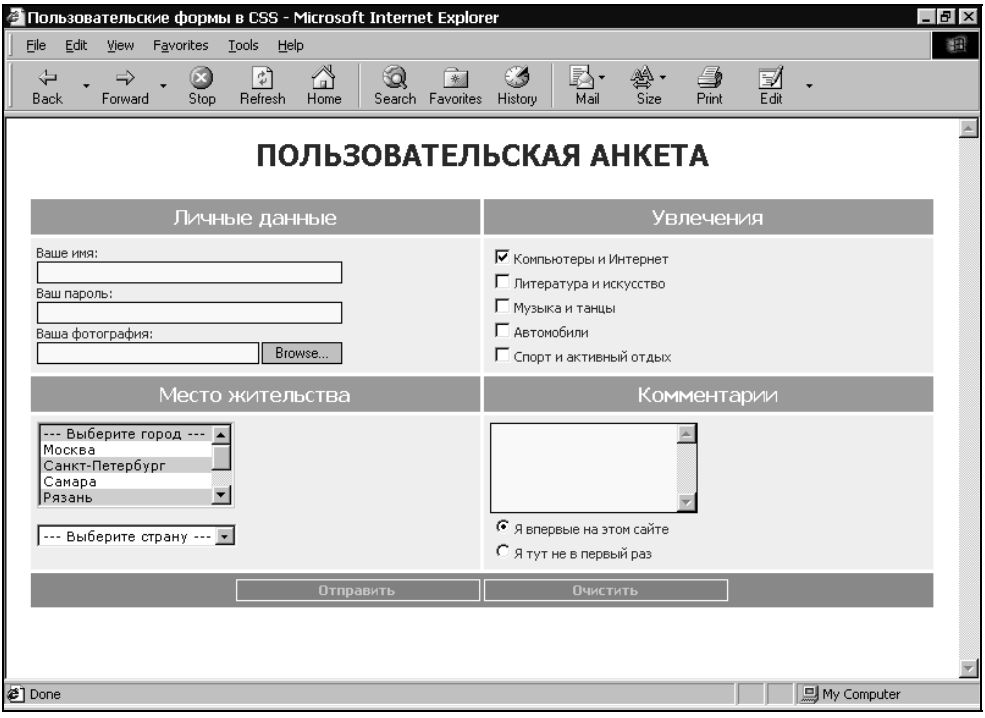


Рис. 9.3. Пользовательские формы в CSS

Чтобы понять превосходство стилевых шаблонов в организации пользовательских форм, сравните рис. 9.3 (применение таблиц стилей CSS) с рис. 8.16 (стандартные элементы форм HTML).

Однако и здесь есть "подводные камни". Если, к примеру, Internet Explorer поддерживает практически все свойства CSS 1 и CSS 2, то другие браузеры (в особенности более ранних версий) могут искажать или попросту не отображать стилевые шаблоны.

Из рис. 9.4 видно, что Opera 6 не отображает цвет заднего фона для элементов списка `SELECT`, не поддерживает свойство `border-style` в текстовых полях `INPUT`, свойство `color` в наименовании кнопок подтверждения и отмены и пр.

Браузер Netscape 6.2 более лоялен к отображению свойств стилевых шаблонов CSS, но почему-то отказывается применять стили для конструкции `INPUT TYPE="file"` (загрузка файла с локального компьютера) (рис. 9.5).

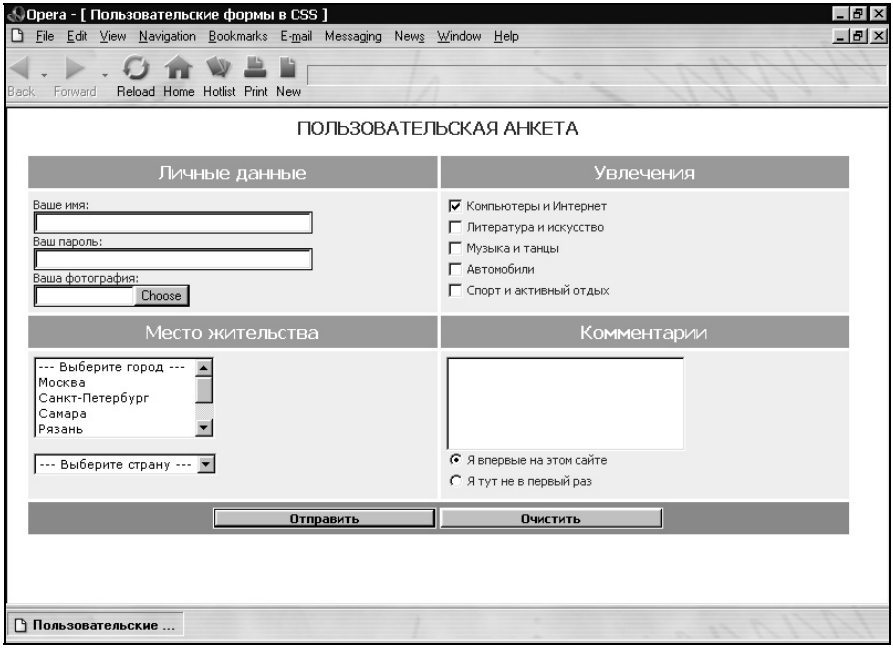


Рис. 9.4. Пользовательские формы в браузере Opera 6 (реализация с CSS)

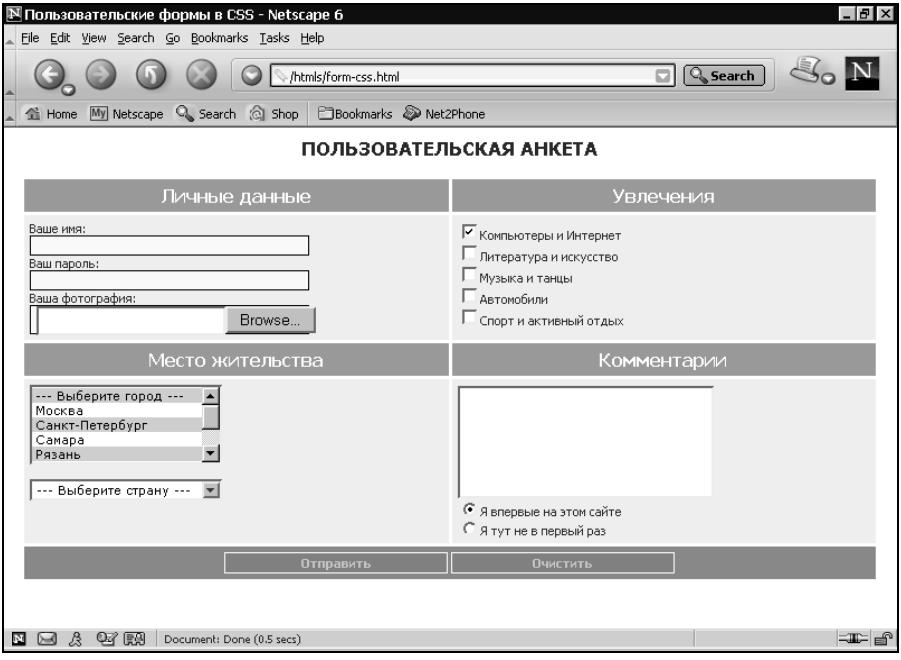


Рис. 9.5. Пользовательские формы в браузере Netscape 6.2 (реализация с CSS)

В отличие от своей более усовершенствованной версии, Netscape Navigator 4.7 выводит на экран полное нарушение стилей и компоновки электронного документа (рис. 9.6).

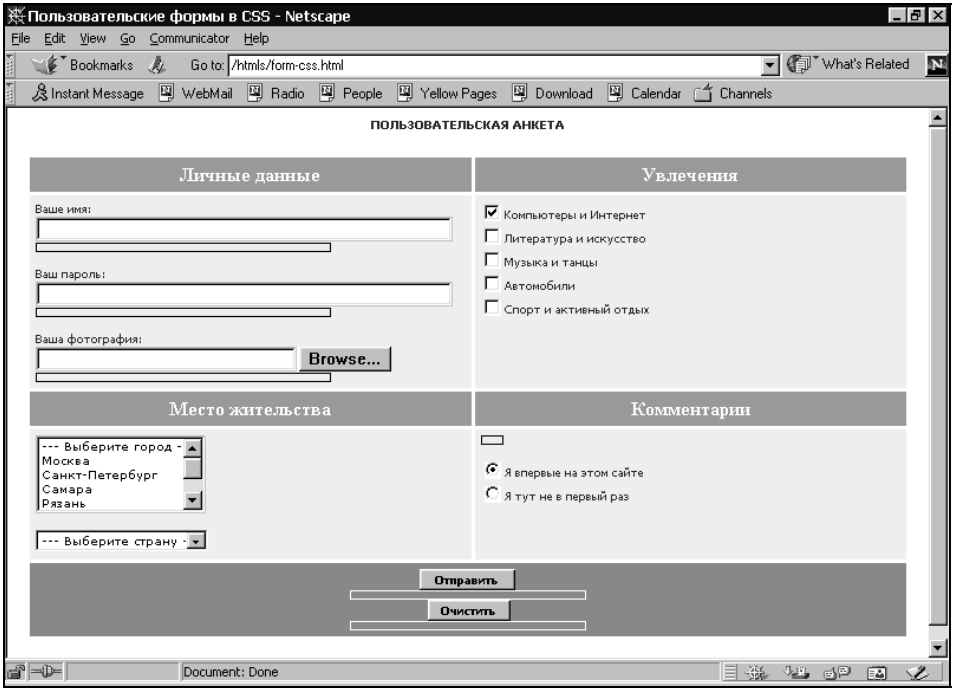


Рис. 9.6. Пользовательские формы в браузере Netscape Navigator 4.7 (реализация с CSS)

Позиционирование объектов

Исходя из концепции языка разметки HTML, все элементы документа выводятся браузером в той последовательности, в какой размещены теговые конструкции в коде. CSS 2 позволяет задавать порядок и последовательность отображения тех или иных HTML-элементов в зависимости от определенных событий на странице или манипуляций, осуществляемых со стороны пользователя.

Другими словами, с помощью средств CSS можно прибегнуть к позиционированию (пространственному расположению) необходимых объектов в пределах электронного документа.

Существует два типа визуального позиционирования элементов: абсолютное и относительное. Для описания каждого типа используются специальные свойства `position`, `top` и `left`.

Абсолютное позиционирование

Абсолютное позиционирование подразумевает четкое фиксирование выбранного элемента на странице, независимо от остальных элементов документа, например:

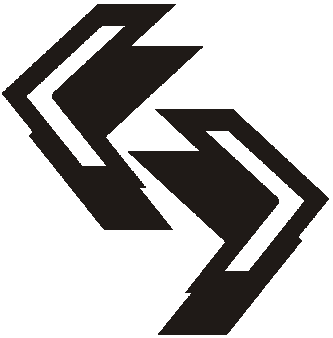
```
<IMG SRC="picture.gif" WIDTH="100" HEIGHT="100" ALT="Рисунок"
STYLE="position: absolute; top: 10px; left: 25px;">
```

В данном случае графическое изображение абсолютно спозиционировано и размещается в 10 пикселах от верхнего и в 25 пикселах от левого края своего родительского элемента (в качестве родительского элемента выступает верхняя левая точка структуры документа).

Относительное позиционирование

Относительное позиционирование позволяет расположить выбранный объект в зависимости от размещения остальных элементов документа (т. е. относительно других объектов страницы), например:

```
<DIV ID="text" STYLE="position: relative; top: 50px; left: 50px;">
```



ЧАСТЬ II

Практикум мастера

Глава 10. Dynamic HTML

Глава 11. Концепция Web-сайта

Глава 12. Композиция Web-сайта

Глава 13. Навигация по Web-сайту

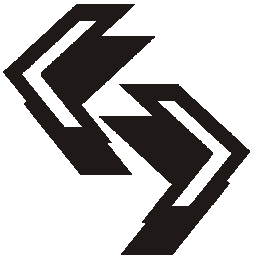
Глава 14. Шаблоны ошибок

Глава 15. Разработка графического макета будущего Web-сайта

Глава 16. HTML-верстка графического макета: шаг за шагом

Глава 17. Продвижение Web-сайта в Интернете

ГЛАВА 10



Dynamic HTML

Общая информация

Dynamic HTML, или динамический HTML, не является отдельным языком программирования или разметки документа. Это всего лишь технология, реализующая электронные документы с динамически изменяющимся содержанием.

Реализация динамического HTML строится на трех компонентах:

- HTML — HyperText Markup Language, простой язык разметки гипертекстовых документов;
- CSS — Cascading Style Sheets, каскадные таблицы стилей HTML-документа;
- JavaScript — клиентский язык программирования (выполняется не на сервере, а непосредственно в браузере пользователя на его локальном компьютере). Возможные аналоги: VBScript, JScript.

Эти три компонента образуют важнейшую структуру под названием Document Object Model, DOM (Объектная модель документа), которая дополняет простоту HTML и изящество CSS возможностью динамического изменения содержания без перезагрузки электронного документа.

К сожалению, объектная модель DHTML (Dynamic HTML), входящая в состав Internet Explorer, начиная с версии 4.01, по многим параметрам не поддерживается браузерами других производителей (например, Opera или Netscape), что накладывает определенные ограничения на процесс разработки интерактивных документов с динамическим содержанием.

Создание визуальных эффектов

Динамический HTML дает возможность разработчику создать в пределах электронного документа рабочий инструментарий, позволяющий пользователю манипулировать содержанием страницы, видом и расположением объектов и элементов и т. д.

Рассмотрим действие Dynamic HTML на примере динамических информационных блоков и применения визуальных фильтров для графических изображений.

Динамические блоки

Область применения *динамических информационных блоков* чрезвычайно широка, поэтому остановимся лишь на одном, достаточно распространенном случае.

Предположим, на сайте расположена карта России с нанесенными на ней названиями городов. Посетитель должен быстро и в удобном виде получить информацию по каждому из городов.

Какие могут быть способы реализации данной задачи? Разрезать карту на части или задать активные области для карты-изображения (Imagemap)? Неудобно, т. к. каждый раз посетитель будет переходить по ссылке и назад, что снова потребует загрузки графического файла.

Поместить нужный текст в поле `ALT`, чтобы тот отображался в виде всплывающей подсказки при наведении курсора? Бессмысленно, т. к. подсказка через несколько секунд исчезнет, а текста может быть много.

Самый подходящий и эффективный способ — прибегнуть к помощи Dynamic HTML в совокупности с Imagemap (листинг 10.1). В этом случае при наведении курсора мыши на заданную активную область карты в определенном месте страницы появляется информация о городе (как текстовая, так и графическая). Такой подход имеет два важных достоинства: корректная работа и в Internet Explorer, и в Netscape Navigator и экономия времени посетителя (графика загружается только один раз, а все действия выполняются непосредственно из кода текущего электронного документа).

Прежде всего необходимо определить JavaScript-сценарий в разделе `HEAD`.

Листинг 10.1. JavaScript-сценарий для динамических информационных блоков

```
<SCRIPT LANGUAGE="JavaScript1.2">
<!--
var ie = document.all ? 1 : 0;
```

```
var ns = document.layers ? 1 : 0;
var topcss = 165;
if (ns)
topcss = 200;
function showLayer(name)
{
if (ie)
document.all[name].style.visibility = "visible";
else if (ns)
document.layers[name].visibility = "show";
}
function hideLayer(name)
{
if (ie)
document.all[name].style.visibility = "hidden";
else if (ns)
document.layers[name].visibility = "hide";
}
//-->
</SCRIPT>
```

Далее определяем координаты информационных слоев (листинг 10.2), которые невидимы до тех пор, пока курсор мыши не переместится над активной областью (размещаются в начале раздела BODY).

Листинг 10.2. Координаты информационных слоев

```
<STYLE TYPE="text/css">
<!--
#link1
     { position: absolute; left: 160px; top: 250px; visibility: hidden; }
#link2
     { position: absolute; left: 160px; top: 250px; visibility: hidden; }
#link3
     { position: absolute; left: 160px; top: 250px; visibility: hidden; }
#link4
     { position: absolute; left: 160px; top: 250px; visibility: hidden; }
//-->
</STYLE>
```

В данном случае в качестве селектора стилевого шаблона используется идентификатор. Поле

```
#link3
     {position: absolute; left: 160px; top: 250px; visibility: hidden; }
```

означает, что при активации области link3 скрытый до этого момента информационный слой, соответствующий ей, появится на странице на расстоянии 160 пикселов от левой границы окна и 250 пикселов — от его верхнего края.

Далее переходим к наполнению самих информационных слоев (листинг 10.3), которые в плане HTML лучше всего реализовать с помощью структурного тега <DIV>.

Листинг 10.3. Наполнение информационных слоев

```
<DIV ID='link1'>Санкт-Петербург</DIV>
<DIV ID='link2'>Ростов-на-Дону</DIV>
<DIV ID='link3'>Красноярск</DIV>
<DIV ID='link4'>Владивосток</DIV>
```

Внутри тега-контейнера <DIV> можно разместить таблицы, графические объекты, нумерованные и маркированные списки и пр.

Наконец, последнее, что нужно сделать, — задать активные области при помощи Imagemap и привязать их к функциям сценария по активации/дезактивации слоев (листинг 10.4).

Листинг 10.4. Определение активных областей Imagemap и привязка к JavaScript-сценарию

```
<IMG SRC="russiamap.gif" WIDTH="400" HEIGHT="230" BORDER="0"
USEMAP="#russia">
<MAP NAME="russia">
<AREA SHAPE="rect" COORDS="63,70,162,85" HREF="link1.html"
onMouseOver="showLayer('link1');" onMouseOut="hideLayer('link1');">
<AREA SHAPE="rect" COORDS="98,137,193,157" HREF="link2.html"
onMouseOver="showLayer('link2');" onMouseOut="hideLayer('link2');">
<AREA SHAPE="rect" COORDS="249,125,326,141" HREF="link3.html"
onMouseOver="showLayer('link3');" onMouseOut="hideLayer('link3');">
<AREA SHAPE="rect" COORDS="289,89,368,105" HREF="link4.html"
onMouseOver="showLayer('link4');" onMouseOut="hideLayer('link4');">
</MAP>
```

Конструкции onMouseOver и onMouseOut являются событиями языка JavaScript (наведение и снятие курсора мыши с активной области на карте). Каждой активной области с заданными координатами должен соответствовать информационный блок со своим идентификатором (link1, link2, link3 и т. д.).

В результате при перемещении курсора, например над надписью "Санкт-Петербург" на карте, аналогичная надпись, но уже в текстовом виде, появля-

ется прямо под изображением карты России (рис. 10.1, листинг 10.5). Как уже было сказано, наполнение слоев можно осуществлять в соответствии с индивидуальными предпочтениями и оформительскими требованиями: вместо текстовых блоков вставлять графику, таблицы, гиперссылки и др.

Рис. 10.1. Применение динамических информационных блоков (в примере использованы некоторые стили CSS)

Листинг 10.5. Применение динамических информационных блоков (итоговый листинг)

```
<HTML>
  <HEAD>
    <TITLE>Применение динамических информационных блоков</TITLE>
<SCRIPT LANGUAGE="JavaScript1.2">
<!--
var ie = document.all ? 1 : 0;
var ns = document.layers ? 1 : 0;
var topcss = 165;
if (ns)
topcss = 200;
function showLayer(name)
```

```
{
if (ie)
document.all[name].style.visibility = "visible";
else if (ns)
document.layers[name].visibility = "show";
}
function hideLayer(name)
{
if (ie)
document.all[name].style.visibility = "hidden";
else if (ns)
document.layers[name].visibility = "hide";
}
//-->
</SCRIPT>
  </HEAD>
<BODY BGCOLOR="#FFFFFF" TEXT="black" LINK="#00FF00" ALINK="#00FF00"
VLINK="blue">
<STYLE TYPE="text/css">
<!--
#link1
     { position: absolute; left: 160px; top: 250px; visibility: hidden; }
#link2
     { position: absolute; left: 160px; top: 250px; visibility: hidden; }
#link3
     { position: absolute; left: 160px; top: 250px; visibility: hidden; }
#link4
     { position: absolute; left: 160px; top: 250px; visibility: hidden; }
//-->
</STYLE>
<DIV ID='link1'>Санкт-Петербург</DIV>
<DIV ID='link2'>Ростов-на-Дону</DIV>
<DIV ID='link3'>Красноярск</DIV>
<DIV ID='link4'>Владивосток</DIV>
<IMG SRC="russiamap.gif" WIDTH="400" HEIGHT="230" BORDER="0"
USEMAP="#russia">
<MAP NAME="russia">
<AREA SHAPE="rect" COORDS="63,70,162,85" HREF="link1.html"
onMouseOver="showLayer('link1');" onMouseOut="hideLayer('link1');">
<AREA SHAPE="rect" COORDS="98,137,193,157" HREF="link2.html"
onMouseOver="showLayer('link2');" onMouseOut="hideLayer('link2');">
<AREA SHAPE="rect" COORDS="249,125,326,141" HREF="link3.html"
onMouseOver="showLayer('link3');" onMouseOut="hideLayer('link3');">
```

```
<AREA SHAPE="rect" COORDS="289,89,368,105" HREF="link4.html"
onMouseOver="showLayer('link4');" onMouseOut="hideLayer('link4');">
</MAP>
</BODY>
</HTML>
```

Визуальные фильтры

В Dynamic HTML под фильтром принято понимать некую функцию, так или иначе преобразующую визуальное представление элемента на Web-странице. Преобразование осуществляется непосредственно в браузере, т. е. на стороне клиента (подключение сервера не требуется).

Следует отметить, что применение визуальных фильтров возможно только в браузерах Internet Explorer 4.x и выше (последняя версия Netscape, по утверждению разработчиков, способна поддерживать только лишь часть функциональных возможностей DHTML). Тем не менее, для горячих поклонников браузера от Microsoft или людей, определяющих предпочтения потенциальной аудитории разрабатываемых Web-узлов в соответствии со статистикой различных серверных анализаторов и результатами исследований, средства динамического языка разметки гипертекста будут как нельзя более кстати и смогут существенно улучшить визуальное представление данных, размещаемых в электронных документах.

Прежде всего, визуальные динамические фильтры можно применить не ко всем элементам HTML-документа (табл. 10.1), а только к тем, которые способны определять блок прямоугольного вида при интерпретации браузером и при этом сами не являются окнами (к примеру, "плавающие" фреймы).

Таблица 10.1. *Элементы визуальных динамических фильтров*

Наименование элемента	Режим применения
BODY	Всегда
BUTTON	Всегда
DIV	При заданных параметрах ширины и высоты или использовании абсолютного позиционирования
IMG	Всегда
INPUT	Всегда
SPAN	При заданных параметрах ширины и высоты или использовании абсолютного позиционирования
TABLE	Всегда

Таблица 10.1 *(окончание)*

Наименование элемента	Режим применения
TD, TH	Всегда
TEXTAREA	Всегда

Формат записи фильтра достаточно прост и аналогичен правилам задания свойств элементов с помощью тега <STYLE>. Запись производится в следующем виде:

```
filter: название_фильтра(параметры)
```

где параметры определяются в стандартном для HTML виде:

```
название_параметра=значение_параметра
```

Прежде чем перейти к рассмотрению фильтров и возможностей их применения, необходимо сказать, что, во-первых, допускается использование сразу нескольких фильтров (если это не противоречит окончательному визуальному результату отображения элемента под воздействием наложенных фильтров), во-вторых, при указании фильтров, не имеющих никаких параметров, присутствие круглых скобок (без пробелов) после названия фильтра обязательно.

Общие свойства и описание некоторых фильтров

При использовании фильтров следует помнить, что некоторые из них имеют общие свойства, влияющие на характер действия заданных параметров фильтра. К таковым относятся enabled (со значениями true и false, соответственно, разрешающим или запрещающим применение присоединенного к элементу документа фильтра), direction (определяет направление действия таких фильтров, как shadow, blur и др.), strength (задает интенсивность действия фильтра со значением от 0 до 255) и т. п. Часть фильтров, помимо общих свойств, имеет различные методы их определения (например, фильтр light).

Характеристика свойств и методов некоторых визуальных фильтров приведена в табл. 10.2.

Браузер Internet Explorer 4.x (и выше) поддерживает достаточно большое количество фильтров, широкие возможности по настройке которых представляют применение этой технологии в более выгодном свете (табл. 10.3).

Таблица 10.2. *Характеристика свойств и методов некоторых визуальных фильтров*

Название фильтра	Свойства	Методы
blur	add, direction, enabled, strength	—
chroma	color, enabled	—
glow	color, enabled, strength	—
invert	enabled	—
light	enabled	addAmbient, addCone, addPoint, moveLight и др.
redirect	enabled	elementImage
shadow	color, direction, enabled	—
xray	enabled	—

Таблица 10.3. *Характеристика визуальных фильтров, работающих в браузере Internet Explorer*

Название фильтра	Описание действия
alpha	Определение степени прозрачности объекта
blendTrans	Настройка контрастности отображения объекта
blur	Размытие объекта
chroma	Установление прозрачности пикселам заданного цвета
dropShadow	Создание сплошного силуэта объекта
flipH	Зеркальное отображение объекта по горизонтали
flipV	Зеркальное отображение объекта по вертикали
glow	Создание эффекта свечения внешних границ объекта
gray	Отображение объекта в серых тонах
invert	Инвертирование цветовой гаммы объекта
light	Создание эффекта освещения объекта
mask	Создание прозрачной маски из непрозрачных пикселов объекта
redirect	Преобразование объекта в элемент DAImage (технология Microsoft DirectAnimation)

Таблица 10.3 *(окончание)*

Название фильтра	Описание действия
revealTrans	Эффект появления/исчезновения объекта
shadow	Создание силуэта объекта вдоль его границы
wave	Эффект искривления объекта по вертикали
xray	Изменение глубины цвета объекта (эффект рентгеновского снимка)

Рассмотрим на примере возможные варианты действия некоторых фильтров.

Фильтр *WAVE*

Фильтр wave создает синусоидальное искривление объекта в вертикальном направлении и имеет свойства add, enabled, freq, lightStrength, phase и strength.

Свойства enabled и strength были описаны выше, а остальные имеют следующее назначение:

- add — определяет необходимость добавления исходного вида объекта в его отфильтрованную интерпретацию;
- freq — задает количество максимумов в волне искривления объекта;
- lightStrength — добавляет эффект трехмерности гребням волны искажения объекта;
- phase — определяет фазу смещения волны (ее значение задается в процентах относительно начальной фазы, равной 0).

На рис. 10.2 показано три вида объекта — слева направо:

- исходный объект без действия фильтра wave
- объект с действием фильтра wave и заданными свойствами strength=5, add=0, lightStrength=20
- объект с заданными свойствами strength=5, add=0, phase=50, lightStrength=20, freq=30

Листинг 10.6. Использование визуального фильтра wave

```
<!-- Рисунок 1 -->
<IMG SRC="flower.jpg" WIDTH="243" HEIGHT="262">
<!-- Рисунок 2 -->
<IMG SRC="flower.jpg" WIDTH="243" HEIGHT="262" STYLE="filter:
wave(strength=5, add=0, lightstrength=20)">
```

```
<!-- Рисунок 3 -->
<IMG SRC="flower.jpg" WIDTH="243" HEIGHT="262" STYLE="filter:
wave(strength=5, add=0, phase=50, lightstrength=20, freq=30)">
```

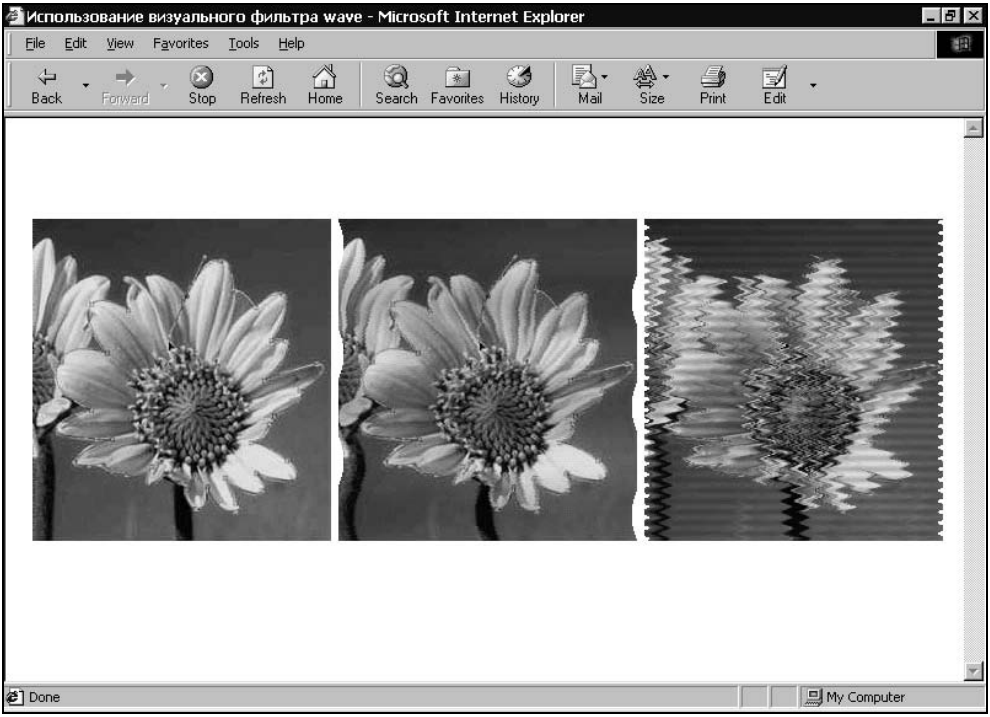


Рис. 10.2. Использование визуального фильтра `wave`

Фильтр *ALPHA*

Данный фильтр использует такие свойства, как `opacity`, `style` и `finishOpacity`. Свойство `opacity` позволяет определить степень прозрачности объекта. Свойство `style` может принимать значения от 0 до 3: нулевое значение соответствует степени прозрачности объекта согласно значению свойства `opacity`.

Конструкция `style=1` задает координаты точек прямой, на которой степень прозрачности соответствует значению свойства `finishOpacity`.

При `style=2` степень прозрачности изменяется от центра объекта к вымышленной окружности, спроецированной на блок отображения объекта (заданный отрезок от значения `opacity` до значения `finishOpacity`). Если свойство `style=3`, прозрачность меняется от центра объекта к его границам.

На рис. 10.3 представлено три примера действия фильтра `alpha`, слева направо:

- `style=0, opacity=60`
- `style=3, opacity=90, finishOpacity=0`
- `style=2, opacity=90, finishOpacity=0`

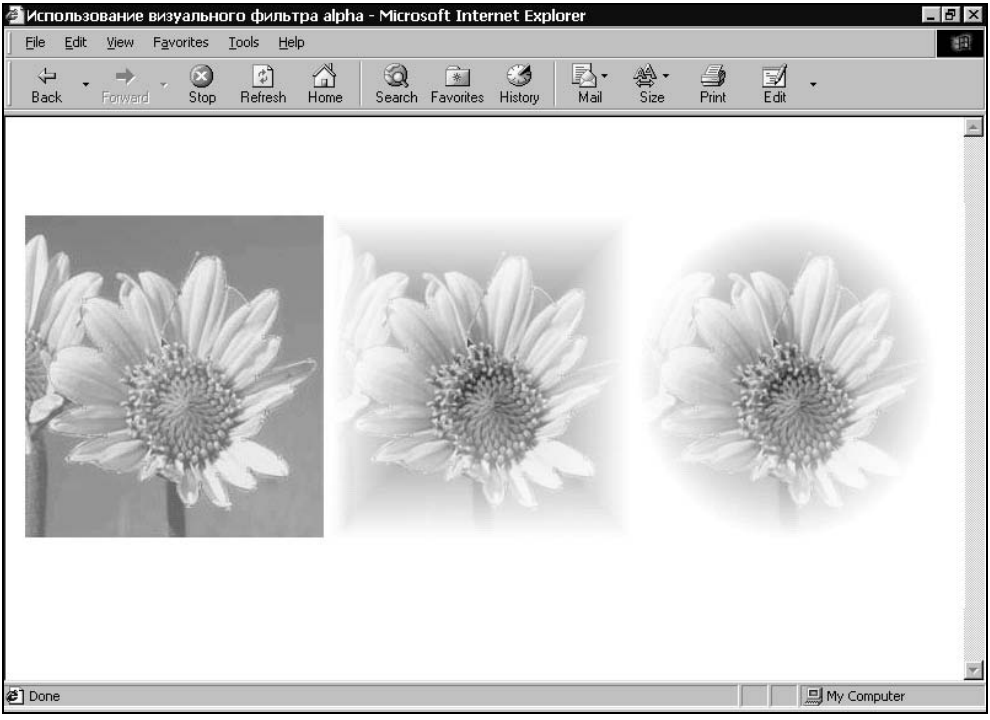


Рис. 10.3. Использование визуального фильтра `alpha`

Фильтр *dropShadow*

Фильтр `dropShadow` создает сплошной силуэт объекта, смещенный в заданном направлении, и образует эффект тени.

Кроме `enabled`, имеет следующие свойства: `color` (определение цвета тени), `positive` (выбор между тенью от прозрачных или непрозрачных точек объекта), `offX` и `offY` (расстояние в пикселах в вертикальной и горизонтальной плоскостях, на котором будет отображаться тень от исходного объекта: положительные значения `offX` и `offY` смещают тень, соответственно, вправо и вниз, отрицательные — влево и вверх.

Фильтр *LIGHT*

Данный фильтр имеет единственное свойство `enabled` и несколько интересных методов, позволяющих варьировать способы освещения выбранного объекта:

- `addAmbient` — помещает над объектом источник рассеянного света и требует задания значений по шкале RGB и степени интенсивности наложения `iStrength` (от 0 до 255);
- `addCone` — направляет на объект конический источник света и имеет следующие значения: координаты источника света `iX1` и `iY1`, номер слоя `iZ`, координаты точки направления света `iX2` и `iY2`, цветовую гамму по шкале RGB (`iRed`, `iGreen`, `iBlue`), интенсивность источника света `iStrength` и величину угла конуса источника света `iAngle`;
- `addPoint` — имеет те же значения, что и `addCone`, кроме `iX2` и `iY2`, и добавляет к объекту точечный источник света.

Это наиболее популярные методы, визуализацию двух первых из них можно увидеть на рис. 10.4.

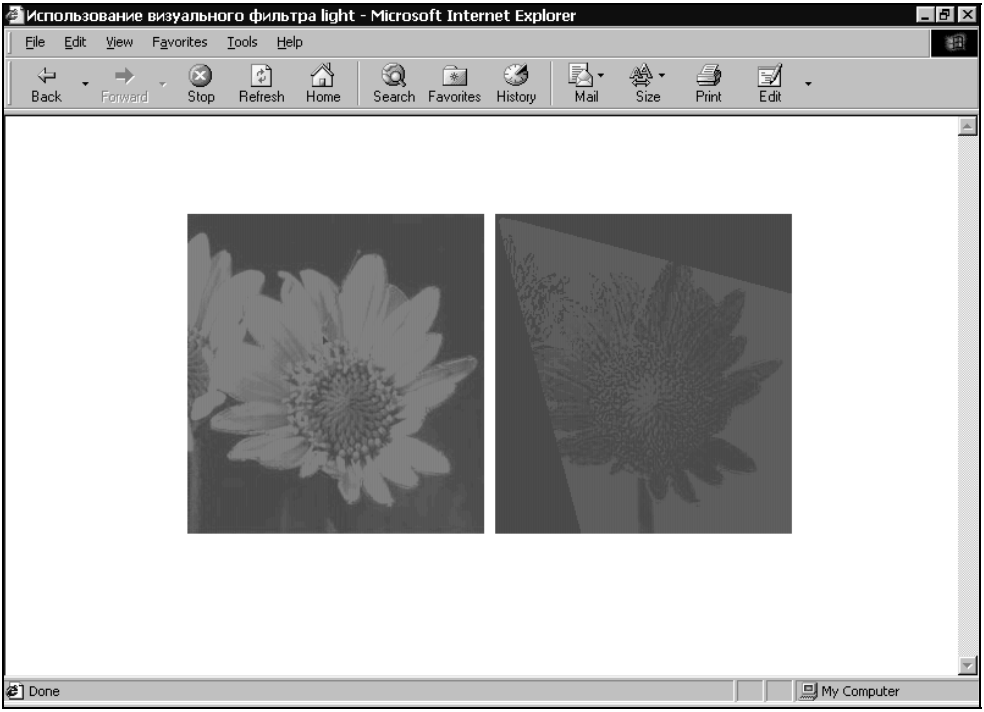


Рис. 10.4. Использование визуального фильтра `light`

Для реализации фильтра `light` в код HTML-документа обычно включают небольшой JavaScript-сценарий (раздел `HEAD`):

Листинг 10.7. Использование визуального фильтра `light`

```
<SCRIPT LANGUAGE="JavaScript">
function lightFilter()
{
ambient.filters.light.addAmbient(255,0,0,255);
cone.filters.light.addCone(0,0,3,255,255,0,0,255,200,30);
}
</SCRIPT>
```

В основной HTML-код добавляем идентификаторы графических изображений `ambient` и `cone` и указываем фильтр `light`, а также инициализируем функцию фильтра `lightFilter()` посредством события `onLoad` в разделе `BODY`:

```
<BODY onLoad="lightFilter();">
<IMG SRC="flower.jpg" WIDTH="243" HEIGHT="262" ID="ambient"
STYLE="filter: light()">
<IMG SRC="flower.jpg" WIDTH="243" HEIGHT="262" ID="cone"
STYLE="filter: light()">
</BODY>
```

Совместное применение нескольких фильтров

Динамический HTML позволяет использовать для одного объекта одновременно несколько фильтров. В этом случае запись фильтров идет через пробел с указанием параметров для каждого из применяемых фильтров.

На рис. 10.5 приведены примеры совместного использования нескольких фильтров, слева направо:

- `wave(strength=10, add=0, phase=15, lightStrength=25) flipH()` — фильтр `wave` без включения исходного объекта, с интенсивностью наложения 10, фазой смещения 15, трехмерным освещением, равным 25, и фильтр `flipH`, переворачивающий объект в горизонтальной плоскости;
- `gray() blur(strength=85, add=1, direction=45)` — фильтр `gray`, убирающий цветовую гамму объекта, и фильтр `blur`, размывающий изображение, с интенсивностью наложения 85, добавлением исходного объекта и с направлением размытия в 45 градусов;
- `xray() alpha(opacity=255, style=2, finishOpacity=0)` — фильтр `xray`, создающий эффект рентгеновского снимка, и фильтр `alpha` со степенью прозрачности объекта 255 и свойством стиля 2 (от центра к воображаемой окружности).

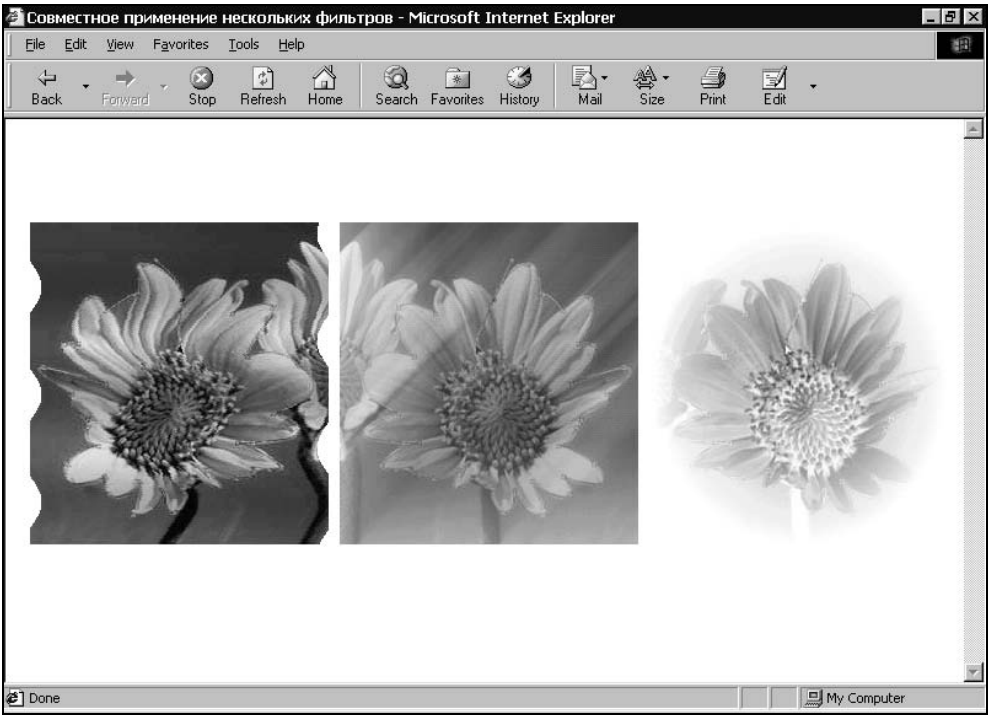


Рис. 10.5. Совместное применение нескольких фильтров

ГЛАВА 11

Концепция Web-сайта

В *части I* книги мы рассмотрели наиболее важные и применяемые элементы языка гипертекстовой разметки HTML. Однако для того, чтобы разработать собственный Web-сайт, недостаточно знать и уметь применять на практике HTML. Нужна некая визуальная основа, от которой следует отталкиваться в процессе создания Web-сайтов с помощью HTML.

В Интернете присутствует огромное количество сайтов разной тематической направленности, с разным оформлением, разной аудиторией и т. п. Как понять, какой именно сайт вам нужен и почему, — об этом пойдет речь в данной главе.

Классификация сайтов

К сожалению, на момент написания книги не существовало ни одной общепринятой стандартизированной классификации Web-сайтов. Поэтому будет приведено условное разделение сайтов, не претендующее на статус аксиомы, однако, отражающее практическую сторону развития интернет-проектов на начало 2004 года.

Корпоративные Web-сайты

Под корпоративным Web-сайтом понимается официальное представительство определенной компании, предприятия, организации в Интернете. Первостепенная задача любой компании сводится к продвижению своих товаров или услуг, а также созданию или стимулированию положительного образа (имиджа) компании.

Аудитория корпоративных Web-сайтов

Аудиторию любого корпоративного Web-сайта можно разделить на 3 категории:

1. Постоянные посетители — они уже знают компанию: что она производит или продает, в каком городе она находится и какие у нее номера телефонов. Это, как правило, клиенты или партнеры компании, которые могут заходить на сайт только ради чтения новостной ленты (быть в курсе событий компании), участия в опросах или маркетингово-рекламных акциях, обсуждения актуальных вопросов на форуме и т. д.
2. Новые посетители — они попадают на корпоративный Web-сайт впервые: по ссылке через поисковую систему, по адресу, указанному в визитке или рекламном объявлении в газете и пр. Они еще не знакомы с профилем фирмы, видом ее деятельности, поэтому для них будут актуальны разделы информационного характера (информация о фирме, история ее развития, успехи и достижения).
3. Вынужденные посетители — этот тип посетителей формируется по конкретной необходимости. К примеру, вам в срочном порядке понадобился контактный телефон директора 3-го мясоперерабатывающего завода г. Малое Дербенево. Вы воспользуетесь поисковыми системами, каталогами интернет-ресурсов и прочими электронными средствами и найдете ссылку на корпоративный Web-сайт упомянутого завода, на котором отыщете искомые контактные данные.

   Другой пример. Вам поручили составить перечень наиболее успешных модельных агентств. Вы пройдете по сайтам этих агентств, соберете нужную информацию, занесете себе в компьютер и, скорее всего, никогда больше не зайдете ни на один из увиденных корпоративных Web-сайтов модельных агентств. Вы — вынужденный посетитель.

Цели и задачи корпоративных Web-сайтов

Можно выделить следующие основные цели и задачи корпоративного Web-сайта:

- предоставление информации — своевременное и регулярное предоставление актуальной и достоверной информации — одна из важнейших задач любого корпоративного Web-сайта. Это повышает авторитет в глазах посетителя, дает понять, что сайт живет и развивается, наконец, подводит человека к мысли о том, что компания успешно работает и преуспевает.

  Другая сторона вопроса — точные статистические данные. Сравните два примера и задайте себе вопрос: "Какая информация вызовет у посетителя большее доверие?"

Пример 1: "Наша фирма — самая крупная в городе, у нас самый большой выбор товаров, самые низкие цены, приходите к нам — не пожалеете!"

Пример 2: "Наша фирма — одна из крупнейших в городе, работает на рынке в течение 5 лет. Мы предлагаем вам широчайший выбор продукции: свыше 5000 наименований товаров на любой вкус! Наши цены вас приятно удивят, а гибкая система скидок убедит вас в том, что вы сделали правильный выбор, став нашим покупателем!";

- создание и стимулирование имиджа — создание положительного благоприятного имиджа компании — другая, не менее важная, задача корпоративного Web-сайта. Это, прежде всего, отсутствие визуальных раздражителей, достоверность информации, внимание к посетителям и пр. Вторым фактором, влияющим на поддержание образа компании, является корпоративный стиль *(см. разд. "Корпоративный стиль" данной главы)*;
- поддержка клиентов — своевременная и квалифицированная поддержка клиентов на корпоративном Web-сайте — залог успеха и формирования положительного имиджа компании. Помощь в получении необходимой информации, ответы на наиболее частые вопросы, консультации и обсуждения в режиме онлайн, простейшая форма обратной связи с посетителем — все это укрепляет позиции компании и создает позитивное отношение к корпоративному Web-сайту.

Корпоративный стиль. Учитывая изложенное выше, можно предположить, что если будет создан отдельный сайт для ЧП Иванов И. И., во владении которого единственный скромный лоток по продаже книжной продукции, то этот сайт можно назвать корпоративным. Предположение, по сути, верное, но по требованиям к корпоративному стилю становится ложным. Другими словами, не каждый сайт не каждой организации можно назвать корпоративным. Чтобы получить этот статус, необходимо удовлетворять следующим требованиям к структуре корпоративного стиля (см. рис. 11.3):

1. Имя компании (как вариант: полное имя плюс аббревиатура).
2. Логотип.
3. Фирменные цвета (фирменная цветовая гамма).
4. Фирменные модули (бланки, документация, визитки, листовки, брошюры, плакаты, постеры, рекламные проспекты и пр.).
5. Фирменный шрифт.
6. Фирменный слоган (девиз).

Примеры корпоративных Web-сайтов

Для более ясного представления внешнего вида и структуры корпоративных Web-сайтов ниже приведено несколько примеров этого типа интернет-проектов.

Address

Поиск:

Вы здесь: М-Стайл / Вакансии / Текущие вакансии /

контакты гостевая книга

КОМПАНИЯ М-СТАЙЛ

Новый день. Новая информация.

ВАКАНСИИ

компания услуги продукты

партнеры вакансии новости

Вакансии

Текущие вакансии

ВАКАНСИИ

Текущие вакансии

Анкета соискателя

Наша компания работает на рынке информационных технологий с 1994 года. Мы являемся региональным информационным центром по распространению и сопровождению справочно-правовой программы КонсультантПлюс. Штат компании - более 200 сотрудников.

Продажи осуществляют **Менеджеры по продажам** СПС КонсультантПлюс. Пополнение и сопровождение программы осуществляют **Специалисты по информационному обслуживанию** (сервисные инженеры). Клиентскую базу обрабатывает информационный отдел, в котором работают **Сотрудники по ведению телефонных переговоров**.

МЕНЕДЖЕР ПО ПРОДАЖЕ СПРАВОЧНЫХ ПРАВОВЫХ СИСТЕМ "КОНСУЛЬТАНТ ПЛЮС"	
Требования к кандидатам	**Обязанности**
■ возраст: 22-37 лет ■ пол: не имеет значения ■ образование: высшее	Выезд к клиенту по предварительной договоренности, демонстрация программы и доведение сделки до продажи.

ГОРЯЧАЯ ЛИНИЯ

На Горячей линии пользователю ответят на все вопросы, связанные с системами КонсультантПлюс, помогут найти документ в установленных у него системах или примут индивидуальный заказ на поиск необходимого документа.

БЕСПЛАТНОЕ ОБУЧЕНИЕ

Целью Программы является обучение специалистов наиболее эффективным методам и навыкам работы с Системами. Программа рассчитана как на начинающих пользователей, так и на тех, кто давно работает с КонсультантПлюс

Программная технология КонсультантПлюс непрерывно

Рис. 11.1. Корпоративный Web-сайт компании "М-Стайл"

Address

DATAARTSTUDIO

НОВОСТИ | РАБОТЫ | КЛИЕНТЫ | ЦЕНЫ | О СТУДИИ | МЫ | FREE

285 392

СТУДИЯ ДИЗАЙНА

НОВОСТИ

DATAART HOLDS QUARTERLY INTER-COMPANY CONFERENCE // 5.2.2004

DataArt Strategic Advisory Board representatives Stuart Robbins and Loretta W. Prencipe took part in the conference. **подробнее**

DATAART AND CIO COLLECTIVE HOLD TECHNOLOGY LEADERSHIP FORUM // 4.2.2004

The Technology Leadership Forum was recently held in the Grand Hotel Europe. **подробнее**

DATAART EXPERT TEAM // 28.1.2004

DataArt is forming an internal group of experts dealing with the latest developments in technology, frameworks, tools, and software

LIVE

// Прямое включение

20 мая 2003 // "Золотой Сайт 2002"

Состоялась церемония награждения победителей Четвертого Открытого Всероссийского Интернет-конкурса

По результатам конкурса мы получили максимальное количество наград:

Гран-При конкурса "Золотой Сайт":

УСЛУГИ

// Студия "DataArt Дизайн" предлагает

Дизайн:

- Разработка фирменного стиля.
- Web-дизайн.
- Flash-анимация.
- Создание баннеров.

Продвижение:

- Рекламные кампании на крупнейших площадках Рунета.

Разработка ПО:

- Разработка, создание и установка стратегических легко настраиваемых программных решений, с использованием новейших технологий (подробнее).

Мы создаем яркие, запоминающиеся web-сайты, которые работают НА ваш бизнес. Мы уверены, вы оцените наши работы по

Рис. 11.2. Корпоративный Web-сайт студии дизайна "DataArt Studio"

Рис. 11.3. Корпоративный Web-сайт компании "Магнитогорский металлургический комбинат"

Промо/презентационные Web-сайты

Для рекламы определенных акций, мероприятий, событий, продвижения конкретных технологий, сервисов, товаров или услуг используются презентационные Web-сайты (промо-сайты).

Промо-сайты, как правило, создаются на базе корпоративных Web-сайтов, но имеют статус самостоятельного интернет-проекта: со своей системой навигации, обратной связью, пользовательской поддержкой и информационными разделами.

Аудитория презентационных Web-сайтов

Аудитория презентационного Web-сайта может включать следующие категории посетителей:

- корпоративные — постоянные посетители корпоративного Web-сайта могут узнать об открытии промо-сайта из новостной ленты, почтовой рассылки, официального пресс-релиза компании и т. д.;
- целевая аудитория, сфокусированная по потребительскому признаку — эти посетители могут не знать о компании, которая открыла промо-сайт, и

ни разу не посещать ее корпоративный Web-сайт. На презентационный Web-сайт такие посетители могут попасть:

- через поисковую систему;
- с новостных лент других сайтов и сайтов электронных средств массовой информации;
- посредством оффлайн-рекламы (телевидение, радио, пресса и пр.).

В любом случае посетитель промо-сайта будет заинтересован в его содержании, т. к. выход на презентационный сайт осуществлялся путем поиска определенной информации в соответствии с потребительскими предпочтениями. К примеру, пользователь Интернета, увлекающийся теннисом, ищет магазин, в котором можно приобрести профессиональные ракетки фирмы Head. Поисковая система выдает ссылку на промо-сайт компании-дистрибьютора, представляющей новую линейку продукции компании Head. В этом случае вероятность перехода по ссылке на промо-сайт Head будет максимальной.

Цели и задачи презентационных Web-сайтов

Можно выделить следующие основные цели и задачи презентационного Web-сайта:

- представление товара — промо-сайт должен выгодно презентовать товар или услугу своему потенциальному покупателю или пользователю. Как правило, используются интерактивные и анимационные ролики, слайд-шоу, демонстрационные показы и пр.;
- убеждение посетителя в преимуществах — промо-сайт должен убедить посетителя в преимуществах презентуемого товара или услуги, предоставить веские аргументы в пользу объекта рекламы. Обычно реализуется посредством психологических рычагов воздействия на потенциального потребителя, сравнительных характеристик товара и пр.;
- возможность попробовать — мало убедить в преимуществах и представить свой товар или услугу во всей красе — надо дать возможность познакомиться с ним на практике. К примеру, для промо-сайта юридических консультаций подойдет бесплатная возможность задать свой вопрос профессиональному юристу в режиме онлайн.

Примеры презентационных Web-сайтов

Для наглядного представления внешнего вида и структуры презентационных Web-сайтов на рис. 11.4—11.6 приведено несколько типичных примеров этого вида интернет-проектов.

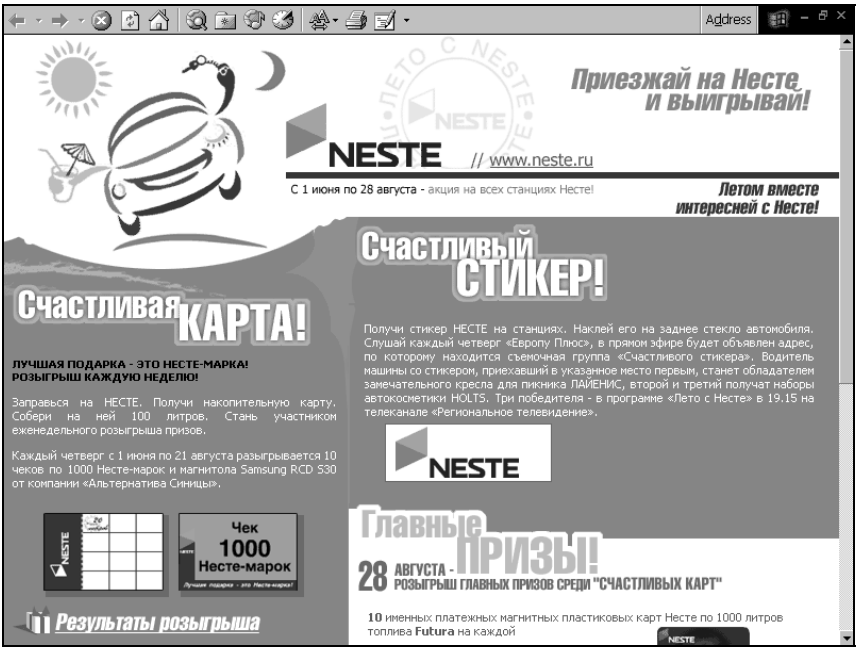


Рис. 11.4. Презентационный Web-сайт рекламной акции "Лето с Neste" от компании "Neste"

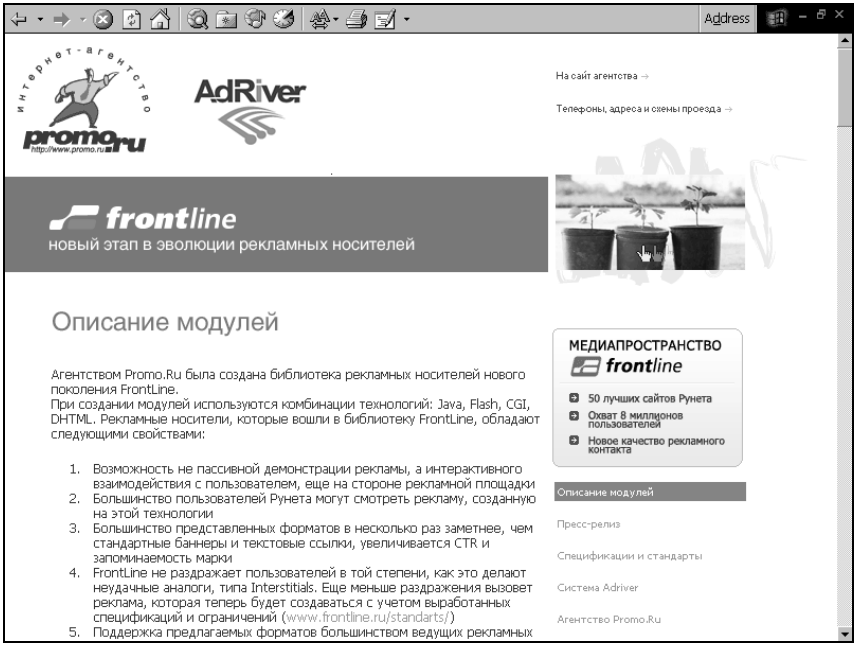


Рис. 11.5. Презентационный Web-сайт рекламной технологии "FrontLine" от интернет-агентства Promo.Ru

Рис. 11.6. Презентационный Web-сайт акции "Романтическая Мистерия" от клуба Fabrique и Mail.Ru

Электронные магазины

Web-сайты электронных магазинов ориентированы на получение прибыли от продажи тех или иных товаров или услуг. На них обычно присутствует новостная лента, общая информация, система голосования и форум, однако, основной акцент поставлен на представлении товара, преподнесении его покупателю в выгодном свете.

Особое внимание уделяется системе обработки заказов, безопасности платежей, конфиденциальности покупательской информации и оперативной квалифицированной помощи в выборе товара или услуги.

В 2000—2002 годах, когда отрасль электронных продаж только начинала развиваться в России, ассортимент товаров, продаваемых в виртуальных магазинах, был достаточно узок: книги, видеокассеты, компакт-диски, компьютеры и комплектующие. Сейчас диапазон продукции существенно расширился и, помимо уже перечисленных товаров, в Интернете можно купить бытовую и оргтехнику, игрушки и посуду, драгоценности и одежду и т. д.

Все электронные магазины можно разделить на 4 категории:

1. Витрина — по сути, она не является полноценным магазином, т. к. в большинстве случаев реализует только рекламно-демонстрационные функции: на каком-либо интернет-ресурсе открывается раздел, полностью посвященный демонстрации товаров или услуг, потенциальный покупатель может ознакомиться с представленным ассортиментом, но чтобы сделать заказ и получить выбранный товар, ему все равно придется прибегнуть к таким традиционным действиям, как звонок по контактному телефону или приезд в офис продавца.

   Это занимает порой достаточно большое время, и в конечном результате может привести к тому, что покупатель передумает осуществлять покупку вообще или решит воспользоваться услугами другой фирмы. Данный вид реализации товаров через Интернет не требует никакого дополнительного программного и аппаратного обеспечения.

2. Торговая площадка — реализована практически так же, как и витрина, но ей присущи некоторые функции полновесного электронного магазина. В этом случае посетитель может ознакомиться с существующим ассортиментом, получить описание потребительских свойств интересующего его товара и, сделав заказ, получить товар — правда, только спустя время, необходимое для ручной обработки заказа.

   Этот вид онлайновых продаж, чаще всего, тоже не требует дополнительного или специализированного программного обеспечения: интерактивная регистрация посетителей, выбор нужного товара, составление и отправка бланка заказа могут быть реализованы посредством таких технологий, как PHP или CGI. Аппаратный комплекс также не ограничивается какими-то особенными критериями.

3. Торговый ряд — сегодня в российской части Интернета существует большое количество служб, представляющих собой так называемые виртуальные торговые ряды, суть которых заключается в предоставлении для любых фирм и организаций бесплатной или платной возможности размещения ассортимента своих товаров или услуг, а также сопутствующей информации: прайс-листов, технического описания, демонстрационного или иллюстративного материала и т. д.

   Такая возможность является арендой виртуального магазина, который, в данном случае, будет считаться посредником между производителем и покупателем. Естественно, степень автоматизации деятельности по сбыту товаров и функциональность такого рода услуг зависит от конкретного торгового ряда, его технической оснащенности и еще целого ряда факторов. При работе с таким торговым рядом могут быть использованы самые разнообразные схемы: начиная с простой регистрации в качестве нового

арендатора виртуального торгового места и загрузки данных о товарах, что не требует никакого специального аппаратного и программного обеспечения, и заканчивая открытием собственного "филиала" на базе головной службы.

Примером последней схемы может послужить система электронных магазинов "Корзина.Ру", которая предлагает своим партнерам установить специальный рабочий сервер с соответствующим ПО, интегрированный в систему реализации товаров. В этом случае, арендатор берет деньги за аппаратное и программное обеспечение, но все операции по работе с клиентами (демонстрация, получение и обработка заказа, определение формы оплаты, доставка и пр.) он берет на себя. Со стороны партнера выполняется всего лишь одна функция: оперативное информирование администраторов арендатора о наличии товара на складе.

4. Автоматизированный магазин — это полновесная виртуальная торговая служба, осуществляющая операции по работе с заказчиком посредством автоматического программного управления. В число выполняемых функций входят такие, как демонстрация товара, предоставление исчерпывающей информации, техническая поддержка клиентов, составление и обработка заказа, выписка счетов, перевод денежных платежей, подбор оптимального способа доставки и др. В данном случае присутствие человека тоже обязательно, однако здесь он осуществляет только общий контроль за работой всей системы.

Аудитория электронных магазинов

Аудитория электронного магазина может включать следующие категории посетителей:

- прямые покупатели — люди, целенаправленно посещающие электронные магазины с целью приобрести определенный товар, при этом получая квалифицированный сервис, техническую поддержку, а также такие немаловажные бонусы, как бесплатная доставка и скидки при заказе через Интернет;
- косвенные покупатели — люди, прибегающие к помощи электронных магазинов при нерегулярной необходимости: поиск подарков к празднику, указание руководства подыскать недорогой товар в Интернете и т. п.;
- корпоративные покупатели — представители компаний, делающие ставку на оперативность обслуживания и минимум собственных трудозатрат. В особенности актуальны для электронных магазинов, продающих дорогие лицензионные товары: программное обеспечение, мультимедиа и пр.

Цели и задачи электронных магазинов

Можно обозначить следующие основные цели и задачи электронного магазина:

- представление товара — полная и достоверная характеристика продаваемого товара, указание технических параметров, особенностей и прочих потребительских свойств;
- квалифицированный сервис — покупатель должен получать качественное обслуживание при осуществлении покупки в электронных магазинах. Консультанты по продажам должны владеть товарной спецификой данного магазина и разбираться в тонкостях разных наименований товаров, курьеры должны вовремя доставлять заказанный товар и быть учтивыми, счета должны выписываться грамотно и т. д.;
- пополнение ассортимента — стимулирует спрос все на новые товары и услуги, дает понять покупателю, что электронный магазин не стоит на месте, а идет в ногу с последними изменениями на товарном рынке;
- стимулирование покупателя — повышает авторитет электронного магазина, его известность и формирует о нем положительное мнение. Обычно реализуется за счет всевозможных подарков, скидок, бонусов, рекламных акций, лотерей, конкурсов и пр.

Примеры электронных магазинов

Для наглядного представления внешнего вида и структуры электронных магазинов на рис. 11.7—11.9 приведено несколько типичных примеров этого вида интернет-проектов.

Онлайн-сервисы

В Интернете существует огромное количество Web-сайтов, которые предлагают посетителям зарегистрироваться, чтобы получить бесплатный (а в ряде случаев и платный) доступ к определенному набору услуг. Как правило, эти услуги направлены на удовлетворение повседневных и бытовых нужд обычного пользователя сети Интернет: поиск информации, работа с текстом и электронной почтой, повышение квалификации и многое другое. Все эти сайты можно отнести к категории онлайн-сервисов.

Аудитория онлайн-сервисов

Аудитория онлайн-сервиса может включать следующие категории посетителей:

- корпоративные пользователи — представители компаний, нуждающихся в получении доступа к услуге в профессиональных целях;

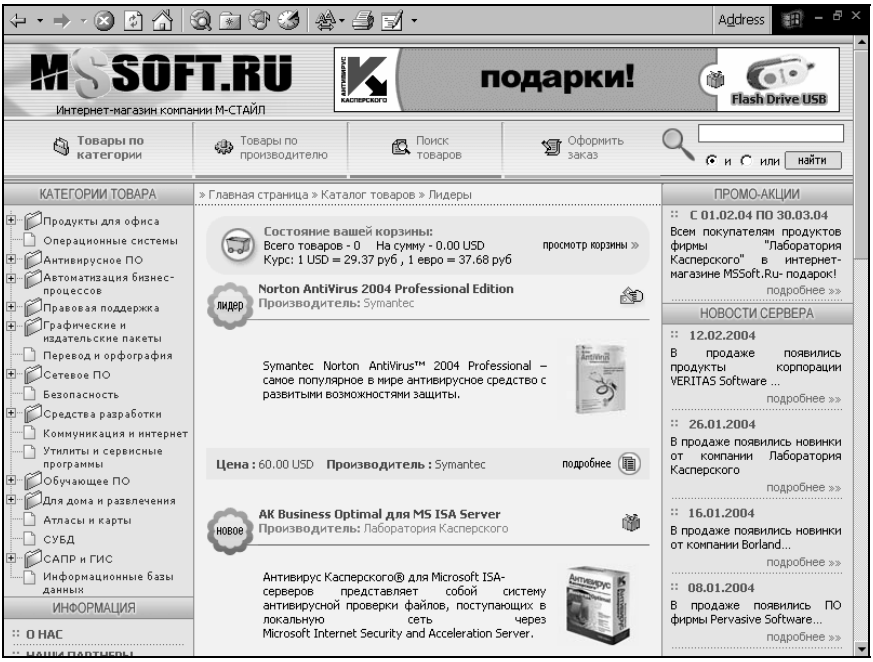


Рис. 11.7. Электронный магазин лицензионного программного обеспечения MSSoft.Ru

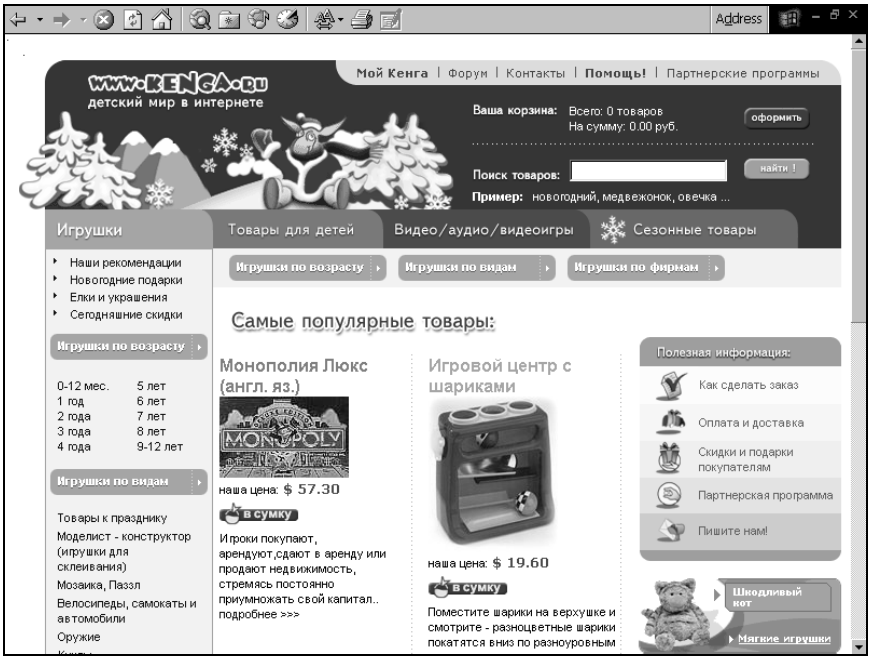


Рис. 11.8. Электронный магазин детских игрушек Kenga.Ru

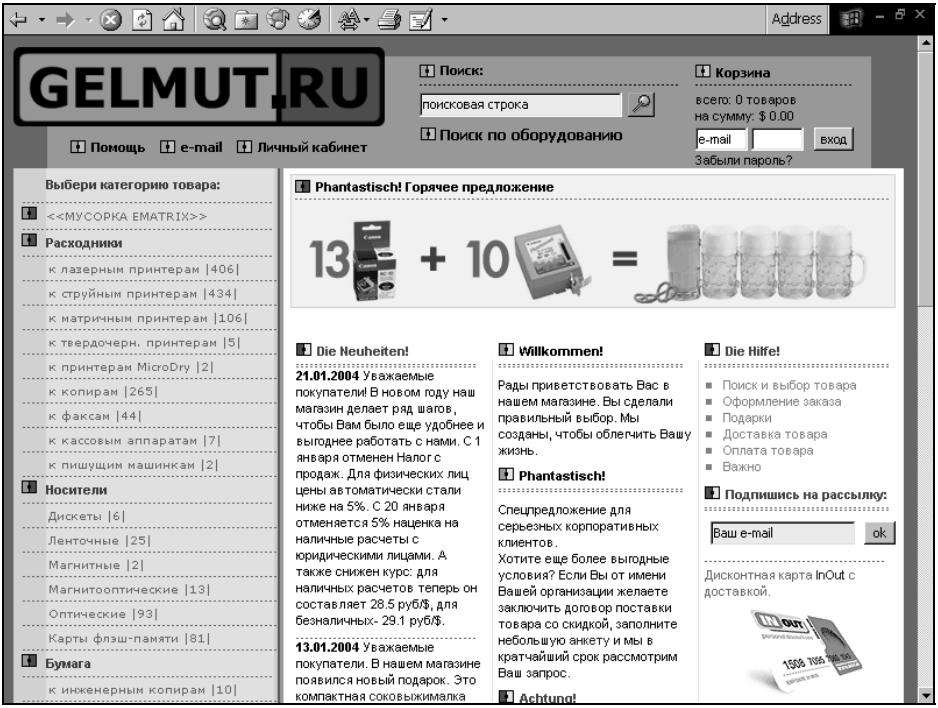


Рис. 11.9. Электронный магазин расходных материалов Gelmut.Ru

- частные пользователи — лица, пользующиеся конкретными услугами в режиме онлайн в личных целях.

Цели и задачи онлайн-сервисов

Можно выделить следующие цели и задачи онлайн-сервиса:

- качество — поддержание высокого качества сервиса в режиме онлайн — задача непростая. Пользователь весьма капризен в своем выборе, и если почувствует, что качество предоставляемых ему услуг начинает портиться, безвозвратно уйдет к другому поставщику аналогичных онлайн-услуг.

  Следует заметить, что в большинстве случаев поддержание высокого качества услуг при отсутствии материальной отдачи со стороны пользователей и их постоянно растущего числа часто приводит к тому, что часть услуг остается бесплатной, а часть (содержащая расширенные возможности пользования) предоставляется исключительно за плату. В качестве примера может послужить система интернет-статистики HotLog, которая на протяжении нескольких лет всем желающим предоставляла подробную бесплатную статистику, а в конце 2003 года ввела в эксплуатацию специаль-

ный тарифный пакет Gold с установленной абонентской платой 999 рублей в месяц;

- совершенствование — необходимо регулярно улучшать работу сервиса, совершенствовать функциональность и увеличивать объем предоставляемых услуг. Для сервиса онлайн-перевода — это пополнение новыми языковыми словарями, для почтовой службы — возможность блокировать спам, для поисковой системы — расширенные возможности для уточнения запросов и т. д.

Примеры онлайн-сервисов

Для наглядного представления внешнего вида и структуры онлайн-сервисов на рис. 11.10—11.12 приведено несколько типичных примеров этого вида интернет-проектов.

Рис. 11.10. Онлайн-переводчик Translate.Ru

Контент-проекты

Под *контент-проектами* принято понимать Web-сайты, специализирующиеся на каком-то определенном информационном содержании (от англ. *content* —

Рис. 11.11. Почтовая система Rambler-почта

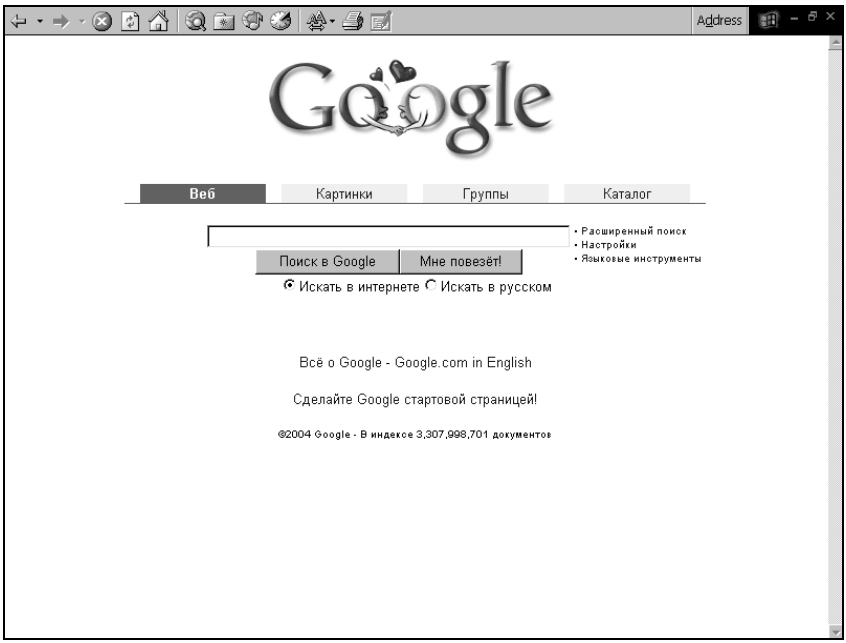


Рис. 11.12. Поисковая система Google

содержание). Это компьютерные библиотеки и серверы документации (тексты, статьи, аналитика, обзоры), каталоги ресурсов и электронные справочники (ссылки, описания, координаты предприятий и компаний), телефонные базы данных, каталоги программ, новостные серверы и электронные средства массовой информации.

Сегодня большинство контент-проектов переживает переходное состояние с точки зрения соответствия своей первоначальной функциональной направленности — несения информации в "чистом" виде. Акцент постепенно сдвигается к развитию различного пользовательского инструментария: системам голосования, опросам, почтовым серверам, услугам хостинга и пр. А последний фактор — признак иного типа Web-сайтов, который называется порталом (о нем речь пойдет ниже).

Аудитория контент-проектов

Аудитория контент-проекта может включать в себя самые разнообразные категории пользователей, объединенных едиными интересами, увлечениями или профессиональными целями.

Посетитель Web-сайта электронной газеты — это школьник и руководитель компании, студент и секретарь-референт и т. д. Пользователем музыкального архива также может быть любой человек, т. к., в данном случае, аудиторию объединяет общее увлечение музыкой.

Цели и задачи контент-проектов

Можно выделить следующие цели и задачи контент-проекта:

- актуальность и достоверность информации — если в каталоге программного обеспечения последние новинки будут датированы прошлым месяцем, актуального содержания на таком Web-сайте посетитель не найдет. Непроверенная, недостоверная или заведомо ложная информация также не добавит авторитета контент-проекту.

  Отсюда вывод: контент-проект должен содержать только актуальную, востребованную своей аудиторией, проверенную и достоверную информацию, будь то новости спорта или новинки музыкального олимпа;

- расширение содержательной части — нельзя останавливаться на какой-то одной теме, нужно постоянно расширять тематический диапазон предоставляемой посетителю информации. К примеру, изначально Web-сайт освещал российский кинематограф, чуть позже стал затрагивать события американской киноиндустрии, впоследствии стал информировать посетителя о мировых новинках, событиях и мероприятиях в мире кино.

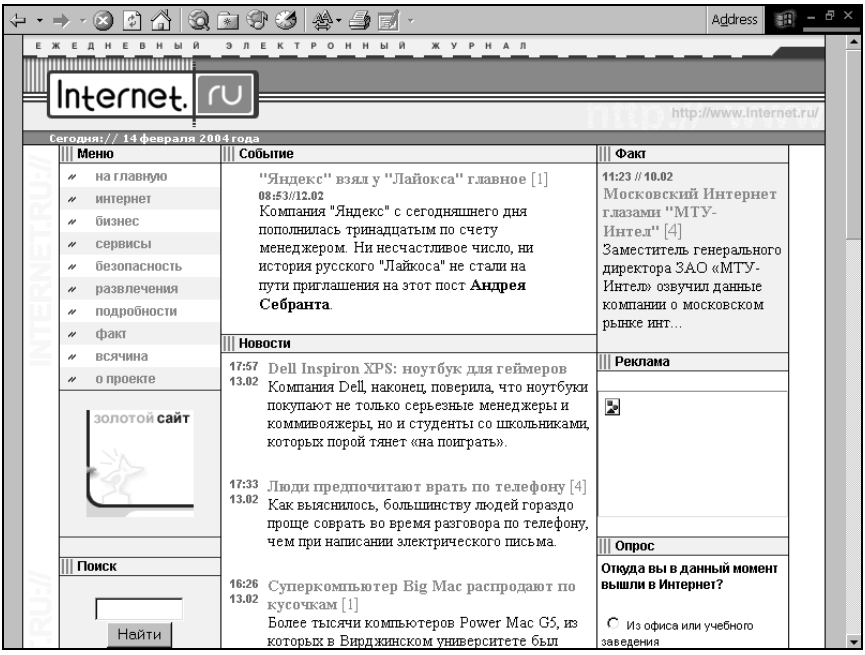


Рис. 11.13. Ежедневный электронный журнал Internet.Ru

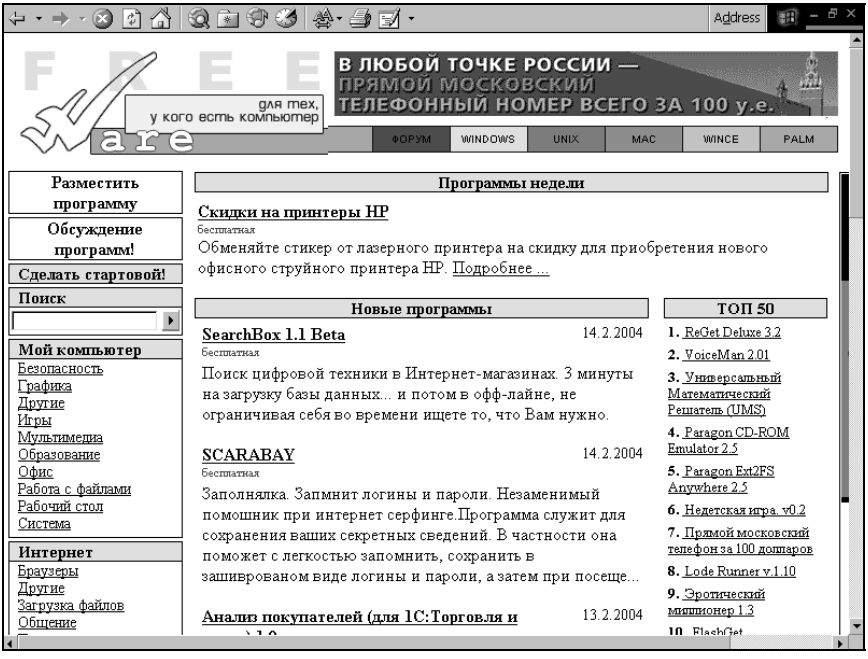


Рис. 11.14. Каталог бесплатных и условно-бесплатных программ FreeWare.Ru

Рис. 11.15. Банк вакансий и резюме Работа.Ru

Примеры контент-проектов

Для наглядного представления внешнего вида и структуры контент-проектов на рис. 11.13—11.15 приведено несколько типичных примеров.

Порталы

В *разд. "Контент-проекты"* было сказано про тенденцию, ведущую к постепенному превращению Web-сайтов, ориентированных на предоставление информации по выбранной тематике, в крупные порталы. Итак, что же такое портал?

Термин "портал" изначально является архитектурным и означает "парадный фасад здания". Также это слово очень любимо писателями-фантастами, разработчиками компьютерных игр и режиссерами фильмов в жанре фэнтази: в созданных ими мирах портал — это средство мгновенного перемещения между мирами.

С точки зрения интернет-тематики, термин "портал" означает некий Web-сайт, с которого большое количество пользователей начинает свое путешествие по глобальной сети.

Первыми сайтами, получившими такой статус, были сайты крупных интернет-провайдеров (AmericaOnLine), каталоги ресурсов (Yahoo!) или поисковые системы (AltaVista). По задумке, зайдя на такой "портал", пользователь получает доступ ко всей необходимой ему в Интернете информации: новости, развлечения, книги, прогноз погоды, программа телепередач, гороскопы и многое другое.

Виды порталов

Интернет-порталы можно разделить на две группы:

- "вертикальные" порталы — ориентированные, прежде всего, на конкретную целевую аудиторию, т. к. посвящены одной обширной теме, которая раскрывается, что называется, "вглубь" (отсюда определение "вертикальный").

  Пример "вертикального" портала: сайт, посвященный мобильной связи, на котором есть каталог новых моделей сотовых телефонов и хит-парад модных мелодий, библиотека логотипов и картинок для телефонов, инструкции и советы по эксплуатации, специализированное программное обеспечение, координаты крупнейших магазинов и операторов сотовой связи, описание стандартов и пр. Тематика одна и та же, а ресурсов — на любой вкус;

- "горизонтальные" порталы — охватывают сразу несколько разноплановых тематических направлений, зачастую совсем не смежных (т. е. развитие идет "вширь", отсюда определение "горизонтальный").

  Пример "горизонтального" портала: каталог интернет-ресурсов + поисковая система + персональная почта + служба знакомств + игровые обзоры + электронный магазин. Добавьте сюда еще прогноз погоды, виртуальные открытки, программу ТВ, заказ "горящих" путевок и получится самый натуральный "горизонтальный" портал.

Аудитория порталов

Логично предположить, что аудитория "вертикальных" порталов формируется по тематическому признаку: автомобили (купля/продажа, ремонт и тюнинг, выставки и автосалоны, сигнализация и тесты, периодика и подписка на рассылку новостей по автомобильной тематике); спорт (расписание турниров, новости, спортивные легенды, репортажи и интервью) и т. д. и т. п.

Аудитория "горизонтальных" порталов строится хаотично, без влияния каких-либо критериев по возрасту, увлечениям или социальному статусу. И это, в принципе, неважно, если нужно просто просмотреть почту, узнать, что сегодня вечером по телевизору, или выяснить, какую погоду на завтра обещают синоптики.

Цели и задачи порталов

Для "вертикальных" порталов основной задачей было и остается более глубокое и детальное раскрытие и освещение выбранной тематики.

Для "горизонтальных" порталов главная задача — не переусердствовать с добавлением все новых сервисов, возможностей и услуг. Ведь пользователь может просто устать от такого разнообразия и уйти на другой — не столь насыщенный сервисами — портал.

Примеры порталов

Для наглядного представления внешнего вида и структуры порталов на рис. 11.16—11.18 приведено несколько типичных примеров.

Рис. 11.16. "Горизонтальный" портал MyWeb.Ru

Коммьюнити

Сайты-коммьюнити появились сравнительно недавно, однако уверенно и высокими темпами набирают популярность и находят новых поклонников и почитателей. Слово "коммьюнити" означает в переводе с английского "сообщество" (community) и подразумевает некий виртуальный центр сосредоточения людей, связанных определенным общим признаком.

Рис. 11.17. "Вертикальный" портал MP3.Ru

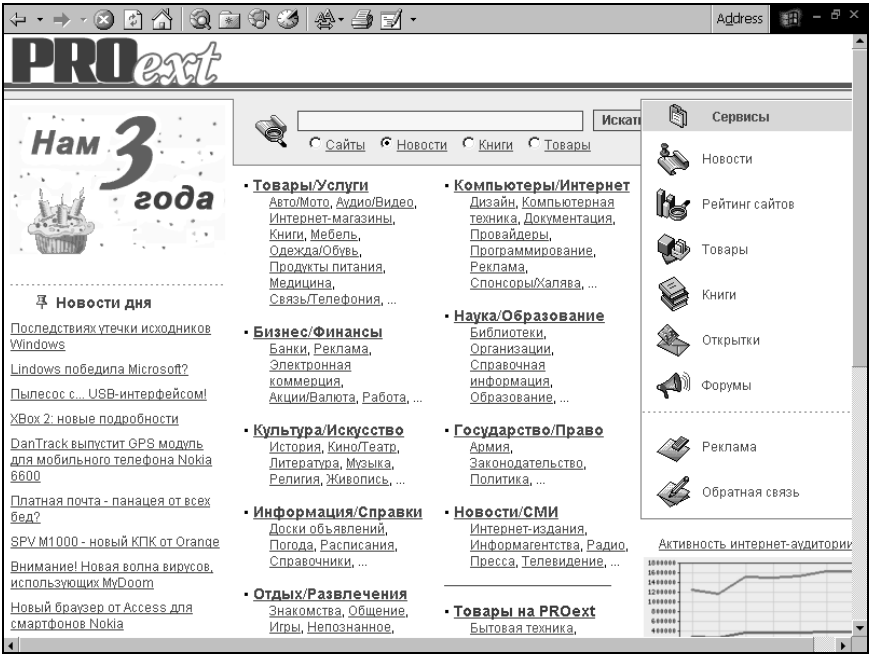


Рис. 11.18. "Горизонтальный" портал PROext.Ru

Средство общения

Если вас спросят, какую информацию можно предоставить в качестве координат для обратной связи или общения с другими людьми, то наверняка вы назовете электронную почту, интернет-пейджер, гостевую книгу, форум и... все! Казалось бы, чего еще надо современному пользователю для полноценного общения в Интернете? Как известно, общение между людьми происходит по нескольким причинам: либо это вынужденная процедура (переписка между рекламным агентом и коммерческой фирмой), либо собеседников связывает нечто общее — интересы и увлечения, работа и дом, общие знакомые и родственники и пр.

Однако и интернет-пейджеры, и форумы, и электронная почта подразумевают определенные рамки, за пределы которых пользователь не может выходить: общаясь по ICQ или AOL, вам придется заинтересовывать своего собеседника или вести себя, как минимум, пристойно, чтобы тот не удалил вас из своего контактного списка; на форумах присутствует жесткий регламент, продиктованный правилами и тематической направленностью форума; электронная почта тоже не может предоставить человеку уверенности в том, что написанное сообщение а) будет интересно адресату; б) дойдет до адресата; в) вызовет желание адресата ответить.

А что делать, если пользователь — человек общительный, впечатлительный, наблюдательный и эмоциональный, готовый прореагировать на любое событие в своей жизни, в жизни своих друзей, знакомых и даже чужих людей, просто ему симпатичных? Вас обрызгал грязью грузовик или вы прокололи колесо на автомобиле, отчего настроение стремительно падает? Вы получили повышение по службе или познакомились с милой девушкой, из-за чего хочется петь и лезть обниматься к прохожим? Ваша пятиюродная бабушка из Могилева наконец-то вышла замуж за Билла Гейтса или день свадьбы был переназначен на 1 апреля? Передавая подобную информацию по электронной почте, интернет-пейджеру или оставляя на форуме, будете ли вы уверены в том, что она интересна хоть кому-нибудь еще, помимо вас?

Здесь и приходит на помощь форма сетевого общения в виде коммьюнити — "Живой Журнал" или просто ЖЖ.

В общих чертах, ЖЖ — это онлайновый дневник, который ведется пользователем. Любой желающий может просматривать сообщения этого пользователя, оставлять свои замечания и пр. На первый взгляд напоминает форум, однако последний обычно объединяет людей по конкретному признаку, отчего круг пользователей заметно сужается (например, конференции по графике и дизайну, программному обеспечению и сетям). То есть, сообщать на форуме по вопросам администрирования локальных сетей под управлением ОС UNIX о том, что ваша кошка Нюра вчера разродилась котятами, будет неуместно, да и просто запрещено.

Прежде всего, ЖЖ хорош тем, что в нем вы можете писать что угодно и сколько угодно — никто не упрекнет вас в болтливости или "засорении просторов Интернета", провайдер не пришлет вам счет за резкое повышение трафика, а администратор сервиса не станет удалять вашу рекламу, фотогалерею в духе "ню" или сборник "1000 и 1 спосаб саздания вэб-странички". Ведение собственного "живого журнала" бесплатно, хотя существует небольшое количество платных *аккаунтов*, предоставляющих более широкий спектр возможностей. Однако и те, что присутствуют в бесплатной версии, могут удовлетворить самого привередливого пользователя.

Что можно делать в ЖЖ:

- добавлять новые и редактировать/удалять старые сообщения;
- настраивать дизайн ЖЖ (цветовая схема);
- определять тип настроения, которое соответствует сообщению;
- указывать ваши музыкальные вкусы на момент написания сообщения;
- вставлять смайлики и использовать HTML-теги (перечень разрешенных тегов прилагается в каждом ЖЖ);
- составлять группы друзей, последние сообщения которых отображаются в специальном разделе аккаунта ЖЖ (весьма удобная опция: не приходится каждый раз обходить ЖЖ всех своих знакомых, что существенно экономит время при наличии списка друзей из нескольких десятков, а то и сотен записей);
- управлять календарем событий;
- редактировать персональные данные (имя, адрес электронной почты, ICQ и пр.) плюс добавлять фотографии (или любые другие графические изображения, которые неизменно появляются рядом со всеми вашими сообщениями и комментариями в ЖЖ);
- назначать статус сообщений (публичный — читают все; приватный — только для личного прочтения; для друзей) и многое другое.

На сегодняшний день насчитывается свыше 500 тысяч различных ЖЖ, из них три тысячи приходится на российских пользователей. А между тем, началось все в марте 1999 года с персонального электронного журнала Брэда Фитцпатрика, студента, который специализировался на вычислительной технике. Методика, основанная на управлении посредством базы данных, которую он использовал для ведения своего личного дневника с 1998 года, со временем переросла в Web-приложение, доступное и его друзьям. Идея оказалась столь успешна, что вскоре ЖЖ стал крупным сервером, команда разработчиков которого все время пополняется для того, чтобы сделать общение пользователей всего мира еще более удобным и приятным.

На данный момент "Живой Журнал" — это программный проект с открытым исходным кодом, финансируемый исключительно самими пользователями и развивающийся при содействии многочисленных добровольцев и разработчиков.

Положительный пример распространения "Живого Журнала" повлек за собой волну клонов, а также успешных имитаций и неудачных подражаний. Однако это позволило впоследствии дать жизнь совершенно новому виду общения в Интернете, новому виду коммьюнити, название которому — "блог" (blog).

Интернет-форумы

Предположим, у вас есть Web-сайт. У сайта есть определенная аудитория посетителей, которые регулярно приходят по обозначенному адресу. Причем посетители приходят на сайт с определенной целью, которую условно можно сформулировать следующим образом: "поиск информации". Разумеется, под этим может подразумеваться все что угодно: начиная с поиска последних обновлений каталога программного обеспечения или свежих анекдотов и заканчивая просмотром наиболее часто задаваемых вопросов (FAQ) или обратной связью с авторами сайта.

Если тематика сайта отвечает пользовательскому спросу, постоянная аудитория объединяется по признаку встречных интересов: например, Вася ищет драйверы для нового модема, купленного на барахолке, а Петя работает на этой самой барахолке и знает, что найти драйверы под продаваемые им комплектующие практически невозможно. Но можно. Поэтому Вася заинтересован в Пете как источнике информации, необходимой для восстановления работоспособности модема и т. п.

Разумеется, два человека могут добавить друг друга в контакт-лист ICQ, поговорить по телефону или встретиться и обсудить проблему нахождения драйверов. А что, если количество подобных проявлений встречного интереса возрастает в десятки, сотни раз? Придется искать надежную телефонную линию на сотню каналов или собираться толпой у метро? А если половина заинтересованных людей проживает на другом конце страны? Именно в такой ситуации на помощь сайту приходит форум.

Форум — это место общения всех поклонников и посетителей определенного сайта. Там любой желающий может задать вопрос и получить ответ, поделиться опытом и спросить совета других участников, сообщить последние новости и просто пообщаться на отвлеченные темы. Разумеется, условия участия в форуме регламентируются правилами конкретного сайта, на котором этот форум установлен, но, в целом, характеристика практически любого интернет-форума выглядит так.

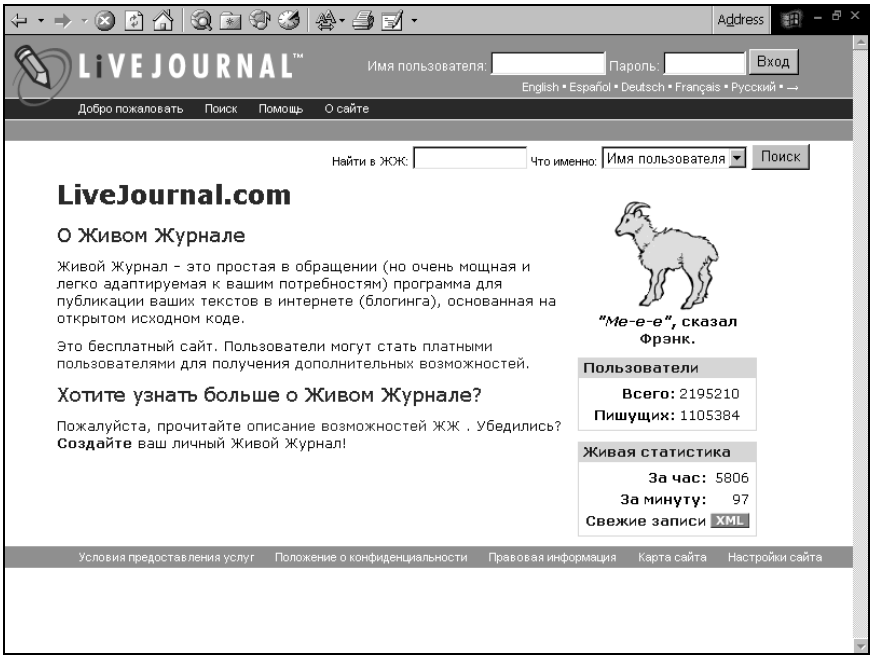


Рис. 11.19. "Живой Журнал" (LiveJournal.Com)

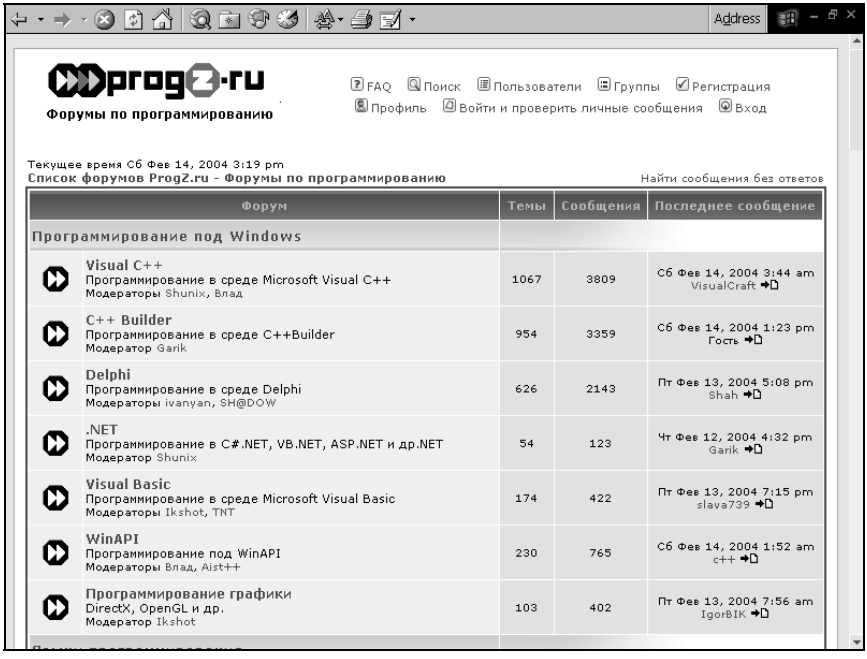


Рис. 11.20. Progz.Ru (весь сайт — форумы по вопросам программирования)

Именно фактор некоего сообщества, коллектива людей, объединенных общими идеями и увлечениями, позволяет поставить форум как коммьюнити в разряд отдельной категории Web-сайтов.

Примеры коммьюнити

Для наглядного представления структуры коммьюнити на рис. 11.19, 11.20 приведено несколько типичных примеров.

Позиционирование Web-сайта

Итак, вы определились с тематикой и категорией своего будущего Web-сайта, наметили цели и задачи присутствия своего ресурса во всемирной сети. Кажется, все уже готово: можно разрабатывать визуальный образ проекта и отбирать интересный контент. На самом деле, между стадией проектирования (разработка графического макета и HTML-верстка) и стадией продвижения вашего проекта существует еще один небольшой по времени, но чрезвычайно важный этап, неверное планирование которого может негативно сказаться впоследствии — в процессе рекламной кампании.

Речь идет о *предварительном позиционировании* пока еще не существующего Web-сайта. Именно предварительном, т. к. мероприятия, относящиеся к этому этапу, выполняются до реального появления проекта в Интернете (другими словами, до того момента, когда проект станет доступен многочисленной аудитории Сети).

Процедура предварительного позиционирования включает в себя следующие направления:

- выбор доменного имени для Web-сайта;
- распределение обязанностей по работе над Web-сайтом;
- работа по исследованию аудитории Web-сайта.

Рассмотрим каждое из них более подробно.

Выбор доменного имени для Web-сайта

Так уж сложилось в жизни, что, для того чтобы вас поняли другие люди, необходимо приводить к *смысловому соответствию* все, что вы делаете: довольно трудно понять, что вы хотите от окружающих, какие цели преследуете, какие задачи ставите перед собой для достижения запланированного и т. д. Если составляющие вашего жизненного (или профессионального) кредо будут находиться в противоречии друг с другом, то вы не только не получите желаемого результата, но и можете пострадать от такой стратегии. А что

касается других людей, то они будут находиться в недоумении и зря потратят свое время на контакт с вами.

Правило смыслового соответствия применимо и к сфере разработки Web-сайтов, в особенности — к вопросу выбора и регистрации доменного имени для вашего будущего проекта. Бытует мнение, что выбор названия для домена не играет существенной роли в аспекте формирования у посетителя имиджа проекта. Такая точка зрения ошибочна, а люди, придерживающиеся ее в процессе разработки интернет-проектов, допускают одну из наиболее серьезных ошибок с позиции психологического восприятия. При выборе доменного имени необходимо следовать следующим критериям: запоминаемость, понятность, однозначность, ассоциация, переадресация.

Запоминаемость

Название домена для вашего проекта должно быть звучным, легко произносимым и по возможности коротким. Невнятное наименование, использующее сложные комбинации букв длиной в половину адресной строки браузера, вызовет множество затруднений у вашего потенциального пользователя: осложнит восприятие на слух (адрес трудно разобрать в телефонном разговоре или устной беседе), потребует много времени, чтобы вписать такой адрес в браузере и др. В связи с этим, домен плохо запоминается, а пользователь не станет лишний раз искать ваш проект в Интернете или по телефонному справочнику в надежде обнаружить нужный адрес. Ему гораздо проще и быстрее обратиться по адресу, который сложно забыть.

Проблема краткости адреса для многих разработчиков стоит особенно остро, ибо далеко не каждый может себе позволить сразу оплатить регистрацию удачного домена, а потом к тому еще и оплачивать хостинг. По этой причине многие регистрируют адреса в бесплатных службах, которые предоставляют, как максимум, домен третьего уровня либо просто отдельную директорию у себя на сервере. Однако советуем на этом не экономить: платный хостинг имеет ряд неоспоримых преимуществ перед бесплатным — стабильность работы, поддержка серверных технологий, техническое сопровождение клиентов, подробная статистика, почтовые серверы, настройка интерфейсов и многое другое. Поэтому лучше пожертвовать частью денег сейчас и спокойно работать над развитием проекта, чем потом мучительно решать проблемы со стабильностью работы бесплатного хостинг-провайдера, поиском внезапно пропавшей корреспонденции с его сервера, нежеланием предоставлять техническую поддержку и пр. Позже вы сами придете к мысли о необходимости собственного домена, но время будет потрачено зря, что может нанести существенный вред вашему Web-сайту. Впрочем, использование услуги платного хостинга необязательно означает, что вы должны регистрироваться в зоне .RU или .COM (домены второго уровня). Иногда вполне достаточно по-

лучить адрес в одной из географических или второстепенных зон (типа .SPB.RU или .COM.RU) и гордиться запоминаемым именем домена.

Пример: сравните между собой два доменных наименования:

`project.ru`

и

`domain.ru/pub/main/users/project/index.html`

Какое из них более запоминаемо и проще для восприятия?

Понятность

Рекомендуется регистрировать доменное имя, аналогичное или весьма схожее по звучанию с названием вашей компании или проекта. Таким образом, ваш потенциальный посетитель, зная название вашей компании, может легко узнать адрес вашего сайта. Формула проста: "название компании/проекта + зона расположения домена". Так можно быстро узнать интернет-адрес официального сайта компании, производящей всемирно известные автомобили Porsche: пишем porsche, а далее после точки — либо com (коммерческий домен), либо de (домен Германии, где производятся эти автомобили). Все, что вам нужно, — это немного фантазии и основные навыки работы с Интернетом.

Многие компании как будто нарочно стремятся запутать пользователя, выбирая для своего доменного имени самые странные, непонятные, глупые, смешные и прочие понятия. За доменом `fool.com`, например, скрывается большой сервер, посвященный информационно-аналитическим материалам по финансам. По адресу `gif.ru` навряд ли можно было найти полную документацию по графическому стандарту GIF, разработанному в 1987 году компанией Compuserv, т. к. этот домен до недавнего времени был занят "Порталом геокультурной навигации" (на данный момент домен недоступен). Наконец, для интернет-портала домен `eprst.ru` мало подходит.

Однозначность

В Интернете существует большое количество проектов, в названии доменов которых обыгрываются различные варианты произношения в английском языке. Например, известная поисковая система и каталог ресурсов GoTo.com выбрала домен `go2.com`, не задумываясь о том, что определенное количество пользователей Интернета плохо знакомо с английским языком и далеко не сразу сможет понять дословный перевод такой комбинации слов (точнее, слова и цифры, где последняя при произношении совпадает со словом "to"). Аналогичный пример: портал WhatUSeek с идентичным доменным написанием. В русской части Интернета такие случаи тоже не редкость: `24x7.ru`

(с английского означает "24 часа в сутки, 7 дней в неделю") и др. С цифрами в названиях доменов вообще отдельная история: не зная, кому принадлежит тот или иной домен, попробуйте догадаться, что можно найти по таким адресам, как `32.ru`, `5000.ru`, `21.ru` и т. д.

Ассоциация

Многие считают ассоциативный ряд, возникающий у пользователей при знакомстве с названием домена, явлением, не способным повлиять на формирование имиджа проекта в целом. Совершенно напрасно, ибо отрицательные ассоциации от домена — это только первый камень в чужой огород. Хуже, когда они закрепляются, становятся устойчивыми в процессе пользования услугами, товарами или данными проекта: в результате, стабильный негативный образ такого интернет-ресурса надолго останется в сознании пользователя. А если пользователь попадется общительный — дурная слава расползется по всему Интернету.

Вот лишь некоторые примеры доменных имен, вызывающих неоднозначные ассоциации: `susanin.net` (поисковая система), `whitehouse.com` (эротика), `musor.net.ru` (измельчители пищевых отходов), `referatov.net` (коллекция рефератов) и пр.

Переадресация

Если вы собираетесь открывать в Интернете Web-сайт какой-нибудь известной компании или в перспективе видите свой проект на вершинах рейтингов, позаботьтесь о том, чтобы зарегистрировать несколько дополнительных доменных имен. Это желательно сделать, если название состоит из нескольких слов (например, Samsung Electronics), подразумевает несколько вариантов написания (например, General Motors Cars) или дано в виде аббревиатуры (например, телеканал РТР). Приблизительные варианты дополнений в этом случае будут выглядеть, соответственно, так:

```
samsung-electronics.com / se.com
general-motors.com / gmc.com
rus-tv.ru / rtr-tv.ru
```

Иначе вы можете столкнуться с результатами деятельности виртуальных хулиганов — киберсквоттеров, занимающихся поиском свободных доменных имен, по написанию схожих с названиями крупных корпораций или популярных сетевых проектов. А после регистрации таких незанятых доменов могут потребовать у вас определенную сумму денег за переуступку. Разумеется, если речь идет о маленьком независимом проекте предприятия среднего звена, на посещаемости и тем более — на финансовом положении компании по-

добные действия со стороны киберсквоттеров никак не скажутся. А если дело касается бизнес-гигантов типа Microsoft?

За примерами далеко ходить не надо. Несколько лет назад один американский журналист зарегистрировал на себя домен `mcdonalds.com` с целью привлечения внимания общественности к проблеме регистрации доменных имен, практически любое из которых можно было тогда приобрести, имея в кармане порядка 100 долларов. Американская общественность обратила внимание на подобный шаг, а вместе с ней на журналиста и его новое доменное имя обратила внимание сама компания MacDonalds, которая предложила ему неплохую денежную компенсацию за отказ от зарегистрированного домена.

Весной 1997 года другой американский гражданин зарегистрировал на себя несколько интересных доменных имен: `encarta.com`, `visualbasic.com` и др. Вам эти названия ничего не напоминают? Счастливчику из Штатов, отхватившему такой лакомый кусочек, теперь они ничего не напоминают, ибо он получил за их переуступку кругленькую сумму от Microsoft. Той же Microsoft пришлось еще раз раскошелиться для получения домена с простеньким именем — `microsoft-.com`. Заметьте, насколько жизненно важен для корпорации Билла Гейтса оказался какой-то маленький, никому не заметный дефис!

Распределение обязанностей по работе над Web-сайтом

Процесс проектирования, реализации, продвижения и поддержки любого Web-сайта подразумевает разветвленную структуру взаимосвязанных действий, функций и задач, поставленных перед авторами ресурса. Это разработка общей концепции проекта и составление схематического плана строения сайта, подготовка и оптимизация графических элементов и проработка навигации сайта, написание скриптов и реализация прототипа проекта в виде HTML-кода, мероприятия по раскрутке и продвижению сайта и регулярное обновление и поддержка интернет-проекта. Каждая из этих функций является отдельной обязанностью для одного или нескольких человек, отвечающих за данную область развития интернет-проекта.

Разумеется, такая система приемлема в случае проектирования крупных многоцелевых проектов, реализация которых позволяет их создателям набирать на работу целый штат профессионалов. Когда же речь заходит о разработке небольшого ресурса или о создании домашней Web-страницы, многие из вышеупомянутых функций предписываются небольшому составу разработчиков, а порой одному человеку.

Приведенные ниже характеристики различных специальностей, которые так или иначе могут быть задействованы в системе управления Web-сайтом, по-

могут разобраться в тонкостях этого непростого дела, подскажут верное решение проблемы нехватки рабочих рук и, самое главное, — поспособствуют координации действий по управлению Web-сайтом:

- художник-дизайнер — т. к. профессиональный дизайнер имеет художественное образование (или как аналог — природный дар к художественно-креативному видению вещей), ему доверяют такую немаловажную, а в некоторых случаях — первостепенную задачу, как разработка общей концепции и стиля будущего проекта. Естественно, он должен уметь свободно работать с различными графическими и издательскими пакетами, иметь представление о художественном вкусе и т. д. Иногда деятельность художника-дизайнера ограничивается проработкой концепции сайта и созданием схематических набросков оформления проекта, что впоследствии переносят в компьютерную среду уже другие специалисты;
- Web-дизайнер — в его обязанности входит перенос созданного художником-дизайнером стилевого оформления проекта на Web-платформу. Другими словами, Web-дизайнер должен уметь разместить отдельные элементы графики на странице без ущерба для самого интернет-проекта. При необходимости он оптимизирует графику, создает рекламные баннеры и анимированные объекты (логотипы, навигационные меню и т. д.), осуществляет графический Slicing (разрезание целого элемента на несколько частей для более удобной компоновки на странице). Иногда в обязанности Web-дизайнера может входить создание графических карт-изображений, тестирование отображения изображений в браузере и пр.;
- HTML-кодер (HTML-верстальщик) — в его задачу входит реализация общей концепции проекта с учетом особенностей стиля и графического оформления в виде HTML-кода. То есть, он должен свободно владеть HTML, Dynamic HTML и CSS, знать все их особенности и "подводные камни", уметь создавать "чистый" код, который будет одинаково отображаться во всех наиболее популярных браузерах, тестировать Web-сайт при различных разрешениях монитора и количестве цветов, проверять все активные элементы Web-страницы (гиперссылки, кнопки, баннеры и т. д.) на функциональность. Иногда HTML-верстальщику требуются базовые знания JavaScript/VBScript, CGI/Perl, Flash;
- Web-программист — требования к нему могут сильно варьироваться в зависимости от конкретных задач, особенностей проекта или его учредителя, но чаще всего данный специалист должен владеть Java, C/C++, MySQL, Perl, PHP, ASP, SSI и др. Разумеется, он обязан представлять себе, что такое HTML, CSS, JavaScript/VBScript, а также иметь (как минимум) начальные навыки работы с ними. Но, как правило, все Web-программисты свободно владеют многими технологиями, суть которых содержится в функциях HTML-кодера или Web-мастера;

- Web-мастер — несколько обобщенное понятие, в большинстве случаев подразумевающее совмещение функций Web-дизайнера, HTML-кодера и Web-программиста. Очень часто на Web-мастера приходится регулярное обновление сайта интернет-проекта;
- администратор сервера — следит за бесперебойной работой Web-сервера интернет-проекта, устраняет возникшие сбои и ошибки, обеспечивает защиту сервера и сохранение данных. Естественно, должен свободно владеть серверными настройками (чаще всего, работа идет под Apache или MS Internet Information Server) и знать установленные операционные системы;
- контент-менеджер (контент-оператор) — в случае с контент-проектом, т. е. когда ресурс прежде всего ориентируется на постоянно изменяющееся содержание сайта, требуются услуги контент-менеджера, который следит за отдельной отведенной ему областью проекта: проверяет входящую информацию, редактирует, дополняет или удаляет имеющиеся данные, обрабатывает пользовательские запросы, ведет колонку вопросов и ответов раздела (FAQ или ЧаВо), консультирует клиентов, контролирует работу средств управления информационным наполнением. В ряде случаев контент-менеджеру необходимо знание баз данных и некоторых языков программирования;
- ведущий проекта (редактор/администратор проекта) — как правило, контролирует всю работу по управлению и сопровождению функционирующего интернет-проекта и имеет в своем подчинении несколько человек, исполняющих обязанности ведущих (редакторов) отдельных разделов проекта. Эта специальность подразумевает контроль за информационным содержанием сайта, нормальной работой всех активных элементов Web-страниц (в случае проблем он обращается к HTML-кодеру или Web-мастеру), сроками обновления отдельных разделов, их техническим и информационным развитием и т. д. Если проект размещает данные других людей (тексты, графика, аудио- и видеоматериалы, программное обеспечение и пр.), то в задачи редактора проекта (или раздела) входит общение с авторами размещенных на сайте проекта материалов, контроль за соблюдением авторских прав и требований (по этой причине ведущего, редактора или администратора проекта/раздела иногда называют копирайтером);
- MailList-менеджер — это человек, который управляет процедурой подписки на какую-либо информацию проекта, а также прямыми рассылками, носящими строго информационный характер. Основным требованием является хорошее знание языка, на котором будет осуществляться рассылка/подписка, умение точно формулировать суть информации, а также способность разрешения затруднений, возникших у подписчиков проекта;

- корректор — в его задачи входит проверка текстового материала (а также текстовых фрагментов на графических объектах — кнопках, баннерах и т. д.) на наличие ошибок. Приветствуется базовое знание HTML и информационных технологий в целом;
- интернет-маркетолог — с точки зрения долголетия проекта, его рентабельности и посещаемости, профессия интернет-маркетолога — одна из самых важных. В его обязанности входит продвижение готового проекта в Интернете, рекламные мероприятия и PR-акции: регистрация сайта проекта в каталогах ресурсов и поисковых системах, участие в сетях баннерообмена и партнерских программах, прямые рекламные рассылки, предоставление временных скидок на услуги и бесплатного материала и т. д. и т. п. Само собой разумеется, интернет-маркетолог должен знать специфику рынка, на конкретном сегменте которого будет продвигаться проект, разбираться в технических особенностях работы с поисковыми роботами в ходе регистрации сайта проекта, уметь составить лаконичное и информативное описание проекта, а также рекламное письмо потенциальному партнеру/пользователю, иметь представление об инфраструктуре и правилах обмена рекламными носителями в Интернете и многое другое. Приветствуется рекламное образование, а также знание некоторых технологий — FTP, HTML и пр.

Как видите, много профессий, хороших и разных. Но грамотное исполнение поставленных задач каждым специалистом — еще не залог успешной работы интернет-проекта. Необходимо одинаково представлять цели проекта и работать, как говорится, в одной команде, т. е. не обособленно, замыкаясь на личных обязанностях, а в цепочке профессионалов, каждое звено которой определяет будущую перспективность, популярность и работоспособность интернет-проекта.

Работа по исследованию аудитории Web-сайта

Одним из важнейших компонентов процесса интернет-проектирования является исследование пользовательской аудитории разрабатываемого проекта. Несмотря на основные цели, которые ставят перед собой руководители интернет-проектов в ходе позиционирования — простая бизнес-визитка компании в Сети или персональный тематический ресурс, рекламные идеи или электронный бизнес и т. п. — главной задачей остается ориентация на своего пользователя (потребителя). Четкое представление своей реальной и потенциальной аудитории позволяет изначально позиционировать проект оптимальным образом, получать отдачу от работы в Сети, привлекать рекламодателей, проводить маркетинговые исследования и многое другое.

Реальная аудитория проекта определяется спустя некоторое время после появления Web-сайта в Интернете (при помощи рейтингов, регистрационных

мероприятий, данных Cookies и IP-адресов, о чем речь пойдет несколько позже). Потенциальную аудиторию проекта можно определить путем следующих действий:

- детальный анализ прямых и косвенных конкурентов Web-сайта — необходимо изучить ваших конкурентов до того, как проект увидит свет. Анализ технологических возможностей, пользовательских предпочтений, динамики посещений, ознакомление с потребительскими отзывами и комментариями — основные действия на данном этапе. Тщательный анализ деятельности конкурентов позволит вам еще на стадии подготовки проекта к "запуску" узнать основные плюсы и минусы сайтов, аналогичных вашему, сопоставить цели и задачи своего Web-сайта с требованиями и предпочтениями потенциального пользователя, усвоить и перенести на свой проект информацию о своих конкурентах.

  По окончании анализа следует определиться с дальнейшими действиями: либо вы посредством своего интернет-проекта выходите на рынок и начинаете предлагать пользователю услуги, как минимум, не худшего качества, чем у ваших конкурентов (желательно лучшего), либо вы откладываете выход проекта в Сеть до тех пор, пока его услуги и/или наполнение (смотря на что вы делаете ставку) не будут соответствовать определенным нормам потребительского качества. В сложившейся ситуации многие разработчики решаются пойти на риск и, несмотря на несоответствие потребительским предпочтениям, анонсируют свой проект. Такой риск абсолютно не оправдан с точки зрения рентабельности: потребитель не станет пользоваться менее качественными услугами, получать менее свежую и интересную информацию, посещать менее удобный и исчерпывающий по своему содержанию проект;

- тщательная проработка контента — если на сайт вашего проекта зашел посетитель — это хорошо. Но этого мало. Нужно, чтобы он заходил туда снова и снова, и с каждым разом не только не разочаровывался в нем, но и повышал оценку вашего проекта. Для этого необходимо тщательно проработать содержание интернет-проекта: информационное наполнение, систему навигации по сайту, максимально увеличить удобство и быстроту доступа к данным и пр.;

- интерактивность — не делайте свой сайт скучным перечнем расценок на услуги или бесконечной базой данных по товарам — вовлеките посетителя (потенциального потребителя/клиента/партнера) в общественную жизнь проекта, стимулируйте его на общение не только с вами как разработчиками ресурса, но и с другими пользователями. Убедите каждого посетителя в том, что именно его мнение необходимо знать руководителям проекта с целью формирования мероприятий по его совершенствованию. Приме-

ром средств, повышающих интерактивность проекта, являются всевозможные опросы и голосования посетителей, розыгрыши и скидки, лотереи и викторины, тесты и анкетирование и др.

Знание потенциальной аудитории вашего интернет-проекта поможет быстро сориентироваться на рынке товаров или услуг, гибко менять схемы работы с пользователями и потребителями, безболезненно адаптироваться к любым изменениям в вашей тематической нише и смело идти в ногу с ее тенденциями.

ГЛАВА 12

Композиция Web-сайта

Понятие композиции Web-сайта

Под композицией Web-сайта принято понимать общее (схематическое) расположение всех составных элементов графического макета будущего Web-сайта. Составными элементами могут быть:

- логотип;
- текст;
- имиджевые элементы;
- декоративные элементы;
- пиктограммы.

Рассмотрим более подробно каждый из элементов.

Логотип

Логотип считается неотъемлемой частью любого Web-сайта независимо от того, корпоративный это сервер или авторский контент-проект. Логотип несет в себе атрибут оригинальности, узнаваемости, неповторимости и, что самое главное, отличия от других Web-сайтов подобной направленности.

Как правило, логотип размещается вверху слева, вверху справа или вверху по центру страницы: в любом случае он будет хорошо виден при заходе на сайт и призван привлекать первоначальное внимание посетителя.

Структурно логотип может состоять из следующих компонентов:

- иллюстрация;
- иллюстрация + название;

- иллюстрация + аббревиатура;
- название;
- аббревиатура.

На рис. 12.1—12.4 приведены примеры логотипов с разным структурным построением.

Рис. 12.1. Логотип поисковой системы и интернет-портала Rambler

Рис. 12.2. Логотип компании "Петерстар"

Рис. 12.3. Логотип компьютерной библиотеки GetInfo.Ru

Рис. 12.4. Логотип хостинг-провайдера Net.Ru

Текст

Текст занимает одну из ведущих позиций в композиции Web-сайта, т. к. он несет в себе информацию, предназначенную посетителям.

Основные типы текста на Web-сайте:

- заголовки;
- подзаголовки;
- основной текст;
- гиперссылки.

Заголовки, в основном, используются для наименования разделов верхнего уровня, поэтому должны быть хорошо видны и читаемы. Реализуются либо графикой с использованием шрифта большого размера, либо посредством группы тегов `<H1>...<H6>`.

Подзаголовки используются для обозначения подразделов или для визуального выделения наиболее значимых частей текстовых блоков. Чаще всего реализуются с помощью тегов `<B>`, `<STRONG>`, `<BIG>` или `<H>` низшего уровня.

Основной текст несет в себе главный смысл конкретного текстового блока: в новостной ленте — это текст новости, в электронном магазине — описание

товара и т. д. Используется обыкновенное начертание (без курсива, подчеркивания и выделения жирным), размер шрифта 10—12 пикселов или в процентном соотношении (70—80%).

Гиперссылки по умолчанию отображаются с подчеркиванием, однако, с помощью стилевых шаблонов CSS их вид можно преобразить: они могут быть наклонными и жирными, подчеркнутыми и перечеркнутыми, только в верхнем регистре, любого цвета и т. д.

Имиджевые элементы

Имиджевые элементы — это объекты графического формата, визуально подчеркивающие прочие элементы композиции, наделенные смысловой или функциональной составляющей (логотипы, символика, блоки услуг, блоки товаров и пр.).

Никакой информации имиджевые элементы не несут, а только дополняют собой основные элементы Web-сайта.

На рис. 12.5—12.7 приведены некоторые примеры использования имиджевых элементов.

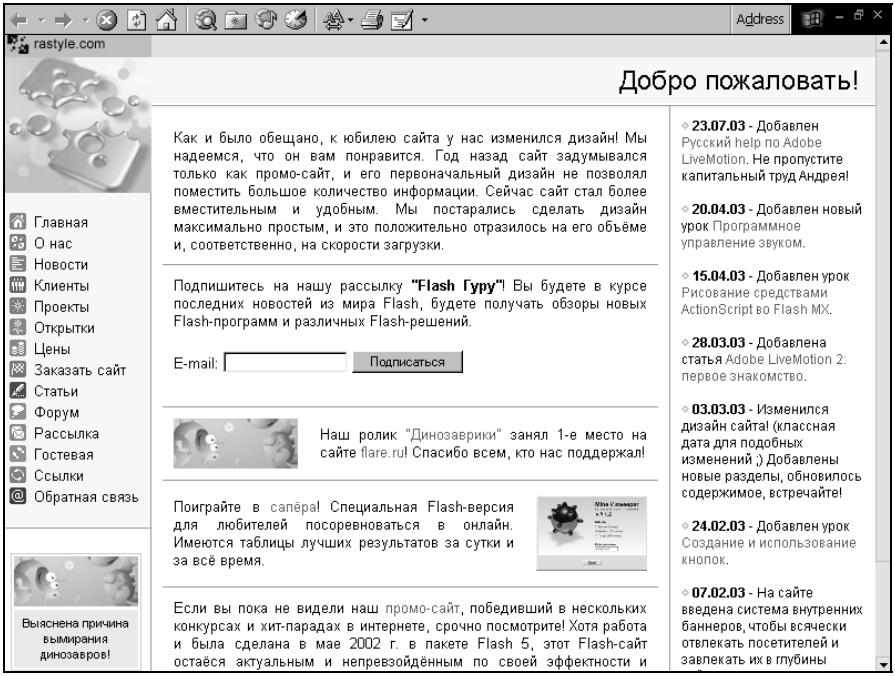


Рис. 12.5. Web-сайт студии дизайна Rastyle (использование имиджевого изображения, визуально дополняющего логотип)

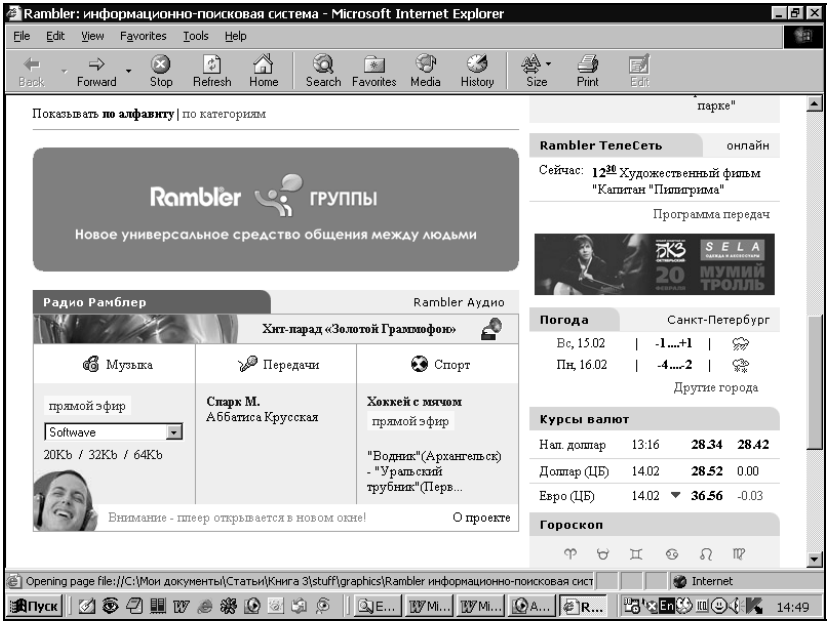


Рис. 12.6. Промо-блок "Радио Рамблер" в поддержку хит-парада "Золотой Граммофон" (в строке с заголовком крупным планом используется фрагмент изображения граммофона, носящий имиджевый характер)

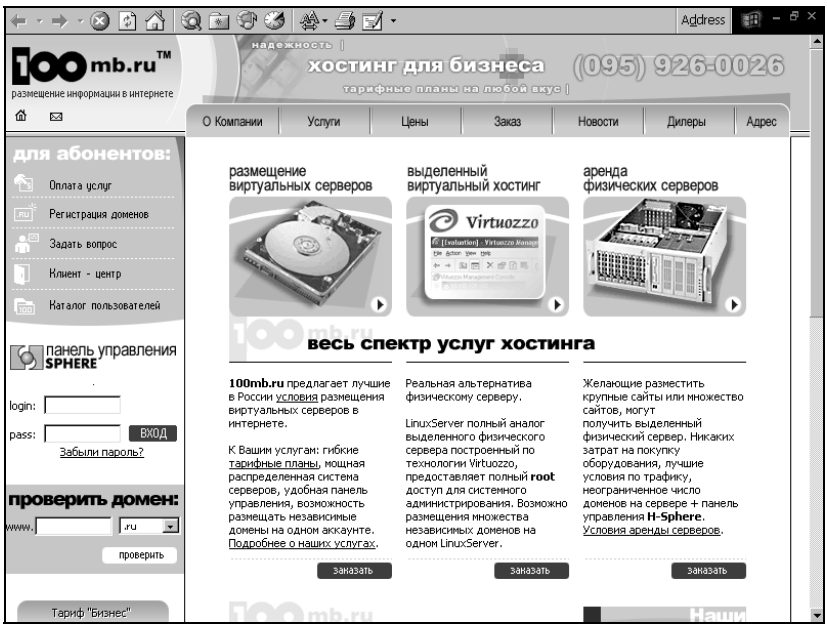


Рис. 12.7. Блок услуг хостинг-провайдера 100mb.Ru снабжен имиджевыми изображениями, визуально подчеркивающими каждую из предлагаемых услуг

Декоративные элементы

Функция *декоративных элементов* уже понятна из названия — украшение отдельных фрагментов Web-сайта для придания им наглядности и привлекательности. В роли декоративных элементов могут выступать всевозможные стрелки, маркеры, точки, пунктиры, разделители, геометрические фигуры и многое другое.

На рис. 12.8—12.10 приведены некоторые примеры использования декоративных элементов.

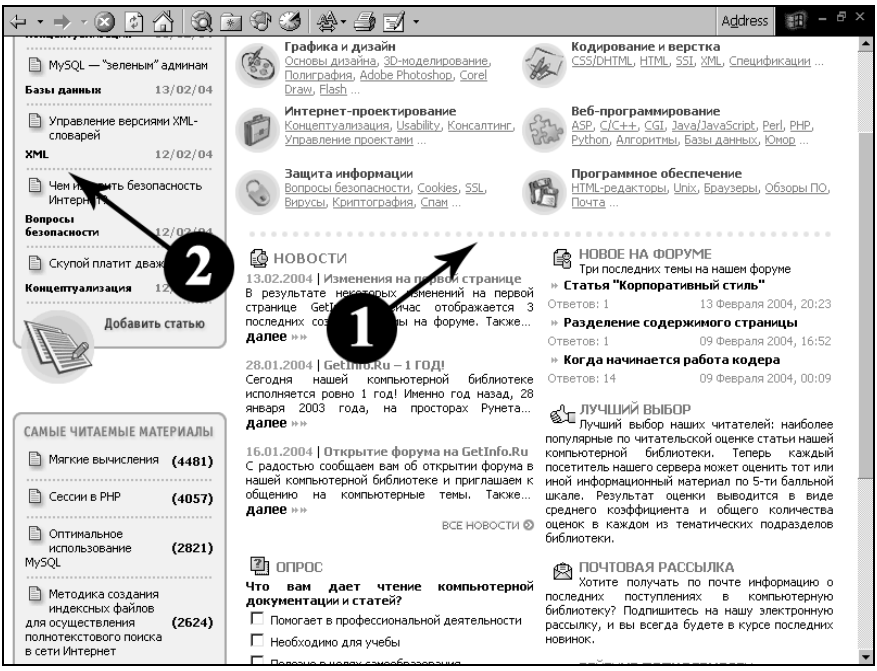


Рис. 12.8. Первая страница компьютерной библиотеки GetInfo.Ru
(1 и 2 — пунктирные разделители, выполняющие декоративную функцию)

Пиктограммы

Каждого, кто не чужд мира компьютерных технологий, сегодня окружает огромное количество изобразительной информации: фотографии и эскизы, реклама и элементы фирменного стиля, имиджевая и деловая графика. Любой пользовательский интерфейс включает в себя эти и многие другие атрибуты, призванные проиллюстрировать предоставляемую информацию, сделать ее более привлекательной и запоминаемой.

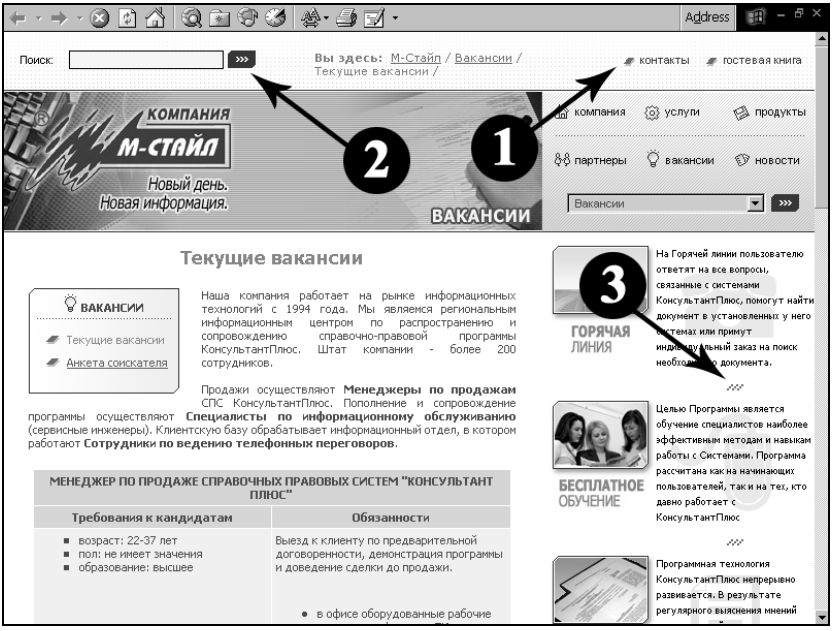


Рис. 12.9. Web-сайт компании "М-Стайл"
(1—3 — декоративные элементы различного предназначения)

Рис. 12.10. Стартовая страница хостинг-провайдера 100mb.Ru
(декоративные элементы: 1 — указатель, 2 — пунктирный разделитель)

Рабочая область любого Web-сайта также подпадает под определение пользовательского интерфейса со своей системой навигации, управлением потоками данных, функциями поиска и сортировки информации и пр. На страницах интернет-проектов находят достойное применение фотографии, рекламные модули, имиджевые заставки, логотипы и прочие компоненты сложившейся инфраструктуры. В составе последней пиктограммы (иконки) играют существенную роль.

Пиктограммы представляют собой небольшие графические изображения тематической направленности.

Требования

Если в среде разработчиков программного обеспечения сложились достаточно строгие требования к созданию иконок (в основном, это формат BMP, размер 16×16 или 32×32 пиксела), то Интернет предоставляет дизайнеру широкое поле деятельности. Пиктограммы для размещения в Интернете могут быть в формате GIF, JPEG, PNG и даже Flash, размеры могут варьироваться от миниатюрных 7×7 до "великанов" 70×70.

Пиктограмма, разработанная для встраивания в программу, ограничена количеством использованных цветов (как правило, 256) — в отличие от Web-иконок, палитра которых может содержать миллионы цветов.

Функции

Пиктограммы, будучи размещенными на страницах Web-сайта, могут выполнять две основные функции:

- навигационная — пиктограмма характеризует определенный раздел сайта и является гиперссылкой (рис. 12.11) на него;
- имиджевая — пиктограмма не является ссылкой, размещена в качестве некоего иллюстративного материала (рис. 12.12), дополняющего конкретный текст или другую графику.

Концепция

Вопрос разработки пиктограмм является актуальным для любого Web-дизайнера. Перед ним возникает целый ряд требований, которым иконки должны удовлетворять. Выделим четыре важных принципа в процессе разработки пиктограмм:

- ассоциативный ряд;
- доступность;

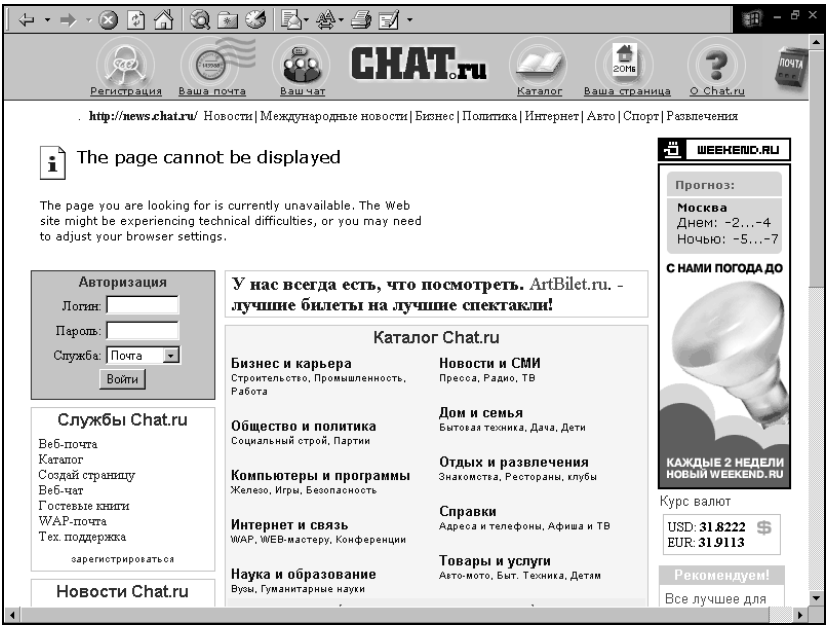


Рис. 12.11. На сервере бесплатной почты и хостинга Chat.Ru каждое тематическое изображение в меню навигации является ссылкой в соответствующий раздел Web-сайта

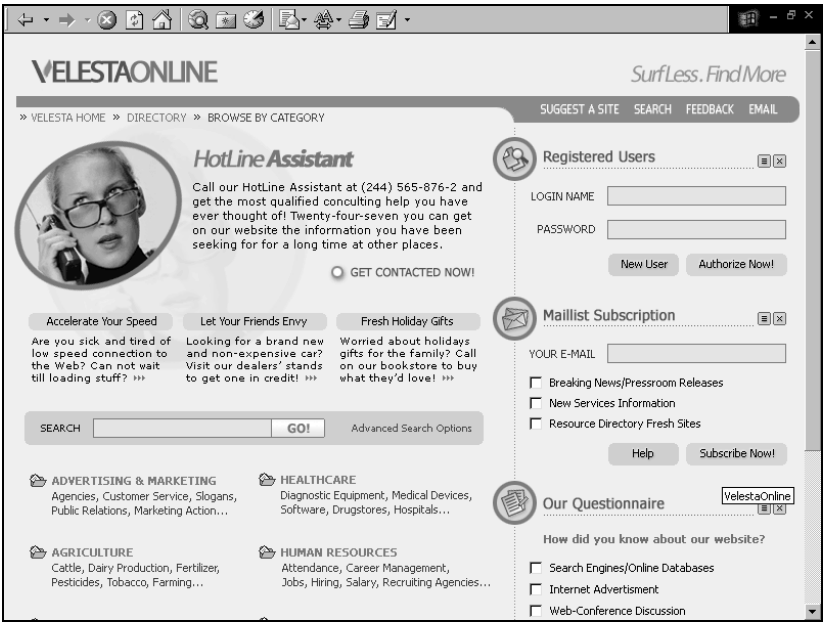


Рис. 12.12. На сайте VelestaOnline представлена имиджевая функция пиктограмм (изображения рядом со словами "Registered Users", "Maillist Subscription" и "Our Questionnaire" только дополняют эти надписи и соответствующие тематические блоки информации сайта)

- эстетика;
- оригинальность.

Далее рассмотрим более подробно каждый из принципов.

Ассоциативный ряд

Прежде чем открывать графический редактор и начинать рисовать иконку, необходимо задуматься над тем, как будущую пиктограмму можно изобразить. Логично, что на иконке для раздела "Тиражирование CD" будет уместно наличие двух или трех компакт-дисков, а на пиктограмме для рубрики "Корм для кошек" — изображение кошачьей мордочки. Но когда дело доходит до распространенных или, наоборот, специфических разделов, следует призвать в помощники ассоциацию.

Другими словами, необходимо перечислить те образы, слова и выражения, которые ассоциируются с предметом разрабатываемой пиктограммы. Например, с чем может ассоциироваться раздел "Форум"?

- общение > люди > изображение человеческих силуэтов;
- обмен информацией > сообщения > изображение листов бумаги;
- поиск информации > вопрос участникам форума > изображение вопросительного знака.

Выше были представлены лишь некоторые варианты составления ассоциативных цепочек, позволяющих, в конечном итоге, определить оптимальный вид будущей пиктограммы. Этот пример достаточно прост, т. к. использует весьма распространенный тематический объект — форум. Гораздо сложнее составить ассоциативную цепочку для специфических пиктограмм, например, такой, как "Гидрофобизирующие материалы, кашированные стеклохолстом" (кстати, реальный пример):

- гидрофобизирующие > не пропускающие влагу > вода > изображение капли;
- стеклохолст > стекло > блестящая поверхность с бликами > холст > лист бумаги > изображение листа бумаги с зеркальными бликами.

Цепочка 1 + Цепочка 2 = Изображение листа бумаги с зеркальными бликами, на фоне которого располагается капля воды.

В данном случае слово "кашированные" представляет собой специфический термин, поэтому не было включено в составление ассоциативной цепочки. Также следует добавить, что такая пиктограмма может носить исключительно имиджевую функцию — в силу своей специфичности.

В завершение разговора о принципе ассоциативного ряда следует упомянуть понятие "семантика" — раздел языкознания, изучающий значение единиц языка, прежде всего, слов. Семантический подход к составлению образа пик-

тограммы позволяет формировать группы значений того или иного тематического направления и выбирать наиболее значимые смысловые элементы для разрабатываемых пиктограмм (рис. 12.13).

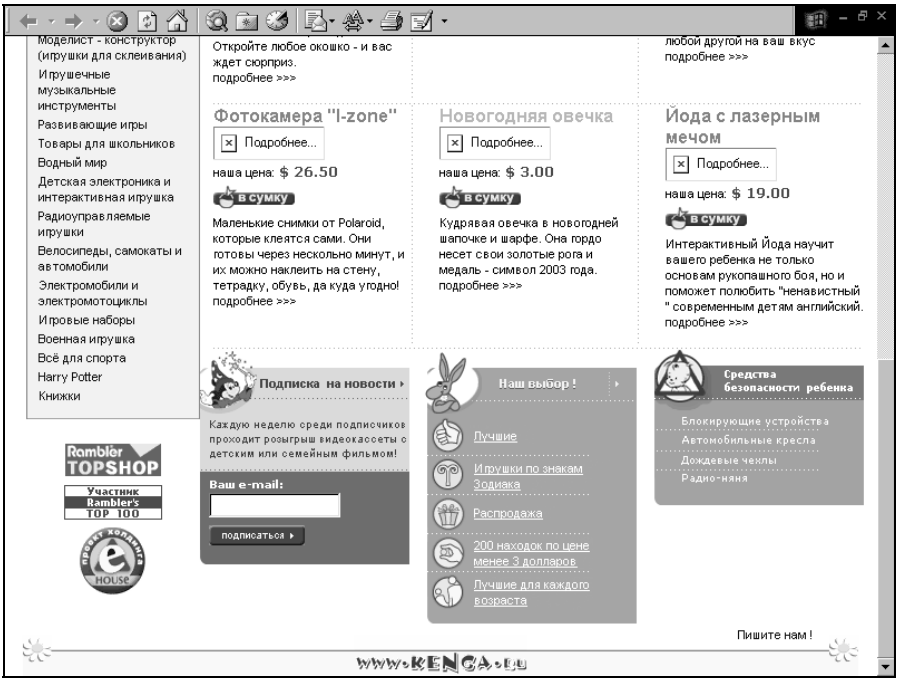


Рис. 12.13. Реализация ассоциативного ряда пиктограмм в рубрике "Наш выбор" интернет-магазина детских игрушек Kenga.Ru

Доступность

Под доступностью пиктограммы подразумевается степень понимания пользователем тематического значения элемента. Другими словами, любая пиктограмма должна быть интуитивно понятна большинству пользователей. Чаще всего использование данного принципа строится на сложившихся стереотипах мышления человека.

Например, изображение принтера у большинства пользователей ассоциируется с печатью документа, а никак не с отделом оргтехники в компьютерном магазине. Вид открытой папки, выполненной в стиле Windows, дает понять, что пользователь, нажав на такую пиктограмму, сможет открыть какой-то документ.

Однако разработчика пиктограммы может ожидать несколько "подводных камней". Во-первых, используя "говорящую", но не обладающую явно выра-

женным значением иконку, можно поставить пользователя в тупик: например, у разных людей изображение раскрытой книги может вызвать совершенно различные смысловые значения — "Обучение", "Библиотека", "Справочник" и т. д. Во-вторых, пользователь может быть незнаком с символикой, использованной на пиктограмме, в том аспекте, в каком его видит дизайнер (рис. 12.14). Одним из самых распространенных примеров является изображение чашечных весов, которые часто используются для разделов по юриспруденции, законодательству, адвокатуре.

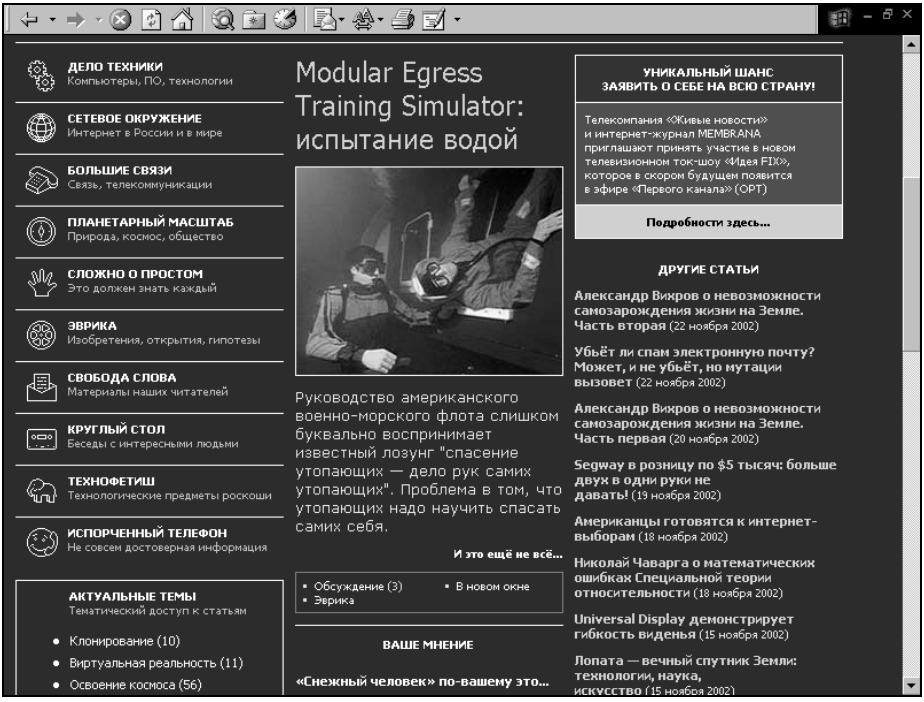


Рис. 12.14. Степень понимания пользователей значения иконок меню научно-популярного интернет-журнала "Мембрана": без дополнительного текста понять смысл пиктограмм достаточно сложно

Необходимо помнить, что устоявшийся образ, признанный людьми в повседневной жизни и быту, не всегда подходит для его реализации в Интернете. Например, на одном западном Web-сайте посетителям была предоставлена возможность самим снимать небольшие видеоклипы. С этой целью дизайнеры имитировали стандартную панель управления видеомагнитофоном с соответствующими функциональными кнопками (пиктограммами). В том числе, присутствовала кнопка начала записи, представленная в виде красного круга. Казалось бы, ассоциация с нужным действием налицо, однако многие поль-

зователи оказались в затруднительном положении. Разработчики сайта провели исследование и выяснили, что причиной затруднений с записью являлась та самая красная кнопка, которая определенной части пользователей больше напоминала кнопку остановки, нежели записи видеоклипа.

Эстетика

Людей всегда привлекает красивое, поэтому они предпочтут более красивую, более ярко выраженную пиктограмму иконке, выполненной на низком художественном уровне. Разумеется, о вкусах не спорят, и то, от чего будет без ума один пользователь, может вызвать недовольную мину у другого. Однако стремление к четкому, ярко выраженному (необязательно яркому!), визуально привлекательному и стильному образу пиктограммы (рис. 12.15) позволит получить дополнительное одобрение со стороны посетителей сайта.

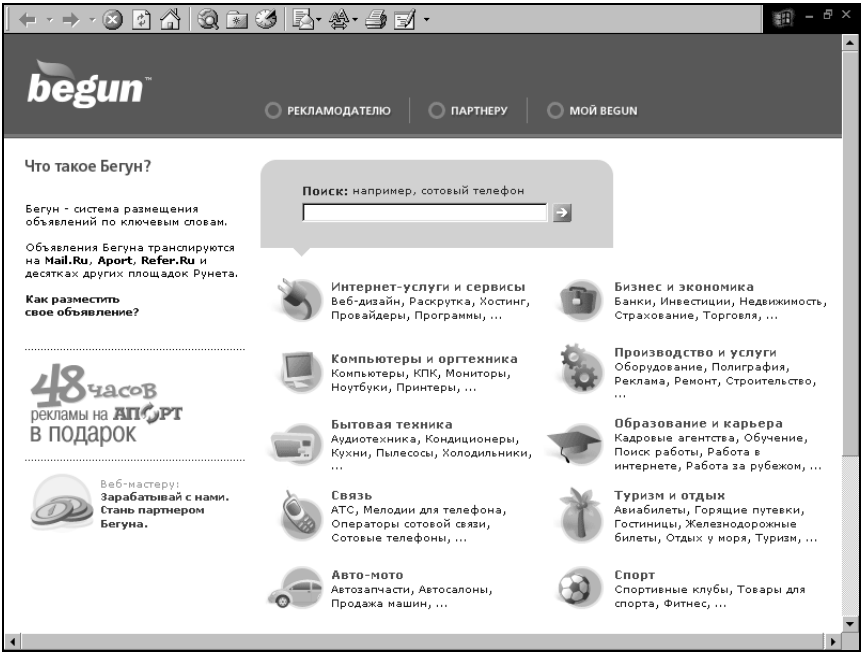


Рис. 12.15. Пиктограммы каталога ресурсов рекламной системы "Бегун", безусловно, не оставят равнодушными пользователей Интернета

Оригинальность

К сожалению, многие разработчики под оригинальностью пиктограммы понимают совсем не то, чего хотел увидеть пользователь. В итоге появляются странные, плохо идентифицируемые, непонятные и просто нелепые иконки.

Часть из них выглядит стильно и привлекает внимание (рис. 12.16), но никаких ассоциаций с предметом раздела сайта они не вызывают. Либо вызывают, но совершенно не те, на какие рассчитывал дизайнер.

Оригинальность пиктограммы — это свежий взгляд на устоявшиеся вещи. К примеру, изображение увеличительного стекла, которое большинство пользователей сопоставит с функцией поиска на сайте, может быть стандартным и невзрачным, а может быть выполнено с определенной долей дизайнерской фантазии. Разумеется, иногда стандартный вид иконки более "к лицу" самому сайту: значок архива в 99% случаев будет более узнаваем, если выполнен в виде стандартной пиктограммы WinZip.

Особенно хочется отметить, что разработка тематических пиктограмм для размещения на Web-сайтах — процесс не только творческий, но и психологический. В конечном итоге разработчику необходимо обработать множество вариантов, дабы отобрать один-единственный, наиболее удачный и понятный большинству пользователей. Сегодня пиктограммы — неотъемлемый атрибут практически любого интернет-проекта, требующий тщательной подготовки и заслуживающий не меньшего внимания, чем остальные компоненты Web-сайта.

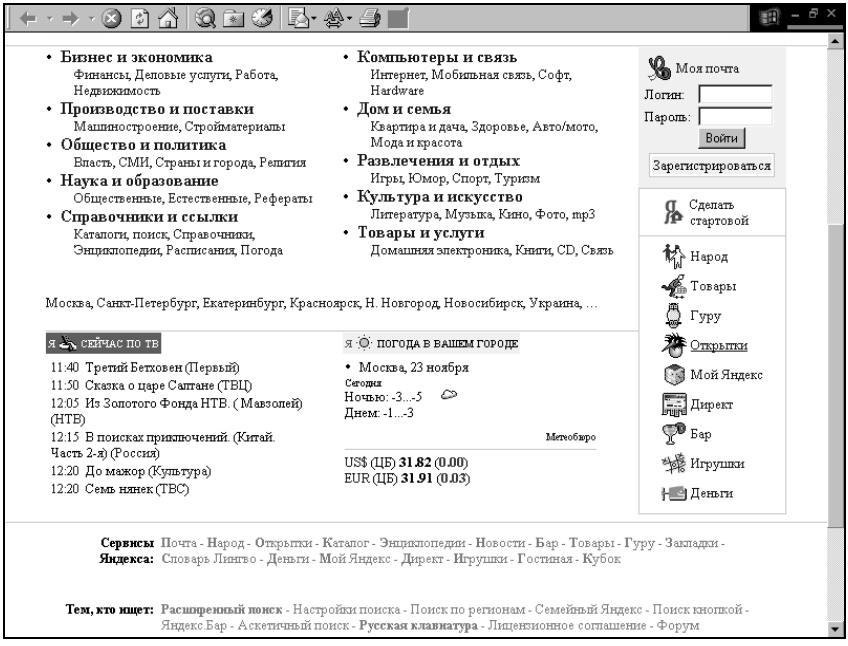


Рис. 12.16. Вид панели представления услуг интернет-портала и поисковой системы Яndex: иконки нарисованы в стиле "pixel design", выглядят свежо и неординарно

Разработка пиктограмм

В предыдущем разделе мы разобрались с тем, какие функции может выполнять пиктограмма на Web-сайте, как строить ассоциативные ряды и каким требованиям должны соответствовать Web-иконки. "А как нарисовать пиктограмму, зная все перечисленное выше?" — возможно, спросите вы.

Рабочий инструментарий

Для работы нам понадобится Adobe Photoshop (желательно не ниже версии 5.5) и развитое образное мышление. Если у вас отсутствует первое, то хорошим аналогом могут послужить Ulead PhotoImpact, Corel PhotoPaint или JASC PaintShop. Отсутствие второго критично и сильно осложнит работу над созданием Web-иконки.

Примитивы

В области 3D-графики и ландшафтного проектирования есть термин "примитивы", обозначающий простейшие геометрические фигуры: эллипс, прямоугольник, треугольник и пр. В плане Web-пиктограмм также есть своеобразные "примитивные" (а точнее, имеющие тенденцию стать стереотипами) образы. Это то, что всегда на виду: конвертики и листы бумаги с карандашом ("Обратная связь"), домики ("Главная страница"), увеличительное стекло или бинокль ("Поиск на сайте") и т. п.

Конверт

Попробуем нарисовать конверт, к примеру, запечатанный. Закроем глаза и мысленно проследим за каждой линией, составляющей образ конверта. Если вы еще пишете бумажные письма, достаньте настоящий конверт и проделайте то же самое, но уже с открытыми глазами. Проследили? Запомнили? Отлично. Тогда открываем Photoshop, создаем прозрачную заготовку с размером 20×20 пикселов (рис. 12.17), устанавливаем масштаб 1600% и выбираем инструмент **Pencil Tool** (клавиша <B> или соответствующий значок на панели инструментов).

Далее при размере деления в 1 пиксел рисуем рамку (рис. 12.18).

Затем рисуем края конверта: верхний с нахлестом на нижний (рис. 12.19).

"Пустые" области внутри конверта можно выделить и залить градиентной заливкой (рис. 12.20).

Телевизор

На заготовке с теми же размерами (20×20 пикселов) рисуем рамку — основу нашего будущего телевизора — и заливаем градиентом (экран телевизора с имитацией блика) (рис. 12.21).

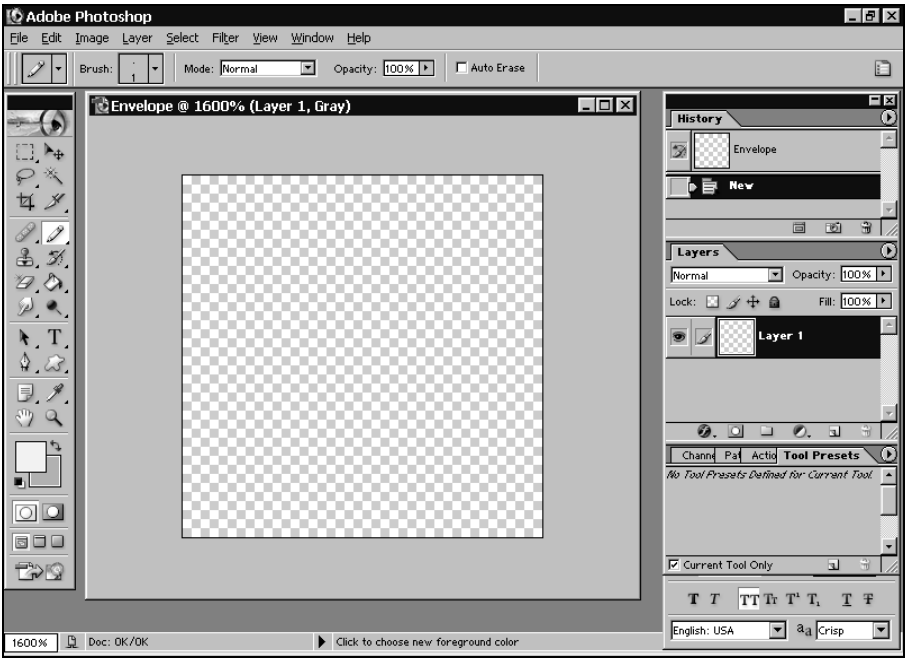


Рис. 12.17. Создание заготовки

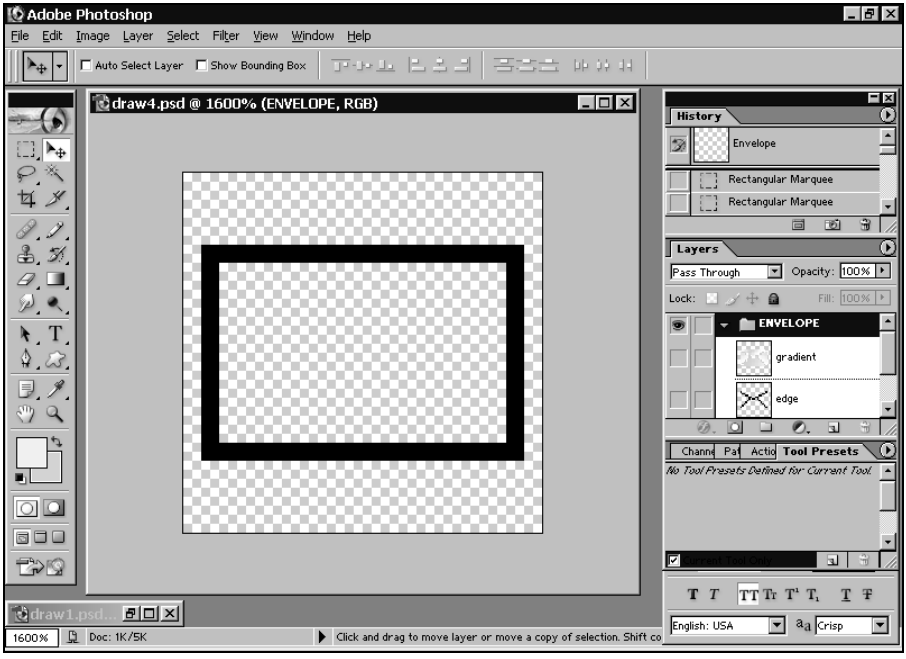


Рис. 12.18. Создание рамки толщиной 1 пиксел

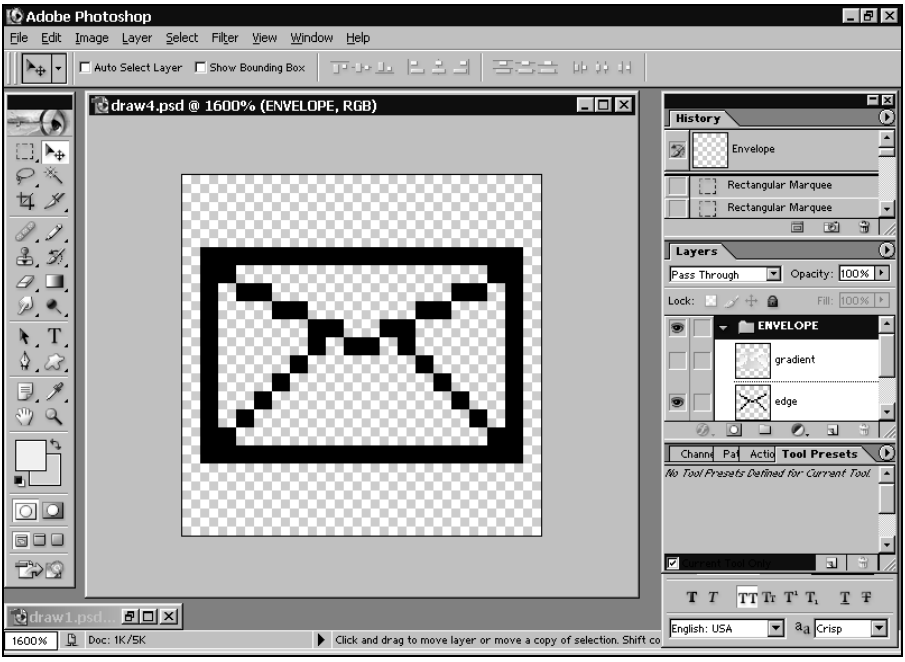


Рис. 12.19. Добавление краев конверта

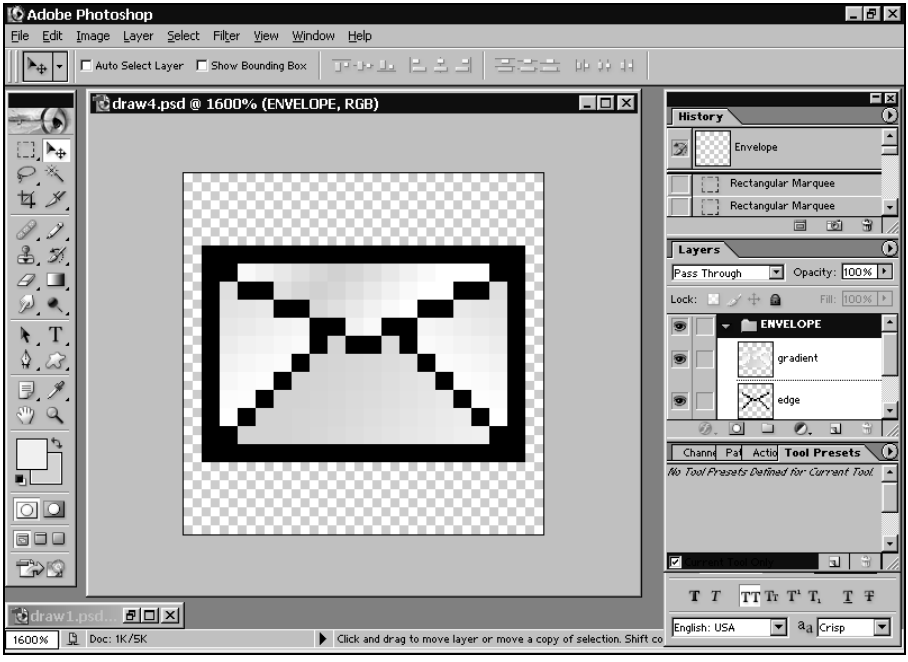


Рис. 12.20. Добавление градиентной заливки

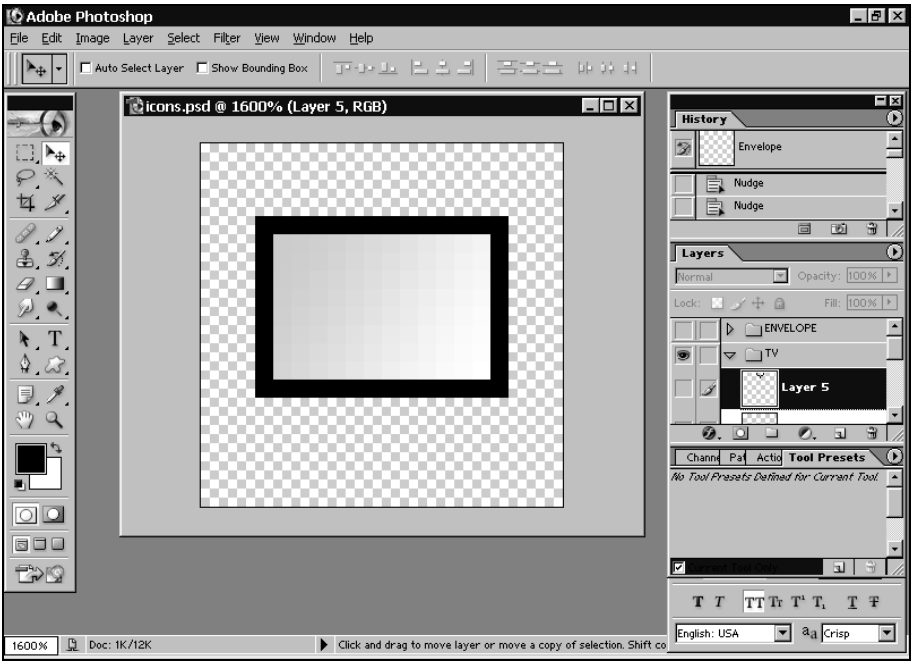


Рис. 12.21. Создание рамки и добавление градиентной заливки

Теперь придадим краям корпуса телевизора закругленности (удалим по одному пикселу с каждого угла каркаса телевизора), добавим приставку с ножками и индикатором включения/выключения, а напоследок сверху прикрепим антенну (рис. 12.22). Старомодно, зато подчеркивает образ телевизора.

Как видно из примеров, техника рисования достаточно проста, что обусловлено влиянием направления под названием "pixel design". Все объекты прорисованы попиксельно, без использования полутонов и закруглений. Попробуем усложнить задачу и нарисуем увеличительное стекло.

Увеличительное стекло

На заготовке размером 50×50 пикселов создаем круг и заливаем его цветом #2A77A8 (рис. 12.23).

На новом слое рисуем второй круг с радиусом, меньшим на 2 пиксела, чем предыдущий (заливаем белым — #FFFFFF). Это у нас окантовка увеличительного стекла, его оправа.

Теперь создаем круг с радиусом на 1 пиксел меньше, чем второй круг с белой заливкой, и делаем поперечный градиент от #F2EFEF к #B0D2FA. Линза с эффектом блика в темной оправе у нас есть (рис. 12.24).

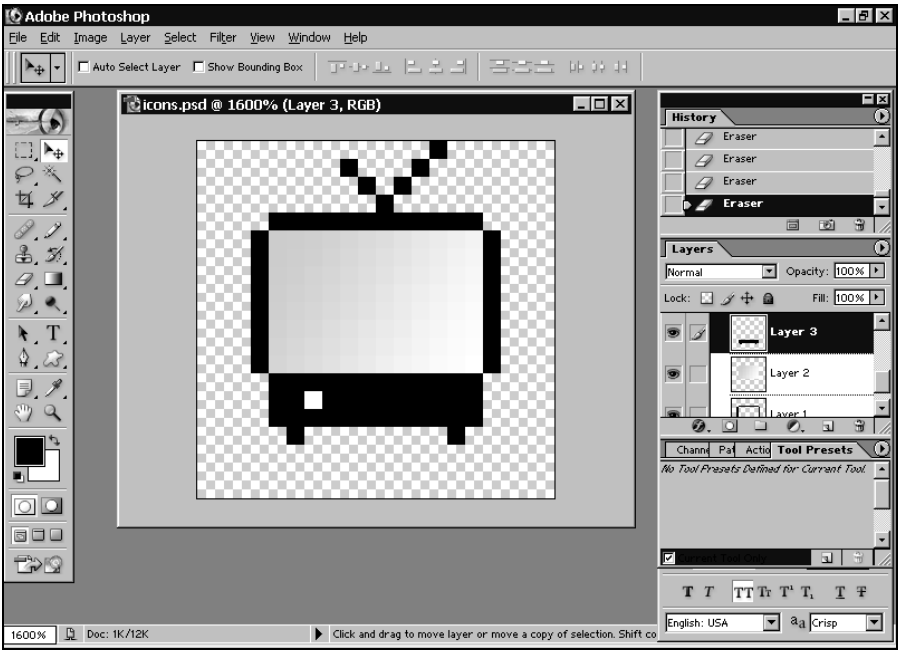


Рис. 12.22. Еще пара штрихов и телевизор готов!

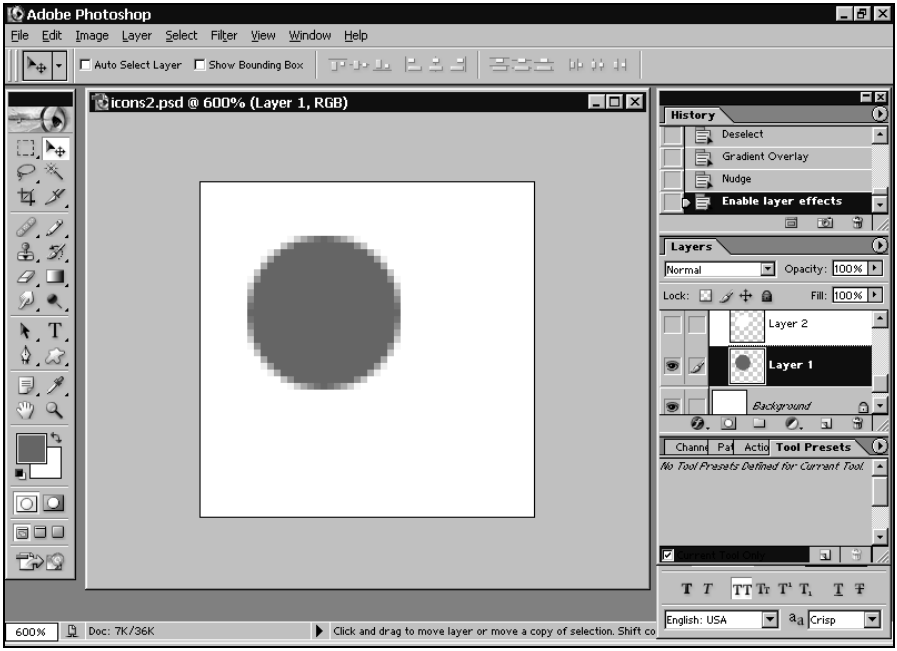


Рис. 12.23. Создание круга и заливка цветом

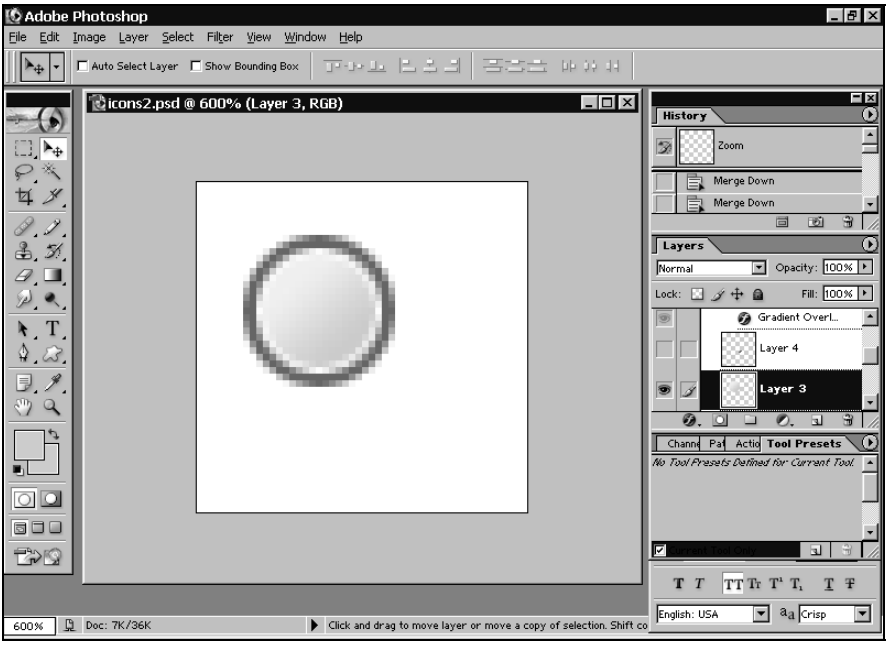


Рис. 12.24. Создание окантовки увеличительного стекла и линзы с эффектом блика

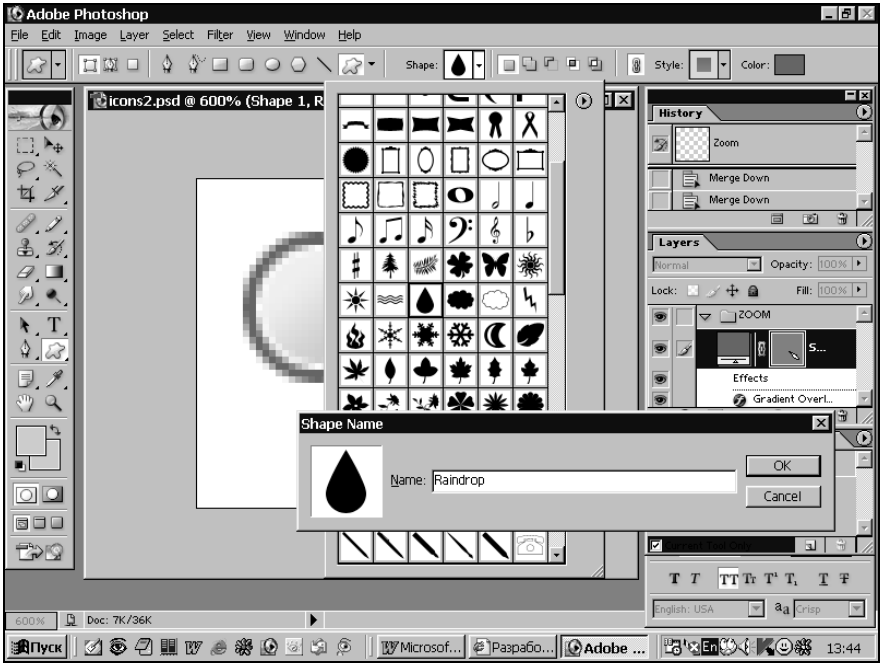


Рис. 12.25. Выбор векторного объекта **Raindrop** (Капля)

Осталось нарисовать ручку. Для этого на панели инструментов Photoshop выбираем опцию автофигур (Custom Shape Tools), а в появившейся библиотеке векторных изображений — объект **Raindrop** (Капля) (рис. 12.25).

Далее с помощью команды **Edit** | **Transform** | **Rotate** в главном меню поворачиваем фигуру до определения нужного ракурса. Градиент (от #367CAC к #6892BF) плюс крепежная деталь, соединяющая ручку со стеклом в оправе — и наше увеличительное стекло готово! (рис. 12.26).

На рис. 12.27 представлены все три пиктограммы в натуральную величину.

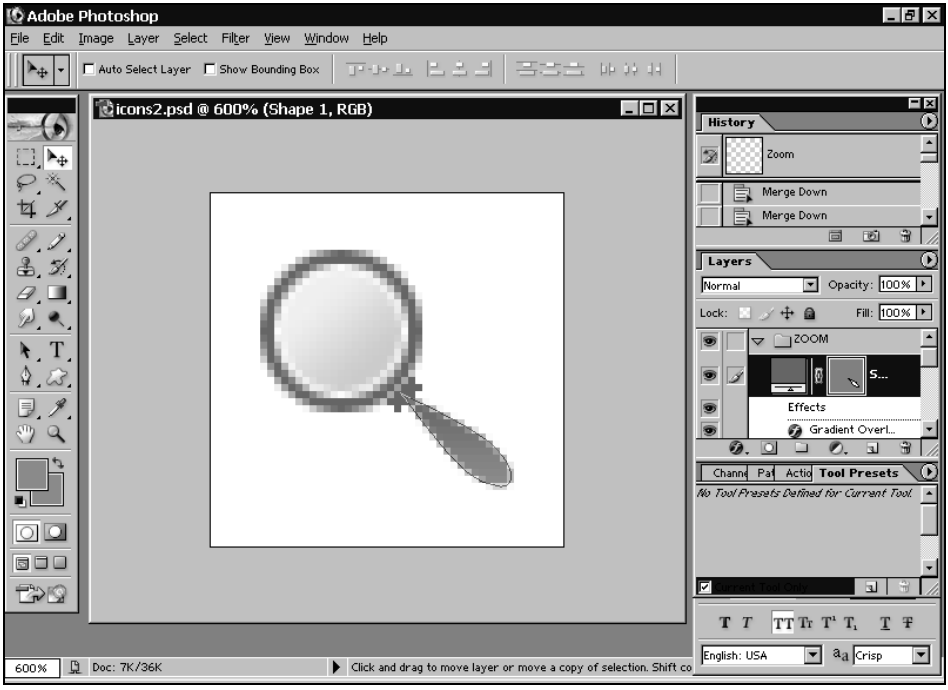


Рис. 12.26. Добавление ручки с крепежной деталью

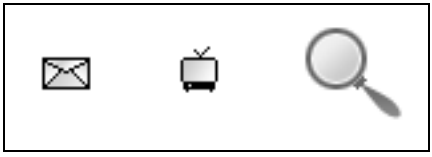


Рис. 12.27. Пиктограммы "Конверт", "Телевизор" и "Увеличительное стекло" в натуральную величину

ГЛАВА 13

Навигация по Web-сайту

Меню навигации

Любой Web-сайт, будь это любительская домашняя страница или сервер электронной коммерции, является источником информации. Независимо от тематики, актуальности, востребованности и практической ценности этой информации, она обычно представлена в определенной логической последовательности для достижения максимального удобства ее восприятия посетителями сайта в Интернете. В структурном плане такая последовательность достигается за счет организации *системы навигации* по сайту.

Понятие системы навигации

Наличие навигационной системы позволяет посетителю визуально определять ценность информации, скрытой в самых отдаленных уголках того или иного интернет-проекта, посредством изучения тематических заголовков. Каждый заголовок в сжатой форме характеризует конкретный раздел или группу подразделов сайта, посвященных определенному вопросу или теме, и является гиперссылкой на их полный вариант. Совокупность таких заголовков составляет упомянутую выше систему навигации по сайту.

Типы навигации

Существует несколько типов организации навигации по Web-сайту. Рассмотрим основные.

Текст

На заре развития Интернета как общедоступного информационного пространства текстовая навигация была единственным способом помочь посети-

телю разобраться, какая информация ему нужна, а какая нет. Текстовые ссылки — самый простой в плане реализации вариант информирования пользователя о том, что его ждет внутри раздела сайта (рис. 13.1).

Однако 90-е гг. XX века породили тенденцию к эстетизации составных элементов электронных документов в Сети, которая выражалась в стремительном повышении пользовательских требований к визуальному оформлению сайта, в том числе к его навигации. Несмотря на растущий интерес к таблицам стилей CSS, позволяющим несколько преобразить и приукрасить стандартный вид шрифтов, текстовая навигация в конце концов уступила "место под солнцем" своему графическому аналогу.

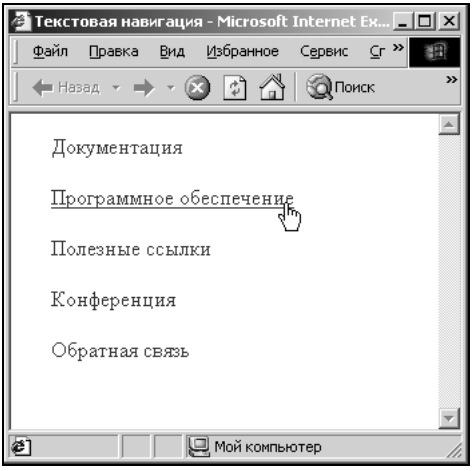


Рис. 13.1. Пример текстовой навигации

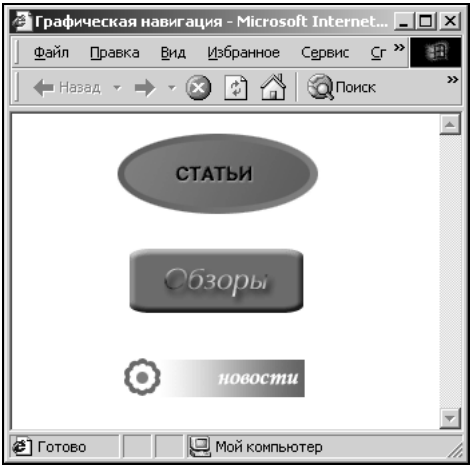


Рис. 13.2. Пример графической навигации

Графика

Графический вариант представления системы навигации (рис. 13.2) является, пожалуй, наиболее распространенным в Интернете. Оно и понятно: современные графические редакторы вкупе со всевозможными надстройками (plug-ins), эффектами и фильтрами способны создавать потрясающие образцы оформительских решений. Применение фотографических изображений, рисунков, а также анимации привлекло огромное количество Web-дизайнеров, которые предпочли визуальную привлекательность аскетичному тексту.

HTML-формы

Спецификация языка гипертекстовой разметки HTML позволяет размещать на Web-страницах ниспадающие и выборочные меню, которые, как правило,

позволяют сэкономить место на странице и представляют собой интуитивно понятные пользователю элементы рабочей среды Windows. Кроме этого, с помощью CSS можно существенно улучшить внешний вид систем навигации, реализованных посредством HTML-форм (рис. 13.3).

Рис. 13.3. Пример навигации с помощью HTML-форм

Java-апплеты

Java-апплеты представляют собой небольшие приложения, которые встраиваются в HTML-код Web-страниц и могут включать как текст, так и графику. Отображаются при наличии в браузере пользователя опции поддержки Java.

Flash

Системы навигации, разработанные на основе технологии Macromedia Flash, могут совмещать в себе растровую и векторную графику, анимацию, аудио и видео, а также реагировать на различные пользовательские манипуляции: нажатие "горячих" клавиш, перемещение курсора и пр. Для их отображения на Web-странице на компьютере посетителя сайта должен быть установлен специальный plug-in — Macromedia Flash Player.

Требования к системе навигации

При разработке системы навигации необходимо следовать некоторым обязательным требованиям, соблюдение которых поможет создать простую и удобную в обращении, интуитивно понятную любому пользователю навигационную структуру. Эти требования могут несущественно варьироваться

в зависимости от особенностей Web-сайта, представленной на нем информации, а также некоторых субъективных факторов разработчика. Однако разрабатываемая система навигации должна в конечном итоге соответствовать ряду критериев удобства пользования навигацией, называемому usability.

Расположение

Пространственное расположение системы навигации на странице играет одну из важнейших ролей в процессе знакомства посетителя с информационными разделами Web-сайта. Зайдя на любую страницу сайта, посетитель должен полностью видеть навигационную систему (или ее основную часть), не прибегая к прокрутке окна браузера. В противном случае посетитель, не найдя и намека на элементы навигации, может покинуть сайт. Вероятность того, что он станет самостоятельно исследовать весь текущий документ, мала.

Первое заблуждение разработчика

Некоторые разработчики "закрывают глаза" на критерий пространственного расположения системы навигации и помещают ее туда, куда им бы хотелось, полагая, что информационное наполнение их интернет-проекта является исключительным и неповторимым. Последний довод стимулирует ложную уверенность разработчиков в способности удержать посетителя при любых обстоятельствах.

Отсюда можно вывести первое правило: *размещать систему навигации следует в верхней части страниц сайта.*

Перемещение навигации по вертикали должно происходить не далее 1/4 видимого пространства окна, при этом необходимо учитывать разницу экранного разрешения пользовательских мониторов. Дальнейшее перемещение навигации повлечет за собой перестановку приоритетов внимания посетителя, которому будет сложно найти "точку отсчета" своих переходов по сайту относительно соседних элементов документа.

Человек устроен таким образом, что просматривать любой документ начинает либо слева направо, либо сверху вниз (за исключением жителей Японии, арабских стран и т. д.). Поэтому оптимальными вариантами размещения навигации можно считать "слева вверху" и "посередине вверху".

Идентифицируемость

Система навигации должна выделяться на фоне остальных элементов сайта и относительно его информационного наполнения. Если все элементы структуры сайта выполнены в едином ключе, пользователь будет испытывать определенные неудобства при попытке выделить среди однообразных, однотипных объектов систему навигации. Хорошая навигация всегда выделяется на

общем фоне сайта (но не бросается в глаза обилием пестрых красок или вызывающих фотографий), давая пользователю наглядное представление о содержании размещенных на сайте разделов.

Выделение системы навигации среди прочего наполнения сайта осуществляется самыми разнообразными способами. Это может быть подчеркивание, курсив и жирное начертание, маркированный список, цветовая заливка и наложение теней, изменение параметров конкретного пункта навигационного меню и т. д. и т. п.

Следует отметить, что наличие пояснительной надписи типа "Меню навигации" или "Содержание" рядом с пунктами навигационной системы — следствие нежелания разработчиков представить важнейший элемент сайта в выгодном для посетителя свете.

Для иллюстрации этого утверждения возьмем обыкновенный пульт дистанционного управления телевизором: при первом же взгляде на него становится понятно, что кнопки "1, 2, 3, 4, 5..." соответствуют определенному телевизионному каналу. При этом ни на одном ПДУ не присутствует пояснений вроде "Кнопки переключения каналов".

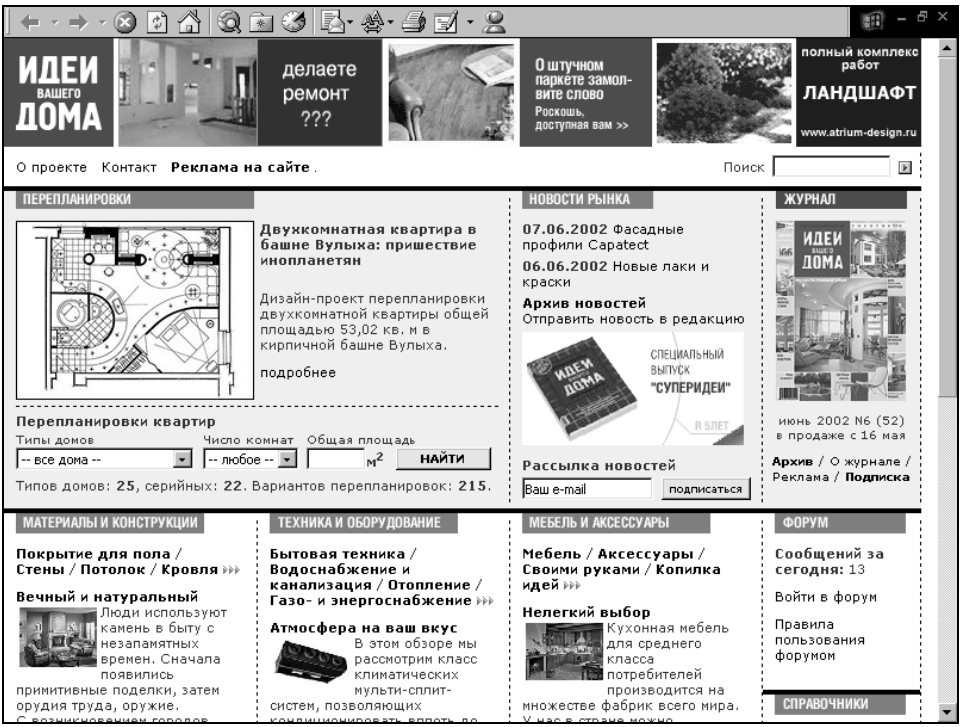


Рис. 13.4. Плохо идентифицируемая система навигации на сервере "Идеи вашего дома"

Примером интернет-проекта, Web-сайт которого содержит плохо идентифицируемую систему навигации, может послужить сервер "Идеи вашего дома" (рис.13.4).

В верхней части Web-сайта расположены небольшие графические элементы, информационное наполнение которых настолько расплывчатое, что становится непонятно: это обычная реклама или пункты меню? Ниже находятся заголовки разделов "Перепланировка" и "Новости рынка", архив и рассылка новостей. И только после этого идут наименования, больше всего похожие на элементы системы навигации. Однако из-за того, что блок навигационного меню по оформлению ничем не отличается от остальных информационных блоков сайта, система навигации становится трудноразличимой.

Второе заблуждение разработчика

Некоторые разработчики делают акцент не на навигационном меню, а на рекламных предложениях, цветных иллюстрациях, разъяснении причин, по которым посетителю следует воспользоваться именно их Web-сайтом, и пр. В результате, обилие ярких, цветных мигающих объектов отвлекает внимание пользователя от навигации, которая, в свою очередь, часто оказывается на удивление незаметной и невзрачной.

Исходя из этого, можно сформулировать следующее правило: *система навигации должна быть легко идентифицируема посетителем сайта.*

Иерархия и наименования

От того, как построена иерархия системы навигации, может зависеть дальнейшее поведение посетителя. Если она покажется непонятной, недостаточно информативной или запутанной, скорее всего он будет склоняться к поиску другого сайта с аналогичным содержанием.

Как же следует строить навигационную иерархию? Прежде всего необходимо выделить те разделы сайта, которые содержат информацию, представляющую наибольший интерес для потенциального посетителя. К примеру, на сайте, посвященном услугам мобильной связи, могут быть следующие разделы: "Отправление SMS", "Форум", "Мелодии" и "Каталог телефонов". Учитывая то, что такой сайт относится к категории контент-проектов, можно утверждать, что наибольший интерес для пользователя будет представлять раздел, где представлены модели телефонов. Функцией отправления коротких текстовых сообщений, как правило, пользуются на сайтах самих сотовых операторов, либо на специализированных серверах с SMS-шлюзом. Поиск мелодии можно отнести к разряду развлечений, в отличие от поиска модели телефона или отправки SMS. Форум будет интересен тем, кто хочет задать вопросы, поделиться опытом или просто пообщаться.

Таким образом, учитывая прогнозируемые пользовательские предпочтения, систему навигации можно построить в следующем порядке:

- каталог телефонов;
- отправление SMS;
- мелодии;
- форум,

где пункт навигационного меню "форум" следует выделить относительно трех предыдущих, т. к. его содержание может напрямую не относиться к тематике сайта.

Если просмотр Web-сайта подразумевает переход от одного подраздела к другому, будет логично представить навигацию именно в той последовательности, как это делает посетитель сайта. Например, сайт, содержащий информацию о программном обеспечении и предлагающий свободные для загрузки дистрибутивы, может построить свою навигацию таким образом:

MICROSOFT NOTEPAD 2002 PROFESSIONAL EDITION

- о программе;
- системные требования;
- условия распространения;
- загрузка.

Вполне разумно предположить, что пользователь сначала пожелает получить общие сведения о программе, затем захочет узнать системные требования, а после этого скачать программу на свой компьютер, предварительно ознакомившись с лицензионным соглашением. И хотя посетитель может начать знакомство с программой с конца навигационного меню, логика переходов по подразделу будет сохранена.

Другой, не менее важный атрибут любой системы навигации — наименование пунктов меню. От того, правильно ли выбрано название того или иного раздела сайта, зависит, поймет посетитель, какая информация кроется за конкретной гиперссылкой, или станет угадывать назначение раздела. Плохо подобранное наименование может сбить посетителя с толку, дезинформировать, наконец, вынудить его перейти в раздел, информация которого не представляет для него никакой ценности. Название должно быть кратким (1—3 слова), исчерпывающим, понятным, легким для восприятия и характеризующим информационное наполнение данного раздела сайта.

Третье заблуждение разработчика

Некоторые разработчики уверены в том, что использование одного слова для характеристики раздела — оптимальный выбор, позволяющий к тому же сохра-

нять единообразие системы навигации (т. е. стараются избегать ситуации, когда один раздел озаглавлен двумя словами, другой — тремя, третий — одним и т. д.). В результате такие "адаптированные" наименования непонятны посетителю, а зачастую просто не отражают суть раздела сайта.

Если Web-сайт посвящен специализированной тематике (например, "Гидрофобизированные теплоизоляционные материалы" или "Система управления дебиторской задолженностью"), следует, по возможности, избегать сложных терминов и формулировок в названии разделов сайта (или по крайней мере минимизировать их количество), помня о том, что аудиторию сайта, помимо специалистов в данной области, могут составлять люди непосвященные, только начинающие работу в этой отрасли, для которых восприятие подобных наименований будет затруднено.

Таким образом, формулируем третье правило: *пункты системы навигации должны быть достаточно информативны и понятны любому посетителю*.

Нелишним будет добавить, что совместное использование понятных и простых наименований и подсказок с помощью HTML-тегов `<ALT>` и `<TITLE>` станет залогом составления эффективного и информативного названия пунктов системы навигации.

Рис. 13.5. Непонятные наименования пунктов навигационного меню на Web-сайте NefteGaz.Ru

Примером сайта, содержащего систему навигации с непонятными наименованиями, является поисковая система по нефтегазовому комплексу NefteGaz.Ru (рис.13.5).

Заголовки всех разделов Web-сайта состоят из одного слова, кроме того, разработчики даже не удосужились сделать подсказку в виде альтернативного текста `ALT`. В итоге, суть таких разделов, как "Магазин", "Каталог" или "Кадры" может быть осмыслена по-разному: а) что за магазин, что там продается и кто является продавцом? б) каталог чего? Компаний, занятых в нефтегазовом комплексе или интернет-ресурсов на эту тему? в) в разделе представлен банк вакансий по всем специальностям или только по нефтегазовой отрасли? Или там размещена информация о вакансиях сервера NefteGaz.Ru?

Доступность

Последний критерий, в соответствии с которым следует разрабатывать систему навигации для любого Web-сайта, — это доступность навигационного меню для пользователя.

Этому критерию полностью соответствует текстовая навигация, т. к. обыкновенный текст отображается всеми существующими браузерами независимо от версии и наличия программных надстроек (plug-ins). В случае отсутствия в браузере поддержки стилевых таблиц CSS или отключения этой опции самим пользователем, текстовые ссылки будут отображаться в соответствии с настройками, установленными по умолчанию, однако меню будет доступным для использования.

Что касается графики, то она также может быть отключена в браузере пользователя или по каким-то причинам изображения, служащие пунктами навигационного меню, не будут загружены (ошибка сервера, прерывание связи и пр.). Однако, если каждому графическому пункту системы навигации будет присвоена информативная и понятная подсказка (альтернативный текст `ALT`), отсутствие изображения не станет помехой для перехода в нужный раздел сайта.

Реализация навигации с помощью HTML-форм также практически лишена каких-либо ограничений. Единственное, о чем следует помнить разработчику, — что некоторые браузеры (за исключением Internet Explorer) могут некорректно обрабатывать HTML-код ниспадающих меню, не заключенный между парным тегом `<FORM></FORM>`. А управление навигацией через HTML-форму посредством различных сценариев должно исключать наличие программных ошибок (если навигацией управляет JavaScript-сценарий, существует вероятность того, что такая опция будет отключена в браузере пользователя).

Если навигация разработана в виде Java-апплета, а опция поддержки Java в браузере посетителя будет отключена, последний не сможет ни увидеть такую навигацию, ни тем более ею воспользоваться.

Наконец, тем разработчикам, кто планирует реализовать систему навигации с помощью технологии Flash, стоит задуматься: как поступит посетитель, если у него не установлен проигрыватель Flash-роликов?

Например, разработчики сайта рекламного агентства "Оскар" (рис. 13.6) решили не брать в расчет посетителей, у которых на компьютере нет проигрывателя Flash-анимации.

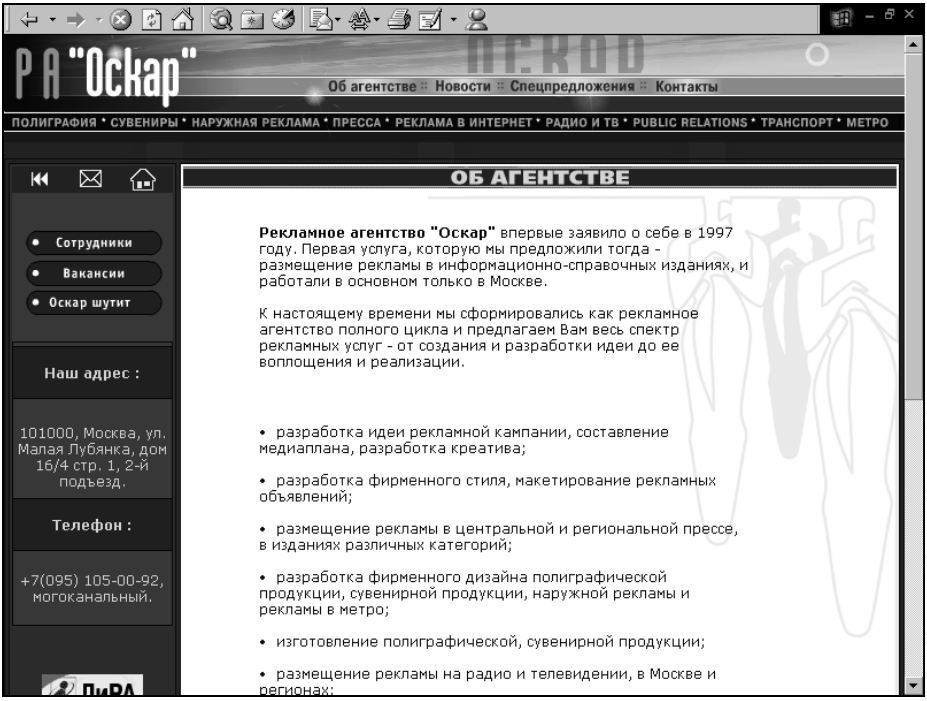


Рис. 13.6. Меню навигации на Web-сайте рекламного агентства "Оскар", реализованное с помощью технологии Macromedia Flash

Если такой пользователь зайдет на Web-сайт агентства, то воспользоваться системой навигации он не сможет. Не сможет он воспользоваться и дублирующей навигацией, расположенной внизу страниц сайта, т. к. она также выполнена по технологии Macromedia Flash.

Четвертое заблуждение разработчика

Некоторые разработчики убеждены, что их потенциальный пользователь использует Microsoft Internet Explorer 6, в котором постоянно включена поддержка

графики и Java и установлен Macromedia Flash Player последней версии. Между тем по-прежнему большинство посетителей Интернета пользуются коммутируемым доступом, имеют медленные или средние по скорости линии, что зачастую лишает их возможности просматривать Flash-ролики и Java-апплеты, и отключают поддержку графики. К тому же часть из них пользуется браузерами других производителей.

В завершение добавим, что перечисленные выше минусы, присущие системам навигации, ни в коем случае не должны ограничивать разработчика Web-сайтов. При этом никогда не забывайте про поговорку: "Береженого Бог бережет" — и не избегайте дублирования систем навигации более простыми, доступными версиями, что позволит вам охватить максимально широкую аудиторию и оставит у посетителя после визита на ваш Web-сайт только приятные впечатления.

Карта сайта

Любая человеческая деятельность подразумевает приверженность определенным стереотипам поведения в конкретных ситуациях. Процесс планирования и разработки Web-сайта не является исключением из правил и сопряжен с использованием ряда функциональных возможностей, в некотором роде ставших стандартными практически для любого Web-сайта. *Карта сайта* — один из наиболее красноречивых примеров.

Вопросы целесообразности

Прежде всего, дадим формулировку понятию "карта сайта".

Карта сайта — это функциональный компонент Web-сайта, предназначенный для четкой логической структуризации содержания. Другими словами, с помощью карты сайта можно значительно облегчить труд посетителя по исследованию составных документов интернет-проекта. Возможно, вы спросите: "А как же навигационная система? Ведь именно она создается в качестве некоего путеводителя по разделам сайта?" Безусловно, роль системы навигации сложно переоценить: не будь на сайте интуитивно понятной и удобной навигации, пользователь вовсе не стал бы находиться на таком ресурсе и, более того, сформировал бы в своем сознании стойкий негативный имидж проекта.

Отсюда следует первое важное заключение, многократно подтвержденное практикой: *"Наличие карты сайта — следствие многоуровневой системы навигации"*. Поэтому в данном случае целесообразность разработки карты сайта зависит именно от того, насколько сложной структурой обладает система навигации.

Многоуровневая навигация

Мы нередко встречаем Web-сайты, использующие двух-, трех- и более уровневую вложенность навигационного меню. В большинстве случаев такой подход продиктован большим объемом информации, который необходимо донести до потенциального клиента или пользователя — причем, в простой и понятной форме.

Один из вариантов — разбить конкретный информационный раздел на несколько частей, присвоив им соответствующие наименования и смысловую направленность. Это второй уровень навигации. Теперь попробуем усложнить ситуацию. Допустим, у нас есть раздел "Контакты", который включает в себя еще три подраздела: "Телефоны", "Карта проезда" и "Обратная связь". Последний, в свою очередь, содержит ссылки на почтовую форму, форум и ICQ-аккаунт менеджера-консультанта. Таким образом, получается третий уровень навигации. В ряде случаев вложенность может быть гораздо больше (например, в описании товаров и/или услуг: "Товары > Программное обеспечение > Freeware > Интернет-приложения > Почта" и т. п.), что приводит к появлению дополнительных пунктов навигационного меню.

Все это уводит посетителя с начальной точки путешествия по сайту: уже через пару минут он может попросту "заблудиться" в закоулках интернет-ресурса. Во избежание подобной ситуации, на сайте в обязательном порядке должны присутствовать следующие вспомогательные функциональные возможности:

- навигационная "крошка" — присутствие в верхней части окна браузера (в видимой посетителю части экрана) гипертекстовой навигационной строки, например:

  » **Почта** » Интернет-приложения » Freeware » Программное обеспечение » Товары

  Данный пример показывает, что посетитель находится в подразделе "Почта", но в любой момент может воспользоваться соответствующей ссылкой (на примере выделены подчеркиванием) и перейти на нужный ему уровень навигации родительского раздела (которым в нашем случае будет являться "Товары");

- дублирование верхних уровней меню и остальных разделов первого уровня — совмещение навигационной "крошки" (возможности быстро перемещаться между уровнями навигации родительского раздела) и остальных пунктов навигационного меню первого уровня, например:

  » **Почта** » Интернет-приложения » Freeware » Программное обеспечение » Товары

  ДРУГИЕ РАЗДЕЛЫ: Услуги :: Прайс-лист :: Заказ online :: Помощь :: Контакты

Запутанная навигация

Второе (не менее важное) заключение можно сделать с помощью следующего примера. Существует некий абстрактный сайт, который содержит три раздела: "Товары", "Услуги" и "Заказ online". При определенных обстоятельствах вспомогательные подразделы, входящие в состав названных разделов первого уровня, могут пересекаться и, более того, дублировать друг друга:

Таблица 13.1. *Пример запутанной навигации*

Раздел "Товары"	Раздел "Услуги"	Раздел "Заказ online"
Общая информация	Общая информация	Каталог товаров и услуг
Каталог товаров	Каталог услуг	Корзина
Прайс-лист	Прайс-лист	Оформление заказа
Бланк заказа	Бланк заказа	

При таком построении системы навигации посетитель может запутаться в предназначении тех или иных пунктов меню, схожих по смыслу, но логически относящихся к разным разделам первого уровня. Подобная запутанная структура меню, конечно, не делает чести разработчикам, однако надо признать, что в ряде случаев такое пересечение может быть вызвано различными соображениями или требованиями, обсуждение которых находится за пределами компетенции исполнителя интернет-проекта (например, пожелание заказчика, требование инвестора и пр.).

Отсюда можно сделать второе важное заключение относительно целесообразности присутствия карты сайта: *"Наличие карты сайта — следствие запутанной системы навигации"*.

Обобщая сказанное выше, можно сделать следующий вывод: присутствие карты сайта может быть обусловлено двумя характерными особенностями структуры навигационного меню: многоуровневым строением и дублирующимися (пересекающимися) пунктами меню смежных разделов. В иных ситуациях карта сайта абсолютно не нужна и только отвлекает внимание посетителя, который ищет в разделе "Карта сайта" дополнительную функциональность.

Приведем простой практический пример, иллюстрирующий данное утверждение. Навигационное меню официального Web-сайта петербургской строительной компании "Экострой" содержит следующие разделы (в скобках указаны разделы второго уровня):

- "Услуги" ("Поставка комплектующих", "Экспертиза проекта", "Разработка проекта", "Строительно-монтажные работы", "Шеф-монтаж", "Оформить заявку");

- "Фасады" ("Общая информация", "Вентилируемые фасады", "Мокрые фасады");
- "Мансарды";
- "Строительный магазин";
- "Полезная информация" ("Теплофизика", "Разное", "Интернет-ссылки");
- "Вопросы и ответы".

Как видно, структура меню достаточно простая, пересекающихся пунктов меню нет, уровень вложенности минимальный. Помимо этого, навигация построена по технологии Dynamic HTML и представляет собой открывающиеся перечни ссылок при наведении курсора на раздел первого уровня. Также предусмотрено постоянное присутствие навигационной "крошки", что является большим плюсом. Все это позволяет посетителю быстро перемещаться по всем пунктам меню из любой точки сайта.

Но несмотря на высокую функциональность и гибкость навигационного меню, на сайте есть раздел "Карта сайта", содержание которого на 95% повторяет совокупность пунктов меню всех вместе взятых разделов, доступных пользователю без перезагрузки страниц проекта (оставшиеся 5% частично компенсируют недостаточность карты сайта некими дополнениями отдельно взятых разделов). В целом, реализация карты сайта выполнена на чрезвычайно низком уровне и не может удовлетворять потребности посетителей в функциональности (согласно статистике сервера, "Карта сайта" входит в тройку наиболее посещаемых разделов).

ГЛАВА 14

Шаблоны ошибок

Наверняка многие из вас сталкивались с ситуацией, когда браузер отказывается загружать определенную страницу Интернета, выводя вместо нее сообщение об ошибке, которой, как правило, присвоен определенный номер. Самой распространенной на сегодня ошибкой можно по праву считать Ошибку 404 — Error 404.

Ошибка 404

Реакцию пользователей на сообщение об Ошибке 404 можно охарактеризовать по-разному: к ней относятся отрицательно или равнодушно, ее боятся или попросту игнорируют. Между тем, далеко не каждый знает, что в честь этой ошибки создаются целые интернет-проекты и каталоги ресурсов, проводятся конкурсы и состязания, наконец, ей даже признаются в любви!

Техническая трактовка

Прежде всего, давайте разберемся, что такое Ошибка 404 и в каких случаях она возникает.

С технической точки зрения, Ошибка 404 (или Error 404) — это кодовый заголовок статуса протокола HTTP (HyperText Transfer Protocol), который, как известно, является чем-то вроде основы взаимодействия удаленных компьютеров всемирной сети. При обращении браузера к Web-серверу, последний посылает кодовый статус запрашиваемого документа. При корректной обработке этого запроса статус идентичен коду 200 OK. Однако пользователь не может видеть подобного сообщения по причине отображения содержания запрошенного Web-сайта в своем браузере. Если запрос документа браузером обработан с ошибками, то пользователь наблюдает сообщение, имеющее статус "404: Not Found".

Что означают эти загадочные цифры — 404? Первая цифра 4 указывает на ошибку программы-клиента, т. е. браузера. В этом случае предполагается, что URL сайта был набран неверно в адресной строке браузера или больше не существует физически на запрошенном сервере. Цифра 0 означает общую ошибку синтаксиса протокола. Наконец, последняя четверка относится к отдельной категории ошибок 40x, которая включает такие распространенные статусы, как "400: Bad Request" и "401: Unauthorized".

Миф Ошибки 404

Некоторые зарубежные источники несколько по-иному трактуют расшифровку сочетания цифр 404. В частности, утверждается, что основная база данных научного центра CERN (European Organization for Nuclear Research), принявшего самое активное участие в развитии World Wide Web, находилась на 4-м этаже в комнате № 404. И когда объем обрабатываемых данных превысил определенные нормы, уже не позволяющие нескольким ученым одновременно обращаться к одному и тому же файлу, разработчики якобы решили выдавать сообщение об ошибке вида "Room 404: file not found".

Однако данное заявление опровергается одним из ученых, входящих в группу разработчиков центра CERN. Согласно его утверждению, "Комнаты № 404 вообще не существовало в лаборатории CERN по причине совершенно иной системы нумерации офисов, по которой первая цифра (прим. автора: "4") означала порядковый номер корпуса здания, а комбинация вторых — номер комнаты. При этом было бы странно говорить о комнате № 04, учитывая тот факт, что нумерация в CERN начиналась с цифры 410".

Причины и последствия

Как уже было сказано выше, согласно спецификации протокола HTTP, Ошибка 404 может быть вызвана либо неверным указанием адреса (URL) запрашиваемого сайта, либо отсутствием документа на Web-сервере. Распространение в России соединения с Интернетом при помощи телефонной линии (dial-up) добавляет еще одну вероятную причину возникновения Ошибки 404 — плохую связь с модемным пулом провайдера.

Как же склонен реагировать пользователь на появление сообщения об Ошибке 404? В большинстве случаев любые попытки загрузить необходимый документ в дальнейшем отклоняются в принципе: люди привыкли ценить время своего пребывания в Интернете и предпочитают поискать аналогичный ресурс, удовлетворяющий их требованиям. При этом далеко не каждый знает о третьей причине вероятной Ошибки 404 — плохой связи, поэтому не станет себя утруждать повторными попытками загрузить желанную страницу.

Памятка пользователю Интернета при возникновении Ошибки 404

При возникновении Ошибки 404 выполняйте следующие действия:

- шаг 1 — попробуйте перезагрузить страницу. Возможно, это просто случайность;
- шаг 2 — поищите ошибки в написании URL или попробуйте модифицировать расширение загружаемого документа (например, поменять `htm` на `html` и наоборот);
- шаг 3 — поднимитесь в структуре URL на один уровень выше и оттуда попытайтесь найти искомый документ, например:

  ```
  http://www.site.ru/docs/users/masha.html
  >>
  http://www.site.ru/docs/users/
  ```

- шаг 4 — попробуйте поискать нужную страницу в поисковой системе: в отличие от забывчивых Web-мастеров, спайдеры более ответственно относятся к содержанию своих индексов;
- шаг 5 — свяжитесь с Web-мастером интернет-ресурса, на котором нет искомой страницы, по электронной почте. Наверняка он отблагодарит вас за то, что вы нашли "битую" ссылку.

Ошибка 404 может возникнуть по вине пользователя Сети, который невнимательно записал со слов друга адрес интересного сайта или набрал URL в адресной строке браузера с грамматическими ошибками. Также "дать маху" способен Web-мастер, который при реорганизации структуры сайта или "чистке" содержимого сервера мог за ненадобностью (как ему казалось) удалить тот или иной документ. Если последний проиндексирован поисковыми системами, то вполне возможно, что несуществующий документ будет выведен в результирующем списке по поисковому запросу пользователя. Или же конкретный интернет-ресурс некогда был добавлен в "закладки" браузера, чтобы обратиться к нему по необходимости в будущем. Таких случаев достаточно много, и все они ведут к возникновению Ошибки 404.

Раздолье на ниве *HTML*

Стандартное сообщение об Ошибке 404 носит весьма аскетичный и строгий характер (рис. 14.1). Но самое главное — оно в большинстве случаев не только огорчает пользователя, но и ставит его в тупик: тот не знает, что ему предпринять и, как правило, покидает сервер, который выдает непонятную ему ошибку. Это большой промах Web-мастера, который таким образом теряет своих потенциальных покупателей, клиентов, заказчиков и т. п.

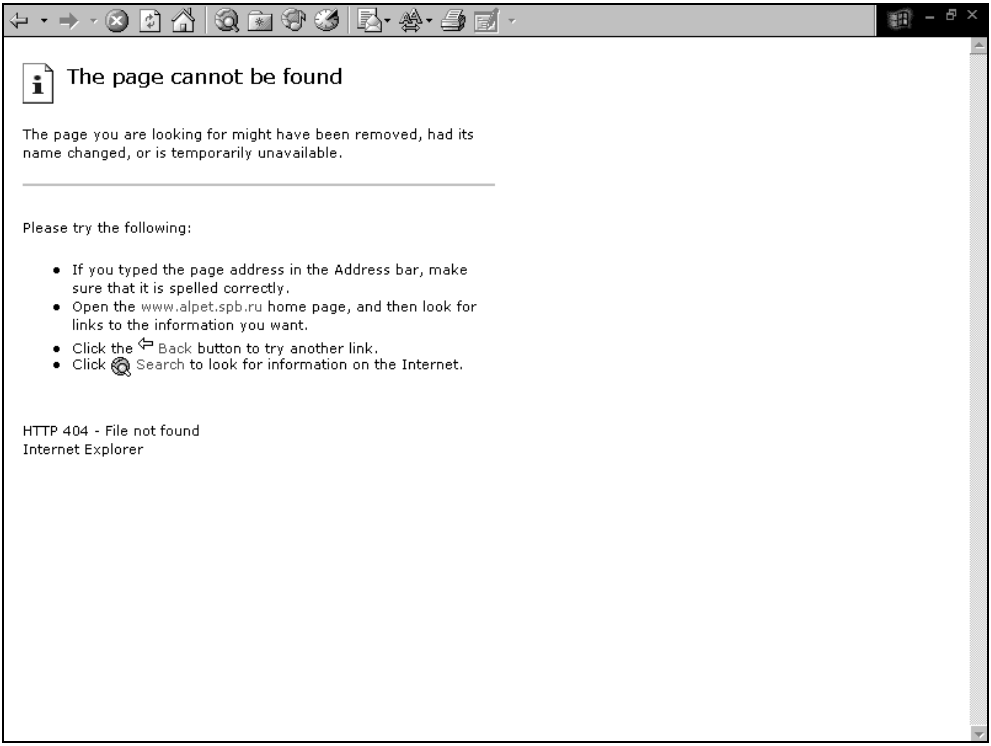


Рис. 14.1. Стандартный вид сообщения "404: Not Found" в браузере Microsoft Internet Explorer

Однако при доступе к некоторым настройкам Web-сервера любой Web-мастер (или любой другой человек, знающий язык гипертекстовой разметки HTML) может оформить внешний вид стандартного сообщения "404: Not Found" по своему вкусу и желанию.

Трудно себе представить просторы для фантазии и красноречия, которые открываются в этом случае перед разработчиками того или иного интернет-ресурса. Теперь можно направить заплутавшегося посетителя в нужном направлении, подсказать ему искомый адрес, наконец — просто посочувствовать и перенаправить на заглавную страницу своего сайта. Другими словами, диалог между поставщиком услуг (которым в какой-то мере является практически любой Web-ресурс) и пользователем не нарушается даже при возникновении ошибки.

Рекомендации по созданию страницы Ошибки 404

При оформлении шаблона страницы для Ошибки 404 следует придерживаться следующих рекомендаций:

- содержание:
  - используйте как можно меньше сложных терминологических выражений и специфических, непонятных простому пользователю формулировок. Излагайте суть ситуации просто и понятно;
  - успокаивайте пользователя и укрепляйте его веру в то, что искомый документ обязательно найдется на вашем сервере;
  - приводите возможные причины возникновения ошибки (технические рассуждения смело заменяйте на предположение о неверном написании URL и т. п.);
- навигация:
  - всегда указывайте ссылки, ведущие на какой-либо из доступных разделов сайта (первая страница, карта сайта и пр.);
  - включайте основную навигацию в структуру сообщения об Ошибке 404: так пользователю будет проще сориентироваться на вашем сервере;
  - указывайте свои координаты для связи;
  - по возможности, разместите на странице с сообщением об ошибке форму поиска по сайту;
- оформление:
  - избегайте тяжеловесных, перегруженных графикой, заполненных Flash-анимацией и Java-апплетами страниц с сообщением о "404: Not Found". Эта страница должна грузиться в считанные секунды;
  - постарайтесь, чтобы оформление страницы Ошибки 404 не сильно отличалось от общего дизайна вашего сайта.

Ошибка 404 как хобби

Одни собирают почтовые марки, вторые увлекаются коллекционированием пуговиц, третьи находятся в вечной погоне за раритетными монетами. А есть люди, которые коллекционируют самые разнообразные варианты Ошибок 404! И тому есть вполне разумное объяснение: бурное развитие Интернета вообще и изобразительного искусства, в частности, повлекло за собой возникновение различных образцов визуальной интерпретации статуса протокола 404: смешных и грустных, для детей и взрослых, поучительных и агрессивных и пр.

Например, каталог ссылок одного из самых авторитетных ресурсов, посвященных Ошибке 404, — "404 Research Lab" (**http://www.plinko.net/404**), на-

считывает свыше 20-ти различных тематических категорий, содержащих множество вариантов отображения Ошибки 404: "Забавные", "Дружелюбные", "Для взрослых", "Полезные", "Философские", "Шокирующие", "Неожиданные" и др. В другом каталоге — 404Lounge.Net (**http://www.404lounge.net**) — расположена одна из крупнейших галерей Ошибок 404, включающая более 700 разнообразных вариаций вывода сообщения "404: Not Found".

На рис. 14.2—14.6 представлены некоторые варианты реализации страницы Ошибки 404.

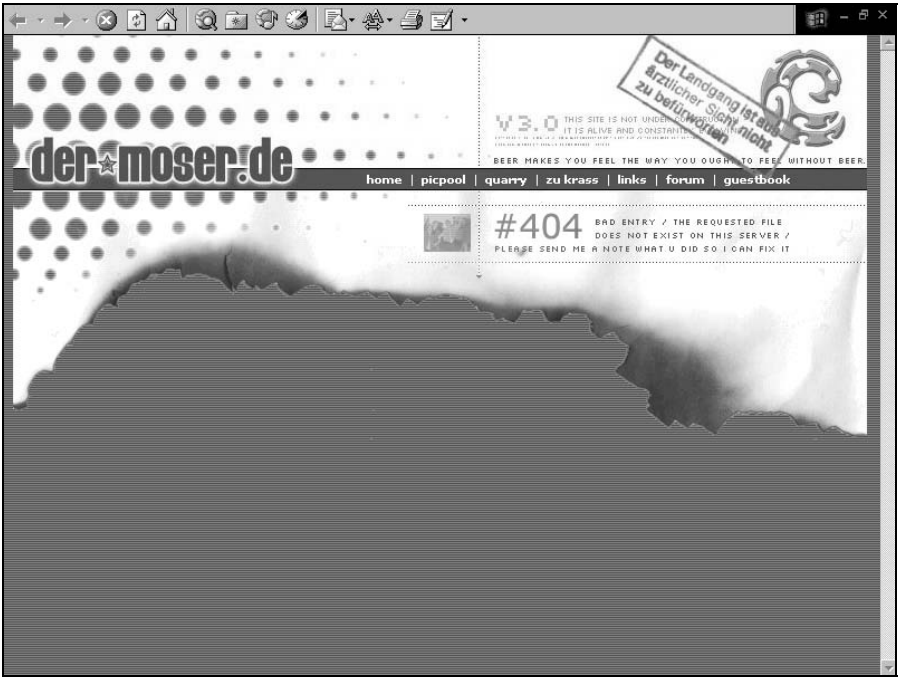


Рис. 14.2. Проявление творческого мышления дизайнера на немецком сайте DerMoser.De (**http://www.der-moser.de**). В сообщении сказано: "Неверный адрес. Запрошенный файл не существует на сервере. Пожалуйста, сообщите мне о том, как вы получили подобное сообщение

Однако открытие каталогов Ошибки 404 и тематических рубрикаторов, новостные ленты событий в мире "404: Not Found" и креативные изыски Web-мастеров многочисленных серверов по оформлению сообщения об ошибке — это все еще как-то можно понять и логически объяснить. А как вы посмотрите на сочинительство хвалебных од и любовных посвящений в честь Ошибки 404? Удивлены? Не верите? Совершенно напрасно, в доказательство чего ниже приводится фрагмент прозаического шедевра под названием

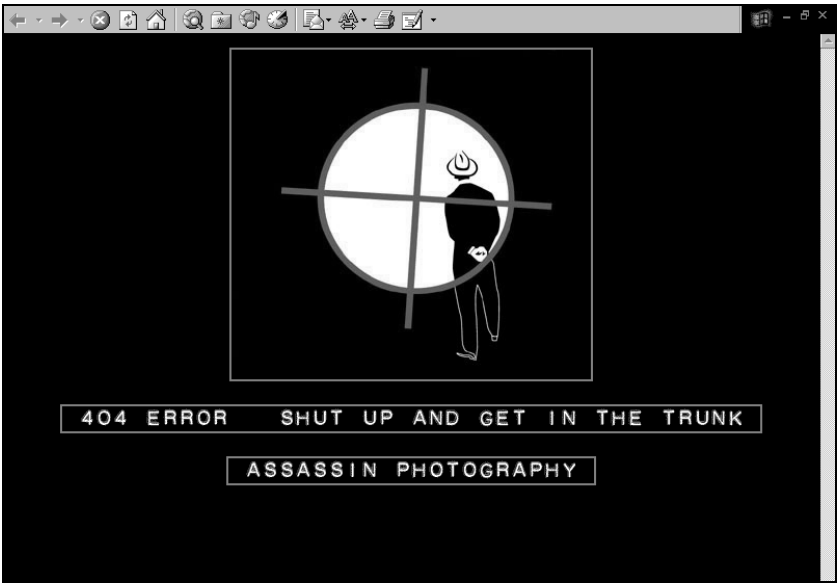


Рис. 14.3. Шокирующий вызов пользователю на сайте Discopix.Com (**http://www.discopix.com**). Надпись на ссылке на главную страницу гласит (перевод адаптирован согласно требованиям цензуры): "Ошибка 404. Замолчите и встаньте на путь истинный"

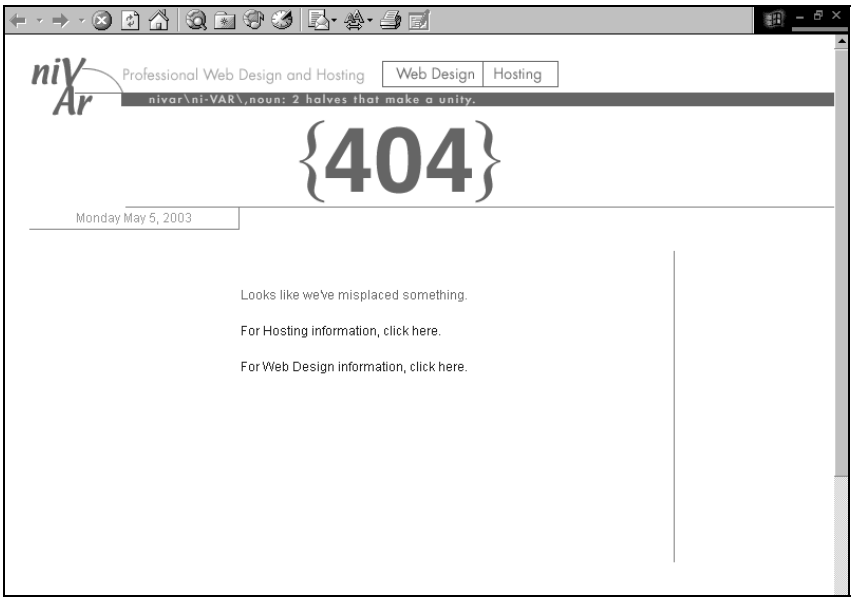


Рис. 14.4. Классический пример "Ошибки 404" на сайте компании "Nivar Professional" (**http://www.nivar.com**): "Похоже, вы не туда попали. Для получения информации о хостинге нажмите здесь. Для информации об услугах Web-дизайна нажмите здесь". Пользователю дается четкая информация о том, что произошла какая-то ошибка, и тут же приводятся ссылки на основные разделы сайта

Рис. 14.5. Использование политических образов в трактовке "Ошибки 404" — далеко не единичный случай. Надпись рядом с Джорджем Бушем, стоящим с битой в позе заядлого бейсболиста, гласит: "Вы промахнулись! Не знаем, что вы искали на stickybuffalo.com, но вы попали мимо. Проверьте правильность URL и попробуйте ударить снова" (Stickybuffalo.Com, **http://www.stickybuffalo.com**)

"404: Чистилище: Любовное посвящение Ошибке 404" ("404 Purgatory: A Loving Tribute to 404 Not Found", перевод автора).

> "Свершилось! Вы познали тайное могущество 404. Вы блуждаете по закоулкам Интернета всю ночь напролет, вы купили себе новый модем и модную эргономичную мышь. Они вам пригодятся. Вы с ловкостью жонглера оперируете семью окнами браузера одновременно и вдруг — 404. Вы не верите своим глазам, ведь этот сайт открывался на прошлой неделе. Вы нажимаете Refresh, с замиранием сердца ожидая чуда. О, нет! Как могли вас предать эти бесчестные закладки браузера?.."

Далее автор начинает пространные рассуждения о причинах столь скоропалительного ухода пользователей с сайтов с Ошибкой 404:

> "Почему мы так быстро оставляем 404? Мы не понимаем, что 404 — это оазис Интернета. Она как передышка в долгом и изнурительном пути Web-серфера, как глоток живительной влаги посреди иссохшей пустыни... Ошибка 404 полна загадки и интриги. Чего мы ждем от браузера? Какую радость нам предстоит познать? Сможем ли мы вновь и вновь вернуться сюда? 404 молча хранит эту тайну, и это заставляет людей снова возвращаться сюда, дабы с каждым разом познать что-то новое и необычное..."

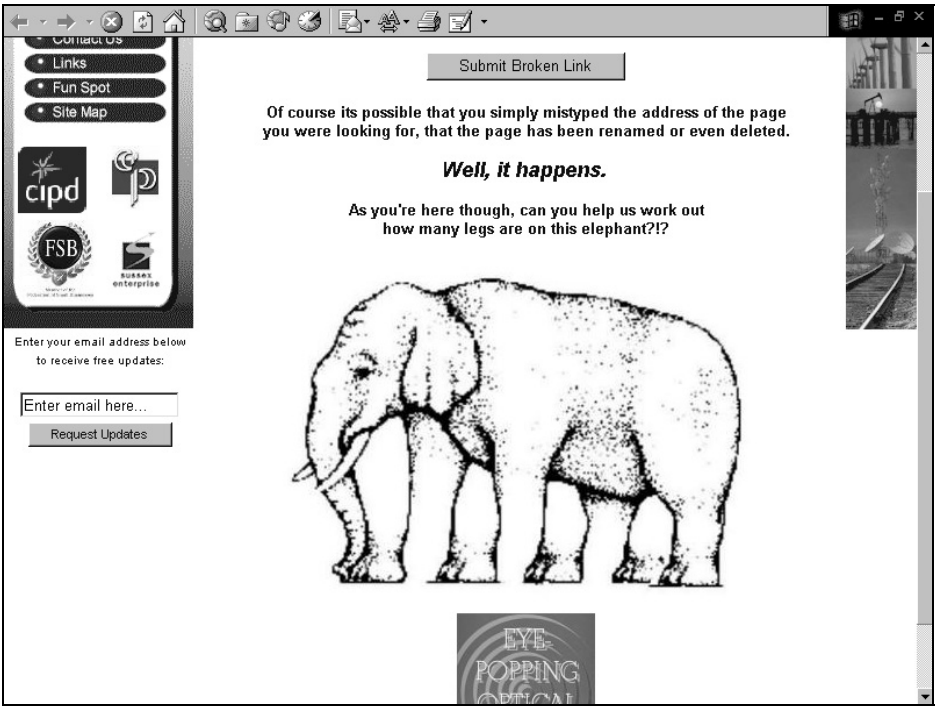


Рис. 14.6. Оригинальный вариант Ошибки 404 представлен на сервере английской компании "Business Improvement Solutions" — текст сообщает посетителю: "Скорее всего, вы просто неверно написали адрес страницы, которую ищете, или же документ был переименован или удален. Но это уже позади, не так ли? А раз вы очутились здесь, не поможете ли нам определить, сколько ног у слона?". Ниже размещен рисунок слона, количество конечностей которого с трудом поддается точному определению. Головоломка, шарада, загадка — назовите, как хотите, но ошибка уже не воспринимается с таким огорчением

И завершает автор свой проникновенный опус следующим заявлением:

"Там, где была тьма, теперь Ошибка 404. И с миром все в порядке"

Безусловно, не стоит относиться к такому толкованию сущности Ошибки 404 всерьез: это всего лишь шутка, добрая ирония по отношению к обычному жизненному явлению. И в следующий раз, когда на экране своего монитора вы увидите "404: Not Found", не расстраивайтесь и не злословьте в адрес нерасторопного Web-мастера: улыбнитесь, вспомнив строки этого "любовного посвящения", и подумайте, как нужно оформить страницу ошибки для вашего Web-сайта (полный текст на английском языке доступен по адресу **http://www.plinko.net/404/tribute.asp**).

ГЛАВА 15

Разработка графического макета будущего Web-сайта

Компоновка макета

Под *компоновкой макета* принято понимать условное (схематическое) расположение блоков информации на будущем Web-сайте. Всего можно выделить четыре блока информации (рис. 15.1). Рассмотрим каждый из них:

- "шапка" — занимает верхнюю часть сайта в окне браузера (как правило, не больше 1/4 видимой части экрана без прокрутки) и может содержать:
  - логотип;
  - слоганы, лозунги, девизы;
  - рекламную информацию (баннеры и текстовые блоки) и др.;
- меню навигации — оно может быть вертикальным, горизонтальным, содержать подменю и т. д. и т. п. (более подробно о навигационных меню было рассказано в *главе 13*;
- содержательная часть — здесь отображается основная и самая главная составляющая любого Web-сайта — информация: новости и каталог продукции, контактные данные и гостевая книга и т. д.;
- "подвал" — нижняя часть сайта в окне браузера, не является обязательным блоком, он лишь подчеркивает четкую структуру сайта и выделяет содержательную часть относительно других блоков информации. Он может содержать:
  - информацию об авторском праве разработчиков Web-сайта (полный текст или ссылка на него);
  - координаты для связи;

- дублирующую навигацию (текстовую);
- рекламную информацию (баннеры и текстовые блоки) и др.

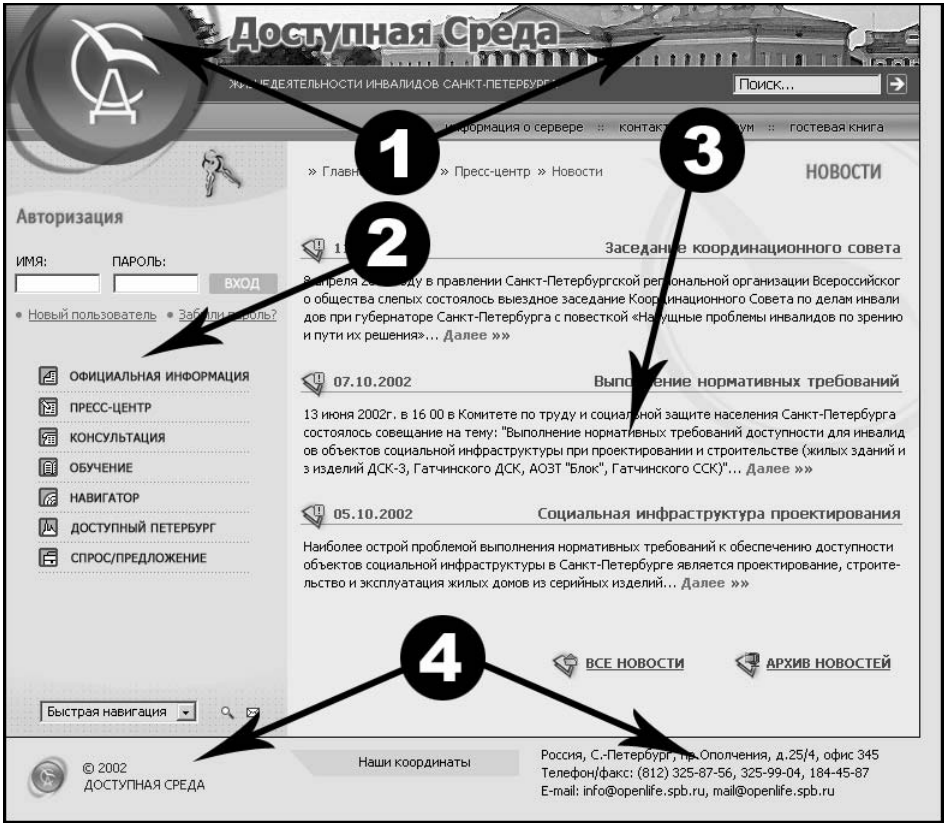


Рис. 15.1. Компоновка блоков на примере одного из Web-сайтов:
1 — "шапка"; 2 — меню навигации; 3 — содержательная часть; 4 — "подвал"

Тип верстки

Еще одним важным моментом, предшествующим началу работы над графическим макетом, является выбор типа верстки. Всего существует два типа:

- "фикс" — в этом случае сайт "затачивается" под конкретное минимальное разрешение экрана и при изменении размеров окна браузера его компоновка остается фиксированной;
- "резина" — структура сайта изменяется пропорционально изменениям размеров окна браузера (т. е. сайт "растягивается").

Логично предположить, что при выборе типа верстки следует исходить из минимального разрешения экрана (табл. 15.1, 15.2).

Таблица 15.1. *Статистика экранного разрешения по данным системы интернет-статистики HotLog (январь 2004 года, российский Интернет)*

Разрешение экрана	Процент использования
1024×768	58,22
800×600	27,08
1280×1024	7,04
1152×864	4,07

Таблица 15.2. *Статистика экранного разрешения по данным исследовательской группы OneStat (январь 2004 года, мировой Интернет)*

Разрешение экрана	Процент использования
1024×768	48,3
800×600	31,7
1280×1024	13,6
1152×864	4

Работа над макетом

Наконец мы подошли к моменту, когда нужно открывать Adobe Photoshop (или любой другой удобный вам графический пакет) и творить.

Составим план создания нашего графического макета.

1. Компоновка блоков: вверху — "шапка", слева посередине — меню навигаци, справа посередине — содержательная часть, внизу — "подвал".
2. Тип верстки: "резина", минимальное разрешение — 800×600.
3. Цветовая гамма: белый, черный, серый, оранжевый, оттенки синего, зеленый.
4. Шрифты: Tahoma.

Итак, задание понятно, приступаем к работе.

Первым делом создаем заготовку размером 780×600 (при разработке макета из ширины в 800 вычитается размер полосы прокрутки в Microsoft Internet Explorer, как правило, 20 пикселов). Основной слой заливаем белым цветом и делаем неподвижный фон: **Layer > New > Layer From Background**.

Далее выбираем инструмент Rectangular Marquee Tool (прямоугольное выделение) и создаем горизонтальную сплошную полосу высотой 6 пикселов и заливкой #2187D6. Копируем полосу, изменяем ее высоту до 2 пикселов и перемещаем под первую полосу, но с зазором между ними в 1 пиксел (рис. 15.2).

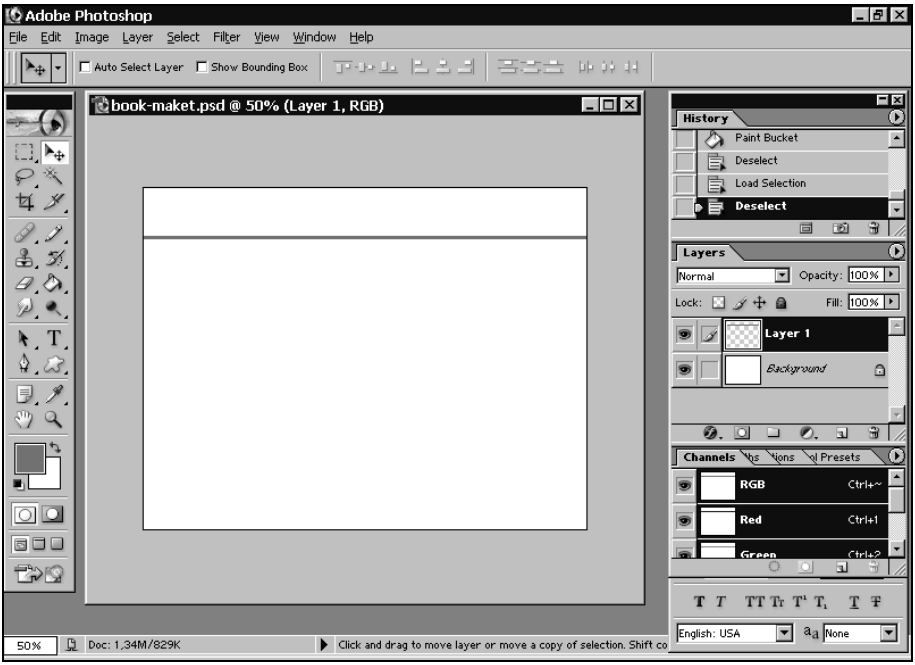


Рис. 15.2. Добавляем полосы

Потом создаем объект прямоугольной формы с верхним левым закруглением (можно использовать автофигуры), заливаем тем же цветом, что и полосы, и придвигаем к предыдущим двум полосам (с выравниванием вправо). Вторую полосу в 2 пиксела копируем, заливаем цветом #15619C и размещаем над первыми двумя полосами, но за прямоугольным объектом с закруглением (рис. 15.3).

Теперь придумываем незатейливый логотип с использованием сочного оранжевого и зеленого цветов и добавляем к нему название нашего сайта черным цветом (рис. 15.4).

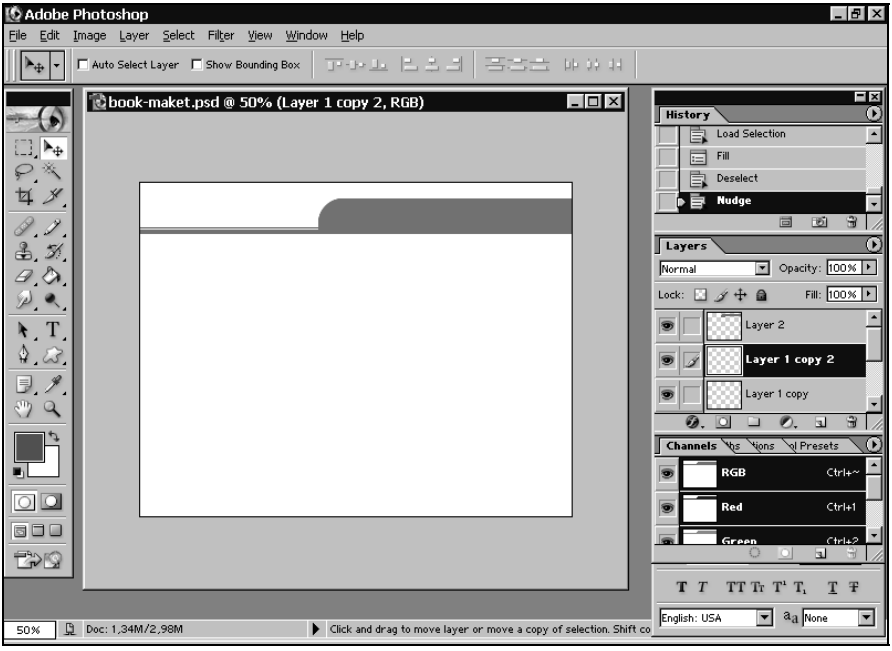


Рис. 15.3. Добавляем прямоугольный объект и третью полосу

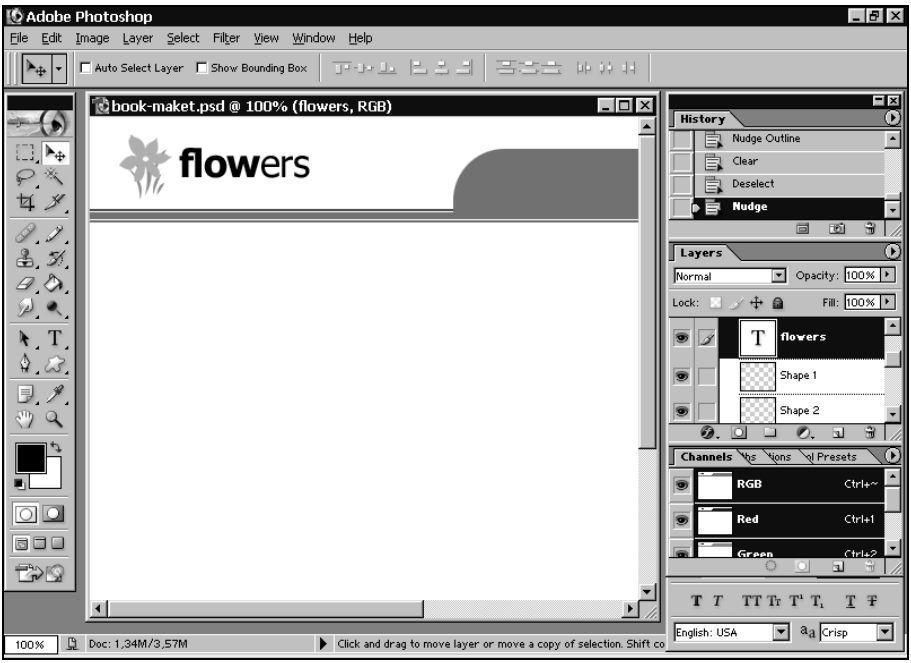


Рис. 15.4. Размещение логотипа

Теперь настала очередь составления меню навигации. Мы разделим его на три категории со своими пунктами. Для каждой категории создаем прямоугольный блок с названием категории, перед пунктами меню и между ними добавим декоративные элементы: буллеты и пунктирные разделители (рис. 15.5).

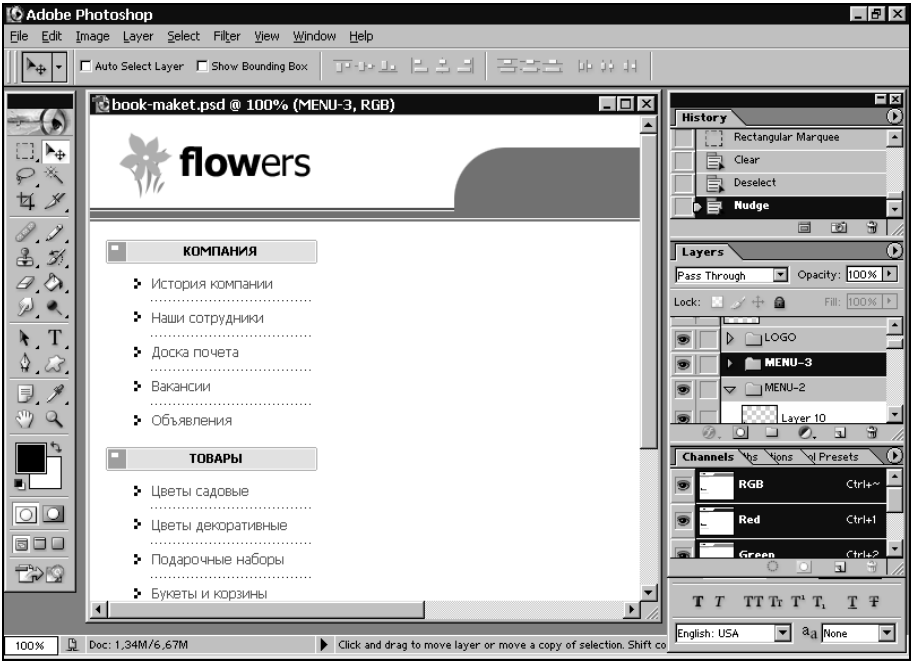


Рис. 15.5. Составление меню навигации

Далее занимаемся "подвалом". Дублируем три линии, которые у нас находятся в "шапке", и переносим вниз. Расстояние между объединенной тройной линией и нижней границей макета заливаем светло-серым цветом (#E6E8E9) для небольшого визуального оттенения цвета основного фона (белого).

Слева в "подвале" размещаем информацию об авторском праве (рис. 15.6).

Справа добавляем дублирующую навигацию (рис. 15.7).

Обратите внимание: в дублирующей навигации, помимо пунктов, соответствующих каждой категории меню, находятся ссылки "Поиск" и "Обратная связь". Именно для них и остается у нас место в верхней правой части макета внутри синего прямоугольного блока с закруглением (рис. 15.8).

Таким образом, на нашем макете остается незаполненной лишь содержательная часть. Отобразим ее на примере страницы с новостной лентой Web-сайта.

В первую очередь покажем "крошку" (рис. 15.9).

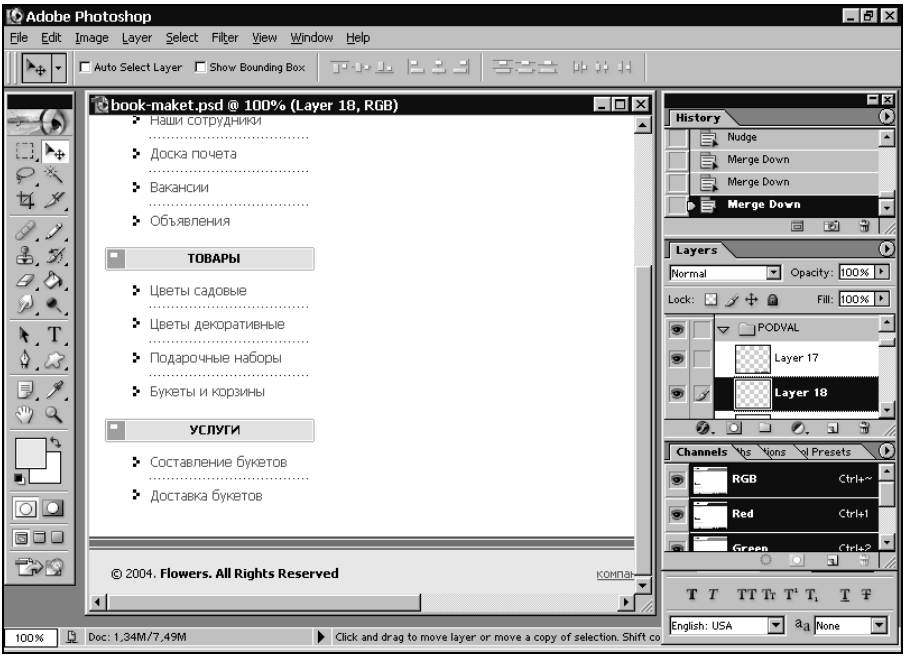


Рис. 15.6. Информация об авторском праве в "подвале" макета

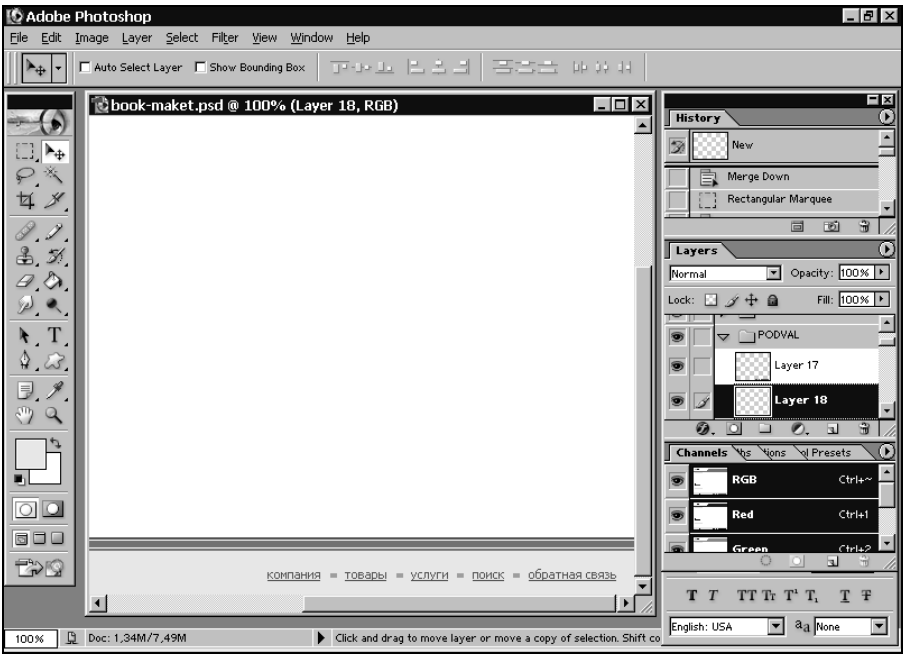


Рис. 15.7. Дублирующая навигация в "подвале" макета

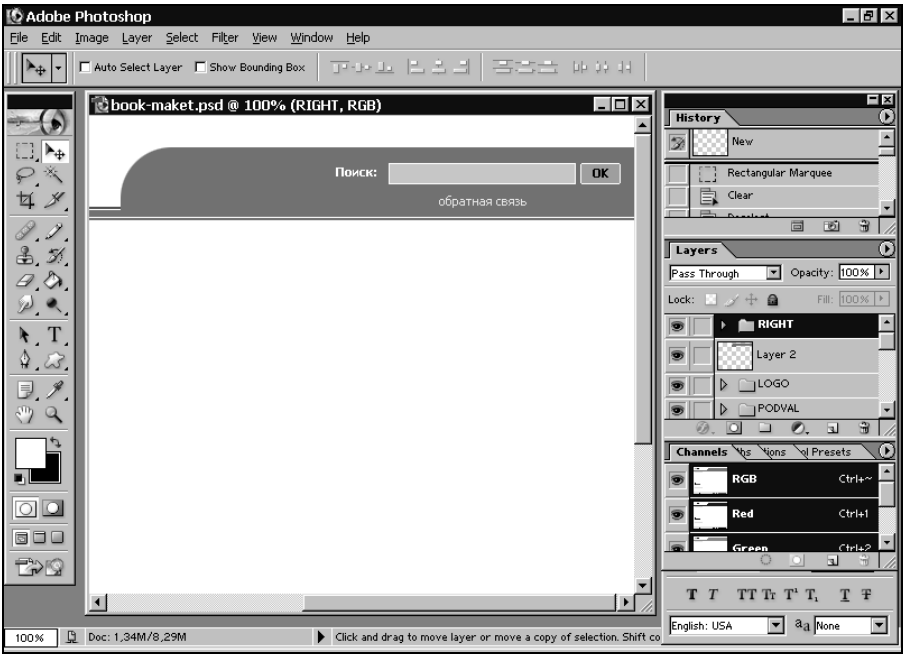


Рис. 15.8. Форма поиска и ссылка обратной связи

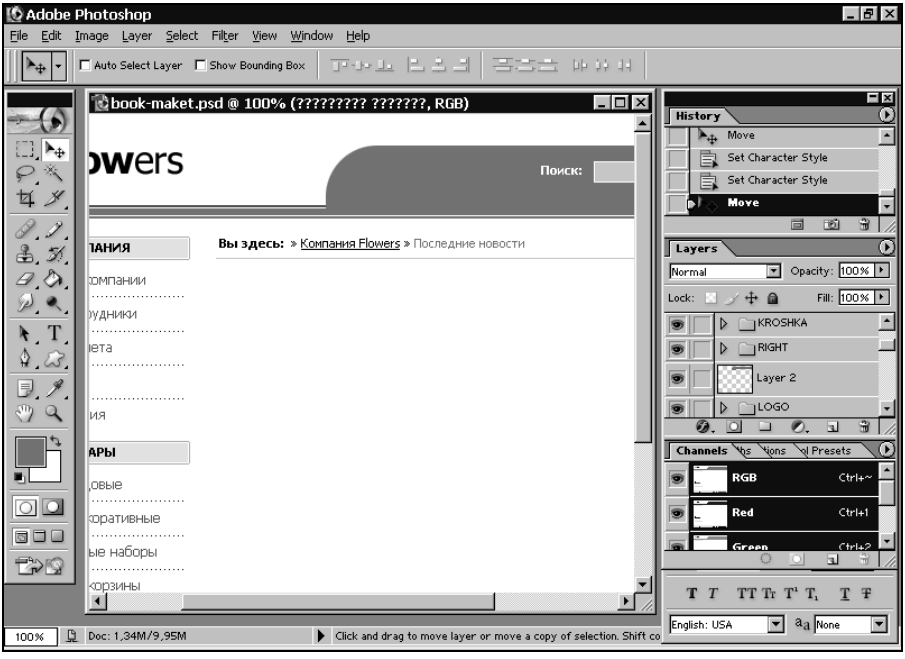


Рис. 15.9. Отображение "крошки"

Далее пойдет заголовок текущего раздела и анонсы новостей со ссылкой на полный текст (рис. 15.10).

Итак, наш графический макет готов к верстке!

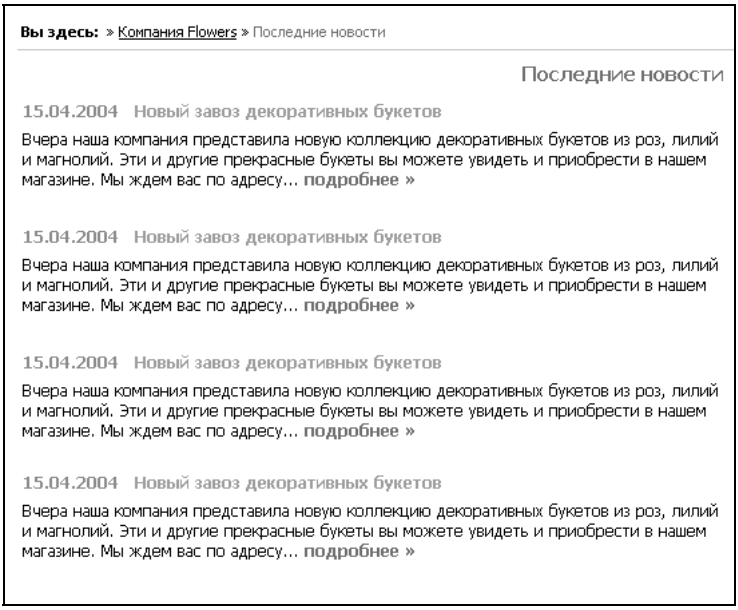
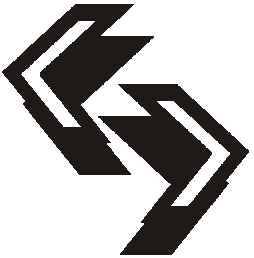


Рис. 15.10. Общий вид содержательной части на примере раздела с новостной лентой



ГЛАВА 16

HTML-верстка графического макета: шаг за шагом

Прежде чем начинать верстку графического макета, необходимо еще раз внимательно окинуть его взглядом, проанализировать каждый блок, каждый элемент, определяя для себя, каким образом и с помощью каких конструкций HTML они могут быть сверстаны.

В данном случае можно с уверенностью сказать, что в основе *верстки* нашего макета будут лежать три несущих таблицы:

- верхняя таблица будет содержать "шапку" сайта;
- средняя таблица разместит в себе меню навигации и содержательную часть (новостную ленту);
- нижнюю таблицу составит "подвал" сайта.

Прежде чем приступать к верстке, остается сформулировать два основных принципа, которых следует придерживаться в течение всего процесса верстки:

- минимальное разрешение 800×600 (соответственно, тестирование фрагментов кода и всего листинга в целом должно осуществляться именно при данном разрешении);
- тип верстки — "резина" (соответственно, необходимо помнить, что, как минимум, один элемент макета должен будет "растягиваться" при изменении размеров окна браузера).

Усвоили? Тогда начнем.

Первая таблица будет располагаться от верхнего края окна браузера до окончания тройной полосы на макете. Соответственно, верхняя несущая таблица будет состоять из трех ячеек:

1. Логотип (фиксированная ячейка).
2. Заглушка между логотипом и закруглением синего прямоугольного блока (фиксированная ячейка).

3. Синий прямоугольный блок без закругления (варьируемая ячейка, визуально состоящая из белого фона сверху и двойной синей полосы; именно в этой ячейке впоследствии будет размещена форма поиска и ссылка на обратную связь).

Таким образом, код верхней несущей таблицы (рис. 16.1) будет выглядеть так.

Листинг 16.1. Код верхней несущей таблицы

```
<HTML>
  <HEAD>
  <TITLE>Flowers</TITLE>
  <LINK REL="stylesheet" TYPE="text/css" HREF="flowers.css">
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=windows-1251">
  </HEAD>
<BODY BGCOLOR="#FFFFFF" LEFTMARGIN="0" TOPMARGIN="0" MARGINWIDTH="0"
MARGINHEIGHT="0" RIGHTMARGIN="0">
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD WIDTH="209">
         <A HREF="http://www.site-name.ru/">
            <IMG SRC="images/logo.gif" WIDTH="209" HEIGHT="93" BORDER="0"
             ALT="Flowers"></A></TD>
      <TD WIDTH="170">
         <IMG SRC="images/top_oval.gif" WIDTH="170" HEIGHT="93" ALT=""></TD>
      <TD BACKGROUND="images/top_var.gif" HEIGHT="93">
 
      </TD>
   </TR>
</TABLE>
</BODY>
</HTML>
```

В листинге 16.1 особое внимание следует уделить следующим моментам:

- `LINK REL="stylesheet" TYPE="text/css" HREF="flowers.css"` — путь к файлу стилевого шаблона CSS, который мы будем составлять параллельно с версткой макета;
- `TD BACKGROUND="images/top_var.gif" HEIGHT="93"` — единственная ячейка в таблице, у которой отсутствует ширина, т. к. она нефиксированная и заполняется файлом заднего фона `top_var.gif`;
- ` ` — пока у нас в этой ячейке нет формы поиска и ссылки на обратную связь, данная конструкция не позволяет верстке "расползтись".

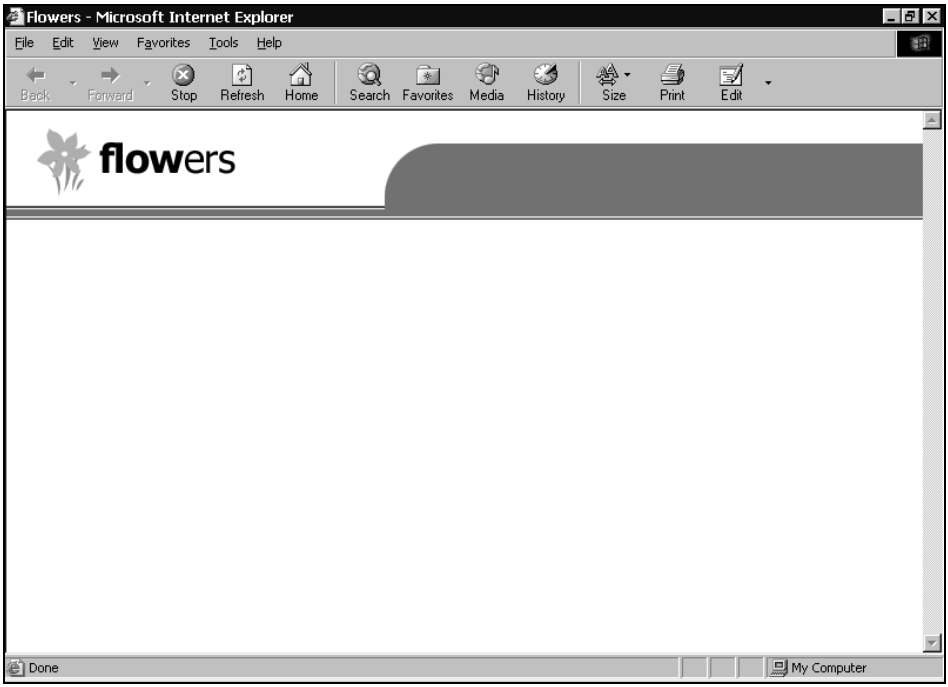


Рис. 16.1. Верхняя несущая таблица в браузере

Средняя таблица будет включать в себя:

- меню навигации (фиксированная ячейка);
- содержательную часть с новостной лентой (варьируемая ячейка).

Однако в этой части макета начинается текст, поэтому перед версткой средней таблицы (которая, к слову говоря, будет содержать другие вложенные таблицы) необходимо составить классы стилей CSS.

Меню навигации:

- заголовок категории: Tahoma, 11 пикселов, цвет — #000000, начертание — жирное, верхний регистр;
- ссылка пунктов меню: Tahoma, 12 пикселов, цвет — #0669B6, начертание — обычное.

Содержательная часть (новостная лента):

- ссылка в "крошке": Tahoma, 11 пикселов, цвет — #000000, начертание — подчеркнутое;
- надпись "Вы здесь": Tahoma, 11 пикселов, цвет — #000000, начертание — обычное;

- неактивный пункт "крошки": Tahoma, 11 пикселов, цвет — #2187D6, начертание — обычное;
- заголовок текущего раздела: Tahoma, 15 пикселов, цвет — #2187D6, начертание — жирное;
- дата новости: Tahoma, 12 пикселов, цвет — #879096, начертание — жирное;
- заголовок новости: Tahoma, 12 пикселов, цвет — #EA930A, начертание — жирное;
- текст анонса новости: Tahoma, 12 пикселов, цвет — #000000, начертание — обычное;
- ссылка на полный текст новости: Tahoma, 12 пикселов, цвет — #2187D6, начертание — жирное.

Чтобы потом не возвращаться к стилевым шаблонам, сразу пропишем классы для "подвала", формы поиска и ссылки на обратную связь:

- информация об авторском праве: Tahoma, 11 пикселов, цвет — #000000, начертание — обычное;
- ссылка дублирующей навигации: Tahoma, 11 пикселов, цвет — #0669B6, начертание — подчеркнутое;
- заголовок формы поиска: Tahoma, 11 пикселов, цвет — #FFFFFF, начертание — жирное;
- ссылка на обратную связь: Tahoma, 11 пикселов, цвет — #D6EDF6, начертание — обычное.

Отсюда получается следующая таблица стилей CSS (`flowers.css`):

Листинг 16.2. Файл стилевых шаблонов CSS в соответствии с разработанным макетом

```
.menutitle
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 11px;
        color: #000000;
        font-weight: bold;
        text-transform: uppercase;
        }
A.menu
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 12px;
```

```
        color: #0669B6;
        text-decoration: none;
        }
A.krlink
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 11px;
        color: #000000;
        text-decoration: underline;
        }
.krhere
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 11px;
        color: #000000;
        }
.kritem
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 11px;
        color: #2187D6;
        }
.header
        {
        font-family: Tahoma, Verdana, Arial;
        color: #2187D6;
        font-size: 15px;
        font-weight: bold;
        }
.newsdate
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 12px;
        color: #879096;
        font-weight: bold;
        }
.newstitle
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 12px;
        color: #EA930A;
        font-weight: bold;
        }
```

```
.newsanons
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 12px;
        color: #000000;
        }
A.newsmore
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 12px;
        color: #2187D6;
        font-weight: bold;
        text-decoration: none;
        }
.copyright
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 11px;
        color: #000000;
        }
A.navd
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 11px;
        color: #0669B6;
        text-decoration: underline;
        }
.searchtitle
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 11px;
        color: #FFFFFF;
        font-weight: bold;
        }
A.feedback
        {
        font-family: Tahoma, Verdana, Arial;
        font-size: 11px;
        color: #D6EDF6;
        text-decoration: none;
        }
```

Теперь можно переходить к верстке составных таблиц средней несущей таблицы. Начнем с меню навигации. По макету видно, что заголовки категорий

меню можно реализовать обычным текстом, а не графикой. Это несколько усложнит верстку, зато облегчит конечный размер HTML-файла.

Поэтому верстаем блок заголовка категории таким образом, чтобы посредине серого фона можно было вписать любой из заголовков (не стоит забывать, что эта таблица будет жестко фиксированной, а именно — шириной в 187 пикселов).

Листинг 16.3. Таблица заголовка категории навигационного меню

```
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD ROWSPAN="3">
         <IMG SRC="images/cat_1.gif" WIDTH="19" HEIGHT="21" ALT=""></TD>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_2.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
   <TR>
      <TH BACKGROUND="images/cat_bg.gif" WIDTH="166" HEIGHT="17">
<SPAN CLASS="menutitle">компания</SPAN></TH>
      <TD>
         <IMG SRC="images/cat_3.gif" WIDTH="2" HEIGHT="17" ALT=""></TD>
   </TR>
   <TR>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_4.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
</TABLE>
```

Однако таблица заголовка категории не подходит для всех пунктов меню: слишком большие зазоры между пунктами, наличие разделителей и т. д. Отсюда вывод: для блока пунктов меню необходима отдельная таблица — тоже фиксированной ширины в 187 пикселов.

Из макета (рис. 16.2) видно, что блок пунктов меню несколько сдвинут к правому краю относительно блока заголовка категории. Это можно реализовать несколькими способами: с помощью прозрачных распорок, увеличением прозрачной части буллетов и т. д. (выбирайте, какой вам по душе).

Еще один немаловажный момент: разделителем выступает точечный пунктир, который будет целесообразнее положить в виде обыкновенного графического объекта, а не файла заднего фона.

Итак, листинг отдельной таблицы для пунктов меню (листинг 16.4) будет выглядеть так (не забываем про классы стилей CSS).

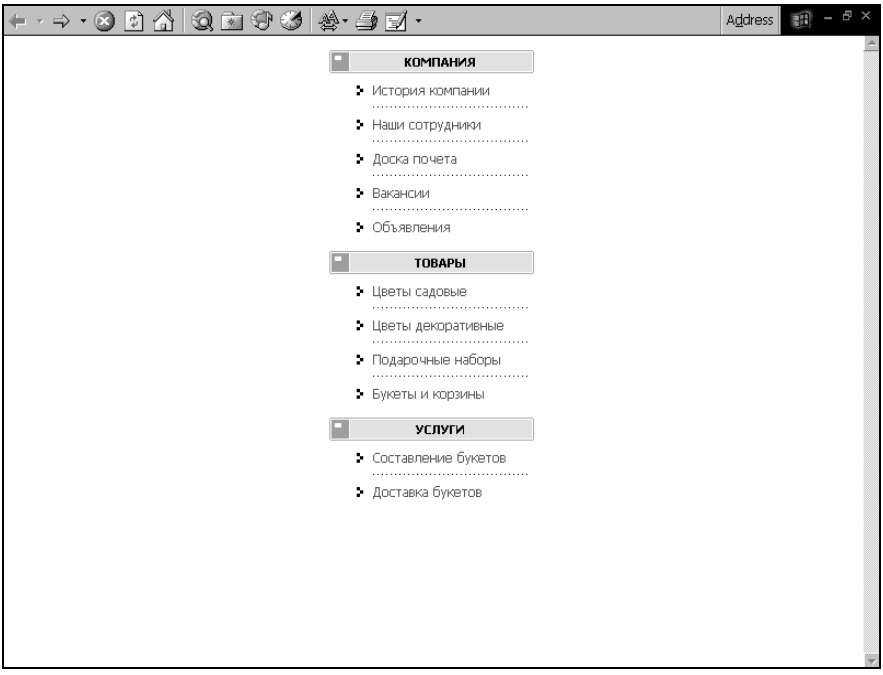


Рис. 16.2. Вид меню навигации в браузере

Листинг 16.4. Таблица пунктов меню

```
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="3"
ALIGN="center">
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
HEIGHT="24" ALT=""><A HREF="history.html" CLASS="menu">
История компании</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="staff.html" CLASS="menu">
       Наши сотрудники</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
```

```
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="fame.html" CLASS="menu">
       Доска почета</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="vacancies.html"
       CLASS="menu">Вакансии</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="announce.html"
       CLASS="menu">Объявления</A></TD>
   </TR>
</TABLE>
```

Теперь соединяем листинг заголовка категорий меню и листинг таблицы пунктов меню:

Листинг 16.5. Полный код отдельной навигационной категории с пунктами меню

```
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD ROWSPAN="3">
         <IMG SRC="images/cat_1.gif" WIDTH="19" HEIGHT="21" ALT=""></TD>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_2.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
   <TR>
      <TH BACKGROUND="images/cat_bg.gif" WIDTH="166" HEIGHT="17">
<SPAN CLASS="menutitle">компания</SPAN></TH>
      <TD>
         <IMG SRC="images/cat_3.gif" WIDTH="2" HEIGHT="17" ALT=""></TD>
   </TR>
```

```
   <TR>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_4.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
</TABLE>
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="3"
ALIGN="center">
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="history.html" CLASS="menu">
       История компании</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="staff.html" CLASS="menu">
       Наши сотрудники</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="fame.html" CLASS="menu">
       Доска почета</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="vacancies.html"
       CLASS="menu">Вакансии</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
```

```
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="announce.html"
       CLASS="menu">Объявления</A></TD>
   </TR>
</TABLE>
```

Добавить остальные категории навигационного меню не представляется сложным.

Теперь переходим к верстке содержательной части нашего макета, которая состоит из "крошки" и новостных анонсов.

Начнем с "крошки" (не забывая о том, что содержательная часть у нас нефиксированная):

Листинг 16.6. "Крошка"

```
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD><SPAN CLASS="krhere"><B>Вы здесь:</B>  &raquo;
       </SPAN> <A CLASS="krlink" HREF="index.html">Компания Flowers</A>
       <SPAN CLASS="krhere">&raquo;</SPAN> <SPAN CLASS="kritem">
       Последние новости</SPAN>
      </TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/spacer.gif" WIDTH="1" HEIGHT="4"></TD>
   </TR>
   <TR>
      <TD BGCOLOR="#C3C8CA"><IMG SRC="images/spacer.gif" WIDTH="1"
       HEIGHT="1"></TD>
   </TR>
</TABLE>
```

В эту же таблицу добавляем еще один ряд, в котором размещается заголовок текущего раздела ("Последние новости"):

```
<TR>
   <TD ALIGN="right"><SPAN CLASS="header">Последние новости</SPAN></TD>
</TR>
```

Наконец мы дошли до новостных анонсов. Судя по структуре блока новостей на макете, для каждого анонса придется создавать отдельную таблицу.

Листинг 16.7. Таблица новостного анонса

```
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="6"
ALIGN="center">
   <TR>
      <TD VALIGN="top"><P ALIGN="justify"><SPAN CLASS="newsdate">
       15.04.2004</SPAN>   <SPAN CLASS="newstitle">
       Новый завоз декоративных букетов</SPAN><BR>
<SPAN CLASS="newsanons">
Вчера наша компания представила новую коллекцию декоративных букетов
из роз, лилий и магнолий. Эти и другие прекрасные букеты вы можете
увидеть и приобрести в нашем магазине. Мы ждем вас по адресу...
<A CLASS="newsmore" HREF="newsmore.html">подробнее &raquo;</A>
</SPAN></P>
      </TD>
   </TR>
</TABLE>
```

Объединив листинги "крошки", заголовка текущего раздела и новостного анонса, мы получаем общий код содержательной части нашего Web-сайта.

Листинг 16.8. Общий код содержательной части Web-сайта

```
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD><SPAN CLASS="krhere"><B>Вы здесь:</B>  &raquo;
       </SPAN> <A CLASS="krlink" HREF="index.html">Компания Flowers</A>
       <SPAN CLASS="krhere">&raquo;</SPAN> <SPAN CLASS="kritem">
       Последние новости</SPAN>
      </TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/spacer.gif" WIDTH="1" HEIGHT="4"></TD>
   </TR>
   <TR>
      <TD BGCOLOR="#C3C8CA"><IMG SRC="images/spacer.gif" WIDTH="1"
       HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD ALIGN="right"><SPAN CLASS="header">Последние новости</SPAN></TD>
   </TR>
</TABLE>
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="6"
ALIGN="center">
```

```
   <TR>
      <TD VALIGN="top"><P ALIGN="justify"><SPAN CLASS="newsdate">
       15.04.2004</SPAN>   <SPAN CLASS="newstitle">
       Новый завоз декоративных букетов</SPAN><BR>
<SPAN CLASS="newsanons">
Вчера наша компания представила новую коллекцию декоративных букетов
из роз, лилий и магнолий. Эти и другие прекрасные букеты вы можете
увидеть и приобрести в нашем магазине. Мы ждем вас по адресу...
<A CLASS="newsmore" HREF="newsmore.html">подробнее &raquo;</A>
</SPAN></P>
      </TD>
   </TR>
</TABLE>
```

Осталось добавить еще несколько таблиц новостных анонсов и наша содержательная часть (рис. 16.3) примет совершенно иной вид.

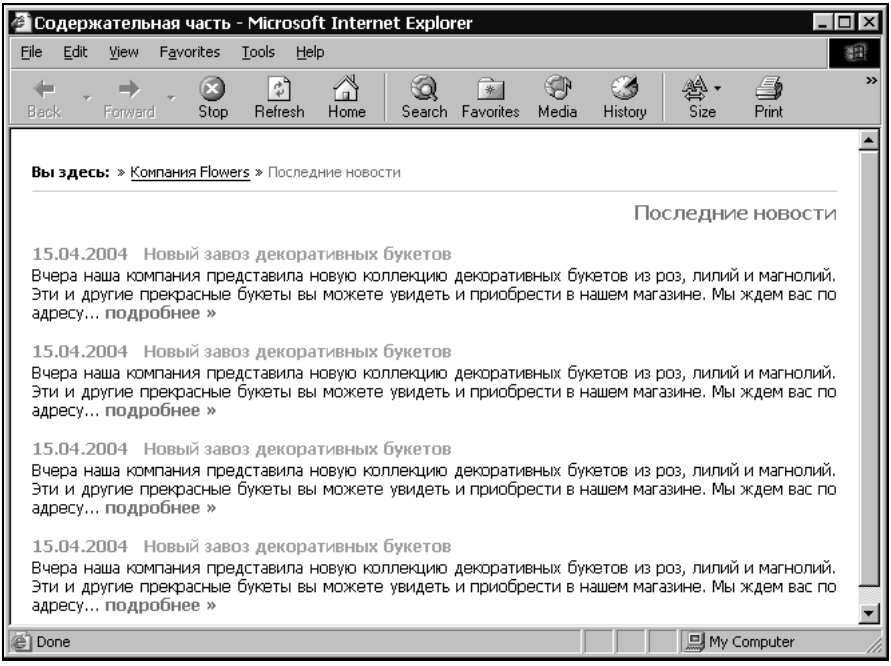


Рис. 16.3. Вид содержательной части Web-сайта в браузере

Итак, обе части средней несущей таблицы у нас уже готовы. Осталось лишь сгруппировать их так, как реализовано на графическом макете. Для этого создается простая табличная конструкция из двух ячеек: в первой будет листинг меню навигации, во второй — содержательная часть.

Итак, код средней несущей таблицы:

```
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="5" CELLSPACING="7"
ALIGN="center">
      <TR>
            <TD VALIGN="top">
<!-- Таблица меню навигации. Начало -->
...код...
<!-- Таблица меню навигации. Окончание -->
            </TD>
            <TD VALIGN="top">
<!-- Таблица содержательной части. Начало -->
...код...
<!-- Таблица содержательной части. Окончание -->
            </TD>
      </TR>
</TABLE>
```

Теперь осталось заполнить места, обозначенные комментариями, и средняя часть нашего Web-сайта готова!

Листинг 16.9. Общий код для средней части Web-сайта (меню навигации плюс содержательная часть)

```
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="5" CELLSPACING="7"
ALIGN="center">
   <TR>
      <TD VALIGN="top">
<!-- Таблица меню навигации. Начало -->
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD ROWSPAN="3">
         <IMG SRC="images/cat_1.gif" WIDTH="19" HEIGHT="21" ALT=""></TD>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_2.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
   <TR>
      <TH BACKGROUND="images/cat_bg.gif" WIDTH="166" HEIGHT="17">
<SPAN CLASS="menutitle">компания</SPAN></TH>
      <TD>
         <IMG SRC="images/cat_3.gif" WIDTH="2" HEIGHT="17" ALT=""></TD>
   </TR>
   <TR>
      <TD COLSPAN="2">
```

```
        <IMG SRC="images/cat_4.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
</TABLE>
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="3"
ALIGN="center">
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="history.html" CLASS="menu">
       История компании</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="staff.html" CLASS="menu">
       Наши сотрудники</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="fame.html" CLASS="menu">
       Доска почета</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="vacancies.html" CLASS="menu">
       Вакансии</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="announce.html" CLASS="menu">
       Объявления</A></TD>
   </TR>
```

```
</TABLE>
<IMG SRC="images/spacer.gif" WIDTH="1" HEIGHT="7" ALT=""><br>
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD ROWSPAN="3">
         <IMG SRC="images/cat_1.gif" WIDTH="19" HEIGHT="21" ALT=""></TD>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_2.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
   <TR>
      <TH BACKGROUND="images/cat_bg.gif" WIDTH="166" HEIGHT="17">
<SPAN CLASS="menutitle">товары</SPAN></TH>
      <TD>
         <IMG SRC="images/cat_3.gif" WIDTH="2" HEIGHT="17" ALT=""></TD>
   </TR>
   <TR>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_4.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
</TABLE>
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="3"
ALIGN="center">
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="garden.html" CLASS="menu">
       Цветы садовые</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="decorative.html" CLASS="menu">
       Цветы декоративные</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="gift.html" CLASS="menu">
       Подарочные наборы</A></TD>
   </TR>
```

```
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="bouquet.html" CLASS="menu">
       Букеты и корзины</A></TD>
   </TR>
</TABLE>
<IMG SRC="images/spacer.gif" WIDTH="1" HEIGHT="7" ALT=""><br>
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD ROWSPAN="3">
         <IMG SRC="images/cat_1.gif" WIDTH="19" HEIGHT="21" ALT=""></TD>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_2.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
   <TR>
      <TH BACKGROUND="images/cat_bg.gif" WIDTH="166" HEIGHT="17">
<SPAN CLASS="menutitle">услуги</SPAN></TH>
      <TD>
         <IMG SRC="images/cat_3.gif" WIDTH="2" HEIGHT="17" ALT=""></TD>
   </TR>
   <TR>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_4.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
</TABLE>
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="3"
ALIGN="center">
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="buoqset.html" CLASS="menu">
       Составление букетов</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center">><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="bouqdelivery.html" CLASS="menu">
       Доставка букетов</A></TD>
   </TR>
```

```
</TABLE>
<!-- Таблица меню навигации. Окончание -->
      </TD>
      <TD VALIGN="top">
<!-- Таблица содержательной части. Начало -->
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD><SPAN CLASS="krhere"><B>Вы здесь:</B>  &raquo;
       </SPAN> <A CLASS="krlink" HREF="index.html">Компания Flowers</A>
       <SPAN CLASS="krhere">&raquo;</SPAN> <SPAN CLASS="kritem">
       Последние новости</SPAN>
      </TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/spacer.gif" WIDTH="1" HEIGHT="4"></TD>
   </TR>
   <TR>
      <TD BGCOLOR="#C3C8CA"><IMG SRC="images/spacer.gif" WIDTH="1"
HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD ALIGN="right"><SPAN CLASS="header">Последние новости</SPAN></TD>
   </TR>
</TABLE>
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="6"
ALIGN="center">
   <TR>
      <TD VALIGN="top"><P ALIGN="justify"><SPAN CLASS="newsdate">
       15.04.2004</SPAN>   <SPAN CLASS="newstitle">
       Новый завоз декоративных букетов</SPAN><BR>
<SPAN CLASS="newsanons">
Вчера наша компания представила новую коллекцию декоративных букетов
из роз, лилий и магнолий. Эти и другие прекрасные букеты вы можете
увидеть и приобрести в нашем магазине. Мы ждем вас по адресу...
<A CLASS="newsmore" HREF="newsmore.html">подробнее &raquo;</A>
</SPAN></P>
      </TD>
   </TR>
</TABLE>
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="6"
ALIGN="center">
   <TR>
      <TD VALIGN="top"><P ALIGN="justify"><SPAN CLASS="newsdate">
       15.04.2004</SPAN>   <SPAN CLASS="newstitle">
       Новый завоз декоративных букетов</SPAN><BR>
```

```
<SPAN CLASS="newsanons">
Вчера наша компания представила новую коллекцию декоративных букетов
из роз, лилий и магнолий. Эти и другие прекрасные букеты вы можете
увидеть и приобрести в нашем магазине. Мы ждем вас по адресу...
<A CLASS="newsmore" HREF="newsmore.html">подробнее &raquo;</A>
</SPAN></P>
      </TD>
   </TR>
</TABLE>
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="6"
ALIGN="center">
   <TR>
      <TD VALIGN="top"><P ALIGN="justify"><SPAN CLASS="newsdate">
       15.04.2004</SPAN>   <SPAN CLASS="newstitle">
       Новый завоз декоративных букетов</SPAN><BR>
<SPAN CLASS="newsanons">
Вчера наша компания представила новую коллекцию декоративных букетов
из роз, лилий и магнолий. Эти и другие прекрасные букеты вы можете
увидеть и приобрести в нашем магазине. Мы ждем вас по адресу...
<A CLASS="newsmore" HREF="newsmore.html">подробнее &raquo;</A>
</SPAN></P>
      </TD>
   </TR>
</TABLE>
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="6"
ALIGN="center">
   <TR>
      <TD VALIGN="top"><P ALIGN="justify"><SPAN CLASS="newsdate">
       15.04.2004</SPAN>   <SPAN CLASS="newstitle">
       Новый завоз декоративных букетов</SPAN><BR>
<SPAN CLASS="newsanons">
Вчера наша компания представила новую коллекцию декоративных букетов
из роз, лилий и магнолий. Эти и другие прекрасные букеты вы можете
увидеть и приобрести в нашем магазине. Мы ждем вас по адресу...
<A CLASS="newsmore" HREF="newsmore.html">подробнее &raquo;</A>
</SPAN></P>
      </TD>
   </TR>
</TABLE>
<!-- Таблица содержательной части. Окончание -->
      </TD>
   </TR>
</TABLE>
```

Наконец мы добрались до верстки "подвала", которая, если посмотреть на графический макет, не представляет особого труда (рис. 16.4). Нижняя несу-

щая таблица будет состоять из двух рядов (верхний — с задним фоном в виде тройной полосы и нижний — с заливкой светло-серым цветом; оба нефиксированной ширины), причем нижний ряд будет содержать еще одну таблицу из двух ячеек: для информации об авторских правах и дублирующей навигации, соответственно.

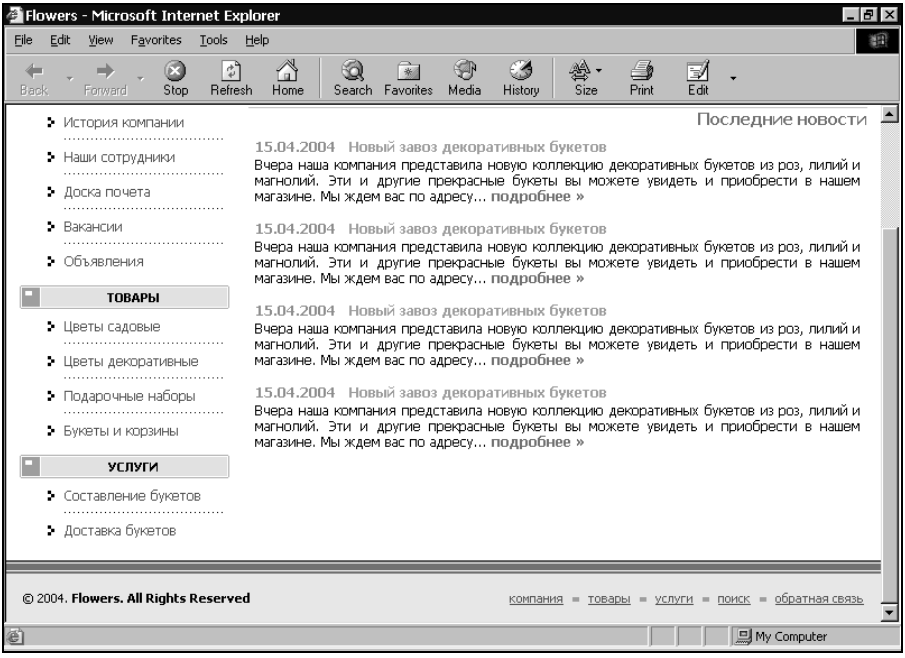


Рис. 16.4. Вид нижней части Web-сайта в браузере

Листинг 16.10. Код нижней части Web-сайта

```
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD BACKGROUND="images/b_bg.gif"><IMG SRC="images/spacer.gif"
       WIDTH="1" HEIGHT="12"></TD>
   </TR>
   <TR>
      <TD BGCOLOR="#E6E8E9">
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="15" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD><SPAN CLASS="copyright">&copy; 2004. <B>Flowers. All Rights
       Reserved</B></SPAN></TD>
```

```
      <TD ALIGN="right"><A CLASS="navd" HREF="/company/">компания</A>
       <IMG SRC="images/b_bullet.gif" WIDTH="22" HEIGHT="10">
       <A CLASS="navd" HREF="/flowers/">товары</A>
       <IMG SRC="images/b_bullet.gif" WIDTH="22" HEIGHT="10">
       <A CLASS="navd" HREF="/services/">услуги</A>
       <IMG SRC="images/b_bullet.gif" WIDTH="22" HEIGHT="10">
       <A CLASS="navd" HREF="/search/">поиск</A><IMG SRC="images/
       b_bullet.gif" WIDTH="22" HEIGHT="10"><A CLASS="navd"
       HREF="/feedback/">обратная связь</A></TD>
   </TR>
</TABLE>
      </TD>
   </TR>
</TABLE>
```

Теперь добавим этот листинг после кода верхней и средней несущих таблиц и получим почти готовую верстку нашего графического макета.

И, наконец, последний штрих — форма поиска и ссылка на обратную связь в верхней правой части макета.

Их мы поместим в одну общую таблицу, состоящую из трех рядов: в первом — прозрачная распорка, фиксирующая таблицу относительно верхней границы окна браузера; во втором — четыре ячейки (заголовок формы поиска, текстовое поле ввода, кнопка подтверждения и пустая ячейка, создающая отступ от правого края окна браузера).

При этом, в третьем ряду таблицы все ячейки останутся пустыми, за исключением второй, где мы разместим ссылку на обратную связь.

Листинг 16.11. Код формы поиска и ссылки на обратную связь

```
<TABLE BORDER="0" CELLPADDING="0" CELLSPACING="3" ALIGN="right">
   <TR>
      <TD COLSPAN="4"><IMG SRC="images/spacer.gif" WIDTH="1"
HEIGHT="25"></TD>
   </TR>
   <TR>
      <TD ALIGN="right"><FORM STYLE="margin: 0px 0px 0px 0px;">
       <SPAN CLASS="searchtitle">Поиск:  </SPAN></TD><TD>
       <INPUT TYPE="text" STYLE="border: 1 solid; color: #FFFFFF;
       background-color: #C3C8CA; font-size: 13px; width: 170px;">
       </TD><TD><INPUT TYPE="image" SRC="images/submit.gif" WIDTH="36"
       HEIGHT="19" ALT="OK" BORDER="0" ALIGN="absmiddle">
       </FORM></TD><TD> </TD>
   </TR>
   <TR>
```

```
      <TD> </TD><TD ALIGN="center"><A CLASS="feedback"
       HREF="/feedback/">обратная связь</A></TD><TD> </TD><TD> 
       </TD>
   </TR>
</TABLE>
```

Особое внимание следует обратить на конструкцию `FORM STYLE="margin: 0px 0px 0px 0px;` — отступы обнулены во избежание переносов, которые производит тег `<FORM>` со значениями по умолчанию.

Вот и все! Верстка графического макета нашего Web-сайта закончена.

Ниже приведены полный листинг верстки и вид страницы в браузере (рис. 6.5).

Рис. 16.5. Вид сверстанного Web-сайта в браузере

Листинг 16.12. Общий код верстки графического макета

```
<HTML>
  <HEAD>
  <TITLE>Flowers</TITLE>
  <LINK REL="stylesheet" TYPE="text/css" HREF="flowers.css">
```

```
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=windows-1251">
  </HEAD>
<BODY BGCOLOR="#FFFFFF" LEFTMARGIN="0" TOPMARGIN="0" MARGINWIDTH="0"
MARGINHEIGHT="0" RIGHTMARGIN="0">
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD WIDTH="209">
         <A HREF="http://www.site-name.ru/">
            <IMG SRC="images/logo.gif" WIDTH="209" HEIGHT="93" BORDER="0"
             ALT="Flowers"></A></TD>
      <TD WIDTH="170">
         <IMG SRC="images/top_oval.gif" WIDTH="170" HEIGHT="93" ALT=""></TD>
      <TD BACKGROUND="images/top_var.gif" HEIGHT="93">

<TABLE BORDER="0" CELLPADDING="0" CELLSPACING="3" ALIGN="right">
   <TR>
      <TD COLSPAN="4"><IMG SRC="images/spacer.gif" WIDTH="1"
       HEIGHT="25"></TD>
   </TR>
   <TR>
      <TD ALIGN="right"><FORM STYLE="margin: 0px 0px 0px 0px;">
       <SPAN CLASS="searchtitle">Поиск:  </SPAN></TD><TD>
       <INPUT TYPE="text" STYLE="border: 1 solid; color: #FFFFFF;
       background-color: #C3C8CA; font-size: 13px; width: 170px;">
       </TD><TD><INPUT TYPE="image" SRC="images/submit.gif" WIDTH="36"
       HEIGHT="19" ALT="OK" BORDER="0" ALIGN="absmiddle">></FORM>
       </TD><TD> </TD>
   </TR>
   <TR>
      <TD> </TD><TD ALIGN="center"><A CLASS="feedback"
       HREF="/feedback/">обратная связь</A></TD><TD> </TD><TD> 
       </TD>
   </TR>
</TABLE>
      </TD>
   </TR>
</TABLE>
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="5" CELLSPACING="7"
ALIGN="center">
   <TR>
      <TD VALIGN="top">
<!-- Таблица меню навигации. Начало -->
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
```

```
      <TD ROWSPAN="3">
         <IMG SRC="images/cat_1.gif" WIDTH="19" HEIGHT="21" ALT=""></TD>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_2.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
   <TR>
      <TH BACKGROUND="images/cat_bg.gif" WIDTH="166" HEIGHT="17">
<SPAN CLASS="menutitle">компания</SPAN></TH>
      <TD>
         <IMG SRC="images/cat_3.gif" WIDTH="2" HEIGHT="17" ALT=""></TD>
   </TR>
   <TR>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_4.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
</TABLE>
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="3"
ALIGN="center">
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="history.html" CLASS="menu">
       История компании</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="staff.html" CLASS="menu">
       Наши сотрудники</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="fame.html" CLASS="menu">
       Доска почета</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
```

```
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="vacancies.html" CLASS="menu">
       Вакансии</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="announce.html" CLASS="menu">
       Объявления</A></TD>
   </TR>
</TABLE>
<IMG SRC="images/spacer.gif" WIDTH="1" HEIGHT="7" ALT=""><br>
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD ROWSPAN="3">
         <IMG SRC="images/cat_1.gif" WIDTH="19" HEIGHT="21" ALT=""></TD>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_2.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
   <TR>
      <TH BACKGROUND="images/cat_bg.gif" WIDTH="166" HEIGHT="17">
<SPAN CLASS="menutitle">товары</SPAN></TH>
      <TD>
         <IMG SRC="images/cat_3.gif" WIDTH="2" HEIGHT="17" ALT=""></TD>
   </TR>
   <TR>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_4.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
</TABLE>
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="3"
ALIGN="center">
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="garden.html" CLASS="menu">
       Цветы садовые</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
```

```
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="decorative.html" CLASS="menu">
       Цветы декоративные</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="gift.html" CLASS="menu">
       Подарочные наборы</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="bouquet.html" CLASS="menu">
       Букеты и корзины</A></TD>
   </TR>
</TABLE>
<IMG SRC="images/spacer.gif" WIDTH="1" HEIGHT="7" ALT=""><br>
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD ROWSPAN="3">
         <IMG SRC="images/cat_1.gif" WIDTH="19" HEIGHT="21" ALT=""></TD>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_2.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
   <TR>
      <TH BACKGROUND="images/cat_bg.gif" WIDTH="166" HEIGHT="17">
<SPAN CLASS="menutitle">услуги</SPAN></TH>
      <TD>
         <IMG SRC="images/cat_3.gif" WIDTH="2" HEIGHT="17" ALT=""></TD>
   </TR>
   <TR>
      <TD COLSPAN="2">
         <IMG SRC="images/cat_4.gif" WIDTH="168" HEIGHT="2" ALT=""></TD>
   </TR>
</TABLE>
```

```
<TABLE WIDTH="187" BORDER="0" CELLPADDING="0" CELLSPACING="3"
ALIGN="center">
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="buoqset.html" CLASS="menu">
       Составление букетов</A></TD>
   </TR>
   <TR>
      <TD COLSPAN="2" ALIGN="center"><IMG SRC="images/dots.gif"
       WIDTH="179" HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/bullet.gif" ALIGN="absmiddle" WIDTH="37"
       HEIGHT="24" ALT=""><A HREF="bouqdelivery.html" CLASS="menu">
       Доставка букетов</A></TD>
   </TR>
</TABLE>
<!-- Таблица меню навигации. Окончание -->
      </TD>
      <TD VALIGN="top">
<!-- Таблица содержательной части. Начало -->
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD><SPAN CLASS="krhere"><B>Вы здесь:</B>  &raquo;
       </SPAN> <A CLASS="krlink" HREF="index.html">Компания Flowers</A>
       <SPAN CLASS="krhere">&raquo;</SPAN> <SPAN CLASS="kritem">
       Последние новости</SPAN>
      </TD>
   </TR>
   <TR>
      <TD><IMG SRC="images/spacer.gif" WIDTH="1" HEIGHT="4"></TD>
   </TR>
   <TR>
      <TD BGCOLOR="#C3C8CA"><IMG SRC="images/spacer.gif" WIDTH="1"
       HEIGHT="1"></TD>
   </TR>
   <TR>
      <TD ALIGN="right"><SPAN CLASS="header">Последние новости</SPAN></TD>
   </TR>
</TABLE>
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="6"
ALIGN="center">
   <TR>
      <TD VALIGN="top"><P ALIGN="justify"><SPAN CLASS="newsdate">
       15.04.2004</SPAN>   
       <SPAN CLASS="newstitle">Новый завоз декоративных букетов</SPAN><BR>
```

```
<SPAN CLASS="newsanons">
Вчера наша компания представила новую коллекцию декоративных букетов
из роз, лилий и магнолий. Эти и другие прекрасные букеты вы можете
увидеть и приобрести в нашем магазине. Мы ждем вас по адресу...
<A CLASS="newsmore" HREF="newsmore.html">подробнее &raquo;</A>
</SPAN></P>
      </TD>
   </TR>
</TABLE>
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="6"
ALIGN="center">
   <TR>
      <TD VALIGN="top"><P ALIGN="justify"><SPAN CLASS="newsdate">
       15.04.2004</SPAN>   <SPAN CLASS="newstitle">
       Новый завоз декоративных букетов</SPAN><BR>
<SPAN CLASS="newsanons">
Вчера наша компания представила новую коллекцию декоративных букетов
из роз, лилий и магнолий. Эти и другие прекрасные букеты вы можете
увидеть и приобрести в нашем магазине. Мы ждем вас по адресу...
<A CLASS="newsmore" HREF="newsmore.html">подробнее &raquo;</A>
</SPAN></P>
      </TD>
   </TR>
</TABLE>
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="6"
ALIGN="center">
   <TR>
      <TD VALIGN="top"><P ALIGN="justify"><SPAN CLASS="newsdate">
       15.04.2004</SPAN>   <SPAN CLASS="newstitle">
       Новый завоз декоративных букетов</SPAN><BR>
<SPAN CLASS="newsanons">
Вчера наша компания представила новую коллекцию декоративных букетов
из роз, лилий и магнолий. Эти и другие прекрасные букеты вы можете
увидеть и приобрести в нашем магазине. Мы ждем вас по адресу...
<A CLASS="newsmore" HREF="newsmore.html">подробнее &raquo;</A>
</SPAN></P>
      </TD>
   </TR>
</TABLE>
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="6"
ALIGN="center">
   <TR>
      <TD VALIGN="top"><P ALIGN="justify"><SPAN CLASS="newsdate">
       15.04.2004</SPAN>   <SPAN CLASS="newstitle">
       Новый завоз декоративных букетов</SPAN><BR>
<SPAN CLASS="newsanons">
```

```
Вчера наша компания представила новую коллекцию декоративных букетов
из роз, лилий и магнолий. Эти и другие прекрасные букеты вы можете
увидеть и приобрести в нашем магазине. Мы ждем вас по адресу...
<A CLASS="newsmore" HREF="newsmore.html">подробнее &raquo;</A>
</SPAN></P>
      </TD>
   </TR>
</TABLE>
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="6"
ALIGN="center">
   <TR>
      <TD VALIGN="top"><P ALIGN="justify"><SPAN
CLASS="newsdate">15.04.2004</SPAN>   <SPAN
CLASS="newstitle">Новый завоз декоративных букетов</SPAN><BR>
<SPAN CLASS="newsanons">
Вчера наша компания представила новую коллекцию декоративных букетов
из роз, лилий и магнолий. Эти и другие прекрасные букеты вы можете
увидеть и приобрести в нашем магазине. Мы ждем вас по адресу...
<A CLASS="newsmore" HREF="newsmore.html">подробнее &raquo;</A>
</SPAN></P>
      </TD>
   </TR>
</TABLE>
<!-- Таблица содержательной части. Окончание -->
      </TD>
   </TR>
</TABLE>
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD BACKGROUND="images/b_bg.gif"><IMG SRC="images/spacer.gif"
       WIDTH="1" HEIGHT="12"></TD>
   </TR>
   <TR>
      <TD BGCOLOR="#E6E8E9">
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="15" CELLSPACING="0"
ALIGN="center">
   <TR>
      <TD><SPAN CLASS="copyright">&copy; 2004. <B>Flowers. All Rights
       Reserved</B></SPAN></TD>
      <TD ALIGN="right"><A CLASS="navd" HREF="/company/">компания</A>
       <IMG SRC="images/b_bullet.gif" WIDTH="22" HEIGHT="10">
       <A CLASS="navd" HREF="/flowers/">товары</A><IMG SRC="images/
       b_bullet.gif" WIDTH="22" HEIGHT="10"><A CLASS="navd"
       HREF="/services/">услуги</A><IMG SRC="images/b_bullet.gif"
       WIDTH="22" HEIGHT="10"><A CLASS="navd" HREF="/search/">
```

```
      поиск</A><IMG SRC="images/b_bullet.gif" WIDTH="22" HEIGHT="10">
      <A CLASS="navd" HREF="/feedback/">обратная связь</A></TD>
   </TR>
</TABLE>
      </TD>
   </TR>
</TABLE>
</BODY>
</HTML>
```

ГЛАВА 17

Продвижение Web-сайта в Интернете

Итак, ваш Web-сайт готов. Позади — большой труд и время, потраченное на разработку своего собственного интернет-проекта. Вы придумали и нарисовали графический макет, продумали структуру и навигацию сайта, сверстали все составляющие страницы и сохранили на своем компьютере.

В принципе, на этом можно было бы и остановиться. Но задайте себе вопрос: "Кто помимо вас — создателя Web-сайта — сможет увидеть ваше детище?". Кто сумеет оценить ваш труд и старания? Ваши друзья и знакомые, родственники и гости. Этим аудитория вашего сайта исчерпывается.

Для достижения большей и разносторонней аудитории ваш сайт нужно разместить в Интернете. Но просто разместить тоже мало, надо, чтобы об этом узнали сотни тысяч пользователей всемирной сети.

Здесь приходит на помощь наука продвижения интернет-проектов в глобальной сети. Мы не оговорились, это действительно целая наука — со своими принципами, правилами и аксиомами, задачами и теоремами и т. д. И как у всякой научной дисциплины, у нее есть базовые основы, овладев которыми вы можете пойти еще дальше в своем совершенствовании и развитии.

Поэтому в этой главе мы уделим немного времени основам продвижения Web-сайтов в Интернете.

Интернет-маркетинг

К сожалению, на сегодняшний день не существует единой классификации маркетинговых процессов в Интернете. Связано это, прежде всего, с тем, что, во-первых, в России нет сложившейся организационной структуры, занимающейся общей стандартизацией сетевых рекламных средств и носителей (в США ситуация обстоит гораздо лучше: созданное Бюро интернет-рекламы,

Internet Advertising Bureau, активно занимается вопросами систематизации средств продвижения сетевых проектов), во-вторых, мнения насчет того, что является признаком проявления сетевой рекламы, а что — нет, сильно расходятся, когда речь заходит о целенаправленном рекламном планировании (причина тому — непонимание или недооценка многих рекламных механизмов, а также отсутствие полной или достоверной статистической информации по степени эффективности определенных видов интернет-рекламы). К наиболее действенным и активно применяемым в интернет-отрасли направлениям сетевого маркетинга можно отнести следующие.

Размещение информации в каталогах интернет-ресурсов

Согласно результатам многочисленных исследований, занесение данных в сетевые информационные хранилища (универсальные и тематические каталоги, классификаторы, "желтые страницы", справочники) входит в первую тройку рекламных средств (вместе с регистрацией в поисковых системах и баннерной рекламой), благодаря действию которых целевая аудитория чаще всего получает информацию об искомых товарах или услугах.

По структуре интернет-каталог представляет собой тематический рубрикатор, использующий нисходящую навигационную иерархию: сначала идут основные разделы каталога, далее — подразделы, последние, в свою очередь, делятся на другие подразделы и т. д. Процесс занесения в базу данных каталога информации о сайте рекламодателя не вызывает особого труда и производится бесплатно (редкое исключение составляют узкоспециализированные справочники). Обычно заносится следующая информация: название, интернет-адрес регистрируемого ресурса, адрес электронной почты, координаты для связи (адрес электронной почты или, в ряде случаев, номера телефонов/факсов и почтовый адрес), имя контактного или отвечающего за регистрацию в каталоге лица, краткое описание сайта, перечень основных ключевых слов и пароль для редактирования/удаления вашей регистрационной записи. По истечении срока проверки вашего ресурса модератором (менеджером проекта) каталога информация заносится (или не заносится, если не соответствует требованиям по размещению информации в базе данных классификатора) в каталог.

Существует мнение, что регистрация в интернет-каталогах не приносит должного эффекта, т. к. в крупных классификаторах ежедневно регистрируются тысячи новых ресурсов, а если учесть, что свежая информация появляется вверху результирующего списка, выдаваемого каталогом, то может случиться так, что размещенный вчера ресурс уже завтра окажется десятым или тридцать пятым. На самом деле это не так.

Во-первых, при подготовке вашего Web-сайта к регистрации в каталоге следует тщательно продумать такие наиважнейшие пункты, как ключевые слова и описание ресурса. Необходимо выбрать небольшое количество ключевых слов или фраз, наиболее ярко характеризующих ваш сайт или область деятельности вашей организации. Подготовить краткое, но вместе с тем исчерпывающее описание (стоит избегать избитых фраз, банальных высказываний, неопределенных утверждений, слишком завышенных характеристик и т. д. — для этого было бы неплохо еще до подготовки текста посетить сайты своих конкурентов, чтобы избежать допущенных ими ошибок, и выявить то, чего на других ресурсах нет). Главное здесь — не количество, а качество, т. к. занести в каталог первые два-три абзаца из рекламного проспекта вашей организации без внимательного изучения и корректировки текста — означает не думать о своем потенциальном потребителе, который будет читать это описание, и, если оно ему не понравится или не вызовет доверия, ни за что не пойдет на ваш сайт.

Если вы подошли со всей ответственностью к подготовке текста описания и ключевых слов, то можете быть на 90% уверены, что пользователь каталога заинтересуется вашими услугами. Недостающие 10% составляет вероятность того, что он не найдет на сайте нужной информации или данное вами в каталог описание будет противоречить содержанию вашего интернет-ресурса. Что же касается угрозы перенесения ваших данных в низ разделов каталога по мере его пополнения, то следует сказать, что, во-первых, действительно заинтересованный потребитель, ценящий свое время и деньги, не остановится на первом попавшемся ресурсе, а обязательно сравнит аналогичные услуги с вашими, а во-вторых, сегодня во всех крупных интернет-каталогах существует функция выбора режима просмотра информации (в алфавитном порядке, по дате появления, по популярности) с целью облегчения поиска в строгом соответствии с запросами пользователя каталога.

Регистрация в поисковых системах

Другим наиболее эффективным способом продвижения интернет-ресурса является регистрация в *поисковых системах*. Поисковые системы, так же, как каталоги, выдают по запросу пользователя список ресурсов и имеют собственный рубрикатор. Принципиальное различие заключается в том, что каталоги могут искать информацию только в пределах своей внутренней базы данных, а поисковые системы осуществляют поиск по всей всемирной сети. Такие системы работают на основе специальной программы-робота, которая регулярно просматривает множество сайтов, заносит информацию о них (ключевые слова, описание и др.) в индексы (программные базы данных), затем через некоторое время просматривает их повторно и, если вновь полученная информация не совпадает с той, что уже есть в индексах, обновляет свои данные.

При регистрации в поисковой системе требуется ввести только адрес и название вашего ресурса, а иногда контактный e-mail. Далее программа-робот все сделает сама. Однако последнее возможно только в том случае, если составные документы вашего сайта содержат метаопределения (конструкции HTML), которые программа-робот просматривает в первую очередь. Если такие инструкции опущены, робот считывает текст, находящийся в самом верху страницы документа, и нет гарантии, что этот текст окажется относящимся к деятельности вашей организации. Добавить метаопределения можно самостоятельно при знании основ HTML или прибегнув к помощи специалиста. Рекомендации по заполнению метаопределений (которые содержат данные о ключевых словах, описание, тип режима индексирования для программы-робота и пр.) аналогичны ситуации с каталогами ресурсов: главное — краткость и доступность изложения. Хотя бывают исключения, когда требуется большое количество ключевых конструкций (например, сайты с многоязыковой поддержкой, где приходится в метаопределениях указывать ключевые слова и описание по всем используемым языкам).

Организация рейтинга

Рейтинговые службы — это сервис, предоставляющий более или менее (в зависимости от технической оснащенности такой службы) подробную и достоверную статистическую информацию о вашем ресурсе. Многие каталоги сегодня, помимо простой регистрации в базе данных, дают возможность владельцам сайтов внести свой ресурс в общий или тематический рейтинг. Чтобы принять участие в рейтинге, нужно заполнить заявку (она аналогична регистрационной форме в каталогах) и получить отдельный программный код, поместив который на страницах своего сайта, вы сможете видеть, какое количество посетителей было в сутки, сколько документов было запрошено и т. д. в виде небольшого графического изображения. При нажатии на это изображение можно перейти на более детальную статистику: количество визитов за определенный период, названия запрошенных документов, данные по используемому посетителями программному обеспечению, географическому расположению и многое другое.

Доски объявлений и форумы

Электронные доски объявлений (Bulletin Boards) и форумы являются местом в Интернете, где практически любой желающий может оставить информацию: ознакомительного, пригласительного, рекламного характера и пр. Данный вид сетевого сервиса представляет собой перечень тем, выставленных для обсуждения, куда можно добавлять ответы, замечания, комментарии и т. д. Доски объявлений и форумы бывают узконаправленные (например,

посвященные только вопросам образования в России) и универсальные, в которых существует определенный рубрикатор.

Размещение рекламной информации на досках объявлений и форумах имеет много преимуществ перед другими средствами продвижения интернет-ресурсов. Во-первых, данная услуга абсолютно бесплатна, что немаловажно, особенно когда у руководителей организаций актуализируется вопрос о том, какие рекламные средства выбрать и при этом иметь поменьше расходов. Во-вторых, добавление информации может не являться разовой акцией: преобладающее количество современных досок объявлений и форумов не имеют никаких лимитов на сообщения пользователей. Другими словами, вы можете одно и то же рекламное сообщение разместить несколько раз и в различных рубриках.

К сожалению, данный вид рекламы имеет свои недостатки. Прежде всего, если вы ожидаете отдачи от размещения рекламы на досках объявлений, нельзя останавливаться, единожды дав свое объявление. Практически все доски и форумы ограничивают время пребывания вашего сообщения в своей базе данных: как правило, от 7 до 20 дней. То есть, по истечении указанного срока ваше сообщение будет удалено. Однако, если в вашем учреждении существует сотрудник, отвечающий за рекламу сайта в Интернете, данный недостаток этого вида рекламы превращается в небольшое неудобство, заключающееся в том, что ему придется раз в 2—3 недели заново давать рекламу в форум или на электронную доску. Второй недостаток (который при желании руководства организации и интернет-маркетолога можно свести на нет) состоит в том, что досок объявлений и форумов в Интернете — сотни тысяч, и порой выбрать действительно нужный сервис не представляется возможным. Чтобы устранить эту проблему, необходимо помнить следующее.

Во-первых, не следует размещать рекламу на досках (форумах) или в их отдельных рубриках, не сходных по тематике с направлением деятельности вашей организации. Пользователь из Екатеринбурга, ищущий ответ на вопрос "Где найти недорогие курсы по компьютерной грамотности?", не станет искать информацию ни на электронных досках Владивостока или других городов, ни в форумах, посвященных любителям французского кинематографа. Зато он точно зайдет на форум по образовательным курсам, выберет из числа имеющихся тем рубрику "Компьютерные курсы" и будет просматривать существующие сообщения (на некоторых досках есть более сложный классификатор, позволяющий пользователю исключительно точно находить нужные сообщения за счет гибкой системы настроек запроса: по городу, по стране, по теме, по возрасту и пр.).

Во-вторых, прежде чем заносить информацию в форумы, нужно отобрать из числа всех досок, посвященных близкому вам вопросу, десять-пятнадцать

наиболее крупных и посещаемых. Согласитесь, что нет смысла давать рекламное сообщение в форум, открывшийся всего неделю назад и содержащий два-три сообщения. Про такой форум, скорее всего, никто не знает, кроме самих его создателей и, соответственно, ваше объявление до целевой аудитории не дойдет. Зато разместить рекламу на досках, имеющих несколько тысяч посещений в день и десятки тысяч сообщений по самым разнообразным темам, имеет большой смысл. Понять, какой форум заслуживает вашего внимания, а какой — нет, можно по количеству присутствующих тем, имеющихся сообщений и, разумеется, по счетчикам посещений, которые обычно стоят на всех досках и форумах.

В заключение разговора об электронных досках необходимо дать еще один совет: отсеяв из числа найденных по нужной тематике, например, пятьдесят не очень популярных форумов, не стоит совсем про них забывать. Не исключено, что через месяц-другой какая-нибудь из таких, ранее являющихся бесперспективными, электронных досок, как говорится, "встанет на ноги": усовершенствуется, обзаведется удобным поиском и обширной базой данных, став одним из наиболее популярных в своем роде интернет-сервисов. Поэтому сотрудник, занимающийся вопросами интернет-маркетинга в организации, должен постоянно отслеживать рынок этого вида сетевой рекламы.

Email-маркетинг

Многие западные эксперты утверждают, что отклик на электронные письма порой бывает гораздо выше, чем на баннерную рекламу или сообщения в форумах. Действительно, при умелом использовании и рациональном подходе к данному механизму, *Email-маркетинг* может стать мощным и весьма эффективным средством продвижения в Интернете. К числу значительных плюсов Email-маркетинга можно отнести: наличие электронного почтового ящика практически у всех пользователей (в отличие от Web-страниц), возможность организации прямых персонифицированных рассылок (Direct Email), четкое разделение и достижение целевой аудитории за счет списков рассылок и дискуссионных листов, возможность включения в письма не только текстовой, но и графической рекламы. Ниже остановимся более подробно на основных видах Email-маркетинга:

- прямые рассылки (Direct Email) — составление и рассылка персонифицированных сообщений является занятием весьма трудоемким, однако вы можете быть уверены, что время потратите не впустую, и письмо придет именно к тому человеку, который заинтересован в ваших услугах. Иначе, если вы попытаетесь привлечь таким образом человека незаинтересованного, ваше письмо будет воспринято как "электронный мусор" (спам) и удалено без прочтения;

- списки рассылок (Mailing Lists) — на данный момент в Интернете существует огромное количество почтовых рассылок, посвященных самым разнообразным темам. Такие рассылки бывают открытые и закрытые, платные и бесплатные. Если учесть, что последние версии браузеров могут в электронных письмах воспроизводить элементы HTML, то подписчики вместе с полезной информацией и текстовой рекламой могут получать и графические рекламные сообщения, т. е. баннеры;
- дискуссионные листы (Discussion Lists) — создаются для обмена информацией, обсуждения вопросов на определенную тему. В отличие от списков рассылок писать в лист могут не только непосредственно его создатели, но и все участники. Как правило, перед тем как сообщение рассылается всем участникам листа, оно проходит верификацию (проверку). Модератор листа (ответственный за содержание и функционирование листов) исключает сообщения, не относящиеся к тематике листа, следит за цензурой общения и т. д.

Контекстная реклама

Начиная с середины 2002 года, стал набирать скорость новый вид продвижения в Интернете — *контекстная реклама*. Суть контекстной рекламы заключается в том, что на определенных рекламных площадках (сайтах), имеющих поисковый механизм, по запросу пользователя выдаются только те рекламные объявления, которые удовлетворяют условию поиска. К примеру, если вы ввели слово "телефон", вам будет показана реклама операторов сотовой связи, а реклама стирального порошка — нет.

На начало 2004 года в российской части Интернета существовало три системы контекстной рекламы: Begun, Яndex.Direct и GoogleAdwords.

Обмен ссылками

Обмен ссылками обычно осуществляется на договорной бесплатной основе между схожими по тематике интернет-сайтами. В отличие от систем баннерного обмена, учет переходов и статистика такой рекламы не ведется, однако обмен ссылками порой дает более высокие результаты, нежели баннеры (например, когда посетители одного сайта хотят узнать другие ресурсы, аналогичные данному, они заходят в особый раздел, обычно называемый "Полезные ссылки" или "Друзья сайта").

Партнерские программы

Наличие рекламного текста на чужих сайтах — не редкость и не новинка. Во всяком случае, на западе этот вид сетевой рекламы развит уже давно и рас-

пространен почти повсеместно. Там этот вид чаще всего называют Affiliate Program, другими словами — партнерская программа. Такая программа подразумевает регистрацию в системе и получение *логина* с *паролем*, а далее вы время от времени (в зависимости от типа партнерской программы) получаете текстовые рекламные сообщения с определенной ссылкой, которые нужно поместить на страницах вашего ресурса. Действует программа так же, как и система баннерообмена: подсчитывается количество переходов по текстовым ссылкам на вашем ресурсе. Доступна передача информации и о вашем сайте. В России одними из промоутеров этого вида рекламы были Артемий Лебедев со своей первой текстовой баннерной системой tx3 и интернет-магазин "Озон". Но если последний предоставлял участникам своей партнерской программы рекламную информацию только о своих товарах, не предлагая возможности обмена данными, то изобретение Лебедева стало полноценной системой баннерообмена — только в текстовом формате. Единственный недостаток рекламных текстов как вида сетевой рекламы заключается в том, что иногда очень сложно разместить такую информацию, не повредив дизайну ресурса.

Спонсорство

Этот вид сетевой рекламы, к сожалению, не пользуется большой популярностью у нас в стране. И совершенно напрасно, т. к. спонсорские акции в Интернете открывают большие возможности развития, получения выгоды и проведения имиджевой рекламы (брендинг). Механизм проведения такой акции аналогичен спонсорским действиям в бизнес-модели, распространенной в реальной жизни: организация решает провести какое-то мероприятие (открытие нового сервиса, техническое переоснащение, организация распродаж и т. п.) и с этой целью просит другую организацию выступить в качестве спонсора и покрыть все или часть расходов на планируемое мероприятие. Взамен она готова на своем Web-сервере в течение определенного времени и с определенной частотой показывать рекламу (текстовую или графическую) этого спонсора. Данный вид рекламы хорош тем, что он не ограничивает спонсора рамками стандартов и форматов рекламных носителей. При размещении рекламы в баннерных сетях или на рекламных площадках необходимо иметь изображение формата 468×60, 100×100, 120×60 пикселов и пр. Обязательно стандарта GIF или JPEG. Размер файла рекламного носителя не должен превышать 12—15 Кбайт. Баннер нужно размещать только вверху страницы. И так далее. А в случае со спонсорством сам спонсор решает, какого формата и размера будет баннер, где его разместить, как часто его показывать и т. п. Другими словами, он полностью контролирует процесс показа своей рекламы.

Баннерная реклама

Несомненно, самым популярным и эффективным видом сетевой рекламы принято считать баннерные показы. Баннер (от англ. banner — знамя) является статичным или анимированным графическим изображением с установленной на него ссылкой, ведущей на тот или иной ресурс Сети. Баннерная реклама имеет два стратегических направления: участие в системах баннерного обмена и аренда рекламных площадей. Суть практически всех систем баннерообмена можно понять уже из их названия — обмен графическими рекламными баннерами. Вы, как владелец или ведущий какого-либо ресурса Интернета, регистрируетесь в такой системе, получаете логин и пароль для редактирования настроек показов и получения статистики, а также готовый код, который необходимо вставить в те страницы, на которых вы хотите видеть рекламу. Помимо всего прочего, вы можете включить в общий баннерообмен и свой баннер(ы). В принципе, это и есть основа системы обмена рекламными баннерами: вы показываете чужие баннеры у себя на страницах, а ваш баннер(ы) показывают на других ресурсах Сети. Если вас не устраивает участие в бесплатном баннерообмене, который, безусловно, не гарантирует, во-первых, 100%-ное достижение вашей целевой аудитории, во-вторых, необходимого количества показов в определенный период времени, можно воспользоваться услугой аренды рекламной площади. В качестве рекламных площадок в данном случае выступают наиболее посещаемые и авторитетные интернет-ресурсы.

Интернет-маркетинг как искусство продвижения проектов в Сети включает в себя немалый набор эффективных инструментов, позволяющих планировать, реализовывать и анализировать сетевые рекламные мероприятия. Это, как уже было сказано выше, Email-маркетинг, позволяющий вести персонифицированную электронную рассылку информационных и рекламных сообщений, — средство продвижения, в умелым руках профессионала способное творить чудеса, а не превратиться в банальный и формирующий негативный имидж рекламный мусор (спам). Это и работа по оптимизации сайтов для подготовки к индексированию их составных страниц поисковыми системами — мощный механизм, при верном использовании которого рутинная прописка метаинструкций превращает проект в один из самых востребованных ресурсов. Это и продвижение путем рекламы на форумах и электронных досках объявлений: при верном подходе к составлению рекламного текста, учете различных психологических, демографических, географических, тематических, социальных и прочих особенностей потребительской аудитории, четкой формулировке цели рекламного воздействия, умении кратко и, в то же время, исчерпывающе охарактеризовать объект рекламы, таланте убеждения потребителя, данный вид сетевой рекламы может значительно повысить посещаемость вашего интернет-проекта или закрепить в сознании потребителя благоприятный образ о вашем бренде, товаре или услуге.

Однако на первой стадии запуска интернет-проекта наиболее оптимальным средством продвижения в Сети является баннерная реклама. Попробуем разобраться, почему.

Финансовые затраты

При отсутствии практического опыта интернет-проектирования и сетевой рекламы разработчик, создавая свой первый проект, делает, что называется, "шаг в неизвестность", поэтому на первоначальном этапе определения степени востребованности его ресурса на интернет-рынке делать существенные денежные вложения в развитие проекта он не может (или вообще решает ограничиться методами, не требующими никаких финансовых затрат). С этой точки зрения баннерная реклама выступает в качестве "палочки-выручалочки" для разработчика, предоставляя широкий спектр рекламных возможностей при минимуме (или полном отсутствии) затрат и достаточно высокой эффективности.

Ресурсоемкость

Баннерная реклама имеет относительно невысокие требования к ресурсам, затрачиваемым руководителем интернет-проекта при продвижении, — человеческим и временным.

Для проведения эффективной баннерной рекламной кампании потребуется минимум специалистов: обязательно наличие маркетолога, который будет продумывать концепцию рекламного воздействия, разрабатывать сюжет и слоган баннера, анализировать психологические особенности восприятия рекламы, учитывать специфику целевой аудитории; и художника-дизайнера, в обязанности которого будет входить визуальная реализация графического рекламного носителя в соответствии с требованиями, разработанными маркетологом. Работа, связанная с выбором объекта размещения баннеров (рекламные площадки или баннерообменные сети), конфигурацией опций таргетинга, анализом эффективности рекламы, настройкой рекламодательских предпочтений, обычно тоже входит в компетенцию интернет-маркетолога. Кроме того, последние обязанности можно возложить на независимого специалиста (интернет-агентства, рекламные службы баннерообменных сетей и т. д.) — разумеется, при определенном финансировании.

Для сравнения с другими средствами рекламного воздействия в Сети в плане требований к человеческим ресурсам, приведем Email-маркетинг. Для эффективного функционирования этого механизма необходимы специалисты по следующим направлениям: поиск целевой аудитории и комплектование базы данных получателей рекламных рассылок (менеджер рассылок); определение смысловой окраски рекламного письма, составление текста сообщения, разработка слогана с учетом особенностей потребительского восприятия и спе-

цифики рыночного сегмента (маркетолог); грамматическая проверка рекламных текстов (это обязанность корректора, который, в отличие от баннерной рекламы, при Email-маркетинге обязателен: если баннер зачастую несет в себе минимум текстовой информации, то рекламные рассылки включают, как правило, не менее трех абзацев по 2—3 предложения в каждом); создание рекламных баннеров (это не опечатка: с появлением возможности пересылать по электронной почте не только текст, но и графику, рекламные рассылки взяли на вооружение часть средств баннерной рекламы для иллюстрирования текстов — в связи с этим потребуется помощь художника-дизайнера). Как видно из перечня некоторых функций Email-маркетинга, вам потребуется в два раза больше специалистов, нежели для работы с баннерами.

Время, как известно, — деньги. Баннерная реклама позволяет значительно сэкономить время, которое требуется для продвижения интернет-проекта, тем самым предоставляя возможность рекламодателю (руководителю рекламируемого проекта) направить усилия своих специалистов на усовершенствование самого интернет-ресурса. Процесс баннерной рекламы можно условно назвать единовременным, т. к. разработка концепции, сюжета, слогана баннера, его графическая реализация, выбор объекта размещения баннера и запуск его в ротацию (рекламную демонстрацию) происходит за определенный период времени, после чего рекламодателю в течение некоторого времени ничего не требуется делать, кроме получения данных по показателям эффективности разработанного баннера. А, например, при Email-маркетинге работа по поиску и комплектованию базы данных адресатов и организация рекламных рассылок продолжается непрерывно на протяжении всего периода проведения рекламной акции.

Технологическая база

Более выгодные условия в плане технологической базы добавляют еще один плюс в пользу баннерной рекламы. Она имеет большое преимущество перед другими видами рекламного воздействия в Сети. Какой технологической базой должен обладать рекламодатель? Той, которая позволит отслеживать процесс работы сетевой рекламы с получением статистики, и гибкого инструментария по управлению рекламными мероприятиями.

Возьмем, к примеру, снова Email-маркетинг: каким образом рекламодатель, используя этот рекламный механизм, может получить данные об эффективности воздействия на потребителя? Единственный показатель для него в этом случае — наличие или отсутствие прироста посетителей на сайте проекта (увеличение объема электронных продаж, количества зарегистрированных пользователей, числа заполненных в онлайне анкет и пр.) в период, сопутствующий и последующий рекламной акции по Email-рассылке. Однако такой показатель носит весьма условный характер, потому как прирост посетителей или увеличение продаж может быть простым совпадением по срокам прове-

дения рекламной рассылки. Например, прирост аудитории на сайте может быть вызван вспышкой интереса определенной части пользователей к услугам, предоставляемым данным проектом, по причинам, носящим сезонный характер (наплыв студентов и учащихся школ на серверы, содержащие банки рефератов и курсовых работ, в конце каждого учебного семестра или четверти вполне естественен и предсказуем). Что касается увеличения объемов электронных продаж на сайте проекта, то изменению кривой заключения сделок могли способствовать такие факторы, как повышение цен на аналогичные товары или услуги на других проектах или временное отсутствие подробного описания этих услуг (товаров) на других серверах.

И даже такое средство отслеживания отклика на рекламные рассылки, как специальные гиперссылки с идентификатором, ведущие на сайт проекта (по которым программа, установленная на сервере рекламируемого интернет-ресурса, подсчитывает количество переходов на сайт рекламодателя из рекламного письма), сегодня становится все менее эффективным. Причина кроется в психологических особенностях восприятия адреса ссылки пользователем, которые, в свою очередь, были сформированы неким собирательным негативным опытом интернет-аудитории. Этот опыт основан на случаях применения некоторыми рекламодателями нечистоплотных, с точки зрения этики, средств привлечения пользователей.

К примеру, какая ссылка вам будет более понятна и вызовет большее доверие:

`http://www.project.ru/services/freebooks.html`

или

`http://www.project.ru/cgi-bin/advert.pl?ID=2465&topic=744`

Скорее всего, первая, ибо она "говорящая": можно предположить, что, нажав на такую ссылку, пользователь попадет на сайт рекламодателя "Project", в раздел "Услуги", содержащий информацию о бесплатных книгах. Второй адрес не несет никакой смысловой нагрузки для пользователя, за исключением названия проекта. Такого рода ссылки использовали (и, к сожалению, по сей день используют) рекламодатели, стремящиеся заработать путем обмана пользователя. Реализуется такой обманный прием примерно следующим образом: на слоган типа "Бесплатные книги! Огромный выбор!" ставится ссылка, приведенная выше в качестве второго варианта, а в настройках программы-обработчика нажатий (в данном случае — это `advert.pl`) указывается путь перехода по ссылке, совершенно не соответствующей ее смысловому описанию (например, не страница с информацией о бесплатных книгах, а страница с исключительно коммерческим предложением). Поэтому подсознательно пользователь отдает предпочтение ссылке, по символам адреса которой можно догадаться, куда она его приведет.

Если же интернет-проект продвигается на рекламных площадях или в баннерообменных сетях (т. е. используется баннерная реклама), в распоряжение рекламодателя предоставляется полностью настраиваемое под индивидуального пользователя специальное программное обеспечение, предназначенное для анализа показателей эффективности, управления таргетингом и пр. Другими словами, если рекламодатель будет наблюдать приток посетителей на сайте своего интернет-проекта, выяснить, вызван он баннерными показами или другими факторами, не составит особого труда: надо будет зайти в специальный защищенный интерфейс на сервере баннерообменной сети и просмотреть статистические количественные показатели эффективности для баннеров, участвующих в ротации (если рекламная кампания проходит на сайте, сдающем рекламную площадь в аренду, рекламодатель получает эти данные через программу, установленную на локальной машине). Если, например, прилив посетителей длится неделю, и данные баннерных показов гласят, что в течение семи дней по баннерам было осуществлено 100 переходов на сайт рекламодателя (приведены абстрактные цифры), значит увеличение числа посетителей напрямую связано с действием баннерной рекламной кампании.

Далее более подробно рассмотрим некоторые особенности баннерной рекламы.

Форматы рекламных баннеров

Баннер — это графическое изображение фиксированного размера, носящее смысловое содержание информационного или рекламного характера и ведущее по гиперссылке на Web-сайт рекламодателя и/или его партнеров.

Раз баннер представляет собой графическое изображение, необходимо определение его размеров (ширины и высоты) в неких единицах измерения. Такой единицей принято считать 1 пиксел. Выбор пиксела в качестве общепринятой единицы измерения баннеров (как графических объектов) обусловлен, прежде всего, национальными предпочтениями измерения. Где-то используются сантиметры, где-то дюймы, поэтому стало актуальным приведение размеров к единой шкале — во избежание путаницы и неудобства при переводе одних единиц измерения в другие. В этом отношении пиксел сумел преодолеть национальный барьер, т. к. уже был взят на вооружение дизайнерами, верстальщиками и другими специалистами для определения размеров графических объектов. В результате не пришлось изобретать новый измерительный стандарт, что повлекло бы за собой временной период адаптации разработчиков и пользователей.

Первый вклад в формирование единой классификационной системы форматов рекламных баннеров был сделан Internet Advertising Bureau совместно с CASIE (The Coalition for Advertising Supported Information & Entertainment). Они предложили восемь баннерных форматов (табл. 17.1).

Таблица 17.1. *Форматы рекламных баннеров*

Наименование формата	Размер (в пикселах)
Full Banner	468×60
Full Banner with Vertical Navigation Bar	392×72
Half Banner	234×60
Square Banner	125×125
Button # 1	120×90
Button # 2	120×60
Micro Button	88×31
Vertical Banner	120×240

Описание форматов

Каждый из форматов имеет свою историю, свои достоинства и недостатки, о которых пойдет речь.

88×31/88×62 (Micro Button/Macro Button). В название данного подраздела не случайно выведено два различных формата. Дело в том, что из-за малого размера эти форматы не могли претендовать на статус баннерных, а предназначались, в основном, для создания так называемых кнопок — небольших графических изображений с логотипом или символикой конкретного интернет-проекта. Изначально такие кнопки делались авторами небольших тематических или контент-ресурсов для того, чтобы в случае заинтересованности их проектом любой желающий, имеющий в своем распоряжении Web-ресурс, мог поместить эту кнопку на одной или нескольких страницах своего сайта. Действительно, изображение столь маленьких размеров сложно отнести к полноценному рекламному баннеру, зато в качестве "лица сайта" формат подходит почти идеально: объема информации требует небольшого (краткое название ресурса и адрес в Интернете, либо одно из двух), доступность изложения — средняя, но вполне оправдывающая объем информационного наполнения (с учетом того, что формат 88×31/88×62 подразумевает минимум графического оформления), тип разверстки — горизонтальный, что соответствует критерию выбора рекламного формата, в отношении полосы прокрутки никаких проблем возникнуть не может (опять же, по причине маленьких размеров).

Если раньше кнопки перехода на интересный ресурс размещались на других сайтах исключительно в одностороннем порядке, то постепенно взамен размещения на своем сайте кнопки другого проекта авторы просили совершить ответную любезность — поставить кнопку их сайта на тот ресурс, баннер

формата 88×31/88×62 которого уже размещен у них. Со временем несистематическое размещение чужих кнопок на своих ресурсах прекратилось, и взаимный обмен стал происходить по определенному признаку — в основном, тематическому. К примеру, автор сайта, посвященного Национальной Баскетбольной Ассоциации США (NBA), предлагает автору ресурса, рассказывающего о баскетболе в целом, обменяться кнопками. Этот момент можно с уверенностью назвать одним из поворотных в истории использования формата 88×31/88×62: была достигнута важнейшая цель обмена информационными носителями — рекламная. И хотя о рекламе как таковой авторы подобных ресурсов тогда еще не помышляли, сущность обмена была в следующем: посетитель приходит на сайт, посвященный NBA, тщательно его изучает, копирует себе на локальный компьютер, однако, на этом интерес к теме не теряет, поэтому пытается обнаружить в пределах данного интернет-проекта информацию или ссылки на аналогичные (или близкие по тематике) ресурсы в Сети. И находит кнопку сайта о баскетболе, автор которого обменялся изображениями формата 88×31/88×62 с разработчиком ресурса об NBA.

Таким образом, путем взаимного размещения информационных носителей, которое изначально задумывалось как простой обмен любезностями между авторами похожих по тематической направленности интернет-проектов, было осуществлено элементарное рекламное воздействие на потребителя — в рассмотренном случае, на любителя игры в баскетбол.

Еще одним проявлением формирующейся баннерной рекламы выступило разделение сайтов, авторы которых обменивались кнопками, по тематическому признаку. Это весьма упрощенный вариант тематического таргетинга, т. е. системы гибких настроек рекламных показов по признаку темы. Конечно, в данном случае никакой речи о системе, тем более обладающей гибкими настройками, нет: однако подобное разделение явилось прототипом такой системы управления рекламными показами баннеров.

Впоследствии формат 88×31/88×62 стал постепенно уходить в тень, т. к. не мог противостоять развивающемуся огромными темпами формату 468×60, а за ним — и 100×100. В условиях роста популярности баннерной рекламы (которая на начальном этапе ассоциировалась только с форматом Full Banner) и ослабления заинтересованности авторов интернет-проектов в обоюдном размещении кнопок (такой подход с развитием больших форматов стал неоправданным и ненужным: большинство стало делать ставку именно на прямую рекламу), некоторые деятели Сети попытались вновь вернуть кнопкам 88×31/88×62 былую популярность, перенеся процесс обмена на современные рельсы. В связи с этим было создано несколько баннерообменных сетей, принимающих в ротацию рекламные баннеры формата 88×31. Старший брат-близнец Micro Button — формат 88×62 — еще раньше сошел с дистанции, и в дальнейшем принятым и повсеместно используемым стандартом остался

88×31 (это касается не только баннеров, но и прочих графических носителей, например, рейтинговых кнопок и счетчиков: если в самом конце 90-х годов XX века такие крупные проекты, как TopList, Aport Ranker и Rambler TOP100, предлагали своим пользователям для установки на страницы Web-сайтов сразу несколько вариантов кнопок — формата 88×31, 88×62 и др., — то к началу 2001 года формат остался всего один — Micro Button 88×31, а еще позже, помимо известных кнопок размером 88×31/88×62, появились другие варианты Micro Button).

Сегодня сетей, принимающих данный формат, не так уж много: они предлагают своим участникам набор стандартных возможностей: таргетинг, управление показами, статистика запросов и пр. Однако особой популярности сети данного формата не получили: разве что немногочисленные мелкие проекты обменивались кнопками между собой. Интернет-общественность вышла на иной уровень, где балом стали править рекламные форматы больших и средних размеров.

125×125/100×100 (Square Banner # 1/Square Banner # 2). Формат квадратных баннеров является, пожалуй, одним из самых обсуждаемых интернет-аналитиками и маркетологами. Дело в том, что он обладает целым рядом преимуществ и недостатков, в равной степени способных дать рекламодателю как положительные, так и отрицательные результаты проведения рекламного воздействия посредством баннеров.

Данный формат (а именно 125×125, т. к. его аналог 100×100 распространен, в основном, в российском сегменте Сети) появился как альтернатива формату Full Banner. В условиях роста популярности 468×60 необходимо было стандартизировать новый формат, обладающий определенными плюсами по отношению к Full Banner. Поэтому ставка была сделана на размер, а точнее — на компактность нового формата.

120×60 (Button # 2). Формат Button # 1 (120×90) мы рассматривать не будем по причине того, что он в российской части Интернета практически не встречается, и тенденций к его развитию и популяризации пока не наблюдается. Чего никак нельзя сказать о формате 120×60, который достаточно развит и по популярности уступает лишь форматам 468×60 и 100×100.

120×240 (Vertical Banner). Вертикальный формат 120×240 также почти не распространен. Зато, начиная с 2002—2003 гг., ему на смену пришло большое количество других вертикальных форматов: 120×400, 120×600, 200×600 и т. д. Такие баннеры получили название "небоскребов".

468×60 (Full Banner). Наиболее эффективным, популярным и повсеместно используемым форматом в баннерной рекламе по праву считается Full Banner (468×60). В качестве доказательства может послужить факт, согласно которому большинство существующих сегодня баннерообменных сетей и рек-

ламных площадок работают с данным форматом. Верстка различных сайтов происходит с учетом показов баннеров именно размером 468×60, основная работа дизайн-студий по разработке рекламных носителей тоже связана с этим форматом и т. п.

Чем же вызвана такая высокая популярность? Во-первых, Full Banner полностью отвечает требованиям всех параметров рекламных форматов: размеры позволяют размещать необходимый объем информации и делать рекламные сообщения доступными для восприятия, а горизонтальная разверстка в несколько раз больше, чем у других форматов. Во-вторых, формат 468×60 способен удовлетворить стремление многих дизайнеров в полной мере показать свое мастерство: на изображении шириной почти в 470 пикселов есть, где размахнуться. В-третьих, распространенность Full Banner в какой-то мере связана с его появлением на рекламном поприще одним из первых. Этот формат с полной уверенностью можно назвать стандартом общепризнанным, повсеместно используемым, к тому же позволяющим получать высокий отклик от потребителей в процессе рекламного воздействия в Интернете.

Типы рекламных баннеров

Графические стандарты. На сегодняшний день для создания графических носителей в Сети используется три основных стандарта: GIF, JPEG и PNG. Вы с ними хорошо знакомы.

Rich-media

Лет десять назад дизайнер был сильно ограничен в своем стремлении донести до потенциального потребителя как можно больше полезной информации: поддерживался формат JPEG и такой же статичный GIF89A. Позже появился новый стандарт графики для Сети — PNG, который позволял использовать достаточно большое количество цветов и выдавал при этом маленький исходный размер файла. Однако в то время баннерная реклама как основополагающее средство сетевого рекламного процесса уже не могла ограничиваться только маленькими размерами исходного файла изображения: назревает практическая необходимость появления динамики в рекламе. Разработанный Compuserve стандарт GIF89A, позволяющий применять технологию чередования кадров, предоставил рекламному дизайнеру широкие возможности по работе с анимационными роликами. Технология анимационного стандарта GIF в несколько раз повысила информативность баннера, сделала рекламные носители динамичными и легко запоминающимися. Эти критерии гораздо больше привлекали внимание пользователей, тем самым повышая эффективность отдачи от рекламных кампаний.

Но жизнь не стояла на месте: появлялись новые технологии, Интернет менялся, менялись сами пользователи, а заодно — и их требования к рекламе в Се-

ти. Простая цикличность кадров рекламного изображения вскоре стала давать обратный эффект: все чаще и чаще стали появляться баннеры с интенсивно мигающими, резко двигающимися и прыгающими элементами, что, само собой разумеется, мало кому нравилось. Кроме того, появилась мода на ОС-бутафорию (рис. 17.1) в контексте представления рекламной информации: почти каждый уважающий себя дизайнер считал обязательным включение в визуальную структуру баннера псевдокнопок управления операционных систем, полос прокруток, ниспадающих системных меню, пиктограмм, курсоров и т. д. и т. п.

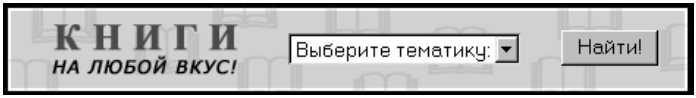


Рис. 17.1. Пример баннера с использованием псевдоэлементов ОС

Можно представить, каков был показатель эффективности таких баннеров на начальной стадии данной моды (и развития баннерной рекламы вообще)! Но именно только на начальной стадии. Скоро пользователь понял, что его обводят вокруг пальца, предлагая немного прокрутить вниз системное окошко с забавным анекдотом или ввести свой знак зодиака для ознакомления со свежим гороскопом. К сожалению, даже сегодня многие дизайнеры (или люди, придумывающие сюжет баннера) этого не понимают и продолжают с упрямством конвейера "лепить" ложные ОС-ориентированные баннеры. Возможно, делается ставка на новичков в Интернете, на тех, кто еще не освоил все премудрости и "подводные камни" Сети. Но владельцы сайта, считающие подобных пользователей основными потребителями рекламируемого таким образом контента, не учитывают банальной истины: разозлив человека, они делают его умнее.

Однако, на наш взгляд, повальный бум на использование ложных ОС-элементов имел и положительную сторону, т. к. приблизил определенный круг людей к гениальной мысли о том, что пользователю нужна РЕАЛЬНАЯ интерактивность. Несомненно, какая бы она ни была, интерактивность между теми, кто дает рекламу, и теми, кто является ее потенциальным потребителем, существенно может повысить показатели эффективности. Даже если такое взаимодействие осуществляется на невербальной основе, а именно — посредством программного обеспечения. Поняв, какие перспективы их ожидают, разработчики стали разрабатывать новые Web-ориентированные технологии и, между делом, интегрировать существующие интерактивные механизмы с развивающимися сетевыми рекламными процессами.

В результате появилась система механизмов, на английском емко и лаконично нареченная rich-media, а у нас получившая довольно расплывчатое поня-

тие "интерактивных рекламных технологий" (также существуют и иные определения). Rich-media представляет собой набор программных средств, обеспечивающих полное взаимодействие рекламодателя (на уровне серверных приложений) и потребителя рекламы. И все это в одном-единственном баннере! Rich-media-механизмы принято делить на три вида: Java, Flash и HTML-ориентированные. Далее более подробно остановимся на каждом их них.

Java-баннеры. По сути, Java-баннеры представляют собой обычные Java-апплеты, встраиваемые в HTML-документы. Чтобы такой баннер корректно отображался в браузере пользователя, необходима специальная программная надстройка (plug-in), которая на сегодняшний день входит в состав всех современных программных пакетов для работы в Интернете. Большими функциональными возможностями Java-баннеры не обладают (простейшая манипуляция объектами, редактирование отдельных элементов, чувствительность к поведению пользователя — щелчок мышью, нажатие клавиши на клавиатуре и пр.), однако нередко требуют много времени на загрузку (ввиду неуемного стремления многих разработчиков Java-баннеров "напичкать" свой рекламный носитель всякого рода эффектами), в связи с чем многие интернет-пользователи просто-напросто выключают опцию поддержки Java-апплетов в браузере.

Практика показывает, что преобладающее количество Java-баннеров — игровой, развлекательной тематики. К примеру, самый первый Java-баннер от компании Hewlett-Packard позволял пользователю сыграть в пинг-понг в пределах рекламного носителя. Такой подход, безусловно, повышал интерес интернет-аудитории к сетевой рекламе как таковой, но, несмотря на достаточно высокий элемент интерактивности, основополагающие задачи, стоящие перед рекламодателем в определенном сегменте рынка, — социологические исследования, опросы клиентов, проведение рекламных и PR-акций с целью привлечения потребителей, а главное, комфорт в процессе взаимодействия с людьми, — с помощью Java-технологии выполнены не были.

Flash-баннеры. Технология Flash Shockwave, разработанная компанией Macromedia, в корне изменила представление пользователей и рекламодателей в отношении интерактивного баннера. Возможности новой технологии были необычайно широкими и позволяли потребителю взаимодействовать с Flash-баннером так, как ему (потребителю) было нужно. Управление звуком (предопределенные звуки и Streaming Sound, загружаемые со стороны сервера по мере необходимости), возможность использования не только растровой, но и векторной графики, вставка форм ввода и обработки информации и последующего ее отправления рекламодателю, высокая чувствительность к манипуляциям мышью и клавишами клавиатуры, возможность динамического

изменения графического и текстового содержания баннера, организация полностью настраиваемой самим пользователем навигации по различным ссылкам — это далеко не весь перечень функциональных возможностей Macromedia Flash.

Казалось бы, идеал достигнут: чего больше можно ожидать от рекламного баннера? Но разработать новую технологию — это одно, а убедить пользователя перейти на нее — совершенно другое. По словам Мариуса Меланда, аналитика Forbes.Com, в первое время после появления технологии Macromedia большинство наиболее популярных интернет-служб и контент-серверов США (включая таких гигантов, как портал Yahoo!) отказались поддерживать новый тип рекламных носителей, мотивируя это тем, что переход на такую технологию, как Flash, вызовет недовольство пользователей, столкнувшихся с проблемой отображения подобной рекламы.

Проблема содержала два основных аспекта: поддержка Macromedia Flash браузерами и психология пользователя. Первый заключался в том, что при загрузке Web-страницы, на которой содержался Flash-ролик, взору пользователя представлялось небольшое окно, которое гласило примерно следующее: "Ресурс, который вы хотите просмотреть, содержит компоненты Macromedia Flash". Далее предлагалось установить программу-интерпретатор с Web-сервера компании-разработчика на компьютер пользователя. И вот здесь актуализируется второй аспект данной проблемы — психология восприятия пользователем необходимости установки дополнительного ПО на свой компьютер.

Во-первых, переход на новые технологии (какими бы они ни были совершенными) по-прежнему вызывает у многих недовольство (основные аргументы против перехода следующие: "что мне лично даст этот Flash, если я знаю два десятка сайтов, которые я посещаю, и безо всякой рекламы?", "если здесь нужна установка дополнительного ПО, лучше я подыщу другой сайт", "переход на технологию, подобную Flash, потребует дополнительных ресурсов для моего компьютера", "завтра придумают еще что-нибудь более эффектное, и мне снова придется пичкать свой браузер модными штучками", "я боюсь устанавливать ПО из Сети, а на лицензионную программу у меня нет денег").

Во-вторых, загрузка Flash-баннеров требует определенного времени, тогда как множество пользователей Сети (особенно в развивающихся странах, в том числе, в России) считают каждую минуту своего пребывания на просторах Интернета. В-третьих, постепенно перед потребителем сетевой рекламы вырисовывается практическая необходимость перехода к широкополосному доступу. Многие не имеют возможности или не видят необходимости в такой, надо сказать, дорогостоящей перемене. По оценке исследовательской компании Jupiter Communications, 77% пользователей Интернета в 2003 году все еще будут входить в Сеть, используя коммутируемые линии.

Все вышеперечисленное можно в какой-то мере назвать предрассудками, однако, с мнением и желанием пользователей необходимо считаться и стараться на практике показать им, какие перспективы открываются перед ними в случае перехода на новую технологию. А излишняя самоуверенность рекламодателя в данном случае может привести к тому, что пользователь нажмет кнопку отмены на инсталляционном окне Macromedia Flash и в результате не увидит рекламного объявления.

HTML-баннеры. Они, с точки зрения дополнительных требований к ПО компьютера пользователя, представляют оптимальный вариант для консервативных или недоверчивых потребителей. Чтобы просмотреть такой баннер, пользователю не нужно устанавливать никаких plug-in, а время, необходимое для загрузки HTML-баннера на странице, сведено к минимуму.

HTML-баннер представляет собой традиционное графическое изображение в формате JPEG или GIF, но в отличие от простого баннера состоит из нескольких объектов: часть которых — графика, часть — интерактивные HTML-формы, обрабатываемые специальным программным обеспечением (CGI/Perl, ASP, PHP и др.). Причем формы, самые, что ни есть, настоящие. Не псевдополосы прокрутки и ниспадающие системные меню, а реальный набор функций, позволяющий пользователю взаимодействовать с рекламодателем. Другими словами, CGI-баннер — это группа обычных HTML-тегов, наряду с графическими объектами, реализующих интерактивные механизмы.

Итак, никаких требований к программному обеспечению и времени загрузки. Но что тогда могут предложить HTML-баннеры взамен? А предложить они, честно говоря, могут достаточно много: организация навигации по баннеру (пользователь самостоятельно выбирает пункт назначения по конкретной ссылке), занесение информации, проведение всевозможных анкетирований, опросов и голосований, вывод запрошенной пользователем информации прямо из кода баннера и многое другое.

Баннерный обмен

Баннерный обмен является оптимальным выбором владельца Web-сайта на первом этапе существования его интернет-проекта или продвижения в Сети конкретных товаров или услуг. Причина тому — возможность бесплатного обмена рекламными носителями. Чтобы понять суть рекламного обмена в баннерных сетях, нужно остановиться на таких важнейших понятиях, как комиссия обмена, разделение на подсети, перевод и накрутка показов, таргетинг, "подъемные" показы, покупка показов и CTR.

Комиссия обмена. Ее можно назвать основополагающим фактором функционирования практически любой баннерообменной сети. Так как большинство баннерных сетей не берет с участника (рекламодателя) плату за осуще-

ствление обмена рекламой с остальными участниками сети, с каждого аккаунта (пользовательского счета рекламодателя в конкретной баннерной сети) за рекламные показы взимается определенный процент комиссии (который и служит, в некотором роде, оплатой рекламного обмена). Обмен подразумевает обоюдное размещение рекламы, поэтому если на сайте одного участника сети было показано 10 баннеров, то на сайтах других участников этой же баннерной сети баннеры первого участника будут продемонстрированы, например, 8 раз. Разницу показов, которая и называется комиссией обмена, баннерная сеть забирает себе. На величину процента комиссии может влиять целый ряд факторов: видимость баннера на страницах сайта-участника сети, самовольное изменение кода, присутствие разделителей между несколькими баннерами, указание под баннером названия баннерной сети со ссылкой на нее, расположение баннеров сети на странице сайта-участника относительно баннеров других сетей обмена рекламными носителями и др.

Перевод и накрутка показов. Функция перевода накопившихся рекламных показов — весьма полезный инструмент баннерной сети. В каких случаях перевод показов может пригодиться участнику системы баннерного обмена?

1. Если участник регистрирует новый аккаунт в пределах той же баннерной сети (по определенным причинам), он может перевести на только что созданный счет все показы, которые были накоплены им на старом аккаунте.
2. Если у интернет-проекта, участвующего в баннерном обмене, меняется владелец, последний вправе сохранить оставшиеся от предыдущего владельца показы в пределах конкретной баннерной сети и даже аккаунта.
3. Если баннерная сеть имеет поддержку конвертации показов, участник одной подсети может перевести накопленные показы в другую подсеть без регистрации дополнительного аккаунта.
4. Баннерные показы могут выступать в качестве ликвидного средства оплаты товаров или услуг. Например, автору статьи можно зачислить на его аккаунт в баннерной сети гонорар в виде определенного количества показов, участник интернет-конкурса может получить приз таким же образом и т. д.
5. Владелец интернет-проекта, который в скором времени будет закрыт, имеет в определенной баннерной сети несколько тысяч накопленных показов. А это и расширение потребительской аудитории, и охват других сегментов рынка, и возможность получить материальную выгоду от рекламных показов, наконец, перспектива продажи накопленных показов другому заинтересованному лицу. Терять такую возможность не хочется, поэтому до закрытия проекта показы либо переводятся на аккаунт другого проекта (партнера, клиента, спонсора, просто хорошего друга и т. п.), либо выстав-

ляются на продажу. В последнем случае, когда покупка накопленных рекламных показов осуществлена, они переводятся на указанный покупателем аккаунт, а старый счет, зарегистрированный проектом-продавцом, закрывается.

Накрутка рекламных показов в баннерных сетях — это большая проблема на современном интернет-рынке рекламы, с которой ведут борьбу лучшие отечественные и зарубежные специалисты. Суть накрутки заключается в искусственном наращивании рекламных показов для конкретного аккаунта. В настоящее время такие мощные и наиболее используемые сегодня рекламные механизмы (на базе которых происходят баннерные показы), как RotaBanner и BannerBank, имеют в своем арсенале эффективные средства борьбы с накрутками рекламных показов.

Таргетинг. Это целевое (от англ. *target* — цель), сфокусированное рекламное воздействие на потребителя. Различается таргетинг *прямой* и *обратный.*

Прямой таргетинг позволяет определять сайты, на которых рекламодатель хочет размещать свои баннеры. Существует несколько типов прямого таргетинга:

1. Тематический таргетинг — наиболее используемый тип целевого воздействия на потребителя, в основе которого лежит тематическая направленность рекламируемого ресурса (товара и/или услуги). Например, существует некий сайт, посвященный пчеловодству, владелец которого принимает решение начать рекламу своего интернет-проекта. В этом случае нет смысла показывать баннеры сайта по пчеловодству на развлекательных ресурсах и серверах знакомств, порталах мобильной связи и автомобильных галереях. Зато было бы уместно разместить рекламу в соответствующих тематических разделах крупных каталогов ресурсов или на страницах общепопулярных служб (почтовые серверы, к примеру), доступных тысячам и миллионам пользователей Интернета ежедневно (среди которых вполне могут оказаться люди, увлекающиеся разведением пчел).

   Однако такой точности рекламного воздействия на потребительскую аудиторию можно достичь только при узкой специализации рекламируемого интернет-проекта. Если вы рекламируете новое лекарство от простуды, тематический таргетинг уже не поможет, т. к. простуженные люди посещают ресурсы любой тематики.

2. Адресный таргетинг — его можно условно назвать более точным вариантом таргетинга по тематике. В этом случае рекламодатель может самостоятельно выбрать из списка участников баннерной сети те сайты, на которых его баннеры не должны будут показываться. Определение неугодных ресурсов идет по физическому адресу в Интернете или номеру

аккаунта. К сожалению, этот тип таргетинга не всегда бывает эффективным и оправдывающим затраченные усилия.

3. Географический таргетинг — в основу этого типа фокусировки рекламы в Сети положен географический признак, т. е. возможность позиционировать баннеры в зависимости от конкретной страны, области, округа, города и т. д. Например, фирма, предоставляющая услуги доступа к Интернету, имеет офисы в Москве и Санкт-Петербурге и обладает технической базой и финансовыми возможностями для того, чтобы осуществлять подключение только в пределах двух столиц и прилегающих к ним областей. Поэтому для нее нет никакого смысла воздействовать на жителей Новосибирска или Киева, даже если это их потенциальные потребители. В связи с этим, фирма разместит рекламу на московских и питерских интернет-ресурсах. И здесь есть свои "подводные камни".

   Во-первых, не факт, что на ресурсы двух столиц, где будет размещена реклама этой фирмы, будут приходить исключительно жители Москвы и Петербурга. С таким же успехом туда может заглянуть житель Ростова-на-Дону, собирающийся в скором времени переехать на постоянное место жительства в столицу России и работать в крупной дизайн-студии и поэтому интересующийся расценками столичных интернет-провайдеров. А если наоборот — фирма работала бы в Ростове-на-Дону и давала рекламу только на город и Ростовскую область, проявление "подводного камня" географического таргетинга тоже налицо: какой-нибудь петербургский аналитик или журналист в целях подготовки финансового отчета по интернет-провайдерам России обязательно бы увидел рекламный баннер этой фирмы, исследуя ростовский рынок интернет-технологий.

   Во-вторых, фокусировка по географическому признаку не принимает во внимание возможные территориальные перемещения потребителя. Например, некий рекламный баннер объявляет о начале грандиозной распродажи меховой и кожаной одежды в Санкт-Петербурге с 15 октября по 25 ноября, и, в соответствии с настройками географического таргетинга, реклама демонстрируется в самом Петербурге, Ленинградской области и, вероятно, в Москве (что вполне возможно в силу небольшого расстояния между Москвой и Петербургом и максимальной концентрации обеспеченных людей в этих двух городах). Люди, проживающие в этих регионах и являющиеся потенциальными покупателями меховой и кожаной одежды, могут в период с 15 октября по 25 ноября (что составляет целых 40 дней), во-первых, уехать отдыхать (например, в Крым), во-вторых, отправиться в деловую поездку или командировку (к примеру, на Урал), в-третьих, решить навестить родных (где-нибудь в Сибири) и т. д. Во всех этих случаях потребители, возможно, имеющие определенный интерес к организованной распродаже (чисто теоретически, т. к. точно о ней знать они не могут),

не увидят рекламные обращения компании-организатора рекламной распродажи.

4. Временной таргетинг — данный тип фокусировки рекламных показов подразумевает воздействие на потребителя в зависимости от числа месяца, дня недели и времени суток. Например, один баннер можно показывать только первого числа каждого месяца (информация о продлении скидок на товары или услуги, напоминание о сроках действия определенного сервиса и т. д.), другой будет демонстрироваться в течение уик-энда (анонс о событиях, происходящих в субботу и пятницу), третий баннер помещается в ротацию исключительно в обеденный перерыв (предложение принять участие в дневном конкурсе от радиостанции) и т. д. Рассмотрим немного подробнее каждое из направлений временного таргетинга:

   - число месяца — как известно, в месяце 30 или 31 день (28/29 для февраля), каждый из которых может быть чем-то примечателен для потребителя: федеральный или народный праздник, приметы и др. Например, назначение показов рекламных баннеров различной направленности на Международный женский день (8 марта) и Пасху (одно из воскресений весны) может дать соответствующие результаты при определенной стратегии воздействия на потребителя. Нужной эффективности рекламодатель может достичь, если решит обыграть ту или иную народную примету в контексте представления рекламной информации (например, Покров: если 14 октября выпадает снег, зима будет снежной, идет дождь — дождливой, а если осадков нет, будет теплая сухая погода);

   - день недели — неделю условно можно поделить на две части: рабочую и уик-энд. Разумеется, для различных категорий людей понятия "рабочей недели" и "выходных" могут существенно различаться. Но, в целом, принято считать рабочими днями понедельник, вторник, среду, четверг и пятницу, а выходными — субботу и воскресенье. Поэтому показывать пользователю в середине рабочей недели рекламный баннер, приглашающий посетить то или иное увеселительное мероприятие, будет малоэффективным рекламным ходом. Аналогичным неудачным примером может послужить реклама вечерних бизнес-семинаров по будням, демонстрирующаяся потребителю в уик-энд;

   - время суток — в дневное время пользователи выходят в Интернет, в основном, со своего рабочего места, ближе к вечеру и ночью, когда расценки на доступ к Сети существенно ниже, — выход в Сеть осуществляется с домашних компьютеров. Также можно предположить, что в течение рабочего дня пользователи посещают Интернет в рабочих целях (поиск информации, работа с сайтом, анализ рынка, исследование

аудитории, чтение и отправка электронной корреспонденции и т. д.), а в обеденный период (в среднем, с 12 до 15 часов) пользуются ресурсами Сети для отдыха и развлечения (чтение новостей, анекдоты, игры, знакомства и пр.). Кроме этого, во многих организациях рабочий день начинается с чтения электронной почты или анализа онлайновых СМИ: утренние часы (в среднем, с 9 до 11 часов) могут принести рекламодателю дополнительную эффективность при соответствующей фокусировке по временному признаку.

- Разумеется, оценка эффективности баннерных показов в соответствии с настройками временного таргетинга носит относительный характер, т. к. нельзя привести поведение людей (в данном случае — потенциальных потребителей) к единому знаменателю. Одни отмечают определенные праздники и знают народные приметы, другие — нет; одни работают пять дней в неделю, а два отдыхают, другие — трудятся посменно; одни пользуются Интернетом бесплатно на рабочем месте, а дома за деньги, другие — наоборот (рабочий трафик ограничен, домашний доступ осуществляется по знакомству на бесплатной основе); одни читают электронную почту и новости утром и посещают развлекательные серверы в обеденный перерыв, другие — делают это, соответственно, вечером и утром и т. д. и т. п.

5. Программный таргетинг — это возможность фокусировать воздействие рекламных обращений рекламодателя в зависимости от программного обеспечения пользователя (потребителя рекламы). В данном случае под программным обеспечением подразумевается операционная система и браузер.

   Известно, что многие элементы структуры Web-страниц по-разному отображаются в различных операционных системах. Например, параметры вывода размера и гарнитуры шрифта для ОС Microsoft Windows не подойдут для корректного отображения в среде Mac OS. Этот аспект чрезвычайно важен, например, при демонстрации HTML-ориентированных баннеров, структура которых включает текстовые блоки информации. С браузерами ситуация еще более сложная: это программное обеспечение для просмотра Web-страниц, в зависимости от модели и версии, может либо некорректно отображать, либо вообще не показывать рекламные баннеры, не поддерживающие различные технологии и спецификации ("плавающие" и обычные фреймы, GIF-анимация, каскадные таблицы стилей CSS, Macromedia Flash и пр.).

   Поэтому, зная, что, к примеру, Netscape Navigator плохо поддерживает стилевые шаблоны CSS, можно первый баннер, использующий эту спецификацию, запустить в ротацию только для пользователей Microsoft Internet Explorer версии 4.x и выше, а второй видоизменить с сохранением общей

рекламной идеи для соответствия стандартам Netscape Navigator и настроить его показы по определенному критерию программного таргетинга.

Единственный недостаток этого типа фокусировки рекламных показов — высокие требования к знанию всех особенностей операционных систем, моделей и версий браузеров. Далеко не каждый рекламодатель знает, в чем разница между Opera Internet Browser и WebTV, MS Windows NT и Red Hat Linux. С другой стороны, не всякий интернет-маркетолог (дизайнер) обладает подобной информацией.

6. Языковой таргетинг — фокусировка по языковому признаку подразумевает собой показ рекламных баннеров на страницах интернет-ресурсов, содержащих информацию на определенном языке. Например, если действие рекламной акции нацелено исключительно на жителей Испании, то было бы целесообразно представить рекламную информацию в испанском секторе Сети. Таким образом, если цель рекламной кампании не включает воздействие на второстепенную часть потребительской аудитории — например, жителей Испании, родившихся за пределами страны и не знающих национального языка, этот тип таргетинга может заметно повысить эффективность баннерных показов рекламодателя.

   Здесь, к сожалению, тоже имеется слабая сторона. Во-первых, чтобы достичь оптимального уровня эффективности при использовании языковых фокусировок, необходимо знать национальные особенности целевой аудитории, соотношение числа прямых потребителей к количеству второстепенных, демографическую и социальную статистику государства (штата, региона, округа и пр.), в сегменте Сети которого будет осуществляться рекламное воздействие (такое знание требует серьезной предрекламной подготовки). Во-вторых, при составлении рекламных сообщений на HTML-носителе необходимо придерживаться специфических языковых кодировок, благодаря которым текстовые блоки рекламного баннера будут корректно отображаться (не каждый браузер имеет мультиязыковую поддержку).

"Подъемные" показы. Когда в баннерной сети регистрируется новый участник, на его аккаунт, как правило, переводится определенное количество так называемых "подъемных" показов. Делается это для того, чтобы дать возможность недавно зарегистрированному ресурсу освоиться в сети, войти в ее рекламный темп, дать возможность для развития.

Покупка показов. Регистрация в баннерной сети и получение персонального аккаунта может подразумевать не только обмен рекламными носителями при определенном проценте комиссии на бесплатной основе, но и коммерческое приобретение рекламных показов. В последнем случае рекламодателю не требуется размещать на сайте своего интернет-проекта специальный

код для отображения баннеров других участников сети. Он просто выкупает необходимое число показов и может распоряжаться ими по своему усмотрению. Возможность подробной статистики и различных типов таргетинга в случае коммерческого приобретения показов обязательна. Некоторые баннерные сети предоставляют услугу фокусировки рекламного воздействия в расширенном режиме только на коммерческой основе.

CTR. Click-Through-Ratio — это отношение числа нажатий на рекламный баннер к количеству показов на сервере баннерной сети. Например, если баннер был показан 100 раз, и 5 человек нажали на него, CTR будет составлять 5%. CTR является одним из наиболее важных показателей эффективности рекламного воздействия. С позиции баннерной сети CTR рассматривается как некий усредненный показатель отклика (эффективности рекламы) на баннеры всех участников (аккаунтов) системы рекламного обмена. Казалось бы, чем выше показатель CTR, тем эффективнее рекламный баннер, однако, если данный показатель меньше 1%, это не значит, что рекламное воздействие было осуществлено впустую. И наоборот, если CTR равняется 10—15%, это еще не является доказательством высокой эффективности баннера.

Создание эффективного рекламного баннера

В этом разделе приводится двадцать рекомендаций по созданию запоминающихся эффективных рекламных баннеров. Эти рекомендации охватывают все аспекты разработки рекламных баннеров: от проектирования и создания (слоган, графика, анимация) до редактирования существующих носителей (изменение объектов, пространственное расположение) и вариантов их размещения.

1. Размер рекламного баннера. Под размером рекламного баннера может подразумеваться формат носителя (показатели ширины и высоты баннера) и объем конечного файла рекламного объекта.

Ранее уже было дано обоснование вариантов использования различных рекламных форматов, в соответствии с которым применение широкоформатных носителей представляет собой оптимальный способ организации рекламных показов. Большой формат дает ряд преимуществ: оперирование гораздо большей рекламной площадью, возможность повышения общей информативности рекламы, наличие определенного разнообразия оформительских и инструментальных (rich-media) средств и пр. Исходя из этого, реклама, размещенная на носителях больших размеров (468×60 и др.), как правило, дает больший отклик, чем размещенная на малоформатных носителях.

Что касается размера конечного файла рекламного баннера, то его, по возможности, необходимо сократить. Причина, по которой следует к этому стремиться, проста: чем больше размер файла рекламы, тем дольше будет

загружаться рекламный баннер. Поэтому, независимо от визуальной приглядности баннера и верно определенной стратегии его показов на рекламных площадках Интернета, пользователь, даже будучи заинтересованным в товаре/услуге, информацию о которых содержит загружающийся баннер, вряд ли станет дожидаться окончания загрузки графического объекта. В этом отношении рекламодателю придется идти на определенные уступки, если он делает ставку на GIF-анимацию (чтобы достичь максимального эффекта, надо будет сделать анимированный баннер динамичным, живым, детальным, что зачастую сопряжено с применением большого количества анимационных кадров). Это, в свою очередь, сильно влияет на конечный размер файла анимированного рекламного баннера. Так же осмотрительно следует поступать и рекламодателю, решившему разместить Flash-баннеры: определенная часть аудитории не сможет или не захочет увидеть его рекламное сообщение *(см. главу 5)*.

2. Графические изображения. При разработке рекламного баннера нельзя ограничиваться только текстом. Удачно составленный слоган — это уже 50% успеха, но чаще всего пользователь обращает внимание на графическое оформление. Оптимальным решением будет разумное использование и графических объектов, и текста: удачно подобранное изображение сможет органично вписаться в компоновку баннера и дополнять смысловое содержание слогана. Под графическими изображениями подразумеваются фотографии, рисунки, схемы, диаграммы и прочий иллюстративный материал. При использовании фотографий рекомендуются портреты (в особенности — женские): это создает ощущение реального образа на баннере. Жанр фотографических изображений не имеет значения за исключением баннеров по художественной тематике. Можно сформулировать ряд общих рекомендаций по выбору фотографий:

- качество — мало кому нравится нечеткое или размытое изображение в телевизоре, так же и с рекламными баннерами: следует выбирать яркие, четкие, детализированные, другими словами — качественные фотографии. Конечно, далеко не каждому доступно применение цифрового фото, однако даже с помощью обыкновенного фотоаппарата-"мыльницы", сканера и специального программного обеспечения по оптимизации отсканированных изображений (например, PhotoPlus) можно получить неплохие фотографии. Если фотоаппарата у вас нет, можете воспользоваться многочисленными фотогалереями в Интернете или на компакт-дисках, которые продаются практически в любом магазине;

Примечание

В последнее время участились случаи, когда разработчики умышленно делают те или иные элементы рекламного баннера нечеткими, размытыми, некачест-

венными для привлечения внимания пользователей. Стоит признать, что в ряде случаев такой подход может быть эффективным, однако повсеместное применение не даст нужного результата.

- авторское право — если вы хотите воспользоваться изображениями, взятыми из Интернета, во-первых, следите за предупреждением о сохранении авторского права (многие фотографические изображения охраняются "Законом РФ об авторском праве и смежных правах" и подразумевают только некоммерческое использование; т. е. если, например, фотография Лондона, снятого с высоты птичьего полета, будет использоваться на баннере туристической компании, такое использование будет считаться коммерческим, т. к. компания станет привлекать с помощью фотографии новых клиентов, а значит — получать определенный доход; если вы включите это изображение в баннер, который будет рекламировать персональную страницу или авторский информационный проект, посвященный крупнейшим городам мира, то такое использование будет считаться некоммерческим);
- индивидуальность баннера — применяя изображения, найденные на многочисленных интернет-ресурсах и в коллекциях, собранных на CD, стоит помнить о том, что использованное на вашем баннере изображение с таким же успехом может появиться в рекламе других сайтов, ибо раз оно свободно для доступа в Сети, и вы смогли его отыскать, почему его не сможет найти кто-либо другой? Создавая баннер, необходимо стремиться к тому, чтобы он стал исключительным, особенным в своем роде, подчеркивал некую индивидуальность рекламируемого сайта, которая будет проходить "красной нитью" через все рекламные носители. Если вы опасаетесь возникновения подобной ситуации, можно воспользоваться услугами платных фотобанков — ресурсов, содержащих огромную базу редких фотографий, за использование которых необходимо заплатить деньги. Если вы хотите, чтобы фотография была эксклюзивной, обратитесь в специализированные фотостудии, профессиональные фотографы которых сделают для ваших баннеров (или сайта) единственные в своем роде фотографии.

3. Анимация. По статистике, анимированные рекламные баннеры получают на 25% больше отклика, чем статичные носители. И это понятно: движение приковывает взгляд пользователя, динамичные сцены и образы фиксируются даже боковым зрением и более четко запечатлеваются в памяти потребителей. Для сравнения статичной и анимированной рекламы приведем пример, связанный с телевидением. Какое рекламное обращение привлечет большее внимание: неподвижная картинка с логотипом рекламодателя и краткой информацией о предлагаемых товарах/услугах или динамично развивающееся действие, состоящее из нескольких ярких, живых сцен с участием людей, природы, животных и пр.? Скорее всего, последнее обращение будет более

эффективным. Вспомните такие ролики, как реклама "Nescafe" ("Сначала был кофе"), "БиЛайн GSM" ("Не звоните, потому что ждете вечерних скидок?") и пр. Общая рекламная идея для представления марки была проведена через всю серию роликов, каждый из которых представлял собой небольшое действие в лицах, динамично отображающее суть рекламного предложения. Наверняка такая реклама надолго останется в памяти людей, даже если они не пьют кофе и не пользуются услугами сотовой связи.

Кроме того, анимированные баннеры позволяют в рамках одного рекламного носителя получать большую информативность сообщения, в отличие от статичного баннера. Возможность чередования кадров позволяет разбить рекламное сообщение на отдельные информационные блоки, призванные постепенно закреплять в памяти пользователя образ рекламируемых товаров или услуг.

Однако при использовании, например, GIF-анимации рекламодатель может столкнуться с такой проблемой: чем насыщеннее, динамичнее и правдоподобнее будет анимированный баннер, тем сложнее его внутренняя структура, а соответственно — выше размер конечного файла. Учитывая, что в российской части Интернета файловым лимитом принято считать 15 Кбайт (для большого формата 468×60) и еще меньший размер для других форматов, рекламодатель вынужден искать "золотую середину": соответствовать требованиям рекламных площадок и баннерных сетей и придать анимированному рекламному носителю максимум динамики, плавности, реалистичности, предоставив при этом приемлемый уровень качества GIF-баннера (оптимизация изображений, корректировка цветовой палитры и пр.).

4. Rich-media. Интерактивные баннеры (rich-media) позволяют, помимо измерения отклика на рекламу (коэффициент CTR), оперировать другими, более детальными величинами. Возможность организации средств обратной связи — консультаций, опросов, исследований — повышает степень взаимодействия между рекламодателем и потенциальным потребителем. Поэтому использование раскрывающихся списков и ниспадающих меню, радиокнопок и чекбоксов весьма приветствуется с точки зрения повышения эффективности рекламного носителя (другое дело, если обменная сеть или рекламная площадка не поддерживают носители rich-media).

HTML-баннер представляет собой код или фрагмент кода обыкновенной Web-страницы, который загружается посредством так называемых "плавающих" фреймов или действия директив SSI (Server Side Includes). Разделим изображение, показанное на рис 17.2, на четыре части: слева логотип компании "Автодром", правая часть состоит из верхнего, нижнего, правого и левого разделителей, а также редактируемой области, в которую будут включены специальные поля формы (радиокнопки "До 1990 года" и "После 1990 года", ниспадающее меню с наименованиями моделей и кнопка подтверждения).

Компоновка составных элементов такого HTML-баннера будет реализована посредством следующего кода:

Листинг 17.1. Примерный вид кода для HTML-баннера

```
<TABLE WIDTH="468" BORDER="0" CELLPADDING="0" CELLSPACING="0">
   <TR>
      <TD ROWSPAN="3">
         <IMG SRC="images/autodrom.gif" WIDTH="186" HEIGHT="60"></TD>
      <TD ROWSPAN="2">
         <IMG SRC="images/l_border.gif" WIDTH="4" HEIGHT="51"></TD>
      <TD COLSPAN="2">
         <IMG SRC="images/t_border.gif" WIDTH="278" HEIGHT="8"></TD>
   </TR>
   <TR>
      <TD WIDTH="270" HEIGHT="43">
<FORM NAME="auto" METHOD="post" ACTION="http://www.server.ru
/cgi-bin/auto.cgi">
<INPUT TYPE="radio" VALUE="till1990" NAME="1990">До 1990 года<BR>
<INPUT TYPE="radio" VALUE="since1990" NAME="1990">После 1990 года
<SELECT NAME="model">
   <OPTION VALUE="Mercedes">Mercedes</OPTION>
   <OPTION VALUE="Toyota">Toyota</OPTION>
   <OPTION VALUE="Opel">Opel</OPTION>
   <OPTION VALUE="Ford">Ford</OPTION>
</SELECT>
<INPUT TYPE="submit" VALUE="OK">
</FORM>
      </TD>
      <TD>
         <IMG SRC="images/r_border.gif" WIDTH="8" HEIGHT="43">></TD>
   </TR>
   <TR>
      <TD COLSPAN="3">
         <IMG SRC="images/b_border.gif" WIDTH="282" HEIGHT="9">></TD>
   </TR>
</TABLE>
```

Рис. 17.2. Внешний вид HTML-баннера

Добавив к вышеуказанному коду остальные необходимые инструкции (<BODY>, <HEAD>, <TITLE> и др.), сохраняем документ, например, под именем banner.html. Далее (как уже было сказано выше) полученный HTML-баннер можно отобразить двумя способами: с помощью встроенного ("плавающего") фрейма и путем SSI-директивы.

Листинг 17.2. Отображение HTML-баннера с помощью "плавающего" фрейма

```
<IFRAME SRC="http://www.server.ru/banners/iframe/banner.html"
NAME="banner" WIDTH="468" HEIGHT="60" SCROLLING="no" FRAMEBORDER="0">
Ваш браузер не поддерживает "плавающие" фреймы.
</IFRAME>
```

Второй вариант встраивания одних документов в другие потребует поддержки SSI (Server Side Includes). Выглядеть это будет примерно так:

```
<!--#include virtual="banner.html" -->
```

5. Цвета. В зависимости от того, какую цветовую палитру вы выберете для своего рекламного баннера, может определяться конечный результат рекламных показов. Согласно распространенному мнению, наиболее заметными считаются баннеры, в основу которых положены яркие цвета. Но многие цвета просто-напросто не сочетаются друг с другом, будучи яркими (например, лимонный и салатный), поэтому, с этой точки зрения, ни о какой эффективности говорить не приходится. Таким образом, следует немного подкорректировать утверждение об использовании цветовой гаммы в рекламных баннерах: для создания эффективных баннеров нужно использовать яркие цвета, хорошо сочетающиеся друг с другом. Но это тоже не до конца верная формулировка: что делать, если используемые цвета и достаточно яркие, и замечательно гармонируют по отношению друг к другу, но абсолютно не заметны друг на друге? Например, некоторые яркие оттенки желтого и зеленого цветов отлично смотрятся вместе, однако, если вы попробуете на ярко-зеленом фоне написать буквы ярко-желтого цвета (или наоборот), получится нечто нечитабельное, что не стимулирует показатели эффективности. Отсюда следует третье правило использования цветовой палитры для рекламных баннеров: мало того, что необходимо выбирать яркие и гармонирующие друг с другом оттенки, нужно следить за тем, чтобы между ними наблюдался определенный контраст. Самый тривиальный, но вместе с тем удивительно эффектный пример контрастности цветов: черное на белом и наоборот. Как говорится, классика жанра! Действительно, мало кто не заметит выделяющегося белого текста, размещенного на черном фоне, среди красочного многообразия, присутствующего в Интернете. Итак, давайте подытожим ска-

занное выше и выделим три основных правила выбора цветовой палитры вашего рекламного баннера:

- яркость (яркие, сочные оттенки предпочтительнее);
- гармоничность (сочетание выбранных оттенков по отношению друг к другу);
- контрастность (взаимовыделяемость избранных оттенков).

В качестве конкретных рекомендаций можно привести некоторые цветовые схемы, оптимальные для применения (разумеется, в зависимости от конкретной ситуации):

- черный на белом (и наоборот);
- черный на желтом;
- синий на белом (и наоборот);
- зеленый на белом (и наоборот);
- красный на желтом;
- красный на белом (и наоборот);
- оранжевый на черном (и наоборот);
- фиолетовый на черном (и наоборот);
- белый на оранжевом.

Разумеется, например, схема "красный на белом" не означает обязательное использование "чистых" цветов: экспериментируйте с палитрой оттенков, стараясь далеко не уходить от выбранного соотношения цветов.

В заключение следует сказать, что человеческое восприятие цветовой гаммы — явление весьма субъективное и плохо поддающееся характеристике с позиции проявления определенных закономерностей. Для одного человека, синий с оранжевым — приятные, радующие глаз, спокойные цвета, другой реагирует на такое соотношение, как бык на красную тряпку. Как известно: сколько людей — столько и мнений, поэтому подстроить цветовое решение рекламного баннера абсолютно под всех потребителей не представляется возможным.

6. Четкость рекламного предложения. Баннер должен содержать четкое, ясное и определенное рекламное предложение, понятное не только целевой аудитории, но и потенциальным потребителям. Такие характеристики, как "неоднозначность", "расплывчатость", "туманность" не должны быть свойственны хорошо составленному рекламному предложению. Многие рекламодатели (а также их маркетологи, баннермейкеры и т. д.) путают два разных понятия, применимых к рекламному содержанию на баннере: таинственность и неопределенность. В результате, на свет появляются баннеры, смысл рекламного предложения которых далек от идеала доступности, простоты

изложения и ясности фраз. Более того, такие баннеры заставляют пользователя думать: "Интересно, а что они ЭТИМ хотели сказать?". Элемент загадки, тайны, интриги в баннерной рекламе мы рассмотрим отдельно, но как можно охарактеризовать рекламу, изначально имеющую статус имиджевой (что не приемлет никаких "тайн мадридского двора") с позиции четкости формулировки рекламного предложения? Например, такую (начало показов по телевизионным каналам приходится на конец ноября 2001 года): с громким визгом и непрерывным гудением, безумно вращающиеся шестеренки, искря и переливаясь, с огромной скоростью несутся по ночному небу, усыпанному то ли звездами, то ли еще чем-то блестящим. Внизу маячат желтые огни ночного мегаполиса, выскакивает надпись "Движ'ОК", сразу за ней откуда-то сбоку выкатываются три то ли пакета, то ли коробки с непонятно чем и, наконец, кульминация — гениальный слоган: "Это драйв!". С первого раза понять, что рекламируют, довольно трудно: фигурирует лишь торговая марка, разобрать, что изображено на пакетах (коробках), нельзя даже при помощи стоп-кадра. А дальше зритель проявляет полет фантазии и начинает "ломать" голову над тем, что же все-таки рекламирует эта марка. Машинное масло? Продукцию подшипникового завода? Новый блокбастер, испеченный на "фабрике грез"? Да все что угодно, потому что суть рекламного предложения в данном примере настолько слабо выражена, что человек, увидевший такую рекламу, сам начинает задумываться над тем, относится он к разряду потенциальных потребителей того, что там прорекламировали, или нет. (Примечание: для тех, кто до сих пор не догадался, что производится под торговой маркой "Движ'ОК", сообщаем — соленый арахис, копченые сухарики и прочая закуска к пиву).

7. Орфография. В последнее время на просторах российского Интернета стали учащаться случаи появления рекламных баннеров, содержащих орфографические ошибки: смешные и грустные, вызывающие жалость и раздражение. Как это ни покажется удивительным, наличие ошибок объясняется двумя причинами: либо уровнем общего образования разработчиков, либо халатностью по отношению к заказам на изготовление рекламы. Первая причина, к сожалению, весьма актуальна для России наряду с дураками и дорогами и заключается в неграмотности людей, разрабатывающих рекламные баннеры (хотя, полное отсутствие ошибок вовсе не означает три высших образования: разработчик может, имея за плечами восемь классов средней школы, создавать грамотные носители посредством принципа "Copy&Paste", т. е. путем выделения заранее составленного другими людьми текста, помещения его в буфер обмена и последующей вставки в нужное место на баннере). В основу второй причины наличия множества орфографических ошибок положено стремление в минимальные сроки изготовить максимальное количество рекламных баннеров, другими словами — возможность получения материальной выгоды для разработчика. Поэтому, когда одному человеку

надо за ночь успеть "наштамповать" десять-двенадцать баннеров, ни о какой орфографии речи уже быть не может: главное — успеть сдать заказ к установленному сроку. Из-за этого процедура проверки текста на предмет ошибок или опечаток игнорируется по причине дефицита времени.

Как было сказано, орфографические ошибки в структуре рекламных баннеров могут быть самые разные и способны вызвать у грамотного пользователя весь спектр чувств — от злости до умиления. Но каким бы ни был широким диапазон переживаемых пользователем эмоций при взгляде на безграмотно составленное рекламное сообщение, следует помнить, что наличие ошибок в тексте рекламного баннера — это признак дурного тона, стимулирующий возникновение скорее негативного образа рекламируемых товаров или услуг, нежели позитивного отношения к рекламодателю. Баннер авторской страницы, содержащий призыв типа "Дабро пажаловать на мой пирсональный сайт!", ничего, кроме неприязни, не вызовет. Реклама серьезного производителя программного обеспечения, анонсируя "новый уникальный суперсовременный програмный комплексс" (пунктуация и орфография сохранены), также заставит потребителя призадуматься.

Иногда на баннерах орфографию нарушают специально, чтобы привлечь внимание пользователя, обыграть какую-то сцену или ситуацию.

8. Соответствие рекламного контента целевой аудитории. По возможности, рекламодателю необходимо стремиться к тому, чтобы реклама его товаров или услуг была размещена не где попало, а там, куда приходит его целевая потребительская аудитория. Размещать баннер сайта о художниках-импрессионистах на ресурсах по автомобильной тематике можно только по двум веским причинам: либо рекламодателю некуда девать деньги, либо он истинно и горячо верит в высокий процент автомобилистов, на досуге обожающих посещать художественные выставки. В любом случае, эффективность рекламы по потребительскому требованию (поиск, переход по разделам сайта, регистрация, участие в опросах и т. д.) гораздо выше той, что определяется случайным образом.

9. Объект внимания. Желательно, чтобы ваш рекламный баннер содержал в себе некий объект, который бы приковывал внимание пользователя, заставлял на нем остановиться даже при беглом взгляде на рекламу. В качестве такого объекта может выступать все что угодно: символы и иероглифы, фотографии и карикатуры, текстуры и особые элементы оформления. Примером использования объекта, притягивающего внимание к рекламному баннеру, можно назвать рекламу в виде карикатуры от участников конкурса на портале Руслана Курепина "Caricatura.Ru".

10. Волшебные фразы. Есть фразы, обладающие практически необъяснимой магической силой, которая заметно увеличивает отклик на рекламу. К их

числу можно отнести такие, как "Click here", "Visit now", "Go! " (на английском) и "Подробнее...", "Жми сюда", "Войти" (на русском). Призыв "Click here" ("Нажмите здесь") является рекордсменом увеличения показателя CTR: по данным различных исследований, коэффициент отклика может повышаться при наличии этой фразы на 30—40%. Динамика эффективности на примере других призывов пока не изучалась, однако можно предположить, что отдача от их использования существенно ниже, чем в случае с "Click here".

В чем же заключается секрет "волшебных" фраз вроде "Click here"? В том, что они стимулируют действие потребителя, призывают его вступить во взаимодействие с рекламным баннером. Это особенно ясно вырисовывается на примере с неопытными пользователями Сети, которые не могут отличить рекламный баннер от фрагмента сайта, чудовищно любознательны, но слегка напуганы вирусами и "бессмертными" окошками pop-up. Поэтому они сами не догадаются, что по рекламному баннеру, содержание которого заинтересовало пользователя, необходимо щелкнуть — именно для них и пишут что-то вроде пояснительных надписей: "Click here", "Подробности здесь" и т. п.

В заключение остается порекомендовать не ограничиваться простой совокупностью букв, слагаемых в словосочетание "Click here", а активно, но в разумной мере, использовать мелкие оформительские "довески" волшебным призывам к действию: стрелки, курсоры, указатели, кнопки и пр.

11. Эротика. Мы уже выяснили, каким образом влияет на эффективность рекламы использование сексуально-эротических мотивов. Нельзя сказать, что использовать эти мотивы в баннерной рекламе — нехорошо. Все зависит от меры откровенности элементов эротики. Под этим подразумевается некая разумная грань (не поддающаяся количественной оценке), за которую содержание баннера с элементами эротики не должно выходить (во всяком случае — не для эротических ресурсов). Если рекламодатель "перегибает палку", баннер может стать похожим на стандартную рекламу третьесортного порносайта. Поэтому, используя элементы эротики при создании рекламных баннеров, старайтесь ими не злоупотреблять.

Отдельно следует сказать о дилемме, стоящей перед любым, не лишенным этических принципов рекламодателем, баннер или баннеры которого разработаны на основе сексуально-эротических мотивов. С одной стороны, основная задача — привлечь пользователя, поэтому сыграть на человеческих инстинктах значит существенно повысить отдачу от рекламных показов. С другой стороны, представив себя на месте простого потребителя, рекламодатель может существенно поменять изначально выбранную рекламную стратегию, ориентированную на действие природных рефлексов.

Несколько правил поведения для тех, кто собирается использовать в баннерной рекламе сексуально-эротические мотивы (Примечание: речь идет не

о ресурсах с эротическим содержанием, для разработки рекламы которых существуют совершенно другие правила):

- знайте чувство меры;
- откровенность эротики — не признак будущей эффективности;
- не забывайте про рекламное предложение;
- постарайтесь использовать элементы эротики в качестве объекта внимания (Рекомендация № 9);
- поставьте себя на место потребителя.

12. Юмор. Он призван вызывать у людей улыбку, которая подразумевает возникновение положительных эмоций, а значит — закрепление в памяти. Услышав на чьем-то дне рождения два дня назад неудачную шутку, вы наверняка ее вскоре забудете, когда как хорошее проявление юмора (анекдот, байка и пр.) обычно запоминается надолго. Так же и в баннерной рекламе: использование смешных ситуаций, обыгрывание анекдотов и фольклора, юмористическое представление заведомо серьезных вещей — все эти элементы рекомендуется использовать в рекламных баннерах. Причем, ни в коем случае нельзя придерживаться точки зрения, в соответствии с которой юмор присущ исключительно развлекательным интернет-проектам. Улыбка на лице потребителя никогда не повредит ни порталу игр, ни серверу финансового агентства, ни электронному средству массовой информации.

Примеров удачного использования юмористических образов — огромное количество. Например, все наверняка видели такой рекламный телевизионный ролик: молодой парень подъезжает на шикарной красной машине к дому, ставит ее на сигнализацию и вразвалочку уходит по своим делам; далее камера переходит на дальний план, и зритель видит висящий на стене этого дома щит: "Опасная зона! Идут взрывные работы" — и тут же слоган: "Все будет хорошо". Забавно, даже смешно, если дать волю фантазии и представить, что могло произойти дальше... Между тем, это реклама принадлежит вполне серьезной страховой компании.

Другой пример — на интернет-тематику. В свое время один из крупнейших интернет-провайдеров Кубани показывал рекламный баннер формата 468×60 следующего содержания: изображение группы "гарных хлопців" в национальных костюмах и надпись "Сала у нас нема...", которая спустя какое-то время сменялась другой — "...Зато есть провайдеры!". Появление доброй улыбки от удачно обыгранной традиции украинского народа гарантировано.

13. Загадка (интрига). Как уже было сказано, рекламодатель зачастую путает понятие загадки в баннере и нечеткой, неопределенной формулировки рекламного предложения. Второе (Рекомендация № 6) вызывает у потребите-

ля недоумение по поводу того, что именно рекламирует баннер. Элемент загадки, таинственности или даже интриги призван заинтересовать пользователя не напрямую рекламируемым товаром или услугой, а тем, КАК они представлены на суд потребителя. Если человек, видя рекламу, заинтригован (что подразумевает в равной степени достаточно высокий уровень заинтересованности и обладание неполной информацией об объекте рекламы), он склонен искать логическое продолжение такой рекламы, чтобы узнать, чем закончилось дело (в случае взаимосвязанной единым сюжетом рекламной серии), что же будет дальше, как станут разворачиваться события и т. д.

Теперь конкретные примеры. Первый снова из мира телевидения: торговая марка "3 корочки" в ноябре 2001 года выпустила рекламный ролик своих сухариков, описывающий полет двух космонавтов, у которых внезапно закончился запас этих самых сухариков и они запросили помощь Земли (а в мягких креслах ЦУПа — центра управления полетами — сидели не кто иные, как сами сухарики "3 корочки"). В начале ролика был сделана пометка "Часть 1". Это подразумевает за собой появление тематического продолжения истории о людях, плохо себя чувствующих без этих сухариков: в другой раз они могут оказаться без жизненно важного продукта где-нибудь в Тихом океане на глубине десяти километров или в чреве огнедышащего вулкана и т. п.

Другой пример касается онлайновой игры, которую некоторое время назад проводили почтовый сервер "Mail.Ru" и компания "Samsung Electronics". Суть игры заключалась в отслеживании баннерной рекламы, размещенной на страницах серверов упомянутых компаний, выявлении ключевых элементов в их оформлении и последующем представлении жюри конкурса всех собранных компонентов. Мало того, что активность пользователя подогревалась перспективой получения ценных призов от "Samsung Electronics", еще присутствовала тайна: никто из участников конкурса заранее не знал, на какой странице из двух крупномасштабных серверов появится баннер с необходимым оформлением.

14. Интерфейс ОС. Имитацию элементов управления операционных систем (в основном, это Microsoft Windows) желательно свести к разумному минимуму, тем более что многие баннерные сети и рекламные площадки не принимают баннеры с изображением псевдоэлементов ОС. Помните, что вероятность того, что пользователь испытает положительные эмоции от невозможности воспользоваться по назначению ложным ниспадающим меню на баннере, чрезвычайно мала.

15. Изменение параметров. Допустим, вы разработали рекламный баннер с определенной структурой, компоновкой составных элементов, слоганом и разместили его на конкретных рекламных площадках или запустили в ротацию на обозначенных интернет-ресурсах. На этом останавливаться нельзя:

спустя некоторое время проследите отклик на баннер и сделайте соответствующие выводы. Возможно, эффективность баннера будет низкой, или реальная аудитория не будет совпадать с целевым пользователем рекламируемых товаров или услуг. Однако это не повод для расстройства или приостановления рекламных показов. Старайтесь постоянно изменять параметры рекламного баннера и его показов согласно следующим принципам:

- компоновка баннерной структуры — к примеру, изначально вы разместили логотип слева, рекламный слоган справа вверху, а перечень товарных характеристик — в нижней правой части баннера. Попробуйте поменять местами эти объекты: логотип посередине, слева — слоган, справа — характеристики. Таким образом, достигается оптимальная компоновка составных элементов рекламного носителя;
- цветовая гамма — первый вариант рекламы использовал спокойные пастельные тона, что не вызывало должной реакции потребителя. Попробуйте добавить один-два ярких или контрастных цвета, однако не дисгармонирующих с изначально выбранной цветовой палитрой. Или кардинально поменяйте ее: в ряде случаев такой подход помогает резко увеличить заинтересованность рекламным баннером (здесь также необходимо помнить о том, что выбор цветовой гаммы баннера должен руководствоваться концепцией рекламируемого товара или услуги: например, в концептуальном плане реклама чего-то экстремального, экстраординарного, острого по ощущениям не приемлет спокойных красок; яркие примеры — "Молотов-коктейль" или программа "Новая атлетика" на канале MTB);
- оформление — если рекламный баннер использует редкие шрифты (Alfredo Heavy Hollow, Bangkok и др.) и при этом отклик на рекламу ниже среднего, возможно, следует использовать более знакомые массовому пользователю начертания. Если на баннере присутствуют фрагменты из какого-то известного кинофильма, но пользователь не идет на сайт рекламодателя, причина может заключаться в неудачно выбранном фрагменте (например, в рекламном баннере фильма "Терминатор" более разумно использовать изображение самого полуробота-получеловека из будущего, нежели женщины Сары О'Коннор, за которой он охотится);
- рекламная площадь — хотя причиной низкого отклика на рекламу может оказаться вовсе не компоновка структуры баннера или его цветовая гамма, а диапазон интернет-ресурсов, на которых этот рекламный баннер показывается. Поэтому необходимо постоянно пересматривать круг сайтов, демонстрирующих вашу рекламу, следить за их изменением во внутреннем или внешнем рейтингах, динамикой визитов, целевой аудиторией и пр.

Также большое значение имеет место пространственного расположения баннеров рекламодателя на страницах сайта издателя.

16. Ассоциации. При разработке рекламных баннеров старайтесь избегать возникновения у пользователя отрицательных ассоциаций. Это может привести, во-первых, к понижению отклика на рекламу, во-вторых, негативно повлиять на имидж компании-рекламодателя (последнее особенно опасно для брендинга).

Рассмотрим два примера рекламы: в первом случае она вызывает отрицательные ассоциации, во втором — положительные.

В качестве примера с негативной ассоциативной связью рассмотрим московскую туристическую компанию под интересным названием "Иван Сусанин Тур". Безусловно, фирма предлагает большой выбор маршрутов для своих клиентов — и для отдыха, и для путешествий, и для осмотра местных достопримечательностей. Однако наверняка мало кому захочется оказаться на месте польских "туристов", много лет назад заблудившихся по вине Сусанина.

Пример положительных ассоциаций: реклама почтовой службы портала "Gala.Net". Один из рекламных баннеров содержал следующую композицию: логотип, изображение актера Алексея Булдакова в камуфляже, далее надпись "Полковнику никто..." и пауза. Потом следует заверение пользователя: "А вам напишут" и краткая характеристика рекламируемого сервиса — "Твой бесплатный почтовый ящик на Gala.Net. 15 Мб". Здесь отчетливо проявляется сразу две ассоциации: с героем Булдакова (фильмы "Особенности национальной охоты", "Особенности национальной рыбалки" и пр.) и песней группы "Би — 2" из кинофильма "Брат-2".

17. Противоречия. Возникновение в сознании потребителя определенных противоречий в процессе ознакомления с рекламным баннером также нежелательно. Смысловое рассогласование может запутать пользователя, сбить его с толку, дезориентировать. Логичным результатом этого будет низкий отклик и формирование негативного имиджа.

По уже сложившейся традиции приведем два схожих примера: из телевизионной и интернет-рекламы. Реклама продукции "Catsan" ("сухой" туалет для кошек) гласит: "Даже в таком юном возрасте кошки определяют местонахождение туалета по запаху. Однако теперь этого запаха нет — представляем вам новые гранулы "Catsan", которые мгновенно впитывают..." и далее. Позвольте, как же теперь бедные кошки смогут определить, где можно, а где — нет, если, как утверждает реклама, запаха больше не существует?

Пример противоречивой рекламы, где рассогласование наступает между рекламным призывом баннера и визуальным восприятием сайта рекламодателя: некая дизайн-студия зарегистрировала домен в зоне .RU и стала привлекать клиентов посредством баннера. На нем говорилось, какие профессионалы своего дела работают в студии, какой огромный опыт находится за их плечами и т. п. Однако заинтересовавшиеся пользователи, перейдя по ссылке, пе-

реживали самые разнообразные ощущения от увиденного на сайте этой студии: от гнева до насмешливой улыбки. Дело в том, что сайт был сделан со множеством орфографических, программных, верстальных и прочих ошибок, возможности обратной связи не было, раздел "Наше портфолио" уже полгода находился в "стадии обновления". Нетрудно догадаться, какой образ сформировали разочарованные посетители сайта этой дизайн-студии.

18. Экспериментализм. Разрабатывая рекламные баннеры, четко ощущайте разницу между подражанием и плагиатом, помните, что грань, находящаяся между ними, чрезвычайно мала и носит весьма условный характер. Если вам понравилась реализация чужого баннера, не надо точь-в-точь копировать удачно созданную рекламу. Лучше тщательно проанализируйте этот рекламный образец, задайтесь вопросами: "Почему он эффективен?", "Что в нем привлекает пользователя?", "За счет чего слоган приковывает внимание?". Когда первопричина будет определена, используйте ее в качестве отправной точки в ходе разработки своих рекламных баннеров. Представьте, что элементы вашего баннера кем-то будут использованы в своих целях: наверняка, это вам не понравится. Стремитесь создать что-то новое, свежее, необычное, отличающееся от существующих аналогов. Другими словами, проявляйте творческий подход к вопросу проектирования и разработки рекламных носителей.

Не бойтесь экспериментировать с компонентами баннера, пробуйте самые неожиданные и, как вам кажется, заведомо неудачные варианты. Такой метод проб и ошибок, в конечном итоге, позволит вам найти оптимальное и наиболее эффективное решение.

19. Этика. Соблюдайте общепринятые нормы этики: ни в коем случае не используйте в своих рекламных баннерах следующие элементы:

- оскорбления и грубость;
- расовые и национальные предрассудки;
- призыв к насилию;
- сплетни и слухи;
- интимные подробности об известных людях (неотъемлемый атрибут "желтой" прессы);
- беспочвенные обвинения;
- непроверенная или не заслуживающая доверия информация.

Помните, что вы в равной степени можете оказаться на месте людей, по отношению к которым ваш рекламный баннер оказался неэтичным. Не пользуйтесь отсутствием свода законов, регулирующих маркетинговые процессы в Интернете. Старайтесь сделать всемирную сеть сферой обитания воспитанных, вежливых и образованных людей.

20. Сохранение исходных файлов. Разрабатывая рекламные баннеры, используйте графические пакеты, которые поддерживают объектное редактирование в файле вашей рекламы (например, Adobe Photoshop). Это позволит, во-первых, при необходимости быстро вносить изменения в рекламный модуль, во-вторых, в случае надобности подтвердить свое авторство по разработанному баннеру. Например, можно создавать рекламный модуль в редакторе Adobe Photoshop, накладывая на холст необходимые объекты (в виде "слоев" — Layers), а далее воспользоваться ПО для оптимизации графических заготовок и создания анимации (в состав Adobe Photoshop входят функциональные средства и для первого, и для второго — в приложении ImageReady). Размещаться в Интернете будет конечный файл: GIF или JPEG, идущие целостным объектом и не позволяющие разбить баннер на части. Поэтому всегда следует сохранять графические заготовки, которыми вы пользуетесь в ходе разработки рекламных баннеров.

Таким образом, вы узнали о наиболее эффективных и доступных способах продвижения своего Web-сайта в сети Интернет, которые помогут добиться успеха и популярности разработанных вами проектов.

Заключение

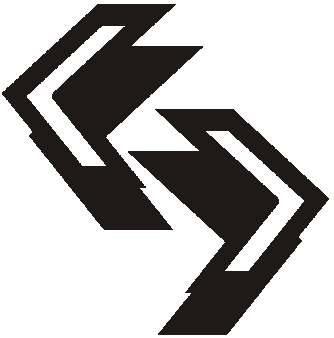
В этой книге был рассмотрен язык разметки гипертекстовых документов HTML, его основные функции, свойства и параметры. Сегодня применение этого языка практикуется во всех без исключения электронных документах — независимо от тематики, величины и коммерческой направленности интернет-проекта.

Чтобы использовать знания HTML на практике, нужно уметь работать с графическим макетом, знать тонкости и нюансы HTML-верстки, уметь продумывать концепцию будущего Web-сайта и делать его максимально удобным для посетителя.

Мало разработать собственный Web-сайт — о его существовании должна узнать многотысячная аудитория Интернета. Об основах продвижения Web-сайта в глобальной сети также было рассказано в этой книге.

Прочитав данную книгу, можно с уверенностью утверждать, что вы освоили азы полного цикла работ в процессе проектирования Web-сайта: от проработки концепции и структуры и создания графического макета до реализации посредством HTML и продвижения Web-сайта в Интернете.

Успехов во всех ваших начинаниях!



Приложения

Приложение 1. Обозначение цветов в HTML

Приложение 2. Список специальных символов HTML

Приложение 3. Полезные ссылки

ПРИЛОЖЕНИЕ 1

Обозначение цветов в HTML

Название	Шестнадцатеричное обозначение	Модель RGB
aliceblue	#F0F8FF	240 248 255
antiquewhite	#FAEBD7	250 235 215
aqua	#00FFFF	0 255 255
aquamarine	#7FFFD4	127 255 212
azure	#F0FFFF	240 255 255
beige	#F5F5DC	245 245 220
bisque	#FFE4C4	255 228 196
black	#000000	0 0 0
blanchedalmond	#FFEBCD	255 235 205
blue	#0000FF	0 0 255
blueviolet	#8A2BE2	138 43 226
brown	#A52A2A	165 42 42
burlywood	#DE8887	222 136 135
cadetblue	#5F9EA0	95 158 160
chocolate	#D2691E	210 105 30
coral	#FF7F50	255 127 80
cornflowerblue	#6495ED	100 149 237
cornsilk	#FFF8DC	255 248 220
crimson	#DC143C	220 20 60

(продолжение)

Название	**Шестнадцатеричное обозначение**	**Модель RGB**
cyan	#00FFFF	0 255 255
darkblue	#00008B	0 0 139
darkcyan	#008B8B	0 139 139
darkgoldenrod	#B8860B	184 134 11
darkgray	#A9A9A9	169 169 169
darkgreen	#006400	0 100 0
darkkhaki	#BDB76D	189 183 109
darkmagenta	#8B008B	139 0 139
darkolivegreen	#556B2F	85 107 47
darkorange	#FF8C00	255 140 0
darkorchid	#9932CC	153 50 204
darkred	#8B0000	139 0 0
darksalmon	#E9967A	233 150 122
darkseagreen	#8FBC8F	143 188 143
darkslateblue	#483D8B	72 61 139
darkslategray	#2F4F4F	47 79 79
darkturquoise	#00CED1	0 206 209
darkviolet	#9400D3	148 0 211
deeppink	#FF1493	255 20 147
deepskyblue	#00BFFF	0 191 255
dimgray	#696969	105 105 105
dodgerblue	#1E90FF	30 144 255
firebrick	#B22222	178 34 34
floralwhite	#FFFAF0	255 250 240
forestgreen	#228B22	34 139 34
fuchsia	#FF00FF	255 0 255
gainsboro	#DCDCDC	220 220 220
ghostwhite	#F8F8FF	248 248 255
gold	#FFD700	255 215 0

(продолжение)

Название	Шестнадцатеричное обозначение	Модель RGB
goldenrod	#DAA520	218 165 32
gray	#808080	128 128 128
green	#008000	0 128 0
greenyellow	#ADFF2F	173 255 47
honeydew	#F0FFF0	240 255 240
hotpink	#FF69B4	255 105 180
indianred	#CD5C5C	205 92 92
indigo	#4B0082	75 0 130
ivory	#FFFFF0	255 255 240
khaki	#F0E68C	240 230 140
lavender	#E6E6FA	230 230 250
lavenderblush	#FFF0F5	255 240 245
lemonchiffon	#FFFACD	255 250 205
lightblue	#ADD8E6	173 216 230
lightcoral	#F08080	240 128 128
lightcyan	#E0FFFF	224 255 255
lightgoldenrodyellow	#FAFAD2	250 250 210
lightgreen	#90EE90	144 238 144
lightpink	#FFB6C1	255 182 193
lightsalmon	#FFA07A	255 160 122
lightseagreen	#20B2AA	32 178 170
lightskyblue	#87CEFA	135 206 250
lightslategray	#778899	119 136 153
lightsteelblue	#B0C4DE	176 196 222
lightyellow	#FFFFE0	255 255 224
lime	#00FF00	0 255 0
limegreen	#32CD32	50 205 50
linen	#FAF0F6	250 240 246
magenta	#FF00FF	255 0 255

(продолжение)

Название	Шестнадцатеричное обозначение	Модель RGB
maroon	#800000	128 0 0
mediumaquamarine	#66CDAA	102 205 170
mediumblue	#0000CD	0 0 205
mediumorchid	#BA55D3	186 85 211
mediumpurple	#9370DB	147 112 219
mediumseagreen	#3CB371	60 179 113
mediumslateblue	#7B68EE	123 104 238
mediumspringgreen	#00FA9A	0 250 154
mediumturquoise	#48D1CC	72 209 204
mediumvioletred	#C71585	199 21 133
midnightblue	#191970	25 25 112
mintcream	#F5FFFA	245 255 250
mistyrose	#FFE4E1	255 228 225
moccasin	#FFE4B5	255 228 181
navajowhite	#FFDEAD	255 222 173
navy	#000080	0 0 128
oldlace	#FDF5E6	253 245 230
olive	#808000	128 128 0
olivedrab	#6B8E23	107 142 35
orange	#FFA500	255 165 0
orangered	#FF4500	255 69 0
orchid	#DA70D6	218 112 214
palegoldenrod	#EEE8AA	238 232 170
palegreen	#98FB98	152 251 152
paleturquoise	#AFEEEE	175 238 238
palevioletred	#DB7093	219 112 147
papayawhip	#FFEFD5	255 239 213
peachpuff	#FFDAB9	255 218 185
peru	#CD853F	205 133 63

(окончание)

Название	Шестнадцатеричное обозначение	Модель RGB
pink	#FFC0CB	255 192 203
plum	#DDA0DD	221 160 221
powderblue	#B0E0E6	176 224 230
purple	#800080	128 00 128
red	#FF0000	255 0 0
rosybrown	#BC8F8F	188 143 143
royalblue	#4169E1	65 105 225
salmon	#FA8072	250 128 114
sandybrown	#F4A460	244 164 96
seagreen	#2E2B57	43 46 87
seashell	#FFE5EE	255 229 238
sienna	#A0522D	160 82 45
silver	#C0C0C0	192 192 192
skyblue	#87CEEB	135 206 235
slateblue	#6A5ACD	106 90 205
slategray	#708090	112 128 144
snow	#FFFAFA	255 250 250
springgreen	#00FF7F	0 255 127
steelblue	#4682B4	70 130 180
tan	#D2B48C	210 180 140
teal	#008080	0 128 128
thistle	#D8BFD8	216 191 216
tomato	#FF6347	255 99 71
turquoise	#40E0D0	64 224 208
violet	#EE82EE	238 130 238
wheat	#F5DEB3	245 222 179
white	#FFFFFF	255 255 255
whitesmoke	#F5F5F5	245 245 245
yellow	#FFFF00	255 255 0
yellowgreen	#9ACD32	154 205 50

ПРИЛОЖЕНИЕ 2

Список специальных символов HTML

Символ	Код	Вид	Описание
			Неразрывный пробел
¡	¡	¡	Перевернутый восклицательный знак
¢	¢	¢	Цент
£	£	£	Фунт стерлингов
¤	¤	¤	Денежная единица
¥	¥	¥	Иена или юань
¦	¦	¦	Разорванная вертикальная черта
§	§	§	Параграф
¨	¨	¨	Трема (знак над гласной в иностранных алфавитах)
©	©	©	Знак copyright
ª	ª	ª	Женский порядковый числитель
«	«	«	Левая двойная угловая скобка
¬	¬	¬	Знак отрицания
		-	Место возможного переноса
®	®	®	Знак зарегистрированной торговой марки
¯	¯	¯	Знак долготы над гласным
°	°	°	Градус
±	±	±	Плюс-минус
²	²	²	Верхний индекс 'два' — "в квадрате"

(продолжение)

Символ	Код	Вид	Описание
³	³	³	Верхний индекс 'три' — "в кубе"
´	´	´	Знак ударения
µ	µ	µ	Микро
¶	¶	¶	Символ параграфа
·	·	·	Точка
¸	¸	¸	Седиль (орфографический знак в иностранных алфавитах)
¹	¹	¹	Верхний индекс 'один'
º	º	º	Мужской порядковый числитель
»	»	»	Правая двойная угловая скобка
¼	¼	¼	Дробь — одна четверть
½	½	½	Дробь — одна вторая
¾	¾	¾	Дробь — три четверти
¿	¿	¿	Перевернутый вопросительный знак
À	À	À	Латинская заглавная буква A с тупым ударением
Á	Á	Á	Латинская заглавная буква A с острым ударением
Â	Â	Â	Латинская заглавная буква A с циркумфлексом (диакритический знак над гласной)
Ã	Ã	Ã	Латинская заглавная буква A с тильдой
Ä	Ä	Ä	Латинская заглавная буква A с тремой
Å	Å	Å	Латинская заглавная буква A с верхним кружком
Æ	Æ	Æ	Латинские заглавные символы AE
Ç	Ç	Ç	Латинская заглавная буква C с седилем
È	È	È	Латинская заглавная буква E с тупым ударением
É	É	É	Латинская заглавная буква E с острым ударением
Ê	Ê	Ê	Латинская заглавная буква E с циркумфлексом
Ë	Ë	Ë	Латинская заглавная буква E с тремой
Ì	Ì	Ì	Латинская заглавная буква I с тупым ударением
Í	Í	Í	Латинская заглавная буква I с острым ударением

(продолжение)

Символ	Код	Вид	Описание
Î	Î	Î	Латинская заглавная буква I с циркумфлексом
Ï	Ï	Ï	Латинская заглавная буква I с тремой
Ð	Ð	Ð	Латинские заглавные символы ETH
Ñ	Ñ	Ñ	Латинская заглавная буква N с тильдой
Ò	Ò	Ò	Латинская заглавная буква O с тупым ударением
Ó	Ó	Ó	Латинская заглавная буква O с острым ударением
Ô	Ô	Ô	Латинская заглавная буква O с циркумфлексом
Õ	Õ	Õ	Латинская заглавная буква O с тильдой
Ö	Ö	Ö	Латинская заглавная буква O с тремой
×	×	×	Знак умножения
Ø	Ø	Ø	Латинская заглавная буква O со штрихом
Ù	Ù	Ù	Латинская заглавная буква U с тупым ударением
Ú	Ú	Ú	Латинская заглавная буква U с острым ударением
Û	Û	Û	Латинская заглавная буква U с циркумфлексом
Ü	Ü	Ü	Латинская заглавная буква U с тремой
Ý	Ý	Ý	Латинская заглавная буква Y с острым ударением
Þ	Þ	Þ	Латинская заглавная буква THORN
à	à	à	Латинская строчная буква a с тупым ударением
á	&##225;	á	Латинская строчная буква a с острым ударением
â	&##226;	â	Латинская строчная буква a с циркумфлексом
ã	ã	ã	Латинская строчная буква a с тильдой
ä	ä	ä	Латинская строчная буква a с тремой
å	å	å	Латинская строчная буква a с верхним кружком
æ	æ	æ	Латинские строчные буквы ae
ç	ç	ç	Латинская строчная буква a с седилем
è	è	è	Латинская строчная буква e с тупым ударением
é	é	é	Латинская строчная буква e с острым ударением
ê	ê	ê	Латинская строчная буква e с циркумфлексом

(продолжение)

Символ	Код	Вид	Описание
ë	ë	ë	Латинская строчная буква e с тремой
ì	ì	ì	Латинская строчная буква i с тупым ударением
í	í	í	Латинская строчная буква i с острым ударением
î	î	î	Латинская строчная буква i с циркумфлексом
ï	ï	ï	Латинская строчная буква i с тремой
ð	ð	ð	Латинские строчные символы eth
ñ	ñ	ñ	Латинская строчная буква n с тильдой
ò	ò	ò	Латинская строчная буква o с тупым ударением
ó	ó	ó	Латинская строчная буква o с острым ударением
ô	ô	ô	Латинская строчная буква o с циркумфлексом
õ	õ	õ	Латинская строчная буква o с тильдой
ö	ö	ö	Латинская строчная буква o с тремой
÷	÷	÷	Знак деления
ø	ø	ø	Латинская строчная буква o со штрихом
ù	ù	ù	Латинская строчная буква u с тупым ударением
ú	ú	ú	Латинская строчная буква u с острым ударением
û	û	û	Латинская строчная буква u с циркумфлексом
ü	ü	ü	Латинская строчная буква u с тремой
ý	ý	ý	Латинская строчная буква y с острым ударением
þ	þ	þ	Латинские строчные символы thorn
ÿ	ÿ	ÿ	Латинская строчная буква y с тремой
ƒ	ƒ	*ƒ*	Знак функции
			Греческие буквы
Α	Α	Α	Греческая заглавная буква альфа
Β	Β	Β	Греческая заглавная буква бета
Γ	Γ	Γ	Греческая заглавная буква гамма
Δ	Δ	Δ	Греческая заглавная буква дельта
Ε	Ε	Ε	Греческая заглавная буква эпсилон
Ζ	Ζ	Ζ	Греческая заглавная буква дзета

(продолжение)

Символ	Код	Вид	Описание
Η	Η	Η	Греческая заглавная буква эта
Θ	Θ	Θ	Греческая заглавная буква тета
Ι	Ι	Ι	Греческая заглавная буква иота
Κ	Κ	Κ	Греческая заглавная буква каппа
Λ	Λ	Λ	Греческая заглавная буква лямбда
Μ	Μ	Μ	Греческая заглавная буква мю
Ν	Ν	Ν	Греческая заглавная буква ню
Ξ	Ξ	Ξ	Греческая заглавная буква кси
Ο	Ο	Ο	Греческая заглавная буква омикрон
Π	Π	Π	Греческая заглавная буква пи
Ρ	Ρ	Ρ	Греческая заглавная буква ро
Σ	Σ	Σ	Греческая заглавная буква сигма
Τ	Τ	Τ	Греческая заглавная буква тау
Υ	Υ	Υ	Греческая заглавная буква ипсилон
Φ	Φ	Φ	Греческая заглавная буква фи
Χ	Χ	Χ	Греческая заглавная буква хи
Ψ	Ψ	Ψ	Греческая заглавная буква пси
Ω	Ω	Ω	Греческая заглавная буква омега
α	α	α	Греческая строчная буква альфа
β	β	β	Греческая строчная буква бета
γ	γ	γ	Греческая строчная буква гамма
δ	δ	δ	Греческая строчная буква дельта
ε	ε	ε	Греческая строчная буква эпсилон
ζ	ζ	ζ	Греческая строчная буква дзета
η	η	η	Греческая строчная буква эта
θ	θ	θ	Греческая строчная буква тета
ι	ι	ι	Греческая строчная буква иота
κ	κ	κ	Греческая строчная буква каппа

(продолжение)

Символ	Код	Вид	Описание
λ	λ	λ	Греческая строчная буква лямбда
μ	μ	μ	Греческая строчная буква мю
ν	ν	ν	Греческая строчная буква ню
ξ	ξ	ξ	Греческая строчная буква кси
ο	ο	ο	Греческая строчная буква омикрон
π	π	π	Греческая строчная буква пи
ρ	ρ	ρ	Греческая строчная буква ро
ς	ς	ς	Греческая строчная буква сигма (final)
σ	σ	σ	Греческая строчная буква сигма
τ	τ	τ	Греческая строчная буква тау
υ	υ	υ	Греческая строчная буква ипсилон
φ	φ	φ	Греческая строчная буква фи
χ	χ	χ	Греческая строчная буква хи
ψ	ψ	ψ	Греческая строчная буква пси
ω	ω	ω	Греческая строчная буква омега
			Стрелки
←	←	←	Стрелка влево
↑	↑	↑	Стрелка вверх
→	→	→	Стрелка вправо
↓	↓	↓	Стрелка вниз
↔	↔	↔	Стрелка влево-вправо
			Прочие символы
♠	♠	♠	Знак масти 'пики'
♣	♣	♣	Знак масти 'трефы'
♥	♥	♥	Знак масти 'червы'
♦	♦	♦	Знак масти 'бубны'
"	"	"	Двойная кавычка
&	&	&	Амперсанд

(окончание)

Символ	Код	Вид	Описание
<	<	<	Знак 'меньше'
>	>	>	Знак 'больше'
ˆ	ˆ	ˆ	Символ циркумфлекса (диакритический знак над гласной)
˜	˜	˜	Тильда
™	™	™	Знак торговой марки
			Знаки пунктуации
•	•	•	Маркер (bullet)
…	…	…	Многоточие
‾	‾	‾	Надчеркивание
⁄	⁄	⁄	Косая дробная черта
			Общая пунктуация
–	–	–	Тире
—	—	—	Длинное тире
‘	‘	‘	Левая одиночная кавычка
’	’	’	Правая одиночная кавычка
‚	‚	‚	Нижняя одиночная кавычка
“	“	“	Левая двойная кавычка
”	”	”	Правая двойная кавычка
„	„	„	Нижняя двойная кавычка

ПРИЛОЖЕНИЕ 3

Полезные ссылки

Ссылки на русскоязычные и зарубежные интернет-ресурсы, посвященные HTML, CSS, Dynamic HTML и вопросам проектирования Web-сайтов.

Русскоязычные ресурсы

- Заметки HTML-кодера — **http://htmlcoder.visions.ru/**
- Спецификация HTML 4.0 (русский перевод официальной версии Консорциума W3C) — **http://www.stack.ru/~julia/HTML4/cover.phtml**
- CIT Forum — **http://www.citforum.ru/**
- GetInfo.Ru — Компьютерная библиотека — **http://www.getinfo.ru/**
- HTML-справочник — **http://html.manual.ru/**
- Report — Сообщество экспертов (HTML) — **http://html.report.ru/**
- WebMascon — Переводные статьи по интернет-проектированию, HTML-верстке и пр. — **http://www.webmascon.com/**

Зарубежные ресурсы

- HTML Goodies — **http://www.htmlgoodies.com/**
- HTMLHelp — **http://www.htmlhelp.com/**
- PageResource — **http://www.pageresource.com/**
- SitePoint — **http://www.sitepoint.com/**
- The Unicode Consortium and The Unicode Standard — **http://www.unicode.org/**
- W3Schools — **http://www.w3schools.com/**
- WebDeveloper — **http://www.webdeveloper.com/**
- Web Developer's Virtual Library — **http://www.wdvl.com/Authoring/**
- World Wide Web Consortium — **http://www.w3.org/**

Предметный указатель

&

> 52
< 52
  44, 101
£ (фунт стерлингов) 52
® (зарегистрированная торговая марка) 52

1

120×240 (Vertical Banner) 351
120×60 (Button # 2) 351
125×125/100×100 (Square Banner # 1/Square Banner # 2) 351

4

468×60 (Full Banner) 351

8

88×31/88x62 (Micro Button/Macro Button) 349

A

A 56, 64, 142
ABBR 29
ABSBOTTOM 120
ABSMIDDLE 120
ACRONYM 29
ACTION 161
ActiveMovie 150
Adobe Photoshop 119, 299
AIFF 147
ALIGN 40, 89, 93, 103, 119, 170
alpha 215
ALT 124, 130, 171
AOL 15
AREA 129
ASCII 51
AU 147

B

B 32, 33
BACKGROUND 25, 95
BASEFONT 33
baseline (параметр) 94
BASELINE (базовая линия) 120
Begun 342
BGCOLOR 95
BGPROPERTIES 25
BGSOUND 154
BIG 34
blank 59
BLINK 34
BODY 18, 24, 107, 116, 139
BORDER 57, 87, 117, 140, 171
BORDERCOLOR 88, 140
bottom (параметр) 94
BOTTOM 120
BR 46, 79, 101

C

CAPTION 86
CELLPADDING 88
CELLSPACING 88
center (параметр) 40
CENTER (тег) 44
CERN 128
CGI (Common Gateway Interface) 127, 155
charset 21
CHECKBOX 169
CHECKED 169
circle 76
CITE 30
CLASS 58, 173
Client-Side Imagemap 128
CODE 30
COL 103

COLGROUP 103
COLOR 35
COLS 136
COLSPAN 96
COMPACT 71, 76, 83
CONTENT 20
COORDS 129
CSS (Cascading Style Sheets) 28, 43, 173,181, 204
CTR 363

D

DD 81
DEL 30, 36
description 22
DHTML (Dynamic HTML) 134, 173, 204
DIR 83
DISABLED 173
disc 76
DIV 42, 43
DL 81
dropShadow 215
DT 81

E

EM 30
E-mail (служба электронной почты) 60
Email-маркетинг 341
EMBED 148, 151
ENCTYPE 163

F

FACE 34
Flash (Macromedia Flash) 276, 354
FONT 33, 34
FORM 161, 164
FRAME 105, 138
FRAMEBORDER 140
FRAMESET 136
FRAMESPACING 140
FTP (File Transfer Protocol) 60

G

Gecko 15
GET 162, 166
GIF (Graphics Interchange Format) 101, 114, 352
GoogleAdwords 342
Gopher 60

H

H1, H2, H3, H4, H5, H6 31
HEAD 18, 19, 24, 183
HEIGHT 92, 118
HotLog 17
HR 48
HREF 58, 59, 63, 130
HSPACE 121
HTML (HyperText Markup Language) 12,18, 24
HTML-баннеры 356
HTML-верстка 306
HTML-документ 18, 26, 28, 46, 63
HTML-код 50
HTML-конструкция 18, 37, 75, 85
HTML-списки 67
HTML-форма 155, 190
HTTP (HyperText Transfer Protocol) 20, 60

I

I 36
IFRAME 145
IMG 116, 119, 124, 128
INPUT 164, 173
Internet Explorer 14, 48, 55, 59, 75, 88, 89, 99, 120, 144, 182
ISMAP 128

J

Java 354
JavaScript 17, 134, 173, 204, 207
JPEG (Joint Photographic Experts Group) 115, 352
JScript 204
justify 40

K

keywords 22
KOI8-R 21

L

left 40
LI 69, 75
light 217
LINK 19
LISTING 49
LiveAudio 150
LOOP 154
Lynx 123

M

MAP 128
MARGINHEIGHT 139
MARGINWIDTH 25, 139
MAXLENGTH 167
MENU 83
META 20
METHOD 162
Microsoft 14
middle (параметр) 94
MIDDLE 120
MIDI 147
Mosaic 15
Mozilla 15
MP3 148
MULTIPLE 175

N

NAME 22, 58, 64, 129, 163, 169
NCSA 128
Netscape 15, 34, 48, 55, 75
Netscape Navigator 14, 15, 89, 99, 120, 186
News (группы новостей) 60
NOBR 47
NOFRAMES 139
NOHREF 130
NOSHADE 49
NOWRAP 96

O

OL 68, 71
Opera 14, 16, 89, 99, 124
OPTION 175

P

P 39, 41, 43
PLAINTEXT 49
PNG (Portable Network Graphics) 116, 352
POST 162
PRE 49

R

RealAudio 148, 151
RGB 24
Rich-media 353, 366
right 40
robots 23
ROWS 136
ROWSPAN 96
RULES 105

S

SCRIPT 24
SCROLLING 138
SELECT 174
SELECTED 175
Server-Side Imagemap 127
SHAPE 129
SIZE 35, 167, 175
SMALL 36
SPAN 36, 42, 43, 103
Spyglass 14
square 76
SRC 116, 138, 170
START 71
STRIKE 30, 36
STRONG 32
STYLE 24, 58, 173, 185
SUB 36, 53
SUP 37, 53

T

TABLE 85, 87, 93, 105
TARGET 58, 129, 142
TBODY 104
TD 86, 93, 96, 108
TEXT 26
TEXTAREA 173
TFOOT 104

TH 86, 93, 96, 108
THEAD 104
TITLE 19, 58, 124, 173
top 94
TOPMARGIN 25
TR 86, 93
TT 37
TYPE 71, 76, 164

U

UL 75
URL 21
USEMAP 128

V

VALIGN 94, 103
VALUE 70, 169, 175
VBScript 204
VSPACE 121

W

WAV 147
wave 213
WBR 47
WIDTH 48, 92, 118
Windows-1251 21
World Wide Web (Всемирная паутина) 12
World Wide Web Consortium (W3C) 13

X

XMP 49

А

Абзац 39
Абсолютное позиционирование 202
Абсолютные единицы измерения 190
Абсолютный путь 59
Авторизация 159
Альтернативный текст 123
Амперсанд 51, 52, 162

Б

Баннерная реклама 344
Баннерный обмен 356
Браузер 14,25, 44, 54
Буквы иностранных алфавитов 55
Буллет (bullet) 75

В

Векторная графика 114
Визуальные фильтры 210
Вложенные списки 80
Вложенные таблицы 108
Внешние ссылки 59
Внутренние ссылки 63

Г

Гиперссылка, ссылка 26, 55
Горизонтальный разделитель, горизонтальная линия, линия 48
Гостевая книга 157
Графика 113
Графическое изображение, рисунок 57, 78, 113
Группировка данных 104

Д

Декоративные элементы 258
Джим Кларк (Jim Clark) 15
Диез 52, 63
Динамические блоки 205
Доски объявлений (Bulletin Boards) 339

Ж

Живой Журнал (LiveJournal) 240

З

Звуковые форматы 147

И

Идентификатор 188
Имиджевые элементы 256
Интернет-каталог, каталог интернет-ресурсов 337
Интернет-маркетинг 336
Интернет-форум, форум, конференция 157, 242
Исследования 160

Й

Йон Штефенсон фон Тежнер (Jon Stephenson fon Tetzchner) 16

К

Карта сайта (Sitemap) 284
Карты-изображения (Imagemaps) 125
Каскадные таблицы 28, 43, 173, 181, 204
Класс 187
Кодовая конструкция 51
Колонтитулы таблицы 104
Комиссия обмена 357
Комментарии 49, 111
Коммьюнити 238
Компоновка графического макета 297
Консорциум W3C 49, 73, 83, 139
Контекстная реклама 342
Контент-проекты 232
Корпоративные Web-сайты 219
Корпоративный стиль 221
Кросс-браузерная совместимость 54
Кэширование 20

Л

Логическое форматирование 28
Логотип 254

М

Марк Андрессен (Marc Andreessen) 15
Маркер 75, 76, 78
Маркированные списки 68, 75
Меню навигации, навигационное меню, меню, навигация 274, 297
Моноширинный шрифт 37
Мультиязыковая поддержка 54

Н

Навигационная "крошка", "крошка" 285
Нумерованные списки 68, 73

О

Обмен ссылками 342
Объектная модель документа (Document Object Model, DOM) 204
Онлайн-сервисы 229
Определение шаблона 186
Организация обратной связи 156
Относительное позиционирование 202
Относительные единицы измерения 190
Относительный путь 60
Отступ 39, 44
Ошибка 404 (Error 404) 288

П

Партнерские программы (Affiliate Program) 343
Перевод показов 357
Пиктограммы, иконки, значки 260
Плавающий фрейм 144
Подвал 297
"Подъемные" показы 362
Позиционирование Web-сайта, позиционирование 244
Поисковые системы 338
Покупка показов 362
Пользовательская форма 155, 190
Пользовательский профиль 161
Портал: "вертикальный", "горизонтальный" 236, 237
Почтовая форма 156
Предварительно отформатированный текст 49
Принудительный перенос строки 43, 46
Пробел 44
Промо/презентационные Web-сайты, промо-сайты, презентационные сайты 223
Прорисовка структуры таблицы 105
Псевдокласс 188

Р

Растровая графика 114
Рейтинговые службы, рейтинг 339
"Резина" 298

С

Седиль 54
Селектор 186
Скрипт 21
Содержательная часть 297
Специальные символы HTML 45, 51, 55
Списки определений 68, 81
Спонсорство 343
Структурное форматирование 39, 48, 190

Т

Таблица 85, 102, 108
Таргетинг 358
Тег 18
Текст 28
Тильда 54
Тип верстки 298
Тип выравнивания текста 47

Ф

Физическое форматирование 28, 43
Фикс 298
Форматирование текста, текстовое форматирование 190
Фрейм 131, 142

Ц

Центрирование 44, 89

Ч

Чат 159
Числовая комбинация 51

Ш

"Шапка" 297
Шестнадцатеричный 24

Э

Электронные магазины, онлайн-магазины 226
Электронный документ 50
Элемент 187
Эффект "обтекания" 90

Я

Яndex.Direct 342