

HTML в Web-дизайне

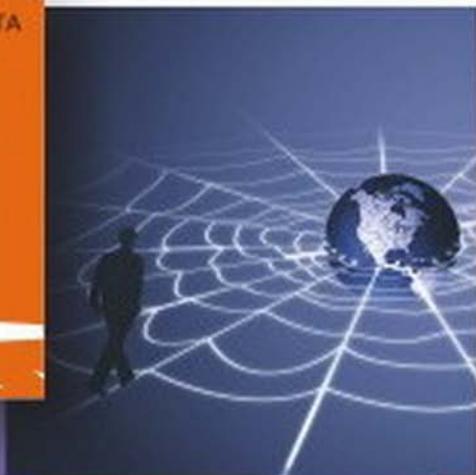
ОСНОВЫ HTML-ДОКУМЕНТА

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	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Евгений Рыбаков</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Татьяна Лапина</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн серии	<i>Иины Тачиной</i>
Оформление обложки	<i>Игоря Цырульниковой</i>
Зав. производством	<i>Николай Тверских</i>

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 ГЭСН 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 <i>HTML</i>	18
Раздел <i>HEAD</i>	19
Раздел <i>BODY</i>	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 <i></i>	71
Маркированные списки	75
Параметры тега <i></i>	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 <code><TABLE></code>	87
Параметры тегов <code><TR></code> , <code><TD></code> и <code><TH></code>	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 <code><FRAMESET></code>	136
Тег <code><FRAME></code>	138
Тег <code><NOFRAMES></code>	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 <code><EMBED></code>	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 <i>ACTION</i>	161
Параметр <i>METHOD</i>	162
Параметр <i>NAME</i>	163
Параметр <i>ENCTYPE</i>	163
Тег <i><INPUT></i>	164
Тег <i><TEXTAREA></i>	173
Тег <i><SELECT></i>	174
Глава 9. Стиливые шаблоны CSS	181
Каскадные таблицы стилей.....	181
Общая информация.....	181
Уровни CSS.....	182
Способы определения таблиц стилей.....	183
Запись шаблона CSS.....	186
Области применения стиливых 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 <i>HTML</i>	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 "СНIP-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

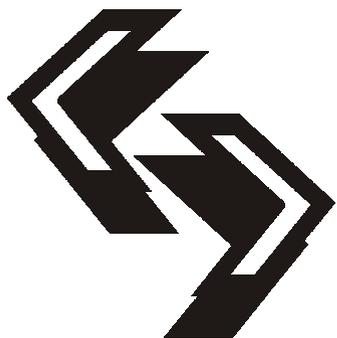
"GetInfo.Ru", "URSA-Россия", "Интернет-магазин программного обеспечения MSSoft.Ru", "Austrian Airlines", "Интернет-справочник Price.Ru", "TravellingRussia", "Техноподиум", "Guides of St.-Petersburg", "Griffin Securities", "Система документооборота Комитета экономики и инвестиций Правительства Ленинградской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

Web-сайт автора (<http://www.alpet.spb.ru>) удостоен 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>.

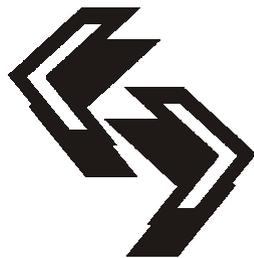


ЧАСТЬ I

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.** Стиливые 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 `` 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 `
`). 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.

МЕТА 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>:

```
<МЕТА 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 главную 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.

МЕТА 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.

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

Упомянутые и малоиспользуемые конструкции типа МЕТА 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
<code>keywords</code>	Указание ключевых слов документа	—
<code>description</code>	Указание краткого описания документа	—
<code>author</code>	Информация об авторе (авторах) документа	—
<code>robots</code>	Составление правил для индексирования документа поисковыми системами (роботами)	<code>ALL</code> <code>NONE</code> <code>INDEX</code> <code>NOINDEX</code> <code>FOLLOW</code> <code>NOFOLLOW</code>
<code>rating</code>	Установление возрастной категории содержания документа	—
<code>document-state</code>	Определение частоты индексирования документа	—
<code>revisit</code>	Определение промежутка времени, через которое должно производиться повторное индексирование документа	—
<code>distribution</code>	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 "диз"). 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 подразумеваемого 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
<code>BGCOLOR</code>	Определение цвета заднего фона
<code>BACKGROUND</code>	Указание фонового рисунка
<code>BGPROPERTIES</code>	Изменение свойств заднего фона (например, фиксирование фонового рисунка)
<code>TOPMARGIN</code> , <code>BOTTOMMARGIN</code> , <code>LEFTMARGIN</code> , <code>RIGHTMARGIN</code> , <code>MARGINWIDTH</code> , <code>MARGINHEIGHT</code>	Определение размера отступов
<code>TEXT</code>	Определение цвета основного текста
<code>LINK</code> , <code>ALINK</code> , <code>VLINK</code>	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!

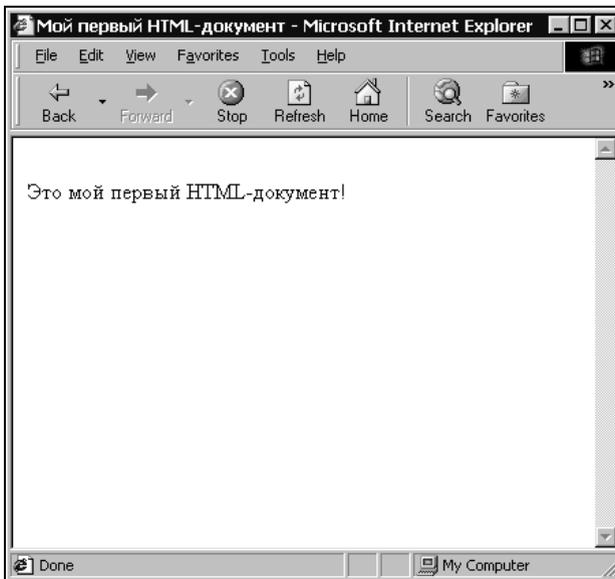
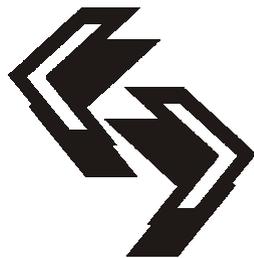


Рис. 1.1. Ваш первый HTML-документ

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 тегам 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).

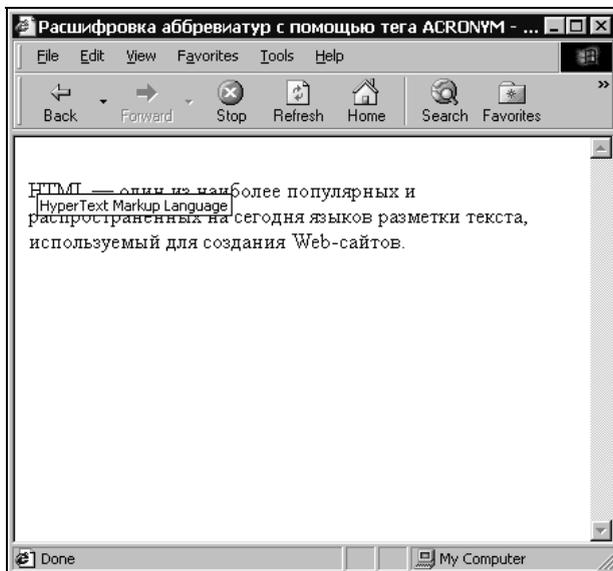


Рис. 2.1. Использование тега `<ACRONYM>`

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 `` и `<I>`;
- ❑ `<CODE><CODE>` — предназначен для визуального выделения небольших фрагментов программного кода. Код, размещенный в этом парном теге, отображается моноширинным шрифтом;
- ❑ `` — используется для обозначения удаленного текста. Имеет необязательные для указания параметры `DATE` и `CITE` (первый показывает дату удаления, второй ссылается на источник причины удаления). Текст, заключенный между тегами ``, отображается зачеркнутым и аналогичен действию тега `<S>` или `<STRIKE>`;
- ❑ `` — используется для интонационного выделения определенного текстового фрагмента (листинг 2.2). Браузерами отображается курсивом (рис. 2.2) (так же, как теги `<CITE>` и `<I>`);

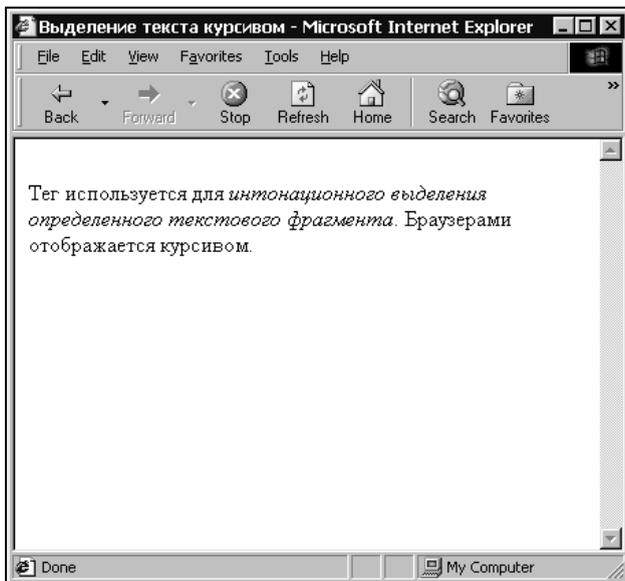


Рис. 2.2. Использование тега ``

Листинг 2.2. Использование тега ``

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