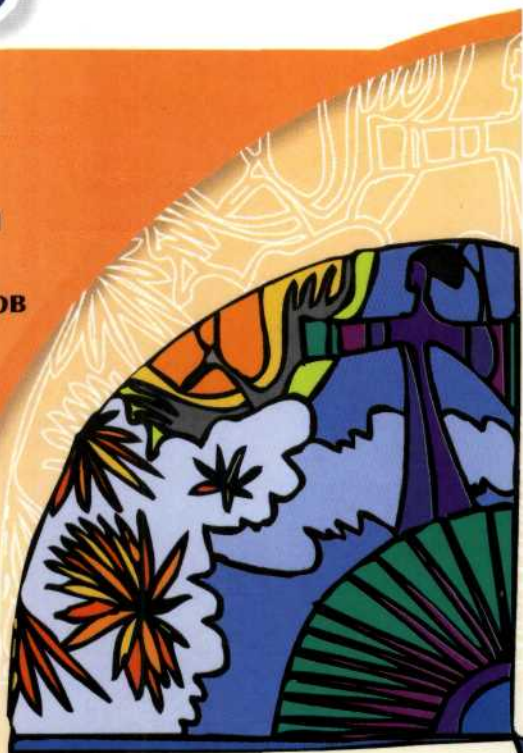


Macromedia DREAMWEAVER

8

Владимир Дронов

- Основы Web-дизайна за 14 занятий
- Создание Web-страниц и Web-сайтов
- Фреймовый, табличный и контейнерный Web-дизайн
- Полезные советы и маленькие хитрости



Владимир Дронов

САМОУЧИТЕЛЬ

Macromedia

DREAMWEAVER

8

Санкт-Петербург

«БХВ-Петербург»

2006

УДК 681.3.06
ББК 32.973.26-018.2
Д75

Дронов В. А.
Д75 Самоучитель Macromedia Dreamweaver 8. — СПб.: БХВ-Петербург, 2006. — 320 с.: ил.

ISBN 5-94157-833-4

Описывается работа в новой версии визуального Web-редактора Macromedia Dreamweaver 8. Книга организована в виде 14-ти занятий по Web-дизайну. Рассмотрены современные стандарты Интернета, интерфейс программы, создание страниц и сайтов, работа с текстами, графикой, таблицами и стилями, различные виды Web-дизайна: фреймовый, табличный и контейнерный. Освещены вопросы использования шаблонов, плавающих и свободно позиционируемых контейнеров, разработки Web-сценариев. Уже с первых страниц начинающий пользователь может быстро приступить к работе и создать простейший Web-сайт. Изучение всего материала книги позволит читателю создавать современные Web-сайты с оформлением, приближенным к полиграфическому. Особую ценность имеют полезные советы автора, маленькие хитрости и нетрадиционные приемы Web-верстки.

Для начинающих Web-дизайнеров

УДК 681.3.06
ББК 32.973.26-018.2

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Евгений Рыбаков</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Алия Шаулис</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн обложки	<i>Игоря Цырульникова</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 19.10.05.

Формат 70×100^{1/8}. Печать офсетная. Усл. печ. л. 25,8.

Тираж 4000 экз. Заказ № 1381

"БХВ-Петербург", 194354, Санкт-Петербург, ул. Есенина, 5Б.

Санитарно-эпидемиологическое заключение на продукцию № 77.99.02.953.Д.006421.11.04 от 11.11.2004 г. выдано Федеральной службой по надзору в сфере защиты прав потребителей и благополучия человека.

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

Оглавление

Ознакомительное занятие	9
Типографские соглашения.....	10
На первом занятии.....	11
КУРС 1. ПРОСТЕЙШИЙ ТЕКСТОВЫЙ ДИЗАЙН.....	13
Занятие 1. Что такое Интернет и как он работает.....	15
Основные принципы работы Интернета	15
Что такое Интернет	15
Сервисы Интернета	17
Клиенты и серверы	18
Протоколы.....	20
Интернет-адреса	22
Основные понятия WWW.....	25
Web-страницы и Web-сайты	25
Web-обозреватели.....	28
Web-серверы	30
Публикация Web-сайта в Интернете. Хостинг-провайдеры	31
На следующем занятии.....	32
Занятие 2. Знакомство с Macromedia Dreamweaver 8	33
Зачем нужны Web-редакторы?.....	33
Среда Dreamweaver 8	34
Выбор рабочей среды.....	34
Главное окно программы.....	35
Управление окнами и панелями Dreamweaver	40
Вызов справки.....	42
Настройка Dreamweaver.....	43
На следующем занятии.....	47

Занятие 3. Работа с текстом.....	49
Создание новой Web-страницы.....	49
Ввод и форматирование текста.....	50
Набор текста Web-страницы.....	51
Сохранение и открытие Web-страниц в Dreamweaver.....	53
Просмотр Web-страницы в Web-обозревателе.....	54
Форматирование абзацев.....	54
Форматирование фрагментов текста.....	61
Работа с кодом HTML.....	66
Три режима отображения Web-страницы.....	66
Введение в язык HTML.....	68
Теги HTML. Форматирование текста.....	68
Вложенность тегов.....	70
Служебные теги. Две секции Web-страницы.....	71
Средства Dreamweaver для работы с тегами HTML.....	72
Создание гиперссылок.....	74
Вставка специальных символов.....	78
На следующем занятии.....	82
Занятие 4. Работа с Web-сайтами.....	83
Подготовка к публикации сайта.....	84
Регистрация сайта в Dreamweaver.....	84
Работа с файлами сайта. Панель <i>Files</i>	91
Взаимодействие панели <i>Files</i> и окна документа.....	94
Проверка Web-страниц.....	95
Проверка правильности HTML-кода.....	95
Проверка гиперссылок.....	97
Публикация сайта.....	98
На следующем занятии.....	103
КУРС 2. БОЛЕЕ СЛОЖНЫЙ ТЕКСТОВЫЙ ДИЗАЙН.....	105
Занятие 5. Работа с графикой.....	107
Графика на Web-страницах. Внедренные элементы.....	107
Работа с графическими изображениями.....	109
Вставка графического изображения.....	109
Параметры графического изображения.....	113
Специальные изображения.....	117
Изображения-гиперссылки.....	117
Активные изображения.....	118
Карты-изображения.....	120
Дополнительные возможности по работе с графикой.....	124
На следующем занятии.....	127

Занятие 6. Работа с таблицами 129

Простые таблицы.....	129
Создание таблиц	129
Работа с таблицами.....	133
Средства HTML, используемые для создания таблиц.....	136
Форматирование таблиц	137
Выделение элементов таблиц.....	137
Параметры ячейки.....	140
Параметры строки.....	142
Параметры таблицы	143
Объединение ячеек таблиц	145
На следующем занятии.....	148

Занятие 7. Использование стилей CSS 149

Понятие о стилях CSS	149
Создание и использование стилей CSS	151
Каскадность CSS и ее использование	156
Краткое введение в язык CSS.....	158
Возможности CSS по оформлению Web-страниц	160
Параметры шрифта.....	161
Параметры фона	163
Параметры абзаца.....	165
Параметры размеров и размещения.....	167
Параметры рамки	169
Параметры маркеров списка.....	171
Управление стилями.....	172
Особые случаи применения стилей CSS	173
Псевдостили гиперссылок	173
Контейнеры	174
Общие параметры Web-страницы.....	176
На следующем занятии.....	177

КУРС 3. ФРЕЙМОВЫЙ И ТАБЛИЧНЫЙ ДИЗАЙН 179**Занятие 8. Фреймовый дизайн 181**

Введение во фреймы и наборы фреймов.....	181
Работа с фреймами в Dreamweaver	183
Создание фреймов	183
Формирование фреймов.....	190
Параметры фреймов и наборов фреймов	191
Параметры наборов фреймов.....	191
Параметры фреймов	193
Создание Web-страниц для набора фреймов	194
Недостатки фреймов и их преодоление	195

На следующем занятии..... 196

Занятие 9. Табличный дизайн..... 197

Режим разметки страниц и его использование 198

Режим разметки страниц..... 198

Работа с таблицами и ячейками разметки 204

Форматирование таблиц и ячеек разметки..... 209

Параметры ячеек разметки..... 209

Задание ширины ячеек..... 210

Параметры таблицы разметки..... 213

Создание страниц, использующих табличный Web-дизайн 214

Недостатки табличного дизайна и их преодоление 215

На следующем занятии..... 217

Занятие 10. Использование шаблонов..... 219

Введение в шаблоны 219

Работа с шаблонами в Dreamweaver 220

Создание шаблона 220

Правка шаблона 222

Создание изменяемых областей 223

Создание Web-страниц на основе шаблонов..... 226

Обновление страниц, созданных на основе шаблонов 228

Управление шаблонами в списке панели *Assets*..... 229

Специальные области шаблонов 230

Необязательные области 230

Повторяющиеся области 233

На следующем занятии..... 235

КУРС 4. КОНТЕЙНЕРНЫЙ WEB-ДИЗАЙН 237

Занятие 11. Плавающие контейнеры 239

Простейший случай контейнерного Web-дизайна 240

Стили CSS, предназначенные для управления контейнерами..... 242

Пример контейнерного Web-дизайна 246

Пример реализации саморастягивающихся контейнеров 251

Недостатки контейнерного Web-дизайна..... 258

На следующем занятии..... 259

Занятие 12. Свободно позиционируемые контейнеры..... 261

Работа со свободно позиционируемыми контейнерами 262

Создание свободно позиционируемых контейнеров 262

Средства HTML и CSS для создания свободно позиционируемых контейнеров..... 266

Параметры свободно позиционируемых контейнеров..... 267

Работа с группой свободно позиционируемых контейнеров.....	269
Использование панели <i>Layers</i>	271
Недостатки свободно позиционируемых контейнеров и их преодоление	273
На следующем занятии.....	274
КУРС 5. ПОСЛЕДНИЕ ШТРИХИ.....	275
Занятие 13. Использование Web-сценариев.....	277
Поведения Dreamweaver и их использование	278
Работа с поведением в панели <i>Behaviors</i>	278
Поведения, поддерживаемые Dreamweaver.....	282
Перенаправление на разные Web-страницы в зависимости от версии Web-обозревателя (<i>Check Browser</i>)	282
Перенаправление на другую Web-страницу (<i>Go To URL</i>)	284
Открытие нового окна Web-обозревателя (<i>Open Browser Window</i>).....	285
Воспроизведение звукового файла (<i>Play Sound</i>)	287
Вывод окна-предупреждения (<i>Popup Message</i>)	287
Задание нового содержимого фрейма (<i>Set Text of Frame</i>)	288
Задание нового содержимого свободно позиционируемого контейнера (<i>Set Text of Layer</i>).....	289
Вывод произвольного текста в строке статуса окна Web-обозревателя (<i>Set Text of Status Bar</i>)	289
Показ и скрытие свободно позиционируемых контейнеров (<i>Show-Hide Layers</i>)	290
Смена изображения (<i>Swap Image</i>)	291
Восстановление всех изначальных изображений (<i>Swap Image Restore</i>)	292
Недостатки Web-сценариев и их преодоление	292
На следующем занятии.....	294
Занятие 14. Маленькие хитрости	295
Подбор цветов для сайта.....	295
Оформление гиперссылок	295
Оформление таблиц	296
Оформление полос прокрутки.....	298
Добавление иконки к пункту <i>Избранное</i>	301
Создание врезки к статье	301
Создание полосы навигации.....	302
Что дальше?	304
Заключение.....	305
Предметный указатель.....	309

Ознакомительное занятие

Все готовы? Тогда начнем наши занятия по Web-дизайну.

Интернет уже прочно вошел в нашу жизнь. Сейчас уже трудно представить, как люди обходились без него раньше, как искали свежие прогнозы погоды и курсы валют, как добывали последние новости, аналитические материалы, не обязательно совпадающие с текущей политикой правительства, документацию по тем же самым компьютерам, свежие версии программ и, что греха таить, рефераты. Да и вообще, как они раньше общались, новое поколение уже слабо себе представляет.

Интернет сейчас — это, прежде всего, Всемирная паутина WWW. Всемирная паутина — это Web-сайты. Web-сайты — это Web-страницы. А Web-страницы — это и есть Web-дизайн.

Вот как раз Web-сайтами, Web-страницами и Web-дизайном мы и займемся. Мы будем овладевать Web-премудростями на четырнадцать занятий, из которых состоит этот самоучитель. А поможет нам ей овладеть замечательный программный пакет Macromedia Dreamweaver 8.

Что такое Macromedia Dreamweaver? Это Web-редактор, иначе говоря, инструмент для создания Web-страниц, невероятно облегчающий нелегкий труд Web-дизайнера. С его помощью прикоснуться к интернет-технологиям может даже начинающий, не искушенный в них пользователь. Ну, а опытный Web-дизайнер выжмет из него все — уж будьте уверены.

Macromedia Dreamweaver — пакет с долгой историей. Первая его версия была выпущена фирмой Macromedia еще в далеком 1998 году. Dreamweaver 1.0 быстро получил популярность в России благодаря своей "благосклонности" к русскому языку и снисходительному отношению к множеству русскоязычных кодировок. (О кодировках мы поговорим на *занятии 2*, а пока что знаете, что кодировка — это способ представления компьютером символов алфа

вита, в нашем случае — кириллицы.) Последняя на данный момент версия Dreamweaver, уже седьмая по счету, носит вновь входящее в моду числовое наименование "8".

Автор этой книги пользуется Dreamweaver, начиная с версии 2.0, и весьма доволен этой программой. В частности, именно с помощью Dreamweaver он создал и поддерживает свой личный сайт, доступный по адресу <http://vgi.volsu.ru:8000/~Vladimir.Dronov>, и несколько других сайтов. Что касается написанных автором книг, то по Dreamweaver эта — пятая.

Как-то так получилось, что Dreamweaver оказался в тени своего более разрекламированного собрата Microsoft FrontPage. Возможно, это потому, что Dreamweaver все-таки предназначен для достаточно опытных Web-дизайнеров, знающих свое дело и занимающихся им профессионально. FrontPage же изначально разрабатывался как инструмент для обычных офисных служащих, занимающихся Web-дизайном от случая к случаю. Что ж, мир велик, и в нем найдется место для любой программы...

Но не следует думать, что Dreamweaver — это что-то запредельно сложное. Он весьма благосклонен к новичкам, в чем мы убедимся уже на *занятии 2*. Но все же полную его мощь раскроет только опытный Web-дизайнер.

Но опыт — дело наживное. Задача автора этой книги — дать вам начальные сведения о Web-дизайне в среде пакета Dreamweaver. Мы будем изучать его, попутно создавая настоящий Web-сайт, посвященный выдуманному студенту Васильеву Василию Васильевичу. Такая практика поможет нам закрепить полученные знания, приобрести нужные навыки и получить тот самый опыт, который позволит вам укротить Dreamweaver.

Но хватит пустой болтовни! Пора начинать первый курс занятий. Тем более что узнать нам предстоит очень много.

Типографские соглашения

Но прежде чем начать изучение Web-дизайна, давайте кое о чем условимся.

В этой книге будут приведены примеры на языке HTML (забегая вперед — этот язык используется для создания Web-страниц). При написании примеров были использованы типографские соглашения, уже ставшие своего рода стандартами в компьютерном книгоиздании. Нам необходимо их знать.

◆ В угловые скобки (<>) заключаются названия параметров или фрагментов кода. В код реального сценария, разумеется, должен быть подставлен реальный параметр или реальный код. Например:

Здесь вместо подстроки Интернет-адрес должен быть подставлен реальный интернет-адрес.

- ◆ В квадратные скобки ([1]) заключаются необязательные фрагменты кода. Например:

```
htm[1];
```

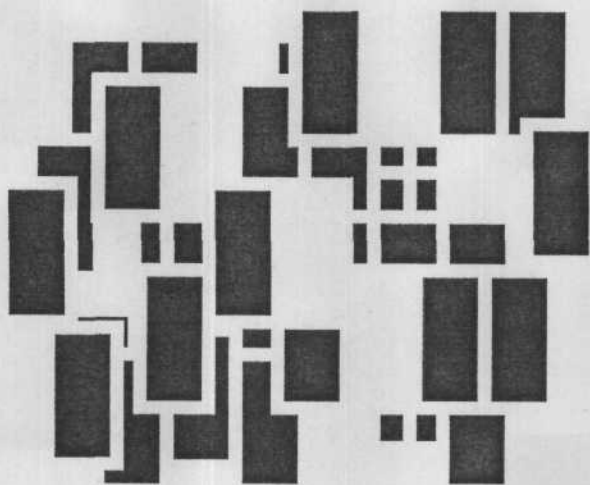
Последняя буква 1 может присутствовать, а может и не присутствовать.

Весь остальной код HTML набирается "как есть".

Теперь можно начинать. И начнем мы с теории.

На первом занятии...

Всякой практике обязательно предшествует теория. Прежде чем знакомиться с Dreamweaver, нам нужно узнать как можно больше о современных интернет-технологиях. Им-то как раз и будет посвящено первое занятие первого курса. Переверните страницу и...

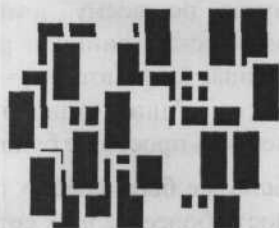


КУРС 1

Простейший текстовый дизайн

- Занятие 1.** Что такое Интернет и как он работает
- Занятие 2.** Знакомство с Macromedia Dreamweaver 8
- Занятие 3.** Работа с текстом
- Занятие 4.** Работа с Web-сайтами

ЗАНЯТИЕ 1



Что такое Интернет и как он работает

Действительно, что такое Интернет? Как он работает? А вот я слышал в автобусе слово "WWW" и не знаю, что это такое... Что представляют собой и как делаются те красивые Web-странички, которые выводит нам Web-обозреватель? И откуда, в конце концов, они берутся?

Терпение, терпение и еще раз терпение. На нашем первом — обзорном — занятии мы получим ответы на все эти вопросы. И начнем с самого первого...

Основные принципы работы Интернета

Сначала мы поговорим о том, что такое Интернет и как он работает — рассмотрим некоторые общие вопросы.

Что такое Интернет

В самом деле, что такое Интернет? Электронный океан, таинственная стихия, заключенная в кремниевых кристаллах и медных проводах современных компьютеров. Несуществующая вселенная, иной раз кажущаяся более реальной, чем наш материальный, "настоящий" мир. То, что вторгается в каждый дом, опутывает всю планету и сознание всех людей тугой медно-кремниевой паутиной; нечто запредельное, непостижимое людскому разуму, никому не видимое, но всеми ощущаемое...

Но довольно! Вы слишком много читаете фантастики (или желтых газет типа "Церковного вестника", что суть та же фантастика, только плохая). На самом деле, все намного проще.

Итак, *Интернет* — это всемирная компьютерная сеть. Ее, кстати, так часто и называют: Всемирная Сеть или даже просто Сеть с большой буквы. Прост

нутая по всему земному шару паутина медных проводов, волоконно-оптических линий и радиоканалов, связывающих друг с другом многочисленные компьютеры, — вот что такое Интернет. Разумеется, все здесь подчиняется общим стандартам (о которых мы поговорим далее), а иначе эта сеть просто не будет работать.

Если же быть совсем точным, то Интернет — это не единая сеть, а совокупность более мелких сетей, связанных друг с другом общими каналами и стандартами. Таких сетей превеликое множество: огромные территориальные сети, раскинувшиеся на целые области, штаты и государства, и ведомственные сети, объединяющие родственные организации, и локальные компьютерные сети отдельных организаций, и так называемые кампусные сети — сети, объединяющие компьютеры одного или нескольких близлежащих районов города. Благодаря проложенным между ними каналам высокоскоростной связи, они составляют единое целое, имя которому Интернет.

Даже частные пользователи, подключающиеся к Интернету по модему, выделенной линии или поддерживающему такую возможность сотовому телефону, тоже по сути дела являются частью Сети. Так что когда мы включаем наш модем и дозваниваемся до нашего *интернет-провайдера* (организации, предоставляющей пользователям доступ в Интернет), то приобщаемся к единому целому. А что, разве это не повод для законной гордости?

Сеть Интернет имеет одну замечательную особенность — она очень устойчива к сбоям. Так, если где-то порвется провод, мы этого не заметим. А все потому, что данные, которые мы запрашиваем, пойдут в этом случае по другому проводу. Специалисты говорят, что Интернет децентрализован — он не имеет единого центра, из которого ведется управление пересылкой данных, поэтому в случае аварии автоматически переконфигурируется и продолжает нормально работать.

Еще одна замечательная особенность Интернета — его глобальность, всемирность. Не вставая из-за компьютера, мы можем совершить путешествие по всему миру, побывать в США, Австралии, Германии, Зимбабве, на Огненной Земле и даже в Антарктиде (да, и туда протянулись вездесущие провода!). Для этого нужно всего лишь набрать нужный нам адрес.

Интернет имеет достаточно долгую и бурную историю. Он появился еще в первой половине 70-х годов XX века, когда американское Министерство обороны финансировало проект создания компьютерной сети, устойчивой к сбоям. Разумеется, создавалась эта сеть для нужд обороны, да и название имела другое — ARPANET. Позднее же, в начале 80-х, эта сеть отошла к ученым, а военные приступили к созданию другой сети, которой пользуются до сих пор. И в то же самое время ARPANET был переименован в Internet, или, если по-русски. Интернет.

Первоначально, еще во времена ARPANET, эта сеть использовалась для пересылки электронной почты и обмена файлами. Web-странички, ради которых мы, в основном, и путешествуем по Сети, появились только в конце 80-х. Именно тогда Интернет и "пошел в народ", перестав быть сетью ученых и превратившись в сеть для всех.

В Россию, точнее, в СССР, Интернет пришел в 1990 году, но популярность среди широких масс компьютерщиков приобрел только в середине 90-х. В настоящее же время в России, наверно, и не найти человека, не слышавшего об Интернете. Вы такого встречали? Автор — еще нет.

На заметку

Говорят, в первой польской энциклопедии, изданной, кажется, в XVII столетии, термин "лошадь" описывался так: "что такое лошадь, знают все". То же самое можно сейчас сказать об Интернете. (Вот только можно ли сейчас сказать то же самое о лошади?..)

Сервисы Интернета

Раз уж мы заговорили об услугах, предоставляемых Интернетом, или, как говорят профессионалы, *сервисах* Интернета, то давайте узнаем о них побольше. В конце концов, нам ими пользоваться...

Итак, самый старый и самый популярный до сих пор сервис Интернета — это электронная почта (e-mail). Ежедневно в мире отправляются и принимаются сотни миллионов электронных писем, и это количество в будущем будет только увеличиваться. В самом деле, электронная почта доступна, удобна, быстра и бесплатна, в отличие от почты "бумажной", которую пользователи Интернета уже успели презрительно прозвать "улиточной" (по-английски — snail mail). Конечно, эти доступность, удобство, быстрота и бесплатность имеют и некоторые недостатки, вроде "спама" — несанкционированных рекламных рассылок, но эти недостатки вполне можно стерпеть.

Еще один сервис Интернета, почти такой же старый, как почта, — это пересылка файлов. Пользователи Интернета называют его FTP (File Transfer Protocol, протокол передачи файлов; почему так — мы узнаем чуть позже). Сейчас FTP уже не имеет той популярности, как на заре существования Интернета, но все еще довольно часто используется.

Третий сервис Интернета — это Всемирная паутина, или WWW (World Wide Web, повсеместно протянутая паутина), или просто Web, те самые Web-страницы и Web-сайты, которые мы просматриваем в Web-обозревателе. Появившийся значительно позже электронной почты и FTP, WWW стала самым популярным сервисом и, собственно, превратила Интернет из сети уче-

Остальные сервисы Интернета (а их немало) мы только упомянем. Это новости UseNet, потоковое вещание, интернет-пейджеры, чаты, нашумевшие в последнее несколько лет файлообменные сети и некоторые другие, менее известные сервисы.

Клиенты и серверы

Но каким образом мы пользуемся всем тем богатством, что дает нам всемирная сеть? С помощью особых программ! Это Web-обозреватель, клиент электронной почты, программа просмотра интернет-телевидения и интернет-радио, ICQ и "чатилка". Все они очень хорошо нам знакомы.

Но программ, используемых для предоставления нам сервисов Интернета, гораздо больше. И очень многие из них нам, если так можно сказать, "не видны", т. е. мы не общаемся с ними напрямую. Вообще, существуют два совершенно разных вида интернет-программ. И сейчас мы о них поговорим.

Программы, относящиеся к первому виду, — это Web-обозреватели, клиенты электронной почты, чатов, интернет-пейджеры, в общем, все те, с которыми мы имеем дело непосредственно. Мы получаем с их помощью различную информацию из Сети и работаем с ней. Такие программы называются программами-клиентами (а компьютеры, на которых они работают, — наши с вами компьютеры! — клиентскими).

Да, но как программы-клиенты получают из Сети нужную нам информацию (Web-страницы, файлы, письма и пр.)? Очень просто — для этого они обращаются к другим программам, относящимся ко второму виду. Это программы-серверы, работающие на серверных компьютерах, где хранится и запрашиваемая клиентами информация. Существуют Web-серверы, серверы электронной почты, чата, интернет-пейджеров, потокового вещания и пр.

На заметку

Очень часто понятие "сервер" распространяется и на серверный компьютер, и на саму программу-сервер. Это, вообще-то, неправильно, т. к. на одном серверном компьютере может быть установлено несколько программ-серверов, но вошло в практику.

Процесс получения информации клиентами от сервера включает пять шагов.

1. Пользователь запрашивает с помощью программы-клиента некую информацию.
2. Клиент устанавливает *соединение* с сервером и посылает тому особый информационный блок, называемый *клиентским запросом*. Этот запрос имеет жестко определенный формат, чтобы сервер его понял.

- Сервер извлекает нужный клиенту файл или фрагмент данных, записанных в файле, и посылает его клиенту в виде другого информационного блока — *серверного ответа*. Разумеется, этот ответ также имеет жестко определенный формат. Если же нужных данных нет, или сервер почему-то не смог понять клиентский запрос, он возвращает *сообщение об ошибке* — информационный блок, содержащий описание возникшей ошибки.
- Клиент получает ответ от сервера, расшифровывает его и выдает полученную информацию пользователю. Если получено сообщение об ошибке, клиент сообщает об этом пользователю либо предпринимает какие-то действия самостоятельно. После принятия ответа от сервера клиент разрывает соединение с ним.

Процесс отправки клиентом данных серверу также включает те же пять шагов.

- Пользователь вводит в программу-клиент отправляемую информацию.
- Клиент устанавливает соединение с сервером и посылает тому отправляемую информацию в составе клиентского запроса. При этом отправляемая информация, как правило, особым образом шифруется.
- Сервер принимает запрос, расшифровывает его и извлекает отправленную информацию.
- Сервер записывает отправленную клиентом информацию в файл, помещает в базу данных или куда-то еще. После этого в случае успешной записи он отправляет клиенту в составе ответа так называемое *подтверждение* — информационный блок, сообщающий о том, что все прошло нормально. Если у сервера возникли проблемы с приемом информации, он отправляет сообщение об ошибке.
- Клиент получает ответ от сервера, расшифровывает его и уведомляет пользователя об успешной или неуспешной отправке данных либо предпринимает какие-то действия самостоятельно. После принятия ответа от сервера клиент разрывает соединение с ним.

Внимание!

Из всего описанного выше видно, что любой обмен данными между клиентом и сервером всегда начинается клиентом. Серверу же в этом процессе отведена подчиненная роль.

Мы только что познакомились с особой *архитектурой* (принципом построения компьютерных систем), называемой *двухзвенной*, или архитектурой "*клиент-сервер*". Эта архитектура использует два вида программ — клиенты и серверы, — выполняющие разные роли. Она используется для реализации

На заметку

Некоторые интернет-сервисы, в частности файлообменные сети (Napster, Gnutella, Kazaa и др.), используют другую архитектуру — *однозвенную*. Здесь все компьютеры, подключенные к Интернету и реализующие этот сервис, фактически равны между собой; любой из них может выступать в роли как клиентского (запрашивать информацию у других компьютеров), так и серверного (предоставлять хранящуюся на нем информацию другим компьютерам). Само собой, здесь используется особое программное обеспечение, которое может работать и как клиент, и как сервер.

Как правило, серверные компьютеры — настоящие монстры, содержащие несколько процессоров, огромные дисковые массивы, мощные каналы для подключения к Интернету и специальное программное обеспечение. Все в них нацелено на то, чтобы обслужить как можно больше клиентов за как можно меньшее время. Но часто, если клиентов и запросов оказывается слишком много, ресурсов серверного компьютера не хватает, начинаются проблемы. Они могут проявляться в том, что сервер просто отказывается обслужить "лишних" клиентов, предлагая им подождать немного, когда нагрузка чуть-чуть снизится, а то и в том, что могучий серверный компьютер просто-напросто "зависает". Такое тоже случается, и не так уж редко...

Ну да не будем о грустном! Не стоит начинать знакомство с таким притягательным миром интернет-технологий со столь печальных вещей, как системные сбои. Чем их меньше, и чем реже они случаются, тем лучше для всех нас.

Протоколы

Люди, чтобы понимать друг друга, должны разговаривать на одном языке. Точно так и с компьютерами, подключенными к сети, неважно какой — всемирной или локальной. Обмен данными по этим сетям должен проходить по единым стандартам, иначе начнется новое вавилонское столпотворение.

Стандарт, по которому кодируются данные для отправки по сети, называется *протоколом*. В Интернете для обмена данными используются несколько протоколов, которые мы здесь вкратце рассмотрим.

Основные протоколы, используемые в Интернете, — это IP (Internet Protocol, межсетевой протокол) и TCP (Transfer Control Protocol, протокол управления передачей). Это так называемые *протоколы низкого уровня*, определяющие самые основные параметры передаваемых данных: длину отдельных порций (*пакетов*) данных, способ кодирования, указания адресов получателя и отправителя, а также защиту от ошибок. Можно сказать, что они занимаются исключительно передачей данных по каналам Интернета, не вникая, что же именно они передают.

Протокол IP занимается тем, что "упаковывает" передаваемые данные в пакеты и помещает в каждый пакет адрес компьютера отправителя и компьютера

ра-получателя. Протокол TCP, базирующийся на IP, обеспечивает гарантированную отправку данных, т. е. следит за тем, чтобы ни один пакет не потерялся в пути, а также разбивает слишком объемные массивы данных на несколько пакетов, а потом собирает их вновь. Эти два протокола настолько взаимосвязаны друг с другом, что часто эту парочку называют одним словом TCP/IP, а иногда даже считают за один протокол.

TCP/IP используется другими протоколами, уже *высокого уровня*. Эти протоколы описывают формат клиентских запросов и серверных ответов: особые команды, пересылаемые клиентом серверу при запросе или передаче данных, и способ представления передаваемой информации. Собственно передачей этих данных занимается TCP/IP.

На заметку

Строго говоря, существуют еще *протоколы физического уровня*, располагающиеся даже "ниже" TCP/IP. Они определяют электрические параметры сигнала, кабелей, разъемов и пр.

Каждый сервис Интернета использует свой собственный высокоуровневый протокол (а то и несколько, предназначенных для разных задач или разработанных конкурирующими организациями). Давайте рассмотрим протоколы, с которыми мы столкнемся в будущем.

Начнем мы, конечно, с WWW. Для передачи данных Всемирная паутина использует протокол HTTP (HyperText Transfer Protocol, протокол передачи гипертекста). Он задает набор команд для запроса данных и управления ими, пересылаемых клиентом (Web-обозревателем) Web-серверу, и способы представления пересылаемых в обе стороны данных. Пожалуй, это самый широкоизвестный протокол Интернета — всем более-менее грамотным интернетчикам знакомы эти четыре буквы.

Сервис пересылки файлов FTP использует протокол, который так и называется — FTP. Он также определяет набор команд для управления файлами на сервере (загрузка с сервера, помещение на сервер, копирование, перемещение, удаление, создание папки и т. д.) и способы кодирования файлов для пересылки по каналам связи. В этом смысле протоколы HTTP и FTP весьма похожи.

А вот электронная почта использует целых два протокола. Первый протокол — SMTP (Simple Mail Transfer Protocol, простой протокол пересылки почты) — используется для пересылки почты клиентом серверу. Для получения же почты от сервера клиент общается с ним по протоколу POP3 (Post-Office Protocol, протокол почты).

Вообще-то существует еще один почтовый протокол — IMAP (Internet Mes-

следник" более старого POP3, он предоставляет больше возможностей, но распространен не так широко.

Каждый высокоуровневый протокол передает свои данные по определенному порту — своего рода воображаемому каналу, созданному стараниями протокола IP внутри реального провода. Любой канал передачи данных Интернета разделен на 65 535 небольших пронумерованных "канальчиков" — именно столько портов предусматривает протокол IP. В табл. 1.1 перечислены некоторые протоколы и "занимаемые" ими по умолчанию порты.

Таблица 1.1. Порты IP, используемые по умолчанию для передачи данных некоторых протоколов высокого уровня

Протокол	Используемый порт
HTTP	80
FTP	21
SMTP	25
POP3	110

Все более-менее серьезные серверы предоставляют возможность изменить порт, используемый протоколом, которые эти серверы обслуживают, на другой. Например, Web-сервер может быть настроен так, чтобы использовать для "общения" с клиентами не 80-й порт, а, скажем, 8000-й. (Автору этой книги время от времени встречаются Web-серверы, настроенные таким образом.) Но так поступают только в крайних случаях, чтобы не вводить в заблуждение пользователей.

Интернет-адреса

Теперь давайте поговорим о том, каким образом идентифицируются компьютеры, подключенные к Интернету. А именно — об интернет-адресах.

Интернет-адрес — это уникальное числовое или строковое значение, позволяющее точно идентифицировать компьютер в Сети. Именно такой интернет-адрес (точнее, два — отправителя и получателя) подставляется в каждый отправляемый по Сети пакет IP, чтобы он успешно дошел до места назначения.

На заметку

Существует, правда, возможность дать одному компьютеру сразу несколько интернет-адресов. Но используется это нечасто и в особых случаях. И в дальнейшем для простоты мы будем считать, что один интернет-адрес — это один компьютер.

На заре эпохи Интернета в качестве интернет-адреса использовался *IP-адрес* — числовое значение, идентифицирующее компьютер для протокола IP. IP-адрес замечательно подходит для компьютеров, но очень плохо — для людей. Он имеет такой вид:

192.168.1.10

Не очень-то наглядно, правда? Именно поэтому с расширением Интернета была введена в строй новая система интернет-адресов, которой мы пользуемся до сих пор. Это так называемые доменные адреса, о которых стоит поговорить подробно.

Но прежде чем мы начнем разговор о доменных адресах, давайте выясним, что такое домен. *Домен*, или *доменная зона*, — это участок Интернета, созданный для удобства управления им. Такой участок может быть крупным или мелким или вообще состоять из одного компьютера. Каждый домен обозначается строкой текста, состоящей из английских букв.

Структура доменов похожа на матрешку: мелкие домены "вложены" внутри крупных, а крупные, в свою очередь, — внутри гигантских. Гигантские домены называются *доменами верхнего уровня*, а вложенные в них более мелкие — *доменами нижнего уровня*.

Домены верхнего уровня бывают интернациональными и национальными. *Интернациональные домены* объединяют компьютеры по какому-то признаку; к ним относятся домены com (коммерческие серверы), edu (образовательные), mil (военные), org (организации, не занимающиеся компьютерами и Интернетом), net (организации, занимающиеся компьютерами и Интернетом) и некоторые другие. *Национальные домены* объединяют компьютеры по территориальному признаку и выдаются целым странам; это домены us (США), uk (Великобритания), fr (Франция), de (Германия), ru (Россия) и др.

Домены нижнего уровня выдаются, как правило, отдельным организациям или, опять же, по территориальному признаку. Их текстовое обозначение часто совпадает с названием этой организации или района.

Если теперь записать обозначения всех доменов, в которых находится нужный нам компьютер, в порядке от более мелких к более крупным, разделив их точками, мы получим *доменное имя* этого компьютера. Так, если у нас сам компьютер имеет имя comp45, отдел, в котором он стоит, — buh (бухгалтерия), организация, включающая этот отдел, — department, а страна — ru (Россия), то мы получим такое доменное имя:

comp45.buh.department.ru

Согласитесь — запомнить это гораздо проще, чем невразумительный IP-адрес.

Да, но проблема в том, что протокол TCP/IP не понимает доменные имена! Что делать? Как преобразовать доменное имя в понятный ему IP-адрес?

Для этого используется особый сервис Интернета, называемый *DNS* (Domain Name System, система доменных имен). Клиент отправляет серверу *DNS* запрос, содержащий доменное имя, и получает в виде ответа IP-адрес, соответствующий этому доменному имени. А уж с IP-адресом он знает, что делать.

Такие серверы *DNS* имеются в каждом домене; кроме того, несколько самых мощных в мире серверов *DNS* (*корневые серверы DNS*) находятся как бы "выше" всех доменов, даже доменов верхнего уровня. И всем им хватает работы.

Всем хороши доменные имена, кроме одного, — они не позволяют задать номер порта IP или хотя бы протокол. Они только задают сам серверный компьютер, а ведь на одном серверном компьютере могут работать несколько программ-серверов. Что делать? Просто указать перед доменным именем обозначение протокола, реализуемого нужным сервером, вот так (обозначение протокола выделено полужирным шрифтом):

```
http://comp45.buh.department.ru
ftp://comp45.buh.department.ru
```

В первом случае мы обращаемся к Web-серверу, а во втором — к серверу FTP, находящимся на одном и том же компьютере comp45.buh.department.ru.

Есть возможность указать прямо номер порта IP, записав его после доменного имени серверного компьютера через двоеточие. Вот так (номер порта выделен полужирным шрифтом):

```
http://comp45.buh.department.ru:8000
```

Многие серверы (почтовые, FTP и др.) требуют от пользователя ввода его имени и, возможно, пароля. Имя пользователя помещается между названием протокола и самим доменным именем и отделяется от последнего знаком "коммерческое эт" (@). Вот два примера задания имени пользователя в доменном имени сервера (выделено полужирным шрифтом):

```
ftp://user@comp45.buh.department.ru
account@server.ru
```

Последний пример демонстрирует нам обычный адрес электронной почты. Заметим, что название протокола здесь не указывается — почтовый клиент и почтовый сервер сами знают, какой протокол использовать.

Ну а пароль пользователя помещается между именем и знаком (@) и отделяется от имени двоеточием — вот так (выделен полужирным шрифтом):

```
ftp://user:password@comp45.buh.department.ru
```

Ну вот, с основными принципами работы Интернета мы ознакомились. Теперь давайте сосредоточимся на WWW — в основном, именно этим сервисом мы будем пользоваться на протяжении всей книги.

Основные понятия WWW

Здесь мы узнаем все о Web-страницах и Web-сайтах, выясним, чем сайт отличается от страницы, поговорим о Web-клиентах и Web-серверах и познакомимся с несколькими новыми понятиями.

Web-страницы и Web-сайты

Что такое *Web-страница*? Ответить на этот вопрос могут многие. Это интернет-документ, предназначенный для распространения через Интернет посредством сервиса WWW. А если уж говорить по-простонародному, это то, что показывает в своем окне программа-клиент для просмотра Web-страниц — *Web-обозреватель*.

С технической точки зрения Web-страница — это обычный текстовый файл, который можно создать в любом текстовом редакторе, том же Блокноте, стандартно поставляемом в составе Windows. Этот файл содержит собственно текст Web-страницы и различные команды форматирования этого самого текста. Команды форматирования называются *тегами*, а описывает их особый язык HTML (HyperText Markup Language, язык гипертекстовой разметки).

Файлы, содержащие Web-страницы, должны иметь расширение htm[l]. Они сохраняются на жестких дисках серверного компьютера. Получив от Web-обозревателя запрос по протоколу HTTP, *Web-сервер* (серверная программа, обеспечивающая работу сервиса WWW) загружает эти файлы и отправляет Web-обозревателю.

Но как Web-обозреватель дает понять Web-серверу, какая Web-страница ему нужна? Очень просто — он пересылает в составе клиентского запроса имя и полный путь файла, в котором она сохранена. Скажем, вот так:

```
http://comp45.buh.department.ru/somepage.html
```

Этот запрос заставит Web-сервер извлечь и отправить Web-обозревателю файл somepage.html.

А что такое *Web-сайт*? Это набор Web-страниц, подчиненных общей тематике и объединенных в единое целое (как — будет рассказано далее в этой книге). Web-сайт также сохраняется на жестких дисках серверного компьютера в виде набора файлов.

необязательно, но так удобнее, особенно если файлов много и все они разных типов.) Как видим, чисто технических отличий у Web-страницы и Web-сайта не слишком много.

А теперь самое время прояснить несколько, опять же, чисто технических моментов, связанных с работой Web-обозревателя и Web-сервера.

Прежде всего, для хранения всех файлов, составляющих сайт (или Web-страницу, если рассматривать ее как "вырожденный" случай сайта), на диске серверного компьютера создается особая папка, называемая *корневой*. Все файлы сайта должны находиться только в этой папке, без малейших исключений. Все, что не находится в ней, автоматически исключается Web-сервером из состава сайта.

Корневую папку сайта на серверном компьютере создает человек, занимающийся настройкой и обслуживанием программы Web-сервера (или же всего серверного компьютера), — *администратор*. При этом он заносит полный путь этой папки в настройки Web-сервера, чтобы последний смог найти ее на диске. После этого корневая папка может быть заполнена файлами нового сайта.

На заметку

Нужно сказать, что все серьезные программы Web-серверов позволяют создавать так называемые *виртуальные папки*. Виртуальная папка может находиться абсолютно в любом месте файловой системы компьютера, Web-сервер будет считать, что она находится в корневой папке сайта, т. е. является его частью. В дальнейшем и мы будем считать виртуальные папки частью сайта, если это не будет оговорено особо.

Когда Web-обозреватель присылает Web-серверу запрос вида:

`http://www.somesite.ru/somepage.html`

Web-сервер ищет файл `somepage.html` в корневой папке сайта и, если находит, отправляет его Web-обозревателю. Если же такого файла нет или Web-сервер почему-то не может его загрузить, он отправляет Web-обозревателю сообщение об ошибке.

Если Web-обозревателю понадобится файл, находящийся не в самой корневой папке сайта, а в одной из вложенных в нее папок, он должен прислать такой запрос:

`http://www.somesite.ru/somefolder1/somefolder2/somepage.html`

В этом случае Web-сервер отправит Web-обозревателю файл `somepage.html`, находящийся в папке `somefolder1/somefolder2`, вложенной, опять же, в корневую папку сайта.

На заметку

Для обращения к файлу, находящемуся в виртуальной папке, используется аналогичный запрос:

<http://www.somesite.ru/somevirtualfolder1/somepage.html>

Так, все прекрасно, все замечательно и все исключительно ясно! Но мы ведь нечасто набираем в поле ввода интернет-адреса Web-обозревателя такие запросы, указывающие непосредственно на нужную нам Web-страницу. Много чаще наши запросы выглядят чуть "скромнее", например, так:

<http://www.somesite.ru>

То есть они не указывают на файл. Как поступает Web-сервер в таком случае?

Дело в том, что одна из страниц сайта задается в качестве так называемой *страницы по умолчанию*. Именно она отправляется Web-обозревателю, если он не прислал запрос на конкретную страницу (и вообще на конкретный файл). Имя файла этой страницы задается администратором Web-сервера в его настройках — как правило, `default.htm[1]` или `index.htm[1]`.

И если мы наберем в поле ввода интернет-адреса нашего любимого Web-обозревателя нечто, похожее на

<http://www.somesite.ru>

Web-обозреватель выведет нам страницу `default.html`, хранящуюся в корневой папке сайта.

Мы можем прислать Web-серверу и такой запрос:

<http://www.somesite.ru/folder1>

В этом случае Web-сервер отправит нам страницу `default.html`, хранящуюся в папке `folder1`.

Ранее мы рассмотрели так называемые *абсолютные* интернет-адреса, содержащие как адрес самого Web-сервера, так и имя файла нужной Web-страницы. Но интернет-адрес файла можно также указать относительно уже открытой в Web-обозревателе (*текущей*) страницы:

`page2.html`

Получив этот запрос, Web-сервер отправит нам страницу `page2.html`, находящуюся в той же папке, что и текущая. Отметим, что имени сервера этот адрес не включает, т. к. подразумевается, что файл `page2.html` находится на том же сервере, что и файл текущей страницы.

`folder/page2.html`

Этот запрос заставит Web-сервер искать страницу `page2.html` в папке `folder`, вложенной в папку, в которой хранится текущая Web-страница.

```
../folder2/page3.html
```

А этот запрос вернет нам страницу `page3.html` из папки `folder2`, находящейся в той же папке, что и папка, в которой хранится текущая Web-страница.

Осталось только сказать, что интернет-адрес, указывающий имя файла относительно файла текущей страницы и не содержащий имени сервера, так и называется — *относительным*.

А теперь давайте поговорим немного подробнее о программах Web-обозревателей и Web-серверов.

Web-обозреватели

Мы уже знаем, что Web-обозреватели — это программы для просмотра Web-страниц и Web-сайтов. Основная их задача — это отправить Web-серверу корректно, в соответствии со всеми стандартами сформированный клиентский запрос, принять серверный ответ и вывести полученную страницу на экран. Для этого окно Web-обозревателя содержит поле ввода интернет-адреса и область, в которую и выводится Web-страница. (Разумеется, оно также содержит заголовок, меню и панели инструментов, как и многие окна приложений Windows.)

Обычно после получения от сервера файла Web-страницы (и всех связанных с ней файлов, т. к. страница может состоять из множества файлов; подробнее об этом мы поговорим в последующих главах) Web-обозреватель сохраняет их на жестком диске клиентского компьютера в особой области, называемой *кэшем*. Этот кэш может иметь вид как обычной папки (кэш Microsoft Internet Explorer или Opera), так и большого файла (кэш Netscape Navigator или Mozilla).

Зачем это нужно? Да хотя бы затем, чтобы мы смогли впоследствии просмотреть данную страницу, не подключаясь к Интернету. Все современные Web-обозреватели поддерживают так называемый *автономный режим* (offline mode), когда они отображают только те страницы, что находятся в кэше. (Кстати — исключительно удобная вещь!) Если же мы попытаемся просмотреть страницу, которой нет в кэше, Web-обозреватель предложит нам подключиться к Интернету и загрузить ее.

Но даже если мы и подключены в данный момент к Интернету, Web-обозреватель все равно активно использует кэш. Перед тем как загрузить какую-либо Web-страницу, он проверяет, не изменилась ли она по сравнению с той, что находится сейчас в его кэше (если, конечно, она там уже есть). Если не

изменилась, он загружает нужную страницу прямо из кэша, что намного быстрее.

Теперь познакомимся с программами Web-обозревателей, имеющими в настоящее время наибольшую популярность. Все они, в общем, следуют одним и тем же стандартам и отличаются друг от друга только поддержкой деталей, не оговоренными в этих стандартах, и удобством для пользователей.

Настоящий король виртуальных просторов — это, конечно, Microsoft Internet Explorer. Он имеется на любом компьютере, работающем под управлением Windows (что, как говорят злые языки, и обусловило его популярность). Однако это очень мощная, быстрая, весьма нетребовательная к ресурсам и исключительно удобная программа. Автор данной книги для просмотра Web-страниц пользуется именно Internet Explorer. В настоящее время доступна версия 6.0 и, по слухам, разрабатывается 7.0, которая войдет в состав новой версии Windows — Windows Vista.

Второе место по популярности занимает самый "младший" Web-обозреватель по имени Firefox. Эта программа распространяется бесплатно, более того, ее исходные тексты открыты для изучения и модификации. Она весьма быстра и компактна, поддерживает все Web-стандарты, нетребовательна к системным ресурсам и имеет множество интересных и весьма полезных возможностей, которыми пока не может похвастаться ни один из ее конкурентов. Совсем недавно вышла версия 1.4, а на момент выхода книги из печати будет доступна 1.5.

Третье место занимает предшественник Firefox под названием Mozilla. Он также распространяется бесплатно, исходные тексты его открыты, а по возможностям он примерно аналогичен Firefox. Самая последняя версия этой программы носит 1.7, и выйдут ли более новые версии — неизвестно.

Некогда властелин WWW Netscape Navigator скромно устроился на четвертом месте. Да, последняя версия Navigator под номером 9.0 выглядит весьма неплохо, поддерживает все современные стандарты Интернета, корректно отображает большинство Web-страниц и не очень требовательна к системным ресурсам. Но все равно время его ушло.

Пятое место оккупировано амбициозной разработкой норвежских программистов Opera. Эта достаточно мощная и очень быстрая программа, поддерживающая все официальные Web-стандарты, тем не менее весьма требовательна к системным ресурсам и не всегда правильно отображает некоторые Web-страницы. Кроме того, эта программа платная, в бесплатной же версии показывает рекламу. Последняя вышедшая в свет версия носит номер 8.50 и, скорее всего, после выхода книги устареет, т. к. новые версии Opera появля-

На шестом месте отдыхает разработка фирмы Apple, производящей широко известные в узких кругах компьютеры Macintosh, — Safari. В настоящий момент имеет хождение версия Safari 2.0. Утверждается, что это самый быстрый в мире Web-обозреватель, быстрее даже Opera. Пока трудно сказать, что в действительности этот Safari собой представляет — у автора нет под рукой компьютера Apple Macintosh, чтобы попробовать Safari в действии.

В настоящее время просторы WWW "бороздят" практически только шесть перечисленных ранее программ. Существует, однако, еще несколько малоизвестных Web-обозревателей, а также довольно многочисленная когорта программ, построенных на основе программного ядра Internet Explorer и расширяющих его возможности. Мы не будем их рассматривать.

Осталось только сказать, что выбор Web-обозревателя — это личное дело каждого. Все они поддерживают одни и те же стандарты (правда, зачастую по-своему) и предоставляют пользователю примерно одинаковый набор возможностей (хотя, не все найдут его удобным). Так что, как в песне поется, "думайте сами, решайте сами"...

Web-серверы

Поскольку мы не только пользователи, но и уже наполовину разработчики, нас будут интересовать не только Web-обозреватели, но и Web-серверы. Давайте поговорим и о них.

"Зоопарк" Web-серверов ничуть не меньше "зоопарка" (или, если учесть, что Web-обозреватели жестоко конкурируют друг с другом, "серпентария") Web-обозревателей, так что мы можем подобрать себе программу "по вкусу". И, в отличие от Web-обозревателей, среди Web-серверов нет безоговорочного лидера — даже самые распространенные из них не занимают больше половины рынка.

Начнем наш краткий обзор с двух пакетов фирмы Microsoft: Personal Web Server и Internet Information Server. Оба этих пакета поставляются в составе Microsoft Windows, первая программа — в составе Windows 98 и ME, а вторая — в составе Windows NT, 2000, XP и 2003. Со своими обязанностями они справляются очень хорошо, не транжирят системные ресурсы, легко настраиваются, поддерживают множество передовых интернет-технологий и при надлежащей настройке легко затыкают за пояс конкурентов. Кроме собственно Web-сервера, они содержат также серверы FTP и почты, а также некоторое количество дополнительных программ.

Web-сервер Apache — пожалуй, самый распространенный. Среди его достоинств: полная бесплатность (более того — его исходные тексты открыты),

держка. По крайней мере, для Web-сайтов с небольшой загрузкой — это идеальный выбор.

В свое время фирма Netscape — разработчик известного Web-обозревателя Navigator — создала и Web-сервер, который так и называется — Netscape Web Server. В смысле производительности и поддержки передовых интернет-технологий он не уступает своему конкуренту от Microsoft, но — увы! — не завоевал его популярности.

Есть еще один весьма примечательный Web-сервер — Sambar. Он поддерживает такое количество интернет-технологий (многие из них — эксклюзивные, не доступные больше ни в одной программе), что просто оторопь берет — как же всем этим богатством воспользоваться и куда его применить? Недостатка у Sambar всего два: малая известность и не очень удобная настройка. (Кстати, автор этой книги пользуется именно этим сервером.)

Менее известные и специализированные серверы мы рассматривать не будем, так как их довольно много. Поговорим лучше о том, как разместить созданный нами сайт в Интернете.

Публикация Web-сайта в Интернете. Хостинг-провайдеры

Итак, предположим, что мы создали свой сайт (а мы его и создадим, пока будем изучать интернет-технологии по этой книге). Теперь нам нужно сделать так, чтобы все желающие увидеть его собственными глазами, а именно — разместить, или, как говорят опытные интернетчики, *опубликовать* его в Интернете. А значит, нам нужно подключение к Интернету и Web-сервер.

Если наш компьютер подключен к Интернету по скоростному каналу или находится в локальной сети, работающей по тем же стандартам, что и Интернет (так называемый *интранет*), то мы можем просто установить на него Web-сервер и сюда же поместить наш Web-сайт. Это самый простой способ, хотя, конечно, нам придется попутно освоить профессию администратора.

Для тех "счастливчиков", что выходят в Интернет по телефонным каналам (как автор этой книги), существуют три способа донести свое Web-творение до страждущих масс. Давайте перечислим их в порядке от простых к более хлопотным.

Большинство солидных интернет-провайдеров, кроме собственно доступа в Интернет, предоставляют своим клиентам и другие услуги: электронную почту, доступ на свой сайт с новостями, документацией и файловым архивом и пр. Так вот, среди этих "пр." есть и такая услуга, как предоставление на жестких дисках серверного компьютера места для размещения Web-сайтов клиентов. В этом и заключается первый способ: выдаете место на диске

сайта на сервере интернет-провайдера и, следуя этим условиям, опубликовать его.

Если же интернет-провайдер почему-то жадничает, можно прибегнуть ко второму способу. В Интернете существует довольно много серверов, предоставляющих место для сайтов бесплатно. Процесс и в этом случае очень прост: заходим на сайт такого сервера, регистрируемся, выясняем условия публикации сайта и публикуем.

Всем хороши бесплатные серверы: и денег не берут, и позволяют публиковать сайты. Но бесплатного сыра много не бывает... Как правило, объем предоставляемого под сайт дискового пространства сильно ограничен, значит, большой сайт таким образом не опубликуешь. Еще администратор может ограничить количество пользователей, которые могут одновременно зайти на наш сайт. Да и с поддержкой некоторых технологий, используемых для создания Web-сайтов, дело может обстоять не очень хорошо.

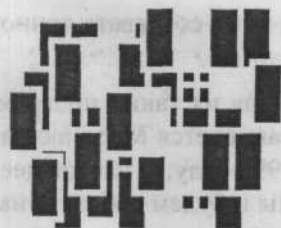
Если же нам нужно нечто большее, чем предложения бесплатных серверов, то придется достать свой кошелек и пойти третьим путем — опубликовать сайт на платном сервере. Благо сейчас их довольно много, и услуги их довольно дешевы.

Кстати, организации, предоставляющие место на своих серверах для публикации Web-сайтов, называются *хостинг-провайдерами*.

На следующем занятии...

Вот и закончился наш краткий курс интернет-технологий. Конечно, многое здесь не описано — все, что вам понадобится для создания и публикации Web-страниц, будет представлено в дальнейшем. А сейчас давайте, не углубляясь в дебри Интернета, перейдем к нашей главной цели — познакомимся с замечательным пакетом Web-редактора Macromedia Dreamweaver 8.

ЗАНЯТИЕ 2



Знакомство с Macromedia Dreamweaver 8

На *занятии 1* мы кое-что узнали о современных интернет-технологиях. Мы выяснили, что такое Интернет и какие сервисы он предоставляет. Мы познакомились с клиентами и серверами Интернета, с его протоколами и интернет-адресами. И, приближаясь к теме предмета, мы выяснили, как работает популярнейший сервис Интернета — WWW, какие клиентские и серверные программы он использует и что собой представляют Web-страницы и Web-сайты.

Следующий шаг — выяснить, с помощью каких программ на языке HTML пишутся эти самые Web-страницы. Да, поскольку они представляют собой обычные текстовые файлы, содержащие код HTML, их можно писать в обычном Блокноте или аналогичной программе. Но на *занятии 1* мы еще упомянули о программах Web-редакторов. Естественно, возникает вполне резонный вопрос...

Зачем нужны Web-редакторы?

Язык HTML, несмотря на столь мудреное название, весьма прост. И написать с его помощью простенькую Web-страничку с парой абзацев текста можно буквально за пять минут, причем большая часть этого времени будет потрачена на набор самого текста этой страницы, а не тегов HTML. Так в чем же проблема?

Проблема в том, что мы пока еще не знаем HTML. И многие его не знают, более того — не хотят его изучать или не имеют на это времени. Но создавать Web-страницы хотят. Поэтому специально для них программисты написали множество программ, предназначенных для создания Web-страниц, — так называемых *Web-редакторов*. С помощью таких программ Web-страницы

можно создавать точно так же, как документы в Microsoft Word, — просто и удобно.

Одна из таких программ написана разработчиками из фирмы Macromedia и называется Macromedia Dreamweaver. Первая ее версия вышла еще в далеком 1998 году; в настоящее же время доступна версия 8. Именно с Dreamweaver мы и будем работать на протяжении всей книги.

Dreamweaver — типичнейший представитель *визуальных Web-редакторов*, работающих по принципу WYSIWYG (What You See Is What You Get, "что ты видишь, то ты и получишь"). При этом пользователь форматирует текст и в окне редактора сразу же видит результаты своих трудов.

Существуют также и *невизуальные Web-редакторы* (они же — *HTML-редакторы*), основанные на другом принципе. Они работают непосредственно с самим HTML-кодом, предоставляя при этом пользователю различные дополнительные возможности: быстрая вставка тегов, удобное задание их параметров, набор предопределенных шаблонов для создания стандартных элементов Web-страницы и пр. В этом смысле они похожи на Блокнот, но значительно расширенный.

Здесь нужно сказать, что практически все серьезные Web-редакторы имеют режим правки непосредственно самого кода HTML (т. е. фактически являются *гибридными Web-редакторами*). Поэтому сейчас практически всегда, когда говорят "визуальный Web-редактор", подразумевают как раз гибридные программы. Разумеется, к их числу относится и Dreamweaver, с которым нам пора познакомиться поближе.

Среда Dreamweaver 8

Итак, начнем рассматривать *рабочую среду* Dreamweaver 8, т. е. наборы окон и различных инструментов, предлагаемых им Web-дизайнеру. Но для начала запустим Dreamweaver.

Запустить программу в Windows проще простого! Нажмем хорошо знакомую нам кнопку **Start** (Пуск), выберем в меню пункт **Programs** (Программы), далее — пункт **Macromedia** и в появившемся подменю — пункт **Macromedia Dreamweaver 8**.

Выбор рабочей среды

Через некоторое время после запуска программы на экране появится небольшое диалоговое окно **Workspace Setup**, предлагающее нам выбрать набор изначально открытых окон и их расположение — собственно рабочую среду

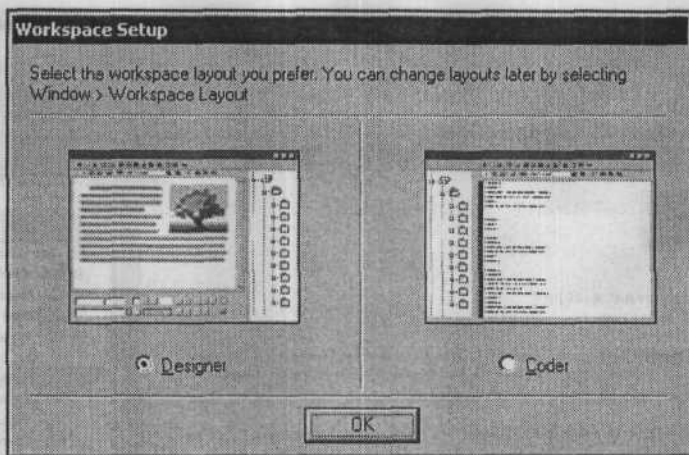


Рис. 2.1. Диалоговое окно Workspace Setup

В этом окне мы видим два небольших изображения, показывающие, какие виды рабочей среды мы можем выбрать с помощью расположенных под ними переключателей. Давайте их рассмотрим.

Переключатель **Designer** выбирает "дизайнерский" вид среды Dreamweaver 8. В этом случае мы сможем работать с Web-страницами в режиме WYSIWYG. Поскольку мы пока не знаем HTML, давайте включим именно его (впрочем, он включен по умолчанию).

Переключатель **Coder** позволяет выбрать "программистский" вид среды Dreamweaver. При этом программа сразу же после открытия очередной Web-страницы переключится в режим редактирования HTML-кода (т. е. будет вести себя как HTML-редактор). Это может пригодиться только тем, кто привык набирать код HTML вручную, — в общем, не наш случай.

Задав нужный вид рабочей среды Dreamweaver, нажмем кнопку **OK**. Через некоторое время мы увидим главное окно. Теперь можно начинать знакомство с этой замечательной программой.

Внимание!

Если вы ошиблись и включили не тот переключатель, можете сменить рабочую среду в самой программе. Для этого нужно воспользоваться подменю **Workplace Layout** меню **Window**. Пункт **Designer** этого подменю включает "дизайнерский" вид среды Dreamweaver 8, а пункт **Coder** — "программистский".

Главное окно программы

Главное окно программы Macromedia Dreamweaver 8 показано на рис. 2.2.

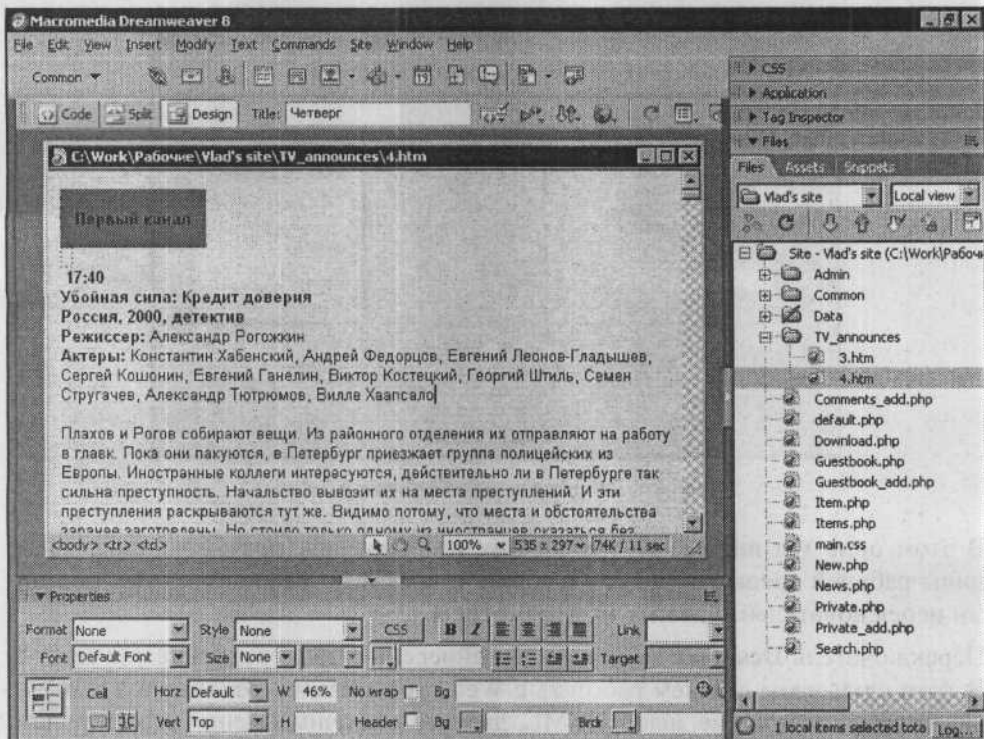


Рис. 2.2. Главное окно Dreamweaver

Главное окно служит "вместилищем" для превеликого множества других окон, содержащих как открытые Web-страницы, так и различные инструменты, предназначенные для работы с ними. Также в главном окне находится строка *главного меню*, с помощью которого мы сможем получить доступ ко всем возможностям Dreamweaver.

Кроме того, если в Dreamweaver еще не открыта ни одна Web-страница, становится видима так называемая *стартовая страница*, отображаемая на фоне главного окна. С помощью стартовой страницы мы можем быстро, не залезая в меню, открыть нужную страницу, создать новую либо перейти на сайт фирмы Macromedia (<http://www.macromedia.com>). После открытия хотя бы одной Web-страницы стартовая страница пропадет.

Внимание!

Отображение стартовой страницы отнимает довольно много системных ресурсов, поэтому, если ваш компьютер недостаточно быстр, лучше отключите ее. Для этого найдите в ее левом нижнем углу флажок **Don't show again** и отметьте его. При следующих запусках Dreamweaver стартовой страницы уже не

Окно документа Dreamweaver служит для отображения открытой Web-страницы (на рис. 2.2 видно, что в Dreamweaver открыта небольшая Web-страничка, чье содержимое автор книги позаимствовал с сайта <http://www.vsetv.com>). В главном окне Dreamweaver может находиться сколько угодно окон документов. Мы можем перемещать, свертывать и разворачивать эти окна, а также изменять их размеры, в общем, проделывать с ними те же манипуляции, что и с любым другим окном Windows. Единственное исключение: мы не можем "вытащить" ни одно из этих окон за пределы главного окна программы.

Выше, ниже и правее окна документа находятся *группы панелей* — небольшие окна, которые могут быть либо "приклеены" к одному краю главного окна, либо свободно "плавать" рядом с ним. В верхней части каждой группы панелей имеется ее *заголовок* — "выпуклая" синяя полоса, на которой написано название панели.

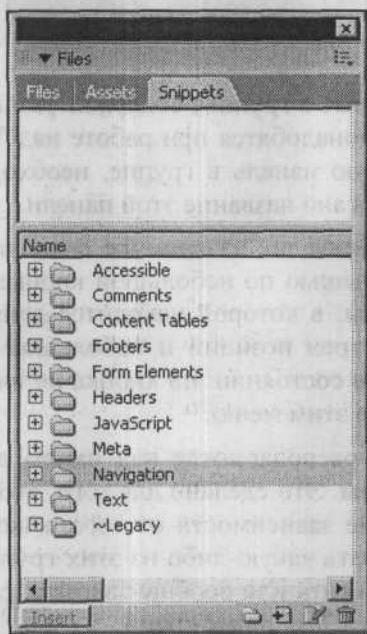


Рис. 2.3. Группа панелей, отделенная от края родительского окна

Изначально все группы панелей, имеющиеся на экране, "приклеены" к какому-либо краю главного окна. Однако мы можем отделить какую-либо группу панелей от края главного окна и превратить ее в отдельное окно (рис. 2.3) (так называемая *плавающая* группа панелей). Для этого каждая из них имеет "ручку" для ее "переноски", находящуюся в правой части заголовка и выходящую за его пределы.

дядущую как набор мелких темных точек. Просто перетащим за эту "ручку" нужную группу подальше от края главного окна и оставим ее там. Разумеется, мы всегда можем присоединить панель обратно к краю главного окна, чтобы она не загромождала документ.

Мы также можем изменять размеры плавающих групп панелей (многих, но не всех). Для этого нужно просто перетащить мышью нижний край окна, в котором находится эта группа.

Чтобы временно уменьшить площадь, занимаемую плавающей группой панелей, мы можем ее сжать, чтобы на экране остался только ее заголовок (рис. 2.4). Для этого достаточно щелкнуть по названию группы панелей, находящемуся в ее заголовке. Чтобы развернуть группу до обычного состояния, нужно снова щелкнуть по ее заголовку.

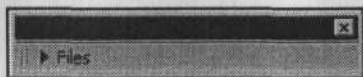


Рис. 2.4. Сжатая группа панелей

Сами *панели*, объединенные в группы, содержат различные инструменты, которые нам обязательно понадобятся при работе над Web-страницами. Чтобы переключиться на нужную панель в группе, необходимо щелкнуть мышью вкладку, на которой написано название этой панели.

Большинство панелей имеют так называемое *дополнительное меню*. Оно открывается при щелчке мышью по небольшой кнопке, расположенной в правом верхнем углу группы, в которой находится данная панель, и имеющей изображение списка из трех позиций и небольшой стрелки, направленной вниз (рис. 2.5). (В сжатом состоянии эта кнопка не видна.) В дальнейшем мы часто будем пользоваться этим меню.

Группы панелей всегда располагаются над окном документа, даже если в данный момент неактивны. Это сделано для того, чтобы мы могли всегда получить к ним доступ, вне зависимости от того, какое окно сейчас активно. Если же мы захотим убрать какую-либо из этих групп, мы можем "вынести" ее за пределы окна документа или вообще закрыть, раскрыв ее дополнительное меню и выбрав пункт **Close panel group**.

Теперь обратим внимание на правый край главного окна, где находится множество всяческих групп панелей. Это так называемый *док* — область, специально для них предназначенная. Док отделен от остального пространства главного окна толстой серой полосой, которую мы можем перетаскивать мышью, изменяя размеры дока. Мы также можем щелкнуть мышью довольно приметную кнопку на доке (рис. 2.6), чтобы быстро скрыть его со всеми

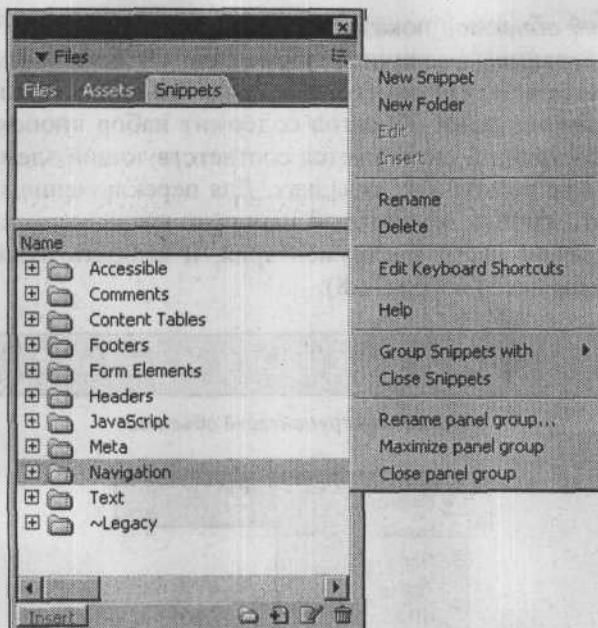


Рис. 2.5. Дополнительное меню группы панелей (открыто)



Рис. 2.6. Кнопка скрытия-раскрытия дока

Еще один док располагается в нижней части главного окна (см. рис. 2.2). В нем находится группа с единственной панелью **Properties** (Свойства), о которой мы еще поговорим. Третий док, в данный момент скрытый, находится в левой части окна; попробуйте переместить туда любую плавающую группу панелей — и вы его увидите.

Еще три панели Dreamweaver достойны отдельного упоминания. Они так сильно отличаются от остальных, что даже носят особое название — *инструментарии*. От обычных панелей они отличаются тем, что имеют постоянные размеры и образуют каждая свою особую группу. Таких инструментариев в Dreamweaver три:

- ◆ инструментарий объектов;
- ◆ инструментарий документа;
- ▲ инструментарий...

Инструментарий объектов показан на рис. 2.7 и служит для быстрого помещения на Web-страницы различных элементов. От остальных инструментариев он отличается тем, что жестко зафиксирован вдоль верхнего края главного окна. Инструментарий объектов содержит набор кнопок, при нажатии которых на Web-страницу помещается соответствующий элемент. Эти кнопки расположены на нескольких вкладках. Для переключения между вкладками нужно нажать кнопку, на которой написано название вкладки и которая расположена в левой части инструментария, и выбрать нужную вкладку в появившемся *меню вкладок* (рис. 2.8).



Рис. 2.7. Инструментарий объектов

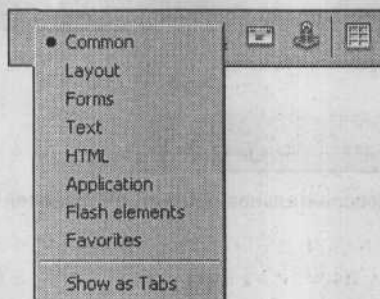


Рис. 2.8. Меню вкладок инструментария объектов

Инструментарий документа позволяет нам выполнять некоторые манипуляции с открытой Web-страницей и самой программой; он показан на рис. 2.9. Мы рассмотрим его позже, когда начнем работу над нашими первыми Web-страницами.

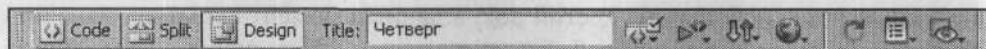


Рис. 2.9. Инструментарий документа

Стандартный инструментарий, предоставляющий доступ к операциям с файлами (создание, открытие и сохранение Web-страницы), буфером обмена и пр., изначально скрыт. Как вывести его на экран, мы рассмотрим чуть позже.

Управление окнами и панелями Dreamweaver

Ясно, что Dreamweaver в состоянии вывести на экран множество разнообраз-

с помощью соответствующих пунктов подменю **Window**. Давайте о них поговорим.

Если мы откроем несколько Web-страниц, разобраться в них будет очень трудно — окна документов перекрывают друг друга, и добраться до нужного удастся далеко не сразу. В этом случае откроем подменю **Window** и посмотрим в самый его низ — там будут находиться пункты, имеющие имена, схожие с именами файлов открытых страниц. Для того чтобы переключиться в окно, где открыт нужный файл, достаточно выбрать соответствующий пункт.

Если же мы раскроем одно из окон документов на весь экран (точнее, на все главное окно), то переключаться между окнами станет еще проще. В этом случае в инструментарии документа появятся вкладки, соответствующие открытым окнам документа (рис. 2.10). Нам останется только щелкнуть нужную вкладку. Кстати, мы можем перетаскивать эти вкладки, меняя их порядок.

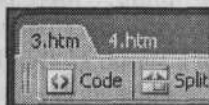


Рис. 2.10. Вкладки, перечисляющие все открытые окна документа

Если нам необходимо держать на виду сразу два окна или больше, стоит воспользоваться пунктами **Cascade**, **Tile Horizontally** или **Tile Vertically** меню **Window**. Первый из них "выкладывает" все открытые окна документов в виде "стопки" в главном окне так, что мы сможем видеть их заголовки и часть содержимого. Второй и третий пункты "выкладывают" в главном окне "мозаику" из окон документов так, чтобы они не перекрывались. Причем второй пункт выкладывает "мозаику" по горизонтали, а третий — по вертикали.

Теперь давайте откроем подменю **Toolbars** меню **View**. В нем находятся три пункта:

- ◆ пункт **Insert** выводит инструментарий объектов;
- ◆ пункт **Document** выводит инструментарий документа;
- ◆ пункт **Standard** выводит стандартный инструментарий.

Инструментарий документа также можно вывести на экран, выбрав пункт **Insert** меню **Window** или нажав комбинацию клавиш <Ctrl>+<F2>.

Если слева от имени одного из этих пунктов стоит галочка, это значит, что соответствующий инструментарий выведен на экран (или, как еще говорят, соответствующий пункт меню "включен"). Чтобы убрать инструментарий, нужно выбрать соответствующий пункт еще раз — и инструментарий исчезнет вместе с галочкой. Такие пункты меню, меняющие свое состояние на противоположное при выборе, называются переключателями.

Если нам понадобится скрыть на время все панели, чтобы без помех посмотреть открытую Web-страницу, нам нужно будет выбрать пункт **Hide Panels** в меню **Window** или одноименный пункт в меню **View**. Вместо выбора этого пункта можно просто нажать клавишу <F4> — так быстрее. Чтобы снова вывести скрытые панели на экран, достаточно в меню **Windows** или **View** выбрать пункт **Show Panels** или, опять же, нажать клавишу <F4>.

На заметку

Вообще, большую часть меню **Window** занимает набор пунктов-выключателей, служащих для вывода на экран или скрытия различных панелей. В дальнейшем они нам очень пригодятся.

Вызов справки

Слышен вопрос озабоченного читателя: а что делать, если я что-то забуду? Как и все серьезные Windows-приложения, Dreamweaver снабжен мощной *справкой*. Для ее вызова достаточно просто нажать клавишу <F1> или выбрать пункт **Dreamweaver Help** в меню **Help**. После этого на экране появится окно справочной системы, показанное на рис. 2.11.

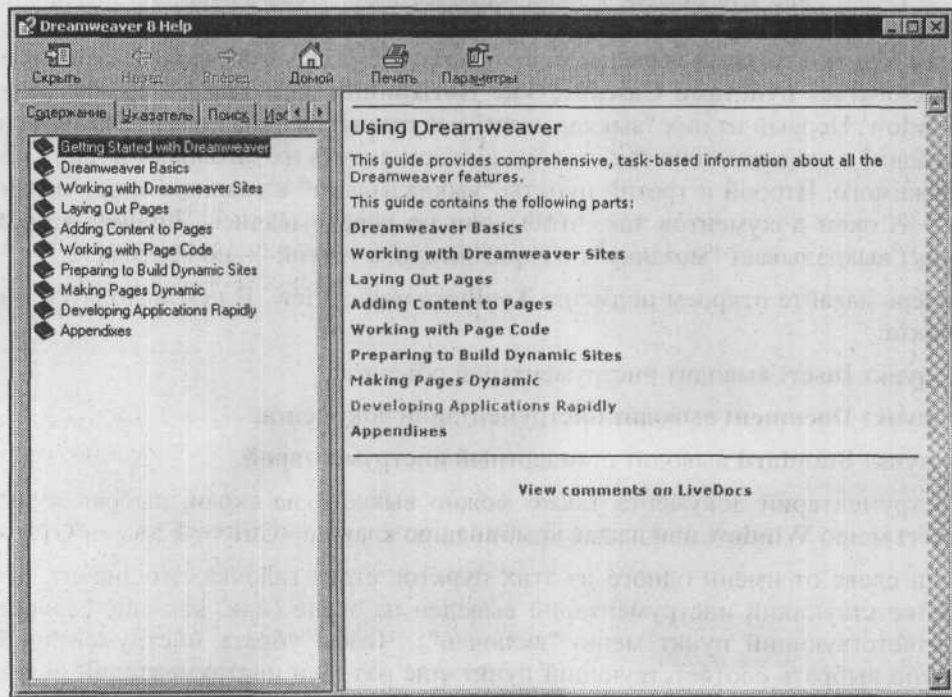


Рис. 2.11. Окно справочной системы Dreamweaver

В левой части окна справки расположен древовидный список тем. Сами же статьи отображаются справа. Мы можем щелкнуть мышью по названию темы, чтобы вывести нужную статью, тем самым развернув или свернув ветвь дерева. Если списка тем почему-то нет, нужно переключиться на вкладку **Содержание**.

Также предусмотрен поиск статей по ключевым словам. Для этого выбираем вкладку **Поиск**, вводим в поле ввода нужное ключевое слово и нажимаем кнопку **Разделы**. После этого в списке, занимающем нижнюю половину вкладки, появятся заголовки найденных статей. Нам остается только выбрать нужную и нажать кнопку **Показать**. Выбранная статья отобразится справа.

Настройка Dreamweaver

А сейчас самое время настроить наш Dreamweaver так, чтобы он нормально понимал русский язык и создавал именно такие Web-страницы, какие нам нужны.

Вся работа будет происходить в диалоговом окне **Preferences**, состоящем из множества вкладок с разными элементами управления. Чтобы вызвать его, выберем в меню **Edit** пункт **Preferences** или нажмем комбинацию клавиш <Ctrl>+<U>. В левой части окна настройки отображен список категорий **Category**, а в правой — настройки, относящиеся к выбранной в нем категории.

Выберем в списке категорий пункт **New Document**. Окно настройки примет вид, представленный на рис. 2.12.

Сначала нужно выбрать в раскрывающемся списке **Default Document Type (DTD)** пункт **HTML 4.01 Transitional**. Этот пункт предписывает Dreamweaver создавать Web-страницы, используя разновидность языка HTML, "понимаемую" всеми Web-обозревателями. Другие пункты этого списка включают использование других разновидностей языка HTML, менее распространенных.

А теперь, прежде чем начать разговор о *русификации* Dreamweaver, немного поговорим об особенностях национального Web-творчества. А именно, о кодировках русского языка и борьбе с ними.

Вероятно, все знают, что каждый символ, который вводится с клавиатуры и отображается на экране, имеет уникальный номер, называемый *кодом символа*. Совокупность таких кодов вместе с описанием, какой код какому символу соответствует, образует *кодировку*, или *кодovou таблицу*. Каждая кодировка имеет свое наименование, например 1251 или КОИ-8.

Поскольку любой язык использует свой набор символов, для каждого языка кодировка, как правило, различна. (Максимальное количество символов в

пейские языки, использующие одну кодировку для всех.) Но на этом путаница с кодировками не кончается. Дело в том, что разные операционные системы используют разные кодировки. Например, западноевропейская версия Windows использует кодировку 1250, русская — 1251, американская версия MS-DOS — 437, а русская — 866. Как видите, русских кодировок уже две. А если добавить сюда еще кодировку, используемую русской версией операционной системы UNIX, — КОИ-8 и русской версией компьютеров Macintosh — MacCyrillic, кодировок станет уже четыре. И это только главные — на памяти автора существовало еще несколько менее распространенных кириллических кодировок ("основная" кодировка ГОСТ, "болгарская", "американская", "югославская" и т. п.). Кроме того, в последнее время появилась кодировка Unicode, поддерживающая ВСЕ имеющиеся на Земле языки. Настоящая тирания кодировок!..

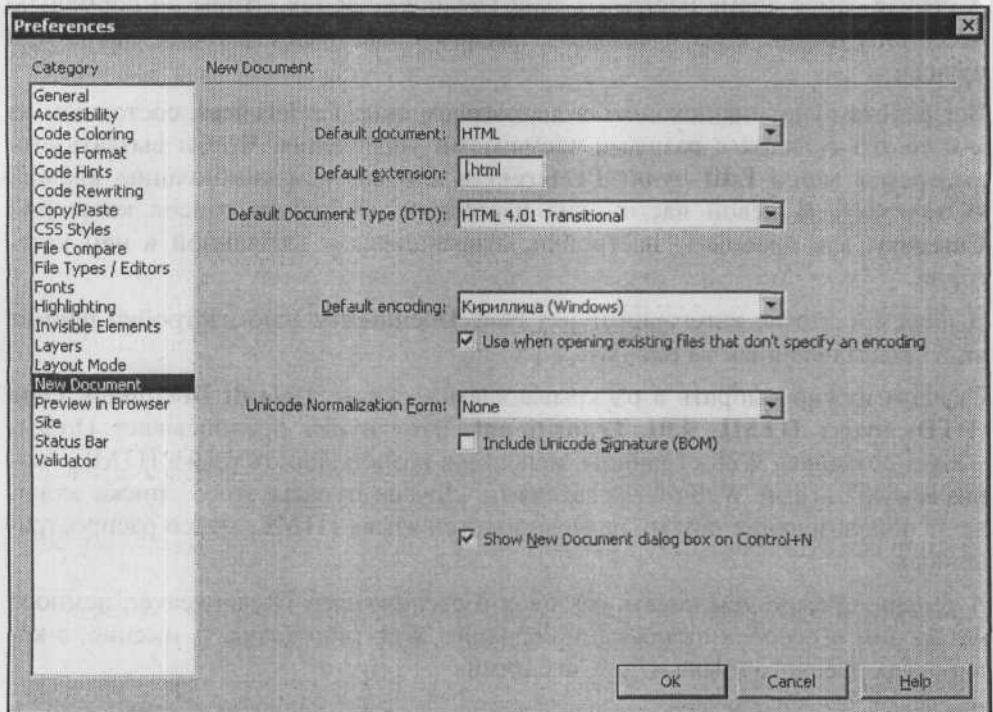


Рис. 2.12. Категория New Document диалогового окна Preferences

Чем все это грозит? А вот чем. Вы, наверно, пытались открыть текстовый документ, созданный в Блокноте, в Norton Commander. Видели, что при этом получается — текст абсолютно нечитаем. А все потому, что русские кодировки 866 (MS-DOS), используемая Norton Commander, и 1251 (Windows),

используемая Блокнотом, не совпадают! В них один и тот же код соответствует разным символам.

Каков же выход?

Выхода нет. Можно надеяться только на то, что какая-то из кодировок станет стандартом и постепенно вытеснит конкурентов. Сейчас на роль такого (негласного) стандарта претендует 1251, хотя интернетчики старого поколения, пользующиеся UNIX-совместимыми системами, все еще часто выбирают КОИ-8. Во всяком случае, сейчас большинство Web-страниц, имеющих в русском сегменте Сети, написано в кодировке 1251.

Здесь стоит упомянуть еще два момента. Во-первых, современные программы Web-обозревателей поддерживают все доступные сейчас кодировки и корректно их распознают. Во-вторых, какой-либо Web-сервер (точнее, его администратор) может потребовать, чтобы публикуемые нами странички были представлены в какой-либо конкретной кодировке, например в КОИ-8. Это стоит иметь в виду, когда мы будем выбирать кодировку для своего Web-творения.

Итак, какие же кодировки поддерживает Dreamweaver? (Имеются в виду, конечно же, русские кодировки.) Все они перечислены в табл. 2.1 и задаются с помощью раскрывающегося списка **Default encoding** категории **New Document** диалогового окна **Preferences**.

Таблица 2.1. Кодировки текста, поддерживаемые Dreamweaver

Имя пункта	Обозначение кодировки	Платформа, которая ее использует
Кириллица (DOS)	866	Русская версия MS-DOS
Кириллица (ISO)	ISO-8859-5	Редко используемая кодировка; автору даже не встречалось ни одной страницы, написанной на ней
Кириллица (КОИ8-R)	КОИ-8	Русские версии UNIX-совместимых систем
Кириллица (Windows)	1251	Русские версии Windows, начиная от версии 3.0
Кириллица (КОИ8-U)	КОИ-8	Украинская версия кодировки КОИ-8
Западноевропейская	Western (Latin1)	Это не русская кодировка, она включена в этот список только для справки. Используется большинством западноевропейских языков

Но какую кодировку выбрать? Ответ прост. Если мы не связаны какими-либо специфическими требованиями администратора Web-сервера на котором

будет опубликован наш сайт, смело выбираем пункт **Кириллица (Windows)**. В противном случае выберем ту кодировку, которую требует сервер. Если мы создаем странички на английском языке, наш выбор — **Западноевропейская**.

Теперь выберем в списке **Category** диалогового окна **Preferences** пункт **Fonts** (рис. 2.13). Теперь мы сможем настроить шрифты, которыми будет отображаться текст наших Web-страниц. В списке **Font settings** выберем шрифтовой набор, который будет использован для отображения наших Web-страниц. Здесь альтернатива еще проще: если текст русский — выбираем пункт **Кириллица**, если английский — **Западноевропейская**.

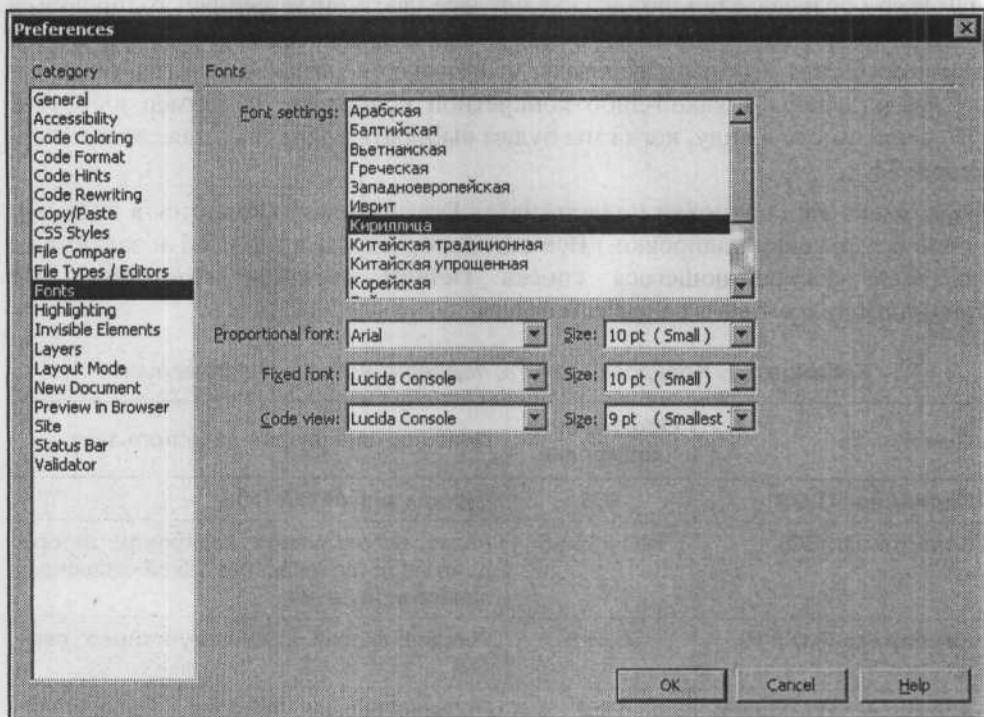


Рис. 2.13. Категория **Fonts** диалогового окна **Preferences**

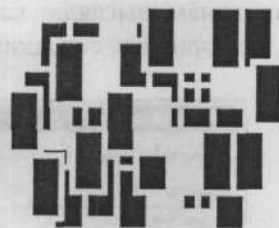
Что касается начертаний и размеров шрифтов, используемых для отображения текста, можно только посоветовать, но никак не порекомендовать. Автор предпочитает в качестве пропорционального шрифта (раскрывающийся список **Proportional font**) Arial, в качестве моноширинного (**Fixed font**) — Lucida Console, а для отображения HTML-кода в режиме HTML-редактора (**Code view**) — тоже Lucida Console. Размеры шрифтов (раскрывающиеся списки **Size**, расположенные правее перечисленных выше списков) автор обычно

ставит равным 10 пунктам (малый размер, **Small**). Но нужно еще раз повторить, что это дело вкуса.

На следующем занятии...

Вот мы и познакомились с Macromedia Dreamweaver 8! Теперь, когда мы, можно сказать, официально представлены друг другу, самое время испытать его в действии.

На следующем занятии мы займемся созданием первой Web-страницы. Самой-самой простой, просто чтобы научиться работать в Dreamweaver. И заодно начнем изучение языка HTML — все-таки без его знания в Web-дизайне никуда. За дело!



ЗАНЯТИЕ 3

Работа с текстом

На этом занятии мы, собственно, и начнем работу с Dreamweaver. На первый раз не будем усложнять себе жизнь и ограничимся простейшей Web-страницей. Пусть это будет небольшая страница гипотетического студента Василия Васильевича Васильева, содержащая... ну что может заинтересовать студента?..

Первая наша Web-страница пока что будет содержать один лишь текст. Мы начнем изучение Dreamweaver с набора и форматирования текста, а потом, на следующих занятиях, начнем работать с графикой и таблицами. От простого — к сложному!

Создание новой Web-страницы

Первое, что мы должны сейчас сделать, — создать новую Web-страницу, которую потом будем постепенно наполнять содержимым.

Для создания новой страницы в Dreamweaver достаточно выбрать в меню **File** пункт **New** или нажать комбинацию клавиш **<Ctrl>+<N>**. На экране появится диалоговое окно **New Document** (рис. 3.1).

Здесь нужно сказать, что Dreamweaver 8 поддерживает создание Web-страниц на основе шаблонов. *Шаблон* — это особым образом подготовленная и сохраненная Web-страница, где уже имеются некоторые элементы, которые могут нам понадобиться (заголовок, сведения об авторских правах, элементы оформления и т. п.). Нам останется только немного их исправить и добавить основное содержимое, ради которого и создается страница.

С одной стороны, шаблоны — это хорошо, они позволяют быстро создать хорошо оформленную страницу. С другой стороны, страницы, созданные по

шаблонам, выглядят как близнецы. Поэтому давайте начнем с чистого листа и, не торопясь, создадим свой неповторимый Web-шедевр.

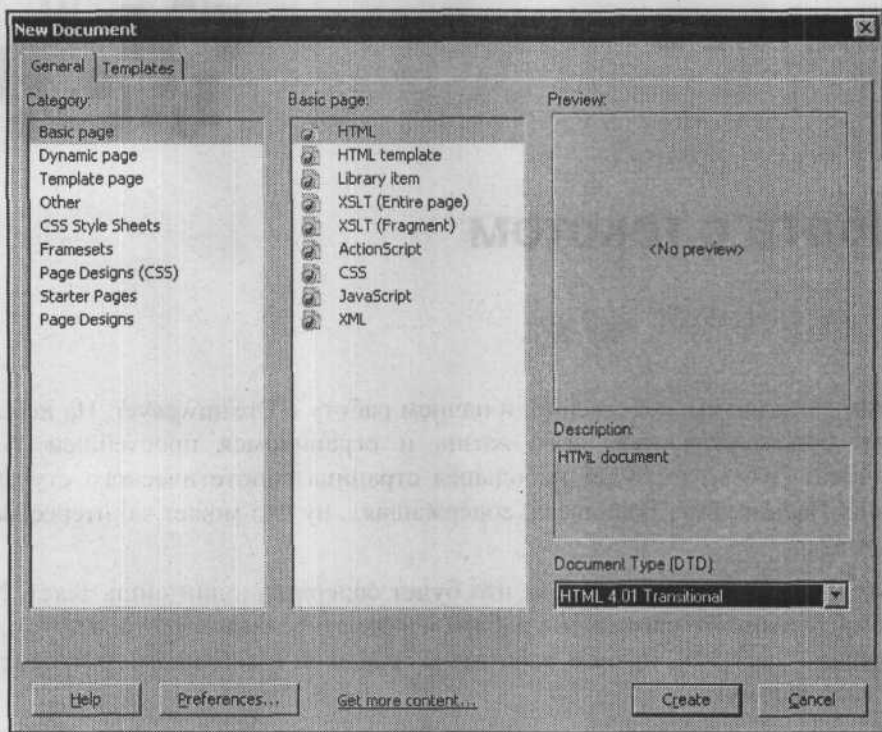


Рис. 3.1. Диалоговое окно New Document

Посмотрим на рис. 3.1. В левой части диалогового окна **New Document** расположен список **Category**, где выбирается категория шаблонов. Сами шаблоны, относящиеся к выбранной категории, перечислены в списке справа. А внешний вид выбранного шаблона во многих случаях можно просмотреть в расположенной еще правее панели предварительного просмотра.

Итак, нам нужен "пустой" шаблон. Выберем в списке **Category** пункт **Basic page**, а в правом списке — пункт **HTML**, после чего нажмем кнопку **Create**. На экране появится пустое окно документа. Новая, пока еще ничего не содержащая Web-страница создана.

Ввод и форматирование текста

Начнем мы с того, что введем в нашу только что созданную Web-страницу текст и отформатируем его — расставим заголовки, выделим части текста

полужирным шрифтом и пр. В этом смысле Dreamweaver мало чем отличается от хорошо знакомых нам текстовых редакторов.

Набор текста Web-страницы

Вот и наступила торжественная минута! Сейчас мы наберем наш первый текст в окне документа Dreamweaver (рис. 3.2) и создадим таким образом нашу первую Web-страницу.

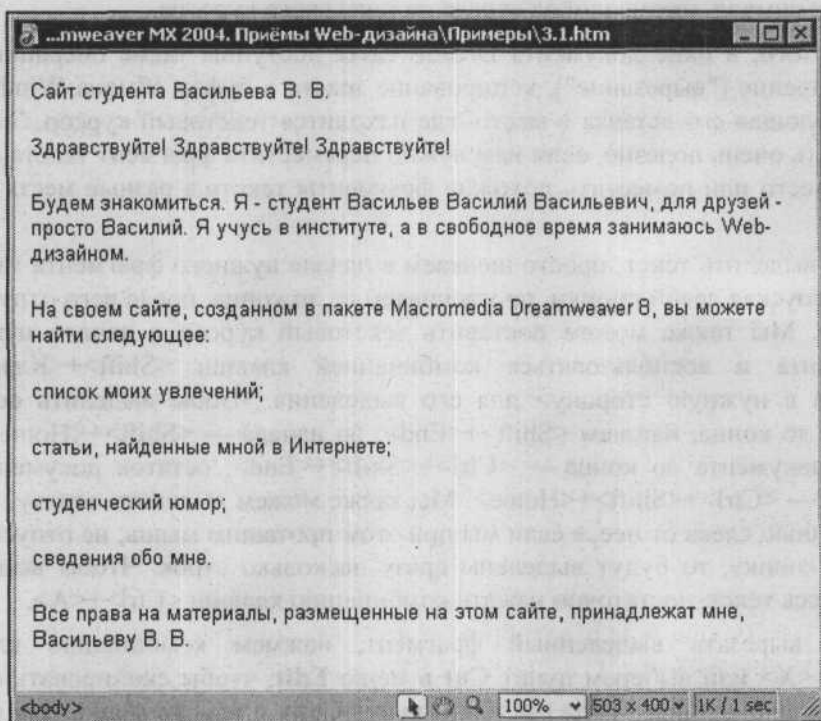


Рис. 3.2. Наш первый текст

Текст набирается с помощью клавиатуры (а вы как думали?), при этом Dreamweaver самостоятельно разобьет текст на строки. Чтобы создать новый абзац, нужно нажать клавишу <Enter>. Если же надо просто перенести текст на другую строку (вставить так называемый *разрыв строк*), то достаточно нажать комбинацию клавиш <Shift>+<Enter>.

Текстовый курсор, т. е. мигающая вертикальная черточка, показывающая место, где будет появляться набираемый нами текст, может перемещаться во всех направлениях с помощью клавиш-стрелок. Также мы можем "листать" текст, нажимая клавиши <PgUp> и <PgDown>, мгновенно перемещаясь к

началу и концу строки клавишами <Home> и <End>. Чтобы быстро переместиться в начало или конец документа, нужно нажать, соответственно, комбинацию клавиш <Ctrl>+<Home> или <Ctrl>+<End>. Мы также можем устанавливать текстовый курсор в произвольное место, просто щелкнув там мышью.

В случае ошибки мы всегда можем удалить неправильно введенный текст, воспользовавшись клавишами <Backspace> и . Первая при нажатии удаляет символ, находящийся слева от текстового курсора, и идеально подходит для удаления только что введенного неправильного символа. Вторая удаляет символ, находящийся справа от текстового курсора.

Кроме того, в окне документа Dreamweaver доступны такие операции, как перемещение ("вырезание"), копирование текста в буфер обмена Windows и последующая его вставка в место, где находится текстовый курсор. Это может быть очень полезно, если нам нужно переместить фрагмент текста с места на место или поместить похожие фрагменты текста в разные места документа.

Чтобы выделить текст, просто щелкаем в начале нужного фрагмента мышью и, не отпуская левой кнопки, протаскиваем ее до конца, после чего отпускаем кнопку. Мы также можем поставить текстовый курсор в начало нужного фрагмента и воспользоваться комбинацией клавиш <Shift>+<Клавиша-стрелка в нужную сторону> для его выделения. Чтобы выделить остаток строки до конца, нажмем <Shift>+<End>, до начала — <Shift>+<Home>, остаток документа до конца — <Ctrl>+<Shift>+<End>, остаток документа до начала — <Ctrl>+<Shift>+<Home>. Мы также можем выделить строку, щелкнув мышью слева от нее, а если мы при этом протащим мышью, не отпуская ее левую кнопку, то будут выделены сразу несколько строк. Чтобы выделить сразу весь текст, достаточно нажать комбинацию клавиш <Ctrl>+<A>.

Чтобы вырезать выделенный фрагмент, нажмем комбинацию клавиш <Ctrl>+<X> или выберем пункт **Cut** в меню **Edit**; чтобы скопировать его — комбинацию клавиш <Ctrl>+<C> или пункт **Copy** в том же меню. Для вставки текста в нужное место расположим там текстовый курсор и нажмем комбинацию клавиш <Ctrl>+<V> или выберем пункт **Paste** в меню **Edit**. Мы также можем стереть выделенный текст, нажав клавишу или выбрав пункт **Clear** все в том же незаменимом меню **Edit**.

Если нам надо всего лишь переместить выделенный фрагмент текста с места на место, вовсе не нужно беспокоить меню **Edit** — мы можем просто "взять" и перетащить его мышью. Если же мы хотим скопировать этот фрагмент в другое место, достаточно при перетаскивании удерживать нажатой клавишу <Ctrl>.

Если щелкнуть правой кнопкой мыши по выделенному тексту, на экране появится *контекстное меню* — небольшое меню, предоставляющее доступ к

операциям, которые можно выполнить над этим текстом. Так, в контекстном меню выделенного текста содержатся, кроме всех прочих, пункты **Cut**, **Copy** и **Paste**.

Если мы сделаем ошибку, не беда. Достаточно воспользоваться пунктом **Undo** <команда> меню **Edit** или комбинацией клавиш <Ctrl>+<Z>, чтобы вернуться к предыдущему состоянию Web-страницы — выполнить *откат*.

Закончили с текстом? Теперь давайте взглянем на инструментарий документа, точнее, на находящееся в нем небольшое текстовое поле (рис. 3.3). В это поле вводится так называемое *название Web-страницы*. Это название отображается в строке заголовка окна Web-обозревателя, когда он отображает страницу, а также записывается в его *"историю"* (список посещенных к данному моменту Web-страниц).

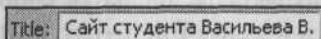


Рис. 3.3. Поле ввода названия Web-страницы в инструментарии документа

Дадим нашей первой Web-странице название "Сайт студента Васильева В. В.", для чего введем его в показанное на рис. 3.3 поле ввода и нажмем клавишу <Enter>.

Вот так, не написав ни строчки HTML-кода, мы получили правильно оформленную Web-страницу.

Сохранение и открытие Web-страниц в Dreamweaver

Здесь мы немного отвлечемся от Web-дизайна и поговорим о том, каким образом выполняется сохранение, открытие и закрытие Web-страниц в Dreamweaver. Это нам обязательно пригодится на дальнейших занятиях.

Итак, для сохранения Web-страницы нужно выбрать пункт **Save** в меню **File** или нажать комбинацию клавиш <Ctrl>+<S>. Если до этого мы не сохраняли нашу Web-страницу (а мы ее еще не сохраняли), на экране появится стандартное диалоговое окно сохранения файла Windows. Остается только ввести в поле ввода этого окна имя файла, в котором будет сохранена страница, и нажать кнопку сохранения. Если же мы сохраняем нашу страницу не в первый раз, Dreamweaver не будет запрашивать у нас имя файла — ведь мы его уже задали, — а "тихо" выполнит сохранение.

Кстати, сейчас самое время сохранить нашу первую Web-страничку в файле. Дадим этому файлу имя 3.1.htm (третье занятие, первый пример), чтобы потом не запутаться.

Пункт **Save All**, находящийся в меню **File**, позволит нам сохранить сразу все открытые Web-страницы.

Чтобы открыть сохраненную ранее Web-страницу, нужно выбрать в меню **File** пункт **Open** или нажать комбинацию клавиш <Ctrl>+<O>. На экране появится стандартное диалоговое окно открытия файла Windows; выбираем в нем нужный файл и нажимаем кнопку открытия.

Когда Web-страница сделана, ее нужно закрыть. Проще всего сделать это, щелкнув кнопку закрытия соответствующего окна документа (не главного окна программы!). Также можно выбрать пункт **Close** меню **File** или нажать комбинацию клавиш <Ctrl>+<W>. После этого страница будет закрыта, но сам Dreamweaver останется открытым. А пункт **Close All** позволит закрыть сразу все открытые Web-страницы.

Сам же Dreamweaver можно закрыть, щелкнув кнопку закрытия его главного окна. Также можно выбрать в меню **File** пункт **Exit** или нажать комбинацию клавиш <Ctrl>+<Q> или <Alt>+<F4>. Перед закрытием Dreamweaver спросит нас, сохранять ли отредактированные, но еще не сохраненные страницы, которые мы к этому времени открыли. Хотя лучше всего сохранять все страницы перед окончанием работы самим — так надежнее.

Просмотр Web-страницы в Web-обозревателе

Хотя Dreamweaver в режиме просмотра страницы и представляет ее почти в таком виде, как она будет показана в Web-обозревателе, все же часто возникает необходимость просмотреть ее в самом Web-обозревателе. Дело в слове "почти": все-таки Dreamweaver не может учесть многие тонкости конкретной программы просмотра Web-страниц. И такая возможность предусмотрена: не закрывая окна документа, мы можем вызвать любой из установленных на вашем компьютере Web-обозревателей и оценить окончательный вид своего творения, так сказать, "в родной обстановке".

Давайте посмотрим нашу Web-страницу в Internet Explorer — Web-обозревателе, установленном, пожалуй, на все компьютеры, работающие под управлением Windows. Для этого нажмем клавишу <F12>. На экране появится окно Web-обозревателя, отображающее нашу страничку.

А теперь давайте займемся ее оформлением.

Форматирование абзацев

Прежде всего, сделаем нашей странице нормальный "кричащий" заголовок, как у газеты. В первой строке как раз набрано Сайт студента Васильева В. В. — это прекрасный кандидат в заголовки.

Чтобы превратить обычный абзац в заголовок, нужно воспользоваться панелью **Properties**, о которой мы упоминали на *занятии 2*. Эта панель называется *редактором свойств* (рис. 3.4). Если она почему-то не видна на экране, раскройте нижний док и проверьте, включен ли пункт-выключатель **Properties** в меню **Window**.

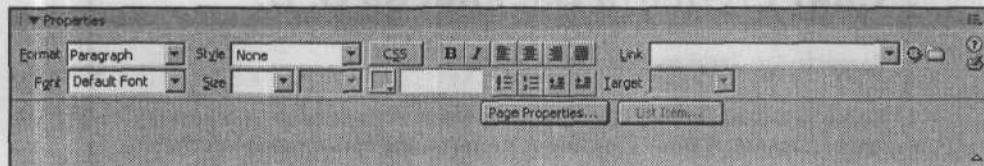


Рис. 3.4. Редактор свойств (панель **Properties**)

Редактор свойств (давайте уж называть эту панель так) можно сворачивать и разворачивать. Изначально он свернут, не показывая некоторых, малоиспользуемых с точки зрения разработчиков Dreamweaver свойств. Но лучше всего сразу же его развернуть, для чего достаточно щелкнуть мышью по кнопке, имеющей вид стрелки, направленной вниз, — она расположена в правом нижнем углу панели. После этого редактор свойств покажет вам все доступные свойства. (На рис. 3.4 редактор свойств находится в развернутом виде.) Заметим, что при этом кнопка разворачивания изменит вид на стрелку вверх, и при щелчке на ней мы, наоборот, свернем редактор свойств.

В редакторе свойств нам понадобится раскрывающийся список **Format**. Пункт **Paragraph** этого списка форматирует текст как обычный абзац. А вот пункты **Heading 1**, ..., **Heading 6** позволяют превратить его в *заголовок* определенного *уровня*. Таких уровней язык HTML предусматривает шесть — от самого главного, которым озаглавливаются сами Web-страницы, до самого нижнего, пригодного для отдельных параграфов.

Нам нужно озаглавить всю Web-страницу, поэтому используем заголовок первого уровня. Поставим текстовый курсор на строку Сайт студента Васильева В. В. и выберем в раскрывающемся списке **Format** пункт **Heading 1**. Получившийся результат показан на рис. 3.5.

Кстати, то же самое мы можем сделать, используя главное меню или контекстное меню окна документа. В первом случае нужно выбрать пункт **Paragraph Format** в меню **Text** и в появившемся на экране подменю — нужный формат абзаца. Во втором случае достаточно щелкнуть правой кнопкой мыши на первой строке, также выбрать пункт **Paragraph Format** и в появившемся подменю — формат абзаца.

Те, кто предпочитает не отрывать пальцы от клавиатуры, могут задать формат заголовка комбинацией клавиш <Ctrl>+<клавиша>, где <клавиша> —

одна из клавиш <I>, ..., . Формат обычного абзаца в этом случае можно задать, нажав комбинацию клавиш <Ctrl>+<Shift>+<P>.

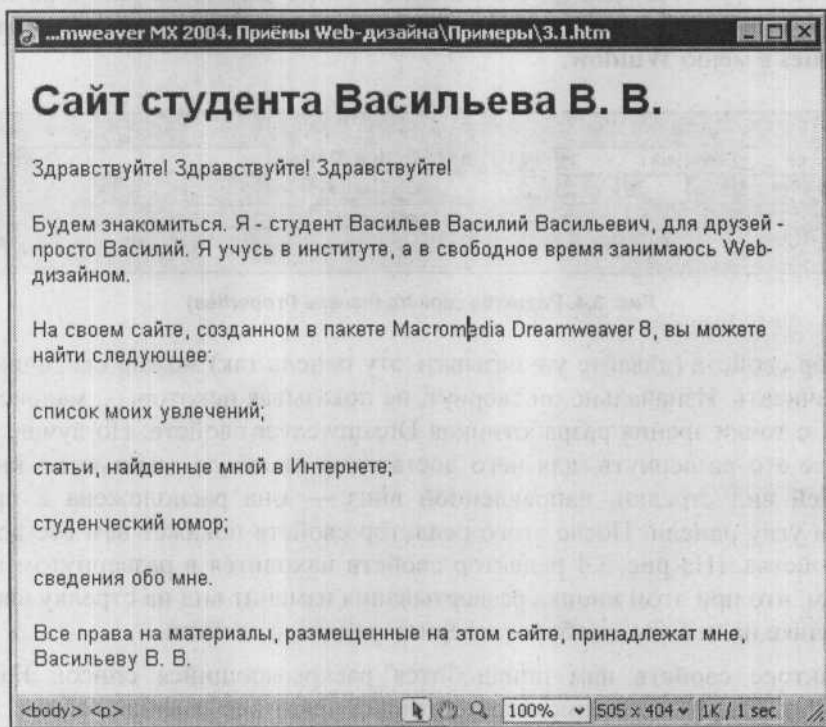



Рис. 3.5. Созданный нами заголовок Web-страницы

Теперь давайте выровняем заголовок по центру. В этом нам помогут кнопки выравнивания , расположенные в редакторе свойств. Эти кнопки задают выравнивание соответственно (порядок перечисления слева направо):

- ◆ по левому краю;
- ◆ по центру;
- ◆ по правому краю;
- ◆ по ширине.

Внимание!

Выравнивание по ширине поддерживается не всеми Web-обозревателями.

Итак, снова поставим текстовый курсор на заголовок и нажмем кнопку выравнивания по центру. Кнопка останется нажатой, и у нас получится выровненный по центру заголовок (рис. 3.6).

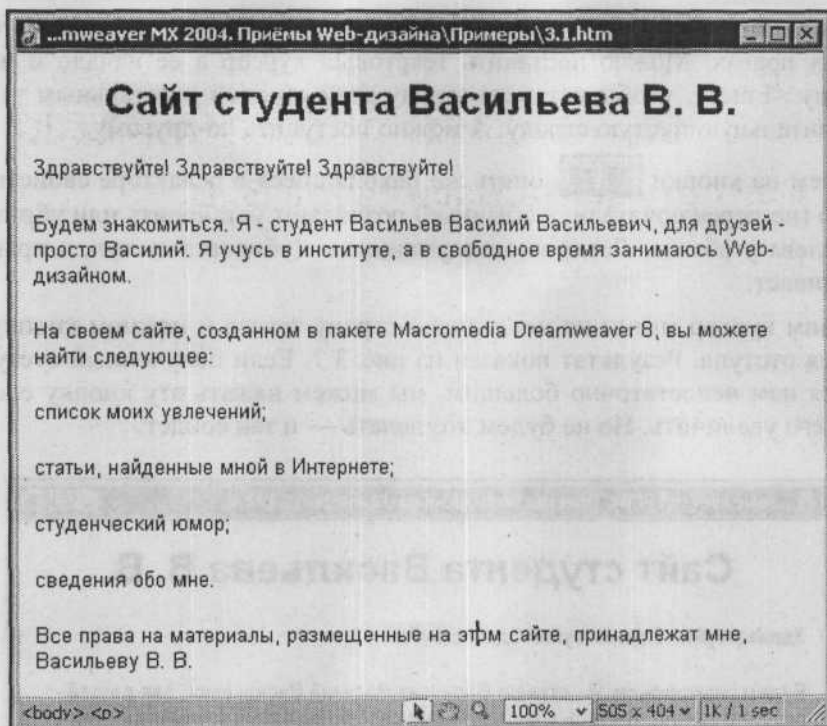


Рис. 3.6. Центрированный заголовок


Кнопки, задающие выравнивание, необычны. В данный момент времени может быть нажата ("включена") только одна из них. Такие кнопки называются *кнопками-переключателями*.

Интересная деталь: если щелкнуть еще раз по нажатой кнопке-переключателю, она "отожмется". В этом случае для абзаца будет установлено выравнивание по умолчанию, обычно — по левому краю.

Задать выравнивание абзаца можно с помощью подменю **Align** меню **Text** или аналогичного подменю контекстного меню, в котором содержатся пункты, задающие различные варианты выравнивания: **Left** (выравнивание по левому краю), **Center** (по центру), **Right** (по правому краю) и **Justify** (по ширине). Эти пункты ведут себя так же, как кнопки-переключатели: при выборе любого из них левее его названия появляется галочка, говорящая о том, что пункт "включен". "Включенным" в данный момент времени может быть только один пункт, поэтому они называются *пунктами-переключателями*.

Осталось только сказать, что для задания выравнивания еще можно нажать комбинацию клавиш <Ctrl>+<Alt>+<Shift>+<клавиша>, где <клавиша> —

Теперь давайте как-нибудь выделим последнюю строку со сведениями об авторских правах. Можно поставить текстовый курсор в ее начало и нажать клавишу <Enter>, чтобы вставить между этой строкой и остальным текстом дополнительную пустую строку. А можно поступить по-другому.

Взглянем на кнопки , опять же находящиеся в редакторе свойств. Эти кнопки (не переключатели, а обычные) позволяют установить или убрать *отступ* слева у абзаца. Левая кнопка уменьшает (убирает) отступ, а правая — увеличивает.

Поставим курсор мыши на последнюю строку текста и нажмем кнопку увеличения отступа. Результат показан на рис. 3.7. Если полученный отступ покажется нам недостаточно большим, мы можем нажать эту кнопку еще раз, чтобы его увеличить. Но не будем это делать — и так сойдет.

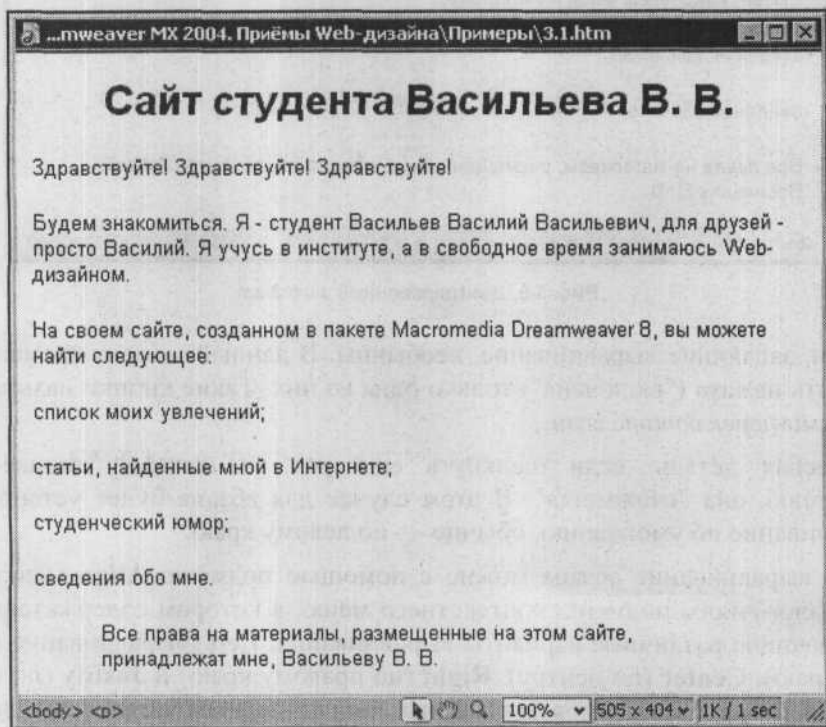



Рис. 3.7. Строка с отступом

Аналогичный результат дают пункты **Outdent** (убрать отступ) и **Indent** (задать отступ) меню **Text** или подменю **List** контекстного меню. Также можно нажать комбинацию клавиш <Ctrl>+<Shift>+<J> для увеличения отступа и

Теперь обратим внимание на список разделов нашего будущего сайта. И вспомним, что "обычные" текстовые редакторы поддерживают создание нумерованных и маркированных списков. Пункты *нумерованных* (упорядоченных) списков, как вы знаете, обозначаются порядковыми номерами, а пункты *маркированных* (неупорядоченных) списков — какими-либо значками. Давайте и мы создадим такой список.

Первое, что нам будет нужно сделать, — это выделить все строки, которые станут пунктами создаваемого списка. У нас их четыре.

Для того чтобы преобразовать выделенные строки в список, воспользуемся кнопками-переключателями редактора свойств . Левая кнопка создает из выделенных абзацев маркированный список, а правая — нумерованный. Если же обе кнопки "отключены", пункт списка, на котором стоит текстовый курсор, превращается в обычный абзац.

Нажмем левую кнопку. Созданный нами маркированный список показан на рис. 3.8.

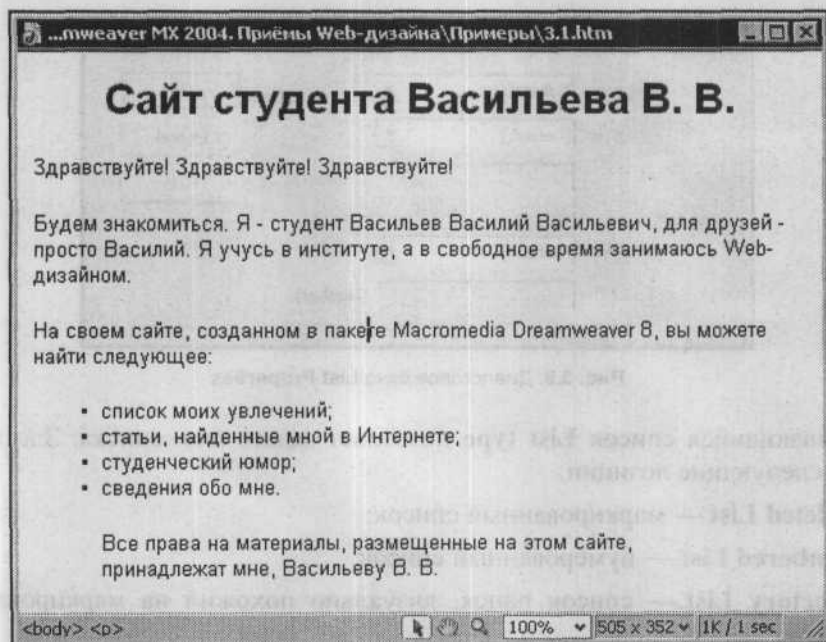


Рис. 3.8. Маркированный список

В меню **Text** и в контекстном меню предусмотрено специальное подменю **List**, выполняющее те же действия. В табл. 3.1 описаны все пункты

переключатели этого подменю. Каких-либо комбинаций клавиш для них не предусмотрено.

Таблица 3.1. Пункты-переключатели подменю **List**

Пункт	Описание
None	Обычный абзац текста
Unordered List	Маркированный список
Ordered List	Нумерованный список
Definition List	Список определений. Каждая нечетная строка становится как бы термином, требующим разъяснения, а каждая четная — собственно разъяснением

Если текстовый курсор стоит на любом пункте списка, становится доступной кнопка **List Item** редактора свойств. Если ее нажать, на экране появится диалоговое окно **List Properties** (рис. 3.9), с помощью которого мы можем задать некоторые параметры списка.

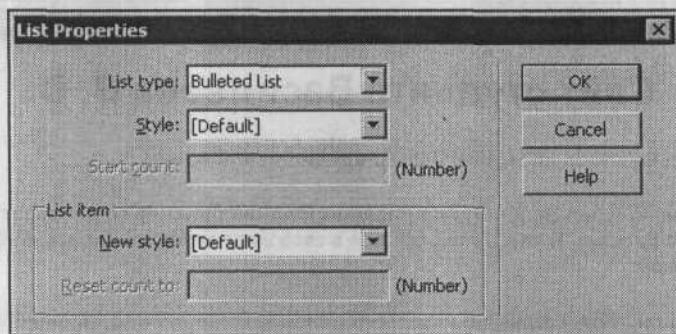


Рис. 3.9. Диалоговое окно **List Properties**

Раскрывающийся список **List type** позволяет задать *тип* списка. Здесь доступны следующие позиции:

- ◆ **Bulleted List** — маркированный список;
- ◆ **Numbered List** — нумерованный список;
- ◆ **Directory List** — список папок, визуально похожий на маркированный список;
- ◆ **Menu List** — список-меню, также визуально похожий на маркированный список.

Если выбран первый или второй пункт, становятся доступными другие элементы управления

Раскрывающийся список **Style** позволяет задать *стиль* нумерации списка или маркера. Если выбран маркированный список, доступны следующие значения:

- ◆ **Bullet** — кружок с заливкой;
- ◆ **Square** — квадратик;
- ◆ **[Default]** — маркер по умолчанию, большинство Web-обозревателей в этом случае выводят кружок с заливкой.

Если же выбран нумерованный список, доступны такие значения:

- ◆ **Number** — арабские цифры;
- ◆ **Roman Small** — малые римские цифры;
- ◆ **Roman Large** — большие римские цифры;
- ◆ **Alphabet Small** — малые латинские буквы;
- ◆ **Alphabet Large** — большие латинские буквы;
- ◆ **[Default]** — нумерация по умолчанию, большинство Web-обозревателей в этом случае нумеруют пункты списка арабские цифры.

Кроме того, если выбран нумерованный список, доступно поле ввода **Start count** (Начать нумерацию с), где мы сможем ввести номер, с которого начнется нумерация строк списка.


Все эти настройки затрагивали список в целом. В нижней половине окна расположена группа элементов управления **List item**, затрагивающая только тот пункт списка, на котором находится текстовый курсор. В эту группу входят раскрывающийся список **New style** и поле ввода **Reset count to**, аналогичные раскрывающемуся списку **Style** и полю ввода **Start count** верхней части окна.

После нажатия кнопки **OK** все заданные нами параметры списка будут применены. Нажатие кнопки **Cancel** позволяет отказаться от них.

Форматирование фрагментов текста

Итак, мы научились форматировать целиком абзацы текста. Теперь поговорим о том, что можно сделать с отдельными фрагментами этих абзацев.

Все текстовые редакторы, например Microsoft Word или поставляемый в составе Windows WordPad, позволяют менять начертание и размер шрифта текста, делать его полужирным, курсивным или подчеркнутым, а также менять его цвет. Многие из этого доступно и в Dreamweaver.

Для начала сделаем некоторые слова на нашей Web-странице полужирными и курсивными. И помогут нам в этом две кнопки изменения *начертания* . Левая (B) кнопка позволяет сделать шрифт полужирным, а правая (I)

Любая из этих кнопок может быть нажата (тогда, соответственно, "полужирность" или "курсивность" будет включена) или отжата ("полужирность" или "курсивность" убирается) независимо от другой. Такие кнопки называются *кнопками-выключателями*.

Выделим слова *Васильев Василий Васильевич* и нажмем кнопку **B**. Выделенные слова станут полужирными. Теперь выделим полностью последний абзац (со сведениями об авторских правах) и нажмем **I**. Последний абзац станет курсивным.

Вместо нажатия кнопки **B** можно выбрать пункт-выключатель **Bold** в подменю **Style** меню **Text** или контекстного меню. В том же подменю существует пункт **Italic** — аналог кнопки **I**. Также можно воспользоваться комбинациями клавиш <Ctrl>+ и <Ctrl>+<I> соответственно.

А вот чтобы включить или отключить подчеркивание текста линией, нам все равно придется воспользоваться пунктом-выключателем **Underline** подменю **Style** меню **Text** или контекстного меню. Ни кнопки, ни комбинации клавиш для этого действия не предусмотрено.

Продолжим изучение подменю **Style**. В нем мы видим пункт-выключатель **Strikethrough**, включив который мы сделаем выделенный фрагмент текста зачеркнутым. Остальные же пункты подменю **Style** перечислены в табл. 3.2. Если хотите, можете поэкспериментировать с ними и посмотреть на результаты.

Таблица 3.2. Остальные пункты-выключатели подменю **Style**

Пункт	Описание
Teletype	Текст, выведенный устройством вывода компьютера ("телетайп")
Emphasis	Обычный курсивный текст
Strong	Обычный полужирный текст
Code	Фрагмент исходного кода программы на каком-либо языке программирования (команд, имен переменных, ключевых слов и т. п.)
Variable	Используется для обозначения в тексте имен переменных программы на каком-либо языке программирования
Sample	Информация, выведенная какой-либо программой пользователю
Keyboard	Текст, который пользователь должен ввести с клавиатуры
Citation	Цитата
Definition	Определение какого-либо термина

Теперь давайте зададим шрифт, которым набран наш текст. Делается это с помощью меню **Font**, которое находится в редакторе

свойств. Для смены самого шрифта используется раскрывающийся список **Font**, а для смены его размера (*кегля*, как говорят полиграфисты) — список **Size**. Но если мы откроем любой из этих списков, то увидим, что количество доступных шрифтов и их размеров очень ограничено. Почему?

Вспомним, зачем создавалась Всемирная паутина WWW. Чтобы любой человек, пользуясь любым компьютером, находясь в любой части света, смог посмотреть любую Web-страницу. А так как разные компьютеры имеют разные параметры видеосистемы, в частности разные стандартные шрифты, то задача одинакового отображения сильно усложняется. Поэтому стандарт языка HTML определяет несколько *стандартных шрифтов*, которые обязательно должны быть установлены на компьютере клиента, и несколько, а именно семь, стандартных размеров этих самых шрифтов.

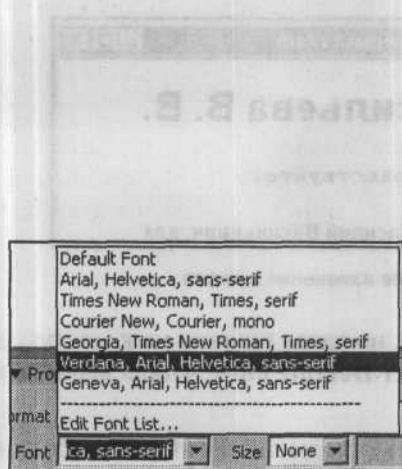


Рис. 3.10. Раскрывающийся список **Font**, находящийся в редакторе свойств

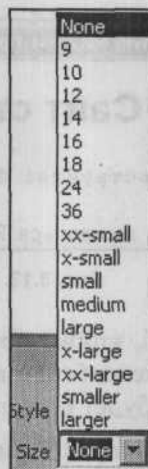


Рис. 3.11. Раскрывающийся список **Size**, расположенный в редакторе свойств

Откроем раскрывающийся список **Font** (рис. 3.10). Каждый его пункт соответствует стандартному шрифту HTML, точнее, целому списку шрифтов, разделенных запятыми. Если первого в таком списке шрифта на клиентском компьютере нет, Web-обозреватель ищет второй, третий и т. д., пока не доберется до последнего шрифта, который уж точно должен там быть.

Пункт **Default Font** этого списка сбрасывает шрифтовые установки, точнее, возвращает их к установкам по умолчанию. Тогда выбор шрифта отдается на откуп Web-обозревателю.

Соответственно в раскрывающемся списке **Size** (рис. 3.11) отображены все доступные размеры шрифтов. Пункты **xx-small**, **x-small**, **small**, **medium**, **large**, **x-large** и **xx-large** переопределяют все остальные стандартные

рядке от самого мелкого до самого крупного. Пункты **larger** и **smaller** задают размер шрифта соответственно на одну ступень больше и меньше. Числовые пункты задают размер шрифта в пунктах; также можно ввести нужный нам размер вручную прямо в список. А пункт **None** сбрасывает размер шрифта в установки по умолчанию.

Давайте увеличим строку **Здравствуйте! Здравствуйте! Здравствуйте!** на одну ступень. Для этого выделим его и выберем в раскрывающемся списке **Size** пункт **larger**. После этого размер шрифта, которым набрано приветствие, несколько увеличится.

Пока первый абзац выделен, давайте изменим его шрифт. Для примера выберем в списке **Format** пункт **Courier New, Courier, mono**. Результат показан на рис. 3.12.

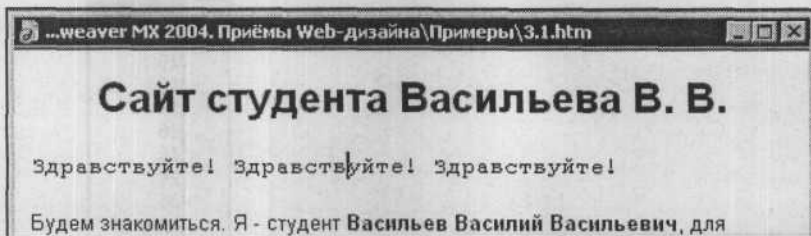
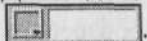


Рис. 3.12. Текст приветствия после изменения шрифта

Да, не сказать, чтобы этот шрифт был плох, но здесь он явно не к месту... Давайте сразу же выберем в списке **Font** пункт **Default Font**, чтобы вернуть все назад. Пусть будет шрифт по умолчанию!

Кстати, задать шрифт мы можем и с помощью подменю **Font**, расположенного в меню **Text** и контекстном меню. Для смены размера шрифта Dreamweaver предоставляет целых два подменю: **Size** и **Size Change**; оба они располагаются в меню **Text**, но в контекстном меню они объединены в одно подменю **Size**. Меню **Size** содержит семь числовых, от 1 до 7, пунктов-переключателей, задающих соответствующие размеры, и пункт **Default**, задающий размер по умолчанию. А меню содержит пункты **+1**, **+2**, **+3**, **+4**, **-1**, **-2** и **-3**, задающие увеличение или уменьшение размера шрифта на соответствующее количество ступеней.

Теперь давайте выделим цветом название программного продукта, с помощью которого делается этот замечательный сайт. Для задания цвета используется так называемый *селектор цвета*. Вот он — .

Селектор цвета состоит из двух частей. Справа находится поле ввода, где вводится код нужного цвета в формате **RGB**. RGB расшифровывается как "Red, Green, Blue" — "красный, зеленый, синий". А сам формат задания цвета

выглядит так — #RRGGBB, где RR — это шестнадцатеричное число от 0 до FF, задающее долю в окончательном цвете красной составляющей, GG — зеленой, а BB — синей. Пример задания цвета в формате RGB — #336699 (это тускло-голубой цвет).

Но вряд ли вы помните наизусть шестнадцатеричные коды ваших любимых цветов. Поэтому в левой части селектора цвета находится кнопка вызова *окна выбора цвета*, которое в раскрытом виде показано на рис. 3.13.

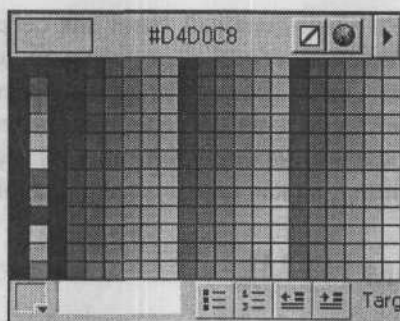


Рис. 3.13. Открытое окно селектора цветов

Большую часть этого окна занимает *палитра*, где, собственно, мы и ищем нужный цвет. Найдя его, просто щелкаем по нему мышью, и окно выбора цвета закрывается.

В небольшом текстовом поле, расположенном в центре верхней серой полосы окна, высвечивается шестнадцатеричный код цвета, над которым мы в данный момент держим курсор мыши. Это очень удобно: впоследствии нам уже не придется долго искать цвет в палитре — мы просто введем в поле ввода селектора код этого цвета в формате RGB.

Если же ни один цвет из палитры нам не подходит, мы можем с помощью того же курсора мыши (имеющего вид пипетки) "взять" нужный цвет откуда угодно: с рабочего стола, из любого окна, из любого элемента управления; для этого достаточно просто щелкнуть мышью по нужному месту экрана.

Правда, выбирать цвет для текста лучше все-таки из палитры. Почему? Чтобы ответить на этот вопрос, снова поговорим о проблеме межплатформенной совместимости, которую пришлось решать создателям языка HTML и WWW.

Как мы уже знаем, разные компьютеры имеют разные параметры видеосистемы. Одни могут отображать всего шестнадцать цветов, а другие — все 16,7 миллионов (что с лихвой перекрывает цветовую разрешающую способность человеческого глаза). Разумеется, при таком богатом множестве компьютерных платформ Web-дизайнеру не стоит и рассчитывать, что все его

цвета и оттенки будут отображены правильно везде. Поэтому стандарт HTML определяет так называемый *безопасный набор цветов*, который гарантированно должен отображаться правильно всеми программами на всех компьютерах. В палитре окна выбора цвета как раз и отображается этот самый безопасный набор цветов.

Что ж, с цветами все ясно. Теперь выделим слова Macromedia Dreamweaver 8 и выберем в палитре, скажем, темно-зеленый цвет. Теперь посмотрим, что у нас получилось.

Честно говоря, выглядит темно-зеленый текст так себе... Давайте лучше убедим эту "зелень", вернув тексту его цвет по умолчанию — черный. Для этого снова выделим слова Macromedia Dreamweaver 8, переключимся на редактор свойств, удалим код цвета в поле ввода селектора цвета, выделив его и нажав клавишу , и напоследок нажмем клавишу <Enter>. И снова сохраним нашу первую Web-страницу, нажав комбинацию клавиш <Ctrl>+<S>.

Работа с кодом HTML

Настала пора подробнее познакомиться с языком HTML. А то что это там Dreamweaver делает за нашей спиной? Надо проверить!

Три режима отображения Web-страницы

И первый вопрос, возникающий перед нами при попытке добраться до кода HTML, — собственно, как нам до него добраться? Очень просто. Ведь окно документа Dreamweaver может показывать редактируемую Web-страницу в трех *режимах отображения*.

Посмотрим на инструментарий документа. В его левой части видны три кнопки-переключателя, показанные на рис. 3.14. Они переключают три доступных режима отображения (перечислены в порядке справа налево): страница, страница и HTML-код, только HTML-код.



Рис. 3.14. Кнопки переключения режимов отображения Web-страницы

Видно, что в данный момент нажата крайняя правая кнопка. Она включает *режим отображения страницы*, предлагаемый Dreamweaver по умолчанию. В этом режиме мы можем работать с Web-страницей по принципу WYSIWYG. Собственно, мы с ним в данный момент и работаем.

Теперь нажмем на среднюю кнопку, включающую *режим отображения страницы и кода HTML*. На рис. 3.15 видно, что из этого получится.

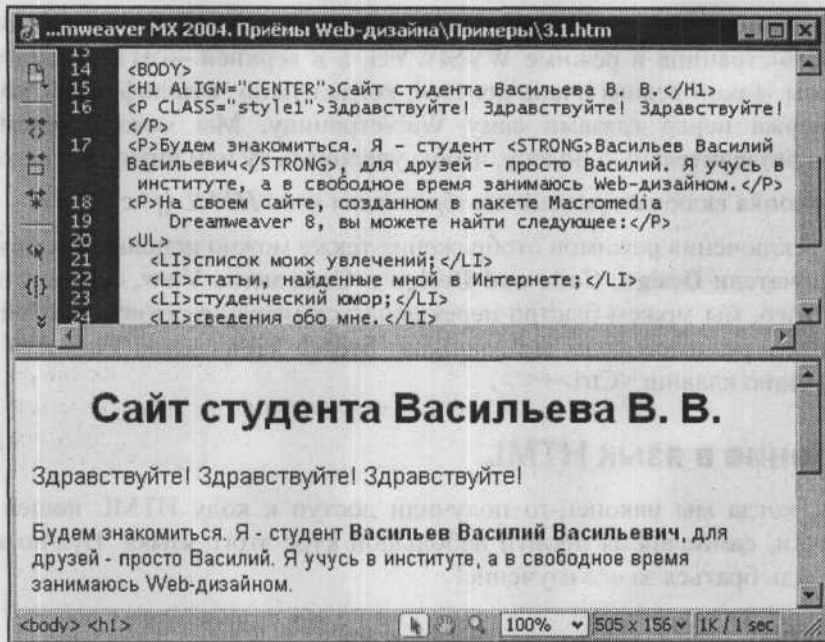


Рис. 3.15. Режим отображения страницы и кода HTML

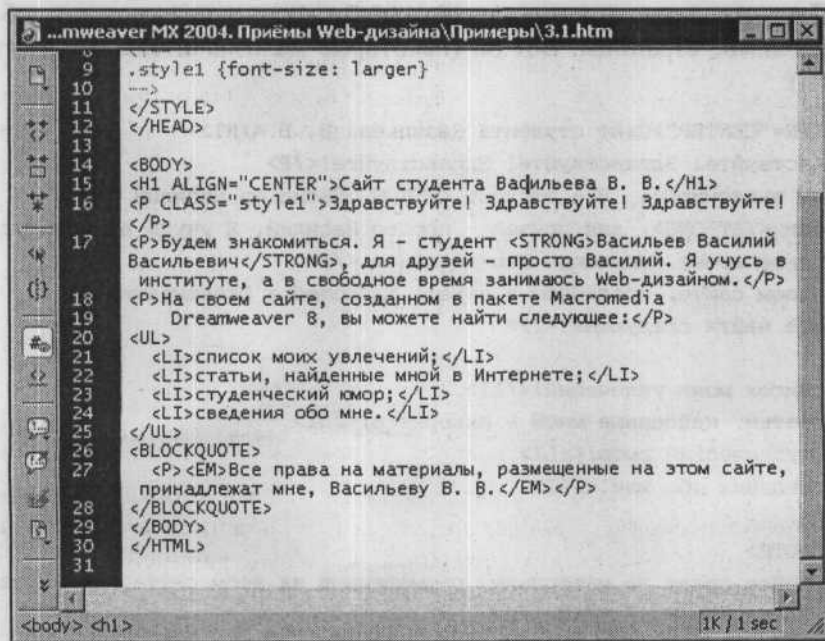


Рис. 3.16. Режим отображения кода HTML

Сейчас окно документа разделено на две части. В нижней части отображается сама Web-страница в режиме WYSIWYG, а в верхней — HTML-код нашей страницы. Такой режим очень полезен, когда нужно "отшлифовать" код, при этом держа перед глазами саму Web-страницу. Мы можем перемещать мышью разделительную полосу, чтобы увеличить ту или иную часть окна.

Левая кнопка включает режим отображения кода HTML (рис. 3.16).

Для переключения режимов отображения также можно использовать пункты-переключатели **Design**, **Code and Design** и **Code** меню **View**, соответственно. Кроме того, мы можем быстро переключаться между режимом отображения страницы и режимом кода, выбрав пункт **Switch Views** меню **View** или нажав комбинацию клавиш <Ctrl>+<'>.

Введение в язык HTML

Теперь, когда мы наконец-то получили доступ к коду HTML нашей Web-странички, самое время пройти небольшой курс этого языка. Нужно же когда-нибудь браться за его изучение!

Теги HTML. Форматирование текста

Давайте переключимся в режим отображения HTML-кода и посмотрим на него. Пока что сосредоточимся на фрагменте кода, описывающего собственно содержимое страницы. Вот он (некоторые малозначительные фрагменты удалены):

```
<H1 ALIGN="CENTER">Сайт студента Васильева В. В.</H1>
<P>Здравствуйете! Здравствуйете! Здравствуйете!</P>
<P>Будем знакомиться. Я - студент <STRONG>Васильев Василий
Васильевич</STRONG>, для друзей - просто Василий. Я учусь в институте, а
в свободное время занимаюсь Web-дизайном.</P>
<P>На своем сайте, созданном в пакете Macromedia Dreamweaver 8,
вы можете найти следующее:</P>
<UL>
  <LI>список моих увлечений;</LI>
  <LI>статьи, найденные мной в Интернете;</LI>
  <LI>студенческий кмор;</LI>
  <LI>сведения обо мне.</LI>
</UL>
<BLOCKQUOTE>
  <P><EM>Все права на материалы, размещенные на этом сайте, принадлежат
  мне, Васильеву В. В.</EM></P>
</BLOCKQUOTE>
```

Здесь мы видим набранный нами ранее текст, "разбавленный" какими-то непонятными словами, взятыми в знаки < и >. Это и есть теги языка HTML, о которых мы упоминали на *занятии 1*, — особые команды, выделяющие фрагменты текста, дающие им особые свойства или просто задающие их внешний вид.

Возьмем, к примеру, теги <P> и </P>. Они создают обычный текстовый абзац. Буква P (от английского paragraph — абзац) это *название тега*, которое помещается между знаками < и >. Собственно, эти знаки и являются признаком тега; больше нигде в тексте они присутствовать не должны.

Тег <P> задает начало фрагмента текста, который станет отдельным абзацем, и поэтому называется *открывающим*. Тег </P> задает конец абзаца и называется *закрывающим*. Заметим, что закрывающий тег имеет то же название, что и открывающий (т. е. P), но между ним и знаком < стоит символ / ("слэш") — отличительный признак закрывающего тега. А сам фрагмент текста, на который оказывает влияние тег, называется его *содержимым*.

Здесь мы сразу же столкнулись с так называемыми *парными* тегами, имеющими открывающий тег, закрывающий тег и содержимое. Повторим для памяти: открывающий тег задает начало содержимого — текстового фрагмента, к которому применяется действие тега, — а закрывающий обозначает его конец.

Теперь обратим внимание на первую строку приведенного выше фрагмента. Нетрудно догадаться, что парный тег <H1>...</H1> задает заголовок первого уровня. Но что это за слово ALIGN="CENTER", помещенное между названием этого тега и знаком >? Выходит, оно тоже часть этого тега?

Да, это так. Здесь мы столкнулись с так называемыми *атрибутами тега* — особыми дополнительными параметрами тегов, задающими некоторые дополнительные условия или значения. В нашем случае парный тег <H1> (будем дальше писать обозначения парных тегов так, все равно закрывающий тег не имеет смысла без открывающего) содержит атрибут ALIGN, задающий выравнивание абзаца-заголовка. Этому атрибуту присвоено значение CENTER — выравнивание по центру. Обратим внимание, что значения атрибутов обязательно помещаются в двойные кавычки.

Атрибуты бывают обязательными и необязательными. *Обязательный* атрибут в любом случае должен присутствовать в теге. *Необязательный* же атрибут присутствовать не обязан; если же он не указан, то Web-обозреватель считает, что ему присвоено некое значение по умолчанию. Так, атрибут ALIGN тега <H1> — необязательный; если он не указан, Web-обозреватель сам решает, какое выравнивание применить к данному заголовку (как правило, по левому краю).

Продолжим изучение HTML-кода нашей страницы. Парный тег `` предписывает Web-обозревателю вывести свое содержимое полужирным шрифтом, а парный тег `` — курсивом. Третий парный тег — `<BLOCKQUOTE>` — задает для своего содержимого отступ. Здесь все просто и понятно.

На заметку

Для создания полужирного текста можно использовать парный тег ``, а для создания курсива — парный тег `<I>`. Но эти теги (как и многие другие) объявлены устаревшими и не рекомендованными к использованию во вновь создаваемых Web-страницах.

А вот в создании списка участвуют целых два парных тега — `` и ``. Первый тег задает сам список и его тип, в нашем случае — маркированный. (Нумерованный список создается парным тегом ``.) А второй тег задает отдельный пункт этого списка. Посмотрим еще раз на приведенный выше листинг — и все станет ясно.

Вложенность тегов

Рассматривая код HTML нашей Web-страницы, мы обязательно заметим, что некоторые теги вложены друг в друга. Так, в последнем абзаце, рассказывающем об авторских правах, тег `` вложен в тег `<P>`, а тот, в свою очередь, — в тег `<BLOCKQUOTE>`. Такая *вложенность* или *иерархия* тегов в HTML встречается на каждом шагу. Давайте выясним, что она дает и зачем применяется.

Вот HTML-код последнего абзаца нашей страницы:

```
<BLOCKQUOTE>
  <P><EM>Все права на материалы, размещенные на этом сайте, принадлежат
  мне, Васильеву В. В.</EM></P>
</BLOCKQUOTE>
```

Здесь мы вложили тег `<P>` в тег `<BLOCKQUOTE>`, и Web-обозреватель отобразит содержимое тега `<P>` с отступом. Кроме того, мы вложили в уже вложенный тег `<P>` тег ``, заставив Web-обозреватель отобразить содержимое тега `<P>` курсивом. Фактически мы применили к последнему абзацу очень сложное форматирование, всего-навсего правильно вложив разные теги друг в друга.

Вот еще один пример вложенности тегов:

```
<P>Будем знакомиться. Я - студент <STRONG>Васильев Василий
Васильевич</STRONG>, для друзей - просто Василий. Я учусь в институте, а
в свободное время занимаюсь Web-дизайном.</P>
```

Здесь слова "Васильев Василий Васильевич", находящиеся в абзаце (тег <P>), будут выделены полужирным шрифтом (тег).

Да, вложенность тегов — хорошая штука! Но здесь легко допустить ошибку. Нужно очень строго следить за порядком вложенности тегов: порядок следования закрывающих тегов должен быть обратным порядку следования тегов открывающих. Иначе говоря, если открывающие теги следуют в порядке

```
<BLOCKQUOTE> <P> <EMP>
```

то закрывающие должны следовать так:

```
</EMP> </P> </BLOCKQUOTE>
```

Напоследок выучим несколько новых терминов (а куда от них деваться?). Тег, в который вложен данный тег, называется *родительским тегом* или *родителем*, а теги, которые в него вложены, — *дочерними тегами* или *потомками*. Например, для тега <P> приведенного выше примера родительским тегом будет <BLOCKQUOTE>, а дочерним — <EMP>.

Служебные теги. Две секции Web-страницы

А теперь обратим внимание на остальную часть HTML-кода нашей страницы, которую мы до сих пор не принимали во внимание. Зачем она нужна? И что за странные теги в ней содержатся?

Давайте "вынесем" еще не рассмотренную часть HTML-кода в отдельный листинг (уже рассмотренный нами фрагмент пропущен и заменен троеточием):

```
<HTML>
<HEAD>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html;
charset=windows-1251">
<TITLE>Сайт студента Васильева В. В.</TITLE>
</HEAD>
<BODY>
. . .
</BODY>
</HTML>
```

Видно, что наш HTML-код, задающий содержимое страницы, помещен в парный тег <BODY> (выделен полужирным шрифтом). Этот тег задает так называемую *секцию тела* Web-страницы — т. е. видимое ее содержимое, которое показывает Web-обозреватель. Все, что мы хотим видеть в его окне, должно находиться в секции тела, т. е. быть содержимым тега <BODY>.

Второй незнакомый нам тег — это парный тег <HEAD> (также выделен полу-

служебная информация. (Не путать с обычным текстовым заголовком, задаваемым одним из тегов <h1>...<h6>!) Эта информация не выводится Web-обозревателем на экран, а используется для его внутренних нужд, в частности, задает некоторые параметры Web-страницы.

Какие же сведения содержатся в секции заголовка Web-страницы? Например, название Web-страницы, задаваемое парным тегом <TITLE>. Также там содержатся сведения о кодировке, используемой для текста страницы; кодировка задается особым тегом <META>, имеющим множество атрибутов. Нетрудно заметить, что это кодировка 1251, о чем говорит значение атрибута CONTENT этого тега.

Тег <META> сразу привлекает к себе внимание. Дело в том, что он *одинарный* — не имеющий содержимого. Одинарные теги служат либо для служебных целей (как <META>), либо для помещения на Web-страницу каких-либо специальных элементов (например, графических изображений, которыми мы займемся на *занятии 5*).

Весь HTML-код — и секция заголовка, и секция тела — Web-страницы должен быть вложен внутрь парного тега <HTML>. Это последний тег, который мы здесь рассмотрим.

Все служебные теги — и <BODY>, и <HEAD>, и <HTML> — называются *невидимыми*. Они никак не отображаются Web-обозревателем, в отличие от *видимых* тегов, изученных нами ранее.

На этом мы пока закончим вводной курс языка HTML. В дальнейшем при изучении Dreamweaver мы познакомимся с другими тегами этого языка. Их будет довольно много.

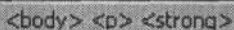
Средства Dreamweaver для работы с тегами HTML

Dreamweaver — пакет, ориентированный больше на подготовленного Web-дизайнера, чем на "чайника". Поэтому он предлагает много возможностей для тех, кто знает язык HTML. Поскольку мы уже познакомились с этим языком, давайте их изучим.

Находясь в режиме отображения Web-страницы, мы можем выделять содержимое отдельных тегов. Так, для выделения всего содержимого родительского тега нужно выбрать в меню **Edit** пункт **Select Parent Tag** или нажать комбинацию клавиш <Ctrl>+<[>. А для выделения первого дочернего тега нужно выбрать в меню **Edit** пункт **Select Child** или нажать комбинацию клавиш <Ctrl>+<]>.

В нижней части окна документа расположена *строка статуса*. Она разделена на три *секции*, из которых для нас представляет интерес самая первая.

секция тегов (рис. 3.17). С помощью этой секции можно выбирать фрагменты текста, являющиеся содержимым того или иного тега HTML. Можно сказать, что секция тегов имеет вид небольшого инструментария, чьи кнопки соответствуют тегам и расположены слева направо в порядке вложенности.



<body> <p>

Рис. 3.17. Секция тегов, расположенная в строке статуса

Предположим, что нам нужно выделить весь текст, помеченный тегом `` (полужирный шрифт). (В нашей первой странице — это имя создателя сайта, студента Васильева.) Для этого сделаем следующее. Поместим текстовый курсор куда-либо на текст Васильев Василий Васильевич, после чего секция тегов примет вид, показанный на рис. 3.17. После этого нажмем кнопку `` этой секции. Весь текст Васильев Василий Васильевич будет выделен.

Если теперь щелкнуть кнопку `<p>`, будет выделен весь текст — содержимое тега `<p>`, т. е. весь абзац. Осталось проверить, как работает кнопка `<body>`. Если ее щелкнуть, будет выделена вся страница (все содержимое тега `<BODY>`).

Еще секция тегов позволяет быстро удалить любой тег, оставив его содержимое. Поставим текстовый курсор на содержимое ненужного тега, щелкнем по нужной кнопке секции тегов правой кнопкой мыши и выберем в контекстном меню пункт **Remove Tag**. Кстати, этот же пункт доступен и в контекстном меню окна документа.

И напоследок — сюрприз! В составе Dreamweaver поставляется полное руководство по языку HTML от издательства O'Reilly. Его содержимое отображается в особой панели **Reference** (рис. 3.18). Чтобы ее открыть, нужно выбрать пункт **Reference** в меню **Window** или нажать комбинацию клавиш `<Shift>+<F1>`. При этом в панели **Reference** будет отображена справка по тегу, на содержимом которого находится текстовый курсор.

Пользуясь панелью **Reference**, мы можем получить справку о любом атрибуте любого тега HTML. Например, давайте узнаем побольше о теге ``. Выберем в раскрывающемся списке **Tag** название этого тега — и в панели появится его описание.

Справа от раскрывающегося списка **Tag** находится другой раскрывающийся список, в котором выбираются атрибуты. Сейчас там выбран пункт **Description** — это знак того, что в панели **Reference** отображаются сведения о самом теге. Если нам нужно просмотреть сведения о каком-либо атрибуте этого тега, мы просто выберем его в этом списке.

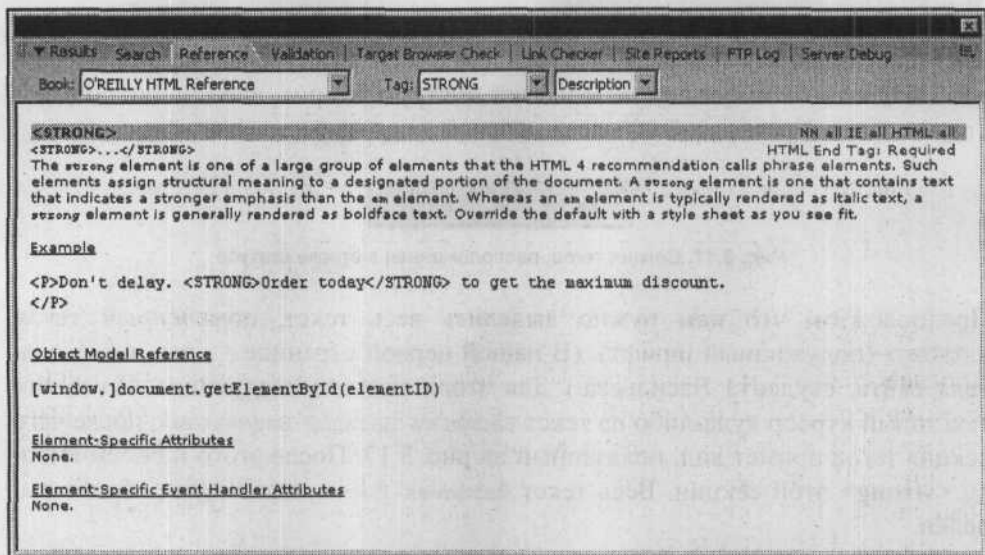


Рис. 3.18. Панель Reference

Создание гиперссылок

На занятии 1, обсуждая отличие Web-страницы от Web-сайта, мы выяснили, что Web-сайт — это набор Web-страниц, связанных друг с другом. Но вот как эти самые страницы связываются, сказано не было. Пора прояснить данный вопрос.

Для этого используются *гиперссылки* — особые связи, ведущие от одной Web-страницы к другой. Они имеют вид фрагментов текста, выделенных особым образом (обычно синим цветом и подчеркиванием — это даже стало своего рода стандартом). Если по такой гиперссылке щелкнуть мышью, Web-обозреватель загрузит страницу, интернет-адрес которой указан в параметрах гиперссылки.

Давайте создадим на нашей первой Web-странице первую гиперссылку, указывающую на сайт фирмы Macromedia (<http://www.macromedia.com>). Для этого мы используем слова Macromedia Dreamweaver 8, которые ранее чуть было не выкрасили в темно-зеленый цвет. Выделим эти слова и посмотрим на редактор свойств. Большое поле ввода **Link** для задания интернет-адреса для гиперссылки заметно сразу (рис. 3.19).

Все, что нам нужно сделать, это ввести в него интернет-адрес сайта Macromedia и нажать клавишу <Enter>. Созданная нами гиперссылка показана на рис. 3.20. Теперь при просмотре в Web-обозревателе пользователь мо-

жет просто щелкнуть мышью по словам Macromedia Dreamweaver 8, чтобы перейти на сайт этой фирмы.



Рис. 3.19. Поле ввода интернет-адреса Link, расположенное в редакторе свойств

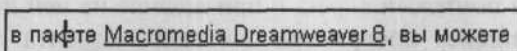


Рис. 3.20. Так в окне документа Dreamweaver выглядит гиперссылка

Впоследствии мы можем исправить нашу гиперссылку, в частности изменить ее интернет-адрес. Для этого достаточно поставить текстовый курсор на текст гиперссылки, изменить адрес в уже знакомом нам поле ввода **Link** (см. рис. 3.19) и нажать клавишу <Enter>. Для удаления гиперссылки и превращения ее содержимого в обычный текст мы должны будем воспользоваться пунктом **Remove Link** меню **Modify** или контекстного меню либо комбинацией клавиш <Ctrl>+<Shift>+<L>.

Отлично! Мы имеем гиперссылку для перехода на сайт Macromedia. Чего еще желать?

Много чего. Например, мы можем сделать так, что при переходе на сайт Macromedia пользователь будет иметь возможность просматривать также и сайт Васильева. Как это сделать? Очень просто: мы откроем сайт Macromedia в другом, новом окне Web-обозревателя.

Поставим текстовый курсор где-нибудь в тексте гиперссылки и снова посмотрим на редактор свойств. На рис. 3.21 показан раскрывающийся список **Target**, служащий для задания цели гиперссылки, указывающей, в каком окне будет открыта новая Web-страница, — в том же или в новом.

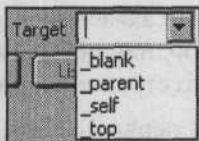


Рис. 3.21. Раскрывающийся список задания цели гиперссылки Target, расположенный в редакторе свойств

Этот список содержит четыре пункта, но сейчас полезными для нас будут только два: **_blank** и **_self**. Первый из них позволит вывести страницу, на которую указывает гиперссылка, в новом окне Web-обозревателя, а второй — в том же окне. (Вместо того чтобы выбирать пункт списка **_self**, мы можем просто оставить его содержимое пустым.) Нам будет нужно выбрать пункт **_blank**

Для задания цели гиперссылки мы также можем воспользоваться подменю **Link Target** меню **Modify** или подменю **Target Frame** контекстного меню. Для этого достаточно просто выбрать соответствующий пункт-переключатель. Пункт **Default Target** этих подменю задает цель по умолчанию (т. е. **_self** — то же самое окно).

Вторая гиперссылка, которую мы сейчас создадим, будет указывать не на Web-страницу, а на почтовый адрес, и при щелчке на ней пользователь попадет в окно открывшейся программы почтового клиента, установленного в системе по умолчанию. Такие гиперссылки еще называют *почтовыми*.

Пусть гипотетический студент Васильев будет иметь такой почтовый адрес:
vasilyev@mail.ru

Этот почтовый адрес мы запишем в поле ввода **Link** вот в таком виде:


mailto:vasilyev@mail.ru

причем между двоеточием после **mailto** и собственно адресом не должно быть пробелов.

В последнем абзаце, где говорится об авторских правах, есть имя автора. Из него-то мы и сделаем почтовую гиперссылку. Выделим имя автора и введем почтовый адрес в приведенном ранее формате в поле ввода **Link**. После нажатия клавиши <Enter> мы получим почтовую гиперссылку, которая по внешнему виду ничем не отличается от обычной.

Третья разновидность гиперссылок, которую мы изучим здесь, позволяет указать на часть какой-либо Web-страницы. Точнее, не на произвольную часть, а на особую метку, которую мы поставим в нужном месте страницы, — *якорь* (по-английски — *anchor*). Сейчас мы выясним, как создавать якоря и указывающие на них гиперссылки.

Для примера давайте создадим новую Web-страницу с большим текстом, напишем в ее начале небольшое оглавление и сохраним ее в файле 3.2.htm. Текст может быть любой — автор, например, использовал фрагмент текста из этой книги. Получившаяся у него страничка показана на рис. 3.22.

Теперь вставим на эту страницу первый якорь, для чего поместим текстовый курсор в начало первого абзаца, после заголовка, и переключимся на вкладку **Common** инструментария объектов. Нажмем кнопку вставки якоря **Named Anchor** . Также можно воспользоваться пунктом **Named Anchor** в меню **Insert** или просто нажать комбинацию клавиш <Ctrl>+<Alt>+<A>.

В результате этих действий на экране появится диалоговое окно **Named Anchor**, показанное на рис. 3.23. В поле ввода **Anchor name** вводим имя создаваемого якоря; в нашем случае это будет *par1*. Имена якорей должны состоять из латинских букв и цифр, причем первым символом должна быть

буква. После этого щелкнем кнопку **ОК**, чтобы вставить якорь, или **Cancel** для отказа от этого.

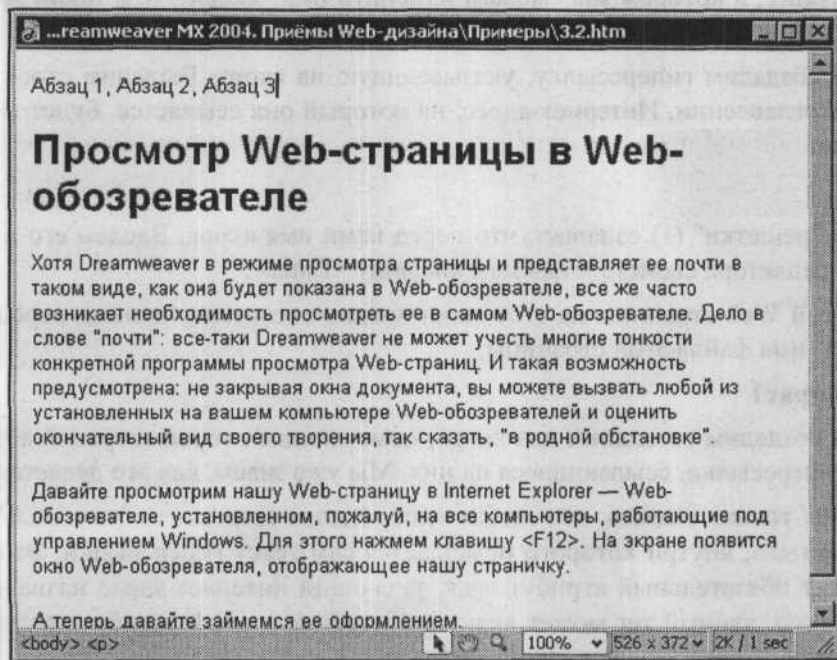


Рис. 3.22. Учебная Web-страница, где будут использованы якоря

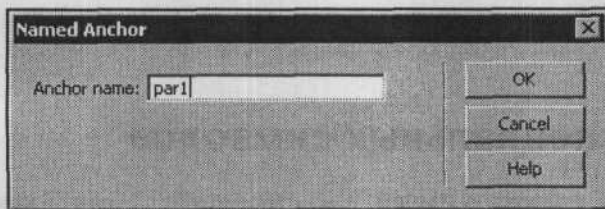



Рис. 3.23. Диалоговое окно Named Anchor

Web-обозреватели никак не отображают якорь, но Dreamweaver, чтобы помочь нам, выведет значок  в позиции, где тот был вставлен. Иначе мы его ни за что не найдем.

Внимание!

Если вы почему-то не видите значка якоря, сделайте следующее. Выберите в меню **Edit** пункт **Preferences**, выберите в списке **Category** диалогового окна **Preferences** пункт **Invisible Elements** и включите флажок **Named anchors**. После этого нажмите кнопку **ОК**.

Мы можем щелкнуть мышью по значку якоря, чтобы выделить его. После этого редактор свойств изменит свой вид — на нем появится большое поле ввода **Name**, в котором мы сможем изменить имя "якоря". Мы также можем удалить выделенный якорь, просто нажав клавишу .

Теперь создадим гиперссылку, указывающую на якорь. Выделим строку абзац 1 в оглавлении. Интернет-адрес, на который она ссылается, будет выглядеть так:

```
#par1
```

Значок "решетки" (#) означает, что перед нами имя якоря. Введем его в поле адреса редактора свойств и нажмем клавишу <Enter>.

Из другой Web-страницы на этот якорь можно сослаться, написав перед "решеткой" имя файла этой страницы:

3.3.htm#par1


Теперь создадим остальные два якоря, помечающие второй и третий абзацы, и две гиперссылки, ссылающиеся на них. Мы уже знаем, как это делается.

Осталось только сказать, что любая гиперссылка создается с помощью парного тега <A>, внутри которого помещается сам текст гиперссылки. Этот тег содержит обязательный атрибут **HREF**, задающий интернет-адрес назначения. Кроме того, данный тег может включать необязательный атрибут **TARGET**, задающий цель гиперссылки. Например:

```
<A HREF="http://www.macromedia.com" TARGET="_blank">Macromedia  
Dreamweaver 8</A>  
<A HREF="mailto:vasilyev@mail.ru">Васильев В. В.</A>
```

Вставка специальных символов

Что бы нам еще сделать с нашей Web-страницей? Давайте как-нибудь отделим первый и второй абзацы от остального текста. Можно сделать это, вставив пустую строку, но мы поступим иначе.

Поставим текстовый курсор в начало третьего абзаца и нажмем кнопку **Horizontal Rule** () , расположенную на вкладке **HTML** инструментария объектов. Как показано на рис. 3.24, между вторым и третьим абзацами появится разделяющая их *горизонтальная линия*.

Для создания горизонтальной линии также можно использовать пункт **Horizontal Rule**, находящийся в подменю **HTML** меню **Insert**.

Горизонтальная линия вставляется в текст с помощью одинарного тега <HR>.

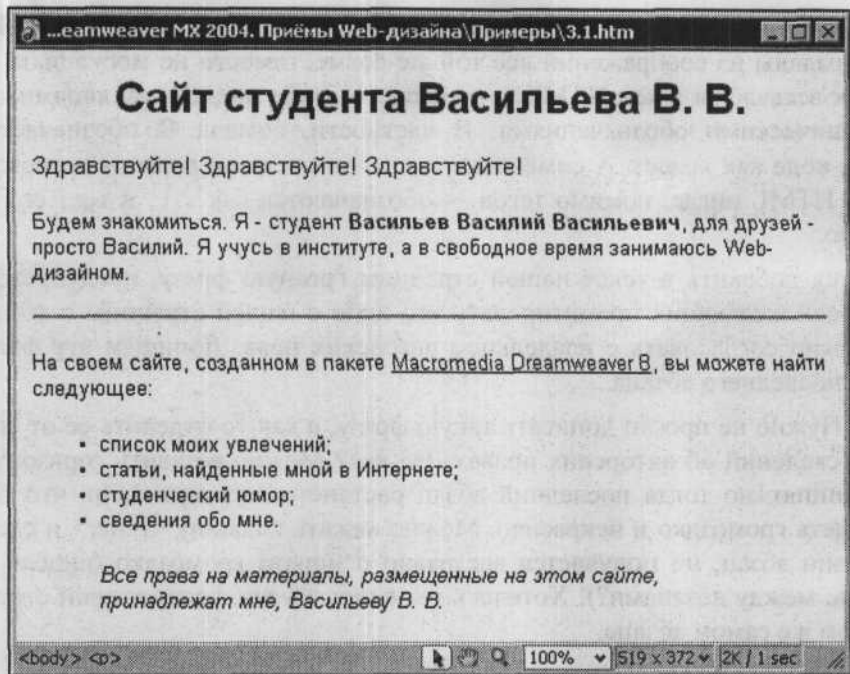


Рис. 3.24. Горизонтальная линия

Теперь посмотрим на последний абзац, содержащий сведения об авторских правах. У нас там тянется длинный невразумительный текст. Давайте заменим его коротким и приметным значком ©, ставшим общепринятым стандартом для обозначения авторских прав.

Сначала выделим слова Все права на материалы, размещенные на этом сайте, принадлежат мне, (не забыв и запятую), которые мы заменим значком ©, и удалим. Далее поставим текстовый курсор в то место текста, где мы хотим поместить этот знак. И выберем пункт **Copyright** в подменю **Special Characters**, находящегося, в свою очередь, в подменю **HTML** меню **Insert**. Dreamweaver выведет небольшое окно-предупреждение, говорящее о том, что данный символ может не отобразиться корректно в случае использования некоторых кодировок. Закроем его, нажав кнопку **ОК**. Если вы не хотите, чтобы оно выводилось на экран в дальнейшем, перед закрытием включите флажок **Don't show me again**. На рис. 3.25 показан только что вставленный нами символ ©.

© Васильев В. В.

Рис. 3.25. Символ © в тексте страницы

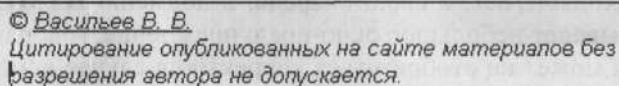
Здесь мы столкнулись с так называемыми *специальными символами HTML*. Эти символы из соображений все той же совместимости не могут быть просто так вставлены в код HTML, а заменяются либо числовыми кодами, либо мнемоническими обозначениями. В частности, символ © обозначается в HTML-коде как ©. А символы < и > — они ведь не должны встречаться в коде HTML нигде, помимо тегов — обозначаются как < и > соответственно.

Остается добавить в текст нашей страницы грозную фразу, предупреждающую всех желающих процитировать что-либо с нашей страницы о том, что это нужно согласовать с владельцем авторских прав. Допишем эту фразу в конце последнего абзаца...

Стоп! Нужно не просто дописать такую фразу, а как-то отделить ее от собственно сведений об авторских правах. Но как? Можно вставить горизонтальную линию, но тогда последний абзац растянется по вертикали, что будет выглядеть громоздко и некрасиво. Можно нажать клавишу <Enter> и сделать еще один абзац, но получается все равно слишком громоздко (видели расстояние между абзацами?). Хотелось бы дописать эту фразу второй строкой, но в том же самом абзаце.

И на этот случай HTML (и, конечно, Dreamweaver) предлагает решение. Речь идет о *разрыве строк*, заставляющем все следующие за ним символы переноситься на новую строку. При этом он не создает нового абзаца, что нам и надо.

Поставим курсор в конец строки со сведениями об авторском праве и нажмем комбинацию клавиш <Shift>+<Enter>. (Существует также возможность вставить разрыв строк мышью, но для этого вам придется снова отправиться в дальнюю дорогу: меню **Insert**, подменю **HTML**, подменю **Special Characters** и, наконец, пункт **Line Break**.) Впишем нужную строку. У нас должно получиться то, что изображено на рис. 3.26.



© Васильев В. В.
Цитирование опубликованных на сайте материалов без
разрешения автора не допускается.

Рис. 3.26. Использование разрыва строк

Разрыв строк вставляется с помощью одинарного тега
.

И последний штрих. В последнем абзаце нашего текста есть имя автора этого сайта. По правилам полиграфии, инициалы нельзя разносить по разным строкам. Web-обозреватель же в полиграфии не силен и при недостатке свободного места все же разнесет их, в результате чего текст будет выглядеть очень

неаккуратно. Чтобы этого не было, между буквами в. инициалов нужно вставить *неразрывный пробел*. Что бы ни случилось, Web-обозреватель никогда не будет переносить строку по этому пробелу.

Прежде всего, уберем пробел между буквами в. и поставим там текстовый курсор. Нажмем комбинацию клавиш <Ctrl>+<Shift>+<Пробел>. (Также можно использовать пункт **Non-Breaking Space** подменю **Special Characters**, подменю **HTML** меню **Insert**.) Неразрывный пробел обозначается в языке HTML кодом ` `.

Другие пункты подменю **Special Characters**, подменю **HTML** меню **Insert** позволят нам вставить в текст Web-страницы другие специальные символы. Все они имеют "говорящие" названия: **Left Quote** — левая кавычка, **Right Quote** — правая кавычка, **Pound** — знак фунта стерлингов и т. д.

А с помощью пункта **Other** подменю **Special Characters**, подменю **HTML** меню **Insert** можно вставить в текст любой доступный в HTML специальный символ. При выборе этого пункта на экране появится диалоговое окно **Insert Other Character**, показанное на рис. 3.27. Нам нужно будет щелкнуть по кнопке, соответствующей нужному символу, и нажать **OK** для его вставки или **Cancel** для отказа от этого.

Вот, собственно, и все о наборе и форматировании текста в Dreamweaver.

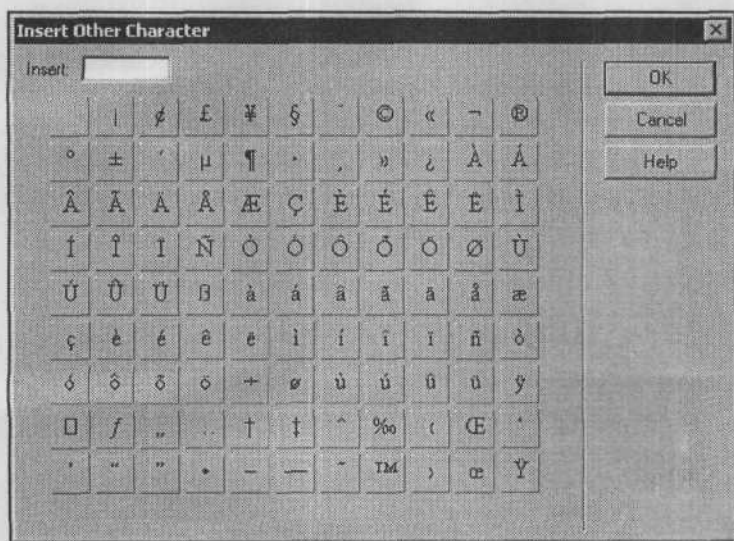
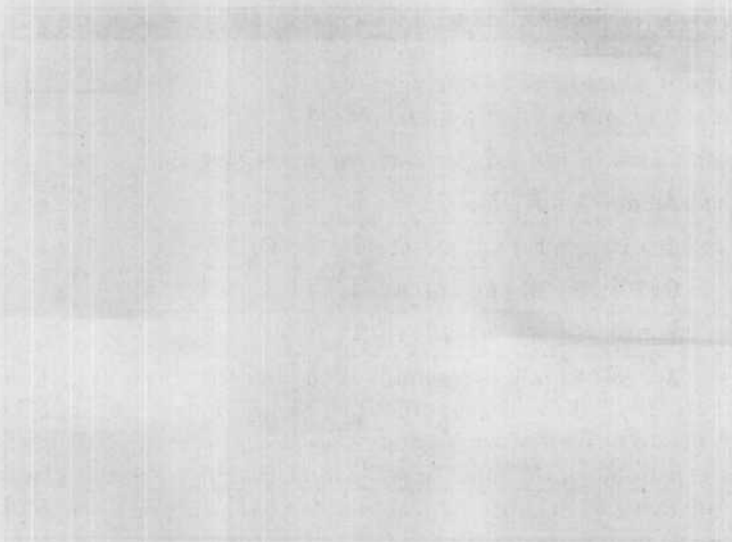


Рис. 3.27. Диалоговое окно Insert Other Character

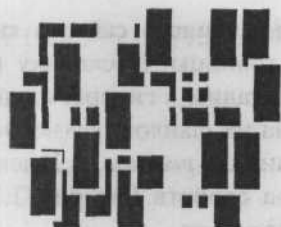
На следующем занятии...

Итак, мы создали первую нашу Web-страницу, которая потом станет главной страницей нашего сайта. Мы научились работать с текстом и гиперссылками, используя для этого стандартные инструменты Dreamweaver. Попутно мы начали изучение языка HTML, ведь без его знания трудно стать хорошим Web-дизайнером.

Теперь, пользуясь полученными знаниями, вы можете сами создать остальные страницы Web-сайта студента Васильева. Пусть это будет вашим домашним заданием. И на следующем занятии мы займемся тем, что опубликуем готовый сайт в Интернете.



ЗАНЯТИЕ 4



Работа с Web-сайтами

Занятие 3 было очень насыщенным. Мы изучили важнейшие возможности Dreamweaver по работе с текстом Web-страниц и гиперссылками, а по ходу дела познакомились с языком HTML. Теперь мы умеем создавать простейшие Web-страницы, содержащие текст, что уже немало.

А на этом занятии мы научимся управляться в среде Dreamweaver не с отдельными страницами, а уже с целыми сайтами. А именно — с нашим сайтом... т. е. с сайтом студента Васильева.

Как вы помните, вашим домашним заданием было создать остальные четыре странички будущего сайта: список увлечений выдуманного студента, список ссылок на статьи в Интернете, странички со студенческим юмором и сведениями о самом Васильеве. Содержимое этих страниц особого значения не имеет; можно вообще создать пустые страницы. Главное сейчас не это.

Назовем файлы, в которых сохранены эти страницы, так:

- ◆ список увлечений — `passions.htm`;
- ◆ список ссылок на статьи в Интернете — `articles.htm`;
- ◆ студенческий юмор — `humour.htm`;
- ◆ сведения об авторе — `about.htm`.

Кроме того, сделаем копию страницы `3.1.htm` и назовем ее `default.htm`. Это будет наша страница по умолчанию, и она же — *главная страница* сайта, т. е. страница, с которой начнется путешествие посетителя по сайту. (Как правило, главная страница сайта является страницей по умолчанию.) Поместим все пять перечисленных выше страниц в особую папку с именем, скажем, `Site1` — так нам будет проще управляться с сайтом.

Мы уже умеем создавать гиперссылки. Давайте откроем в Dreamweaver главную страницу `default.htm` (копию `3.1.htm`) и превратим пункты списка разде-

лов нашего сайта в гиперссылки, указывающие на соответствующие Web-страницы. Поскольку все страницы нашего сайта находятся в одной папке, зададим в гиперссылках относительные интернет-адреса, фактически — имена их файлов: `passions.htm`, `articles.htm` и т. д. (Об интернет-адресах мы узнали на *занятии 1*.) Именно эти адреса мы и введем в поле ввода **Link** редактора свойств (см. рис. 3.19). После этого сохраним страницу `default.htm` и закроем ее.

Все, подготовительные работы закончены. Настала пора простереть руку над только что созданными Web-страничками и торжественно изречь: "Нарекаю тебя сайтом!"

Подготовка к публикации сайта

Шутка, конечно, но доля правды в этом есть. Дело в том, что созданные нами ранее Web-страницы Dreamweaver не рассматривает как Web-сайт. Для него это всего лишь набор разрозненных страничек, не объединенных друг с другом ничем, кроме гиперссылок. Фактически Dreamweaver даже не в курсе, что мы сделали сайт!

Так что первое, что мы должны сделать, — сообщить Dreamweaver о создании сайта, т. е. выполнить его *регистрацию*. Поскольку мы сохраняли все Web-страницы в одной папке — `Site1` — этот процесс не займет много времени.

Регистрация сайта в Dreamweaver

Регистрация сайта в Dreamweaver начинается выбором пункта **Manage Sites** в меню **Site**. На экране появится диалоговое окно **Manage Sites**, показанное на рис. 4.1. Большую часть этого окна занимает список уже созданных сайтов (на рис. 4.1 показаны сайты, созданные автором книги для работы). Чтобы добавить сюда и наш сайт, нажмем кнопку **New** и в появившемся на экране небольшом меню выберем пункт **Site**.

После этого на экране появится диалоговое окно **Site Definition**, состоящее из двух вкладок. Если оно открыто на вкладке **Basic**, переключимся на вкладку **Advanced** — она предоставляет больше возможностей по настройке нашего сайта. И проверим, выбран ли в списке **Category** пункт **Local Info**. Окно **Site Definition** должно иметь такой вид — рис. 4.2.

Элементы управления категории **Local Info** позволяют задать сведения о тех файлах нашего сайта, с которыми в данный момент идет работа. Назовем их *локальной копией* сайта. Напротив, файлы, которые уже опубликованы на Web-сервере, пусть называются *удаленной копией* сайта.

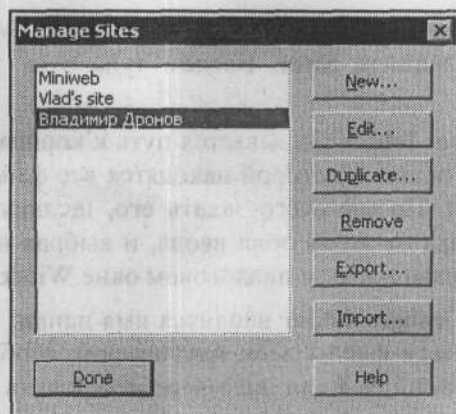


Рис. 4.1. Диалоговое окно Manage Sites

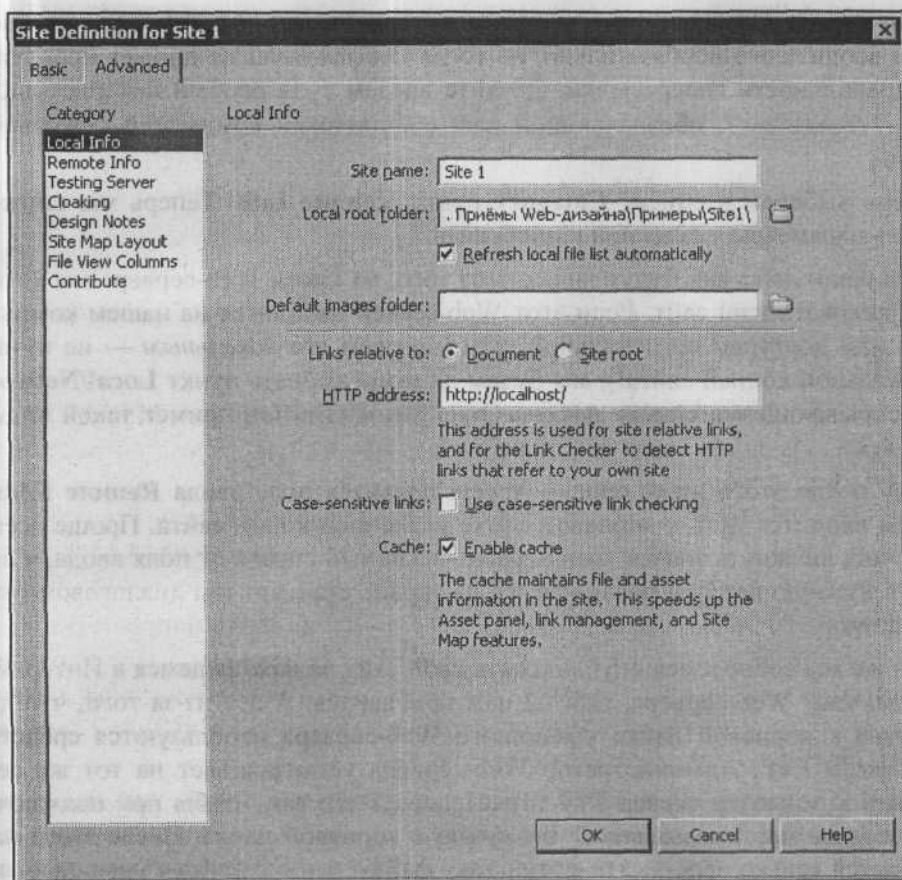


Рис. 4.2. Диалоговое окно Site Definition (категория Local Info)

В поле ввода **Site name** вводится имя сайта, которое будет отображаться в списке сайтов окна **Manage Sites**. Введем туда *site 1*, как показано на рис. 4.2.

В поле ввода **Local root folder** указывается путь к корневой папке локальной копии сайта, т. е. как папке, в которой находятся все файлы и папки этой самой локальной копии. Проще всего задать его, щелкнув по значку папки, расположенному справа от этого поля ввода, и выбрав нужную папку в появившемся на экране стандартном диалоговом окне **Windows**.

В поле ввода **Default images folder** вводится имя папки, в которую по умолчанию будут помещаться файлы всех графических изображений, помещенных нами на Web-страницах. Если же это поле оставить пустым, эти файлы окажутся в корневой папке сайта. Оставим же его пустым, тем более что графикой на Web-страницах мы будем заниматься только на *занятии 5*.

В поле ввода **HTTP address** вводится интернет-адрес нашего сайта. В принципе вводить его необязательно, но тогда **Dreamweaver** не сможет проверить на правильность гиперссылки. Давайте введем туда особый интернет-адрес `http://localhost/`, обозначающий наш собственный компьютер (*локальный хост*).

Теперь выберем в списке **Category** пункт **Remote Info**. Теперь мы сможем задать параметры удаленной копии сайта.

Здесь наши действия будут зависеть от того, на каком Web-сервере мы хотим опубликовать наш сайт. Если этот Web-сервер находится на нашем компьютере или доступен по локальной сети (назовем его *локальным* — не путать с локальной копией сайта!), мы будем должны выбрать пункт **Local/Network** в раскрывающемся списке **Access**. Окно **Site Definition** примет такой вид — рис. 4.3.

Сразу после этого ниже списка **Access** появится поле ввода **Remote folder**. В нем вводится путь к корневой папке удаленной копии сайта. Проще всего, конечно, щелкнуть значок папки, расположенный справа от поля ввода, и выбрать нужную папку в появившемся на экране стандартном диалоговом окне **Windows**.

Если же мы собираемся опубликовать свой сайт на находящемся в Интернете (*удаленном*) Web-сервере, работы нам прибавится. А все из-за того, что для доступа к корневой папке удаленного Web-сервера используются средства протокола FTP. Администратор Web-сервера устанавливает на тот же серверный компьютер сервер FTP и настраивает его так, чтобы при подключении к нему мы "оказывались" бы прямо в корневой папке. После этого нам останется только переписать в эту папку файлы нашего сайта с помощью любой программы FTP-клиента.

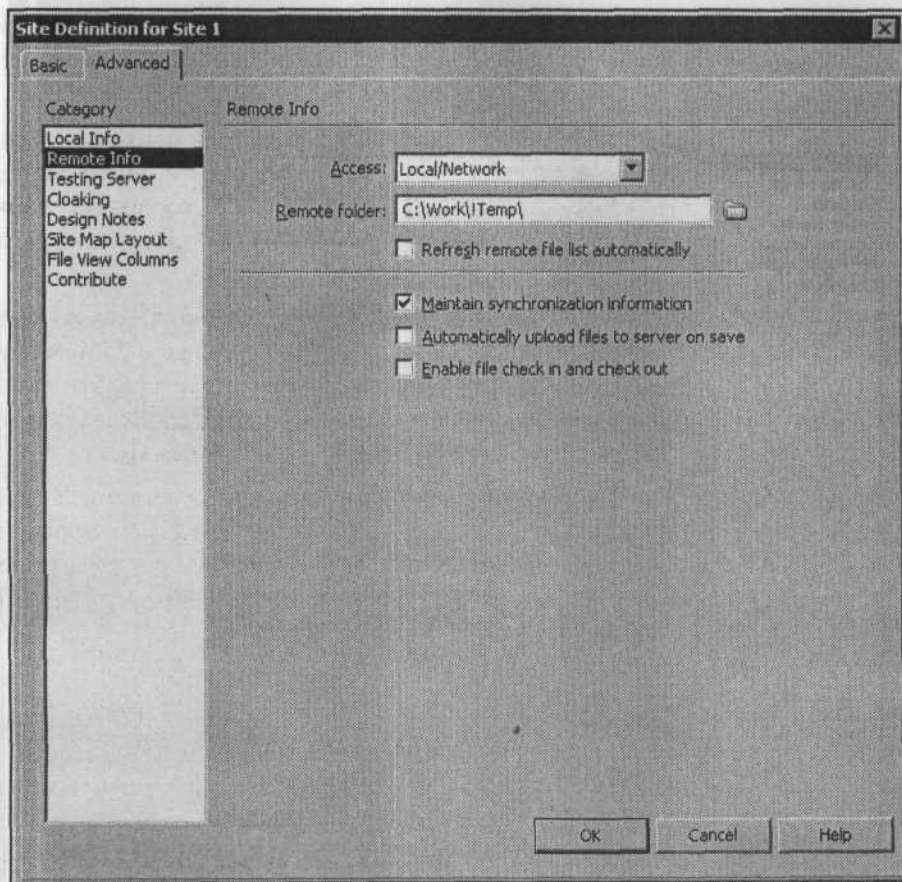


Рис. 4.3. Диалоговое окно Site Definition (категория Remote Info, выбран режим отправки файлов по локальной сети)

Но нам это не придется делать самим — Dreamweaver сам поддерживает протокол FTP и умеет управлять файлами на FTP-сервере. Нам нужно только выбрать пункт **FTP** в раскрывающемся списке **Access** и ввести следующие параметры в соответствующие поля ввода окна **Site Definition** (см. рис. 4.4):

- ◆ **FTP host** — интернет-адрес FTP-сервера вида **ftp.somesite.ru** (без указания протокола — Dreamweaver подставит его сам);
- ◆ **Host directory** — путь к корневой папке нашего сайта. Поскольку FTP-сервер, как правило, сразу подключает нас к этой папке, оставим это поле пустым или введем в него значок \ ("обратный слэш");
- ◆ **Login** — имя пользователя, под которым мы подключаемся;
- ◆ **Password** — пароль (при его вводе в поле отображаются, в зависимости от установленной на компьютере версии Windows, звездочки или точки)

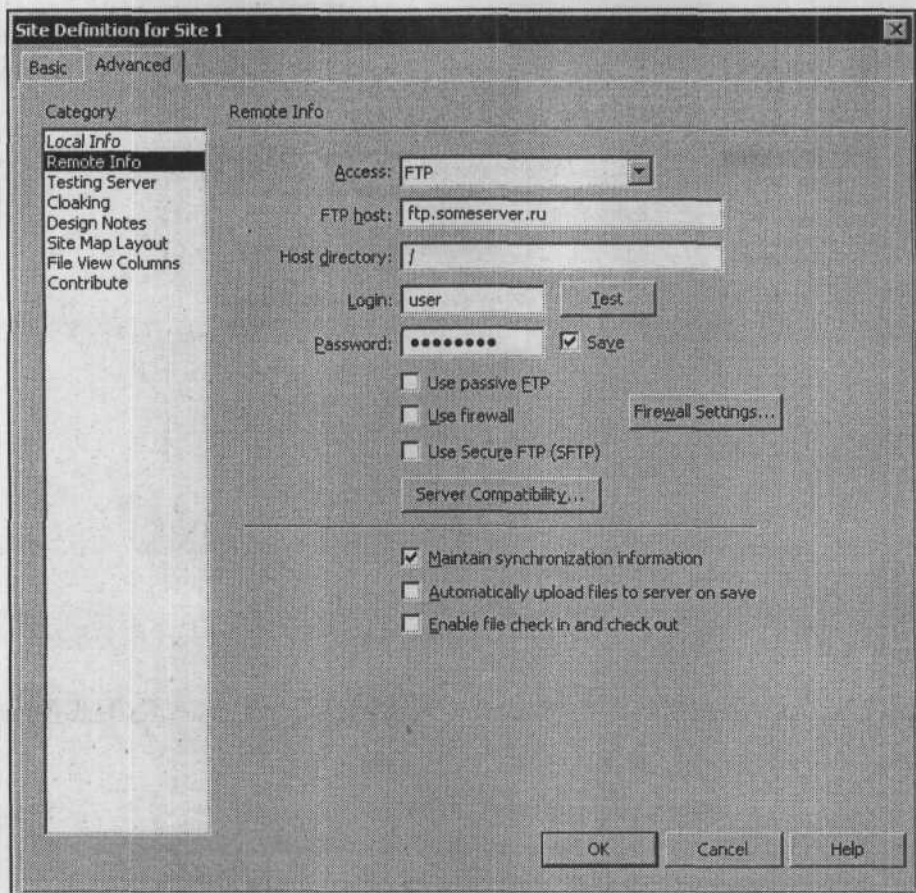


Рис. 4.4. Диалоговое окно Site Definition (категория Remote Info, выбран режим отправки файлов по протоколу FTP)

По умолчанию Dreamweaver запоминает введенный нами пароль и автоматически вводит его при подключении к FTP-серверу. Если вы из соображений безопасности не хотите хранить этот пароль, а собираетесь вводить его при каждом подключении, отключите флажок **Save**.

Нажав кнопку **Test**, мы можем проверить правильность введенных нами параметров. Dreamweaver попытается подключиться к FTP-серверу и, в зависимости от успеха или неуспеха, выведет соответствующее окно-предупреждение.

Внимание!

Если попытка подключения не удалась, но данные введены заведомо правильно, можно попробовать нажать кнопку **Server Compatibility...** чтобы вывести на

экран одноименное диалоговое окно (рис. 4.5). В этом окне можно попробовать отключить флажок **Use FTP performance optimization** и включить флажок **Use alternative FTP move method**, не забыв после этого нажать кнопку **OK**.

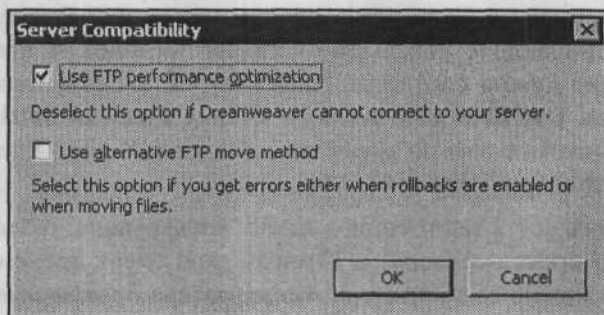
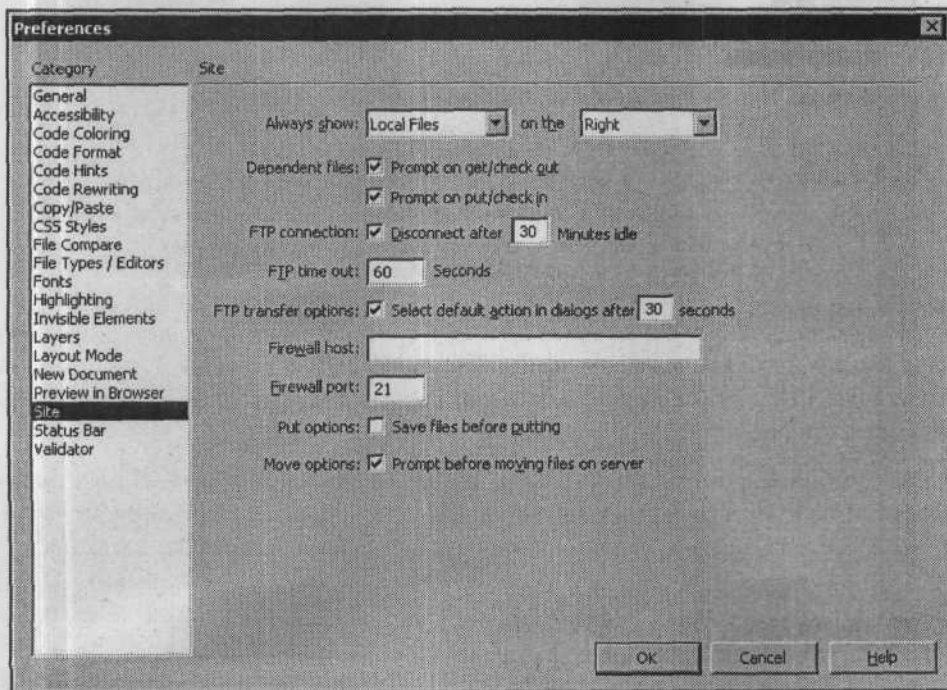


Рис. 4.5. Диалоговое окно **Server Compatibility**

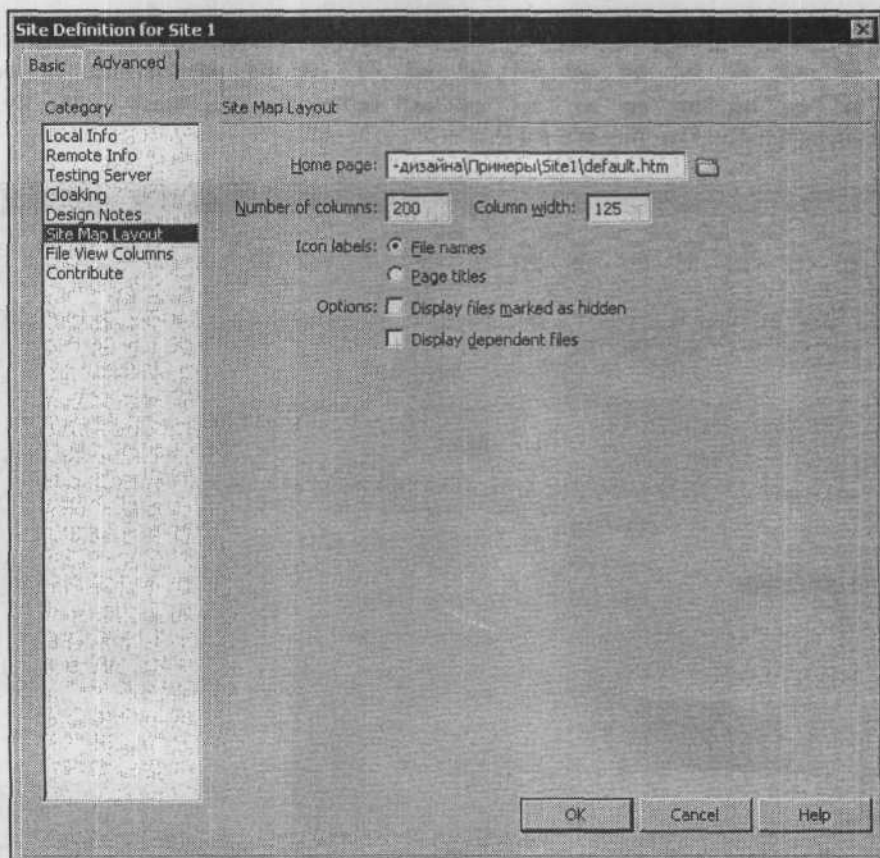
Если FTP-сервер, с помощью которого мы собираемся публиковать сайт, поддерживает только *пассивный протокол FTP* (особая разновидность протокола FTP, используемая из соображений безопасности), нам будет нужно включить флажок **Use passive FTP**.



Далее включим флажок **Use Secure FTP (SFTP)**, если для доступа на FTP-сервер используется защищенная версия протокола FTP — SFTP (Secure FTP, защищенный FTP).

Если же мы выходим в Сеть через *прокси-сервер* или *брандмауэр* (особые программы, защищающие локальную сеть организации от атак из Интернета), то нам будет нужно соответственно настроить Dreamweaver. Для этого включим флажок **Use firewall** и нажмем кнопку **Firewall Settings**. На экране появится уже знакомое нам по *занятию 2* диалоговое окно **Preferences**, переключенное на категорию **Site** (рис. 4.6).

Здесь нас интересуют в настоящий момент только поля ввода **Firewall host** (интернет-адрес прокси-сервера) и **Firewall port** (порт прокси-сервера). Оба этих параметра можно выяснить у администратора прокси-сервера. Задав их, нажмем кнопку **ОК**, чтобы закрыть диалоговое окно **Preferences**.



На заметку

Если выход в Сеть осуществляется через прокси-сервер или брандмауэр, то в диалоговом окне **Site Definition** также не помешает включить флажок **Use passive FTP**.

Остается только выбрать в списке **Category** диалогового окна **Site Definition** пункт **Site Map Layout** (рис. 4.7), кое-что проверить и задать некоторые дополнительные настройки.

Прежде всего, проверим, подставил ли Dreamweaver в поле ввода **Home page** правильное имя файла Web-страницы по умолчанию. (Dreamweaver пытается сам угадать имя файла страницы по умолчанию и, к сожалению, иногда ошибается.) Если нужно, исправим его вручную или щелкнем по значку папки справа от поля ввода и выберем нужный файл в появившемся на экране диалоговом окне **Choose Home Page**, очень похожем на стандартное диалоговое окно открытия файла Windows.

Закончив ввод параметров нашего сайта, нажмем кнопку **OK** диалогового окна **Site Definition**. После этого Dreamweaver выполнит сканирование списка файлов и создаст в памяти их список. Далее нажмем кнопку **Done** диалогового окна **Manage Sites**. Регистрация сайта закончена.

Работа с файлами сайта. Панель **Files**

А теперь мы познакомимся с одним очень полезным и удобным инструментом Dreamweaver для работы с файлами, составляющими Web-сайт. Это панель **Files**. Чтобы вывести ее на экран, нужно включить пункт-выключатель **Files** в меню **Window** или нажать клавишу <F8>. Сама эта панель показана на рис. 4.8.

Большую часть панели **Files** занимает иерархический список файлов и папок, составляющих сайт. Также панель **Files** имеет свой собственный небольшой инструментарий с элементами управления, предоставляющими быстрый доступ к наиболее часто употребляемым командам, и строку статуса.

Список файлов и папок панели **Files** похож на аналогичный список Проводника: его можно сворачивать и разворачивать, а также выделять нужный файл или папку щелчком мыши. Если нужно выделить несколько файлов, достаточно щелкнуть по ним, удерживая нажатой клавишу <Ctrl>. Чтобы выделить все файлы списка, можно нажать комбинацию клавиш <Ctrl>+<A> или выбрать в подменю **Edit** дополнительного меню панели **Files** пункт **Select All**.

Раскрывающийся список, расположенный слева в инструментарии панели **Files**, позволяет быстро выбрать любой из зарегистрированных в Dreamweaver сайтов для отображения в панели. Это значит, что мы можем зарегистри-

стрировать в Dreamweaver сколько угодно сайтов и с легкостью переходить от одного к другому. (Но это, конечно, в будущем...)

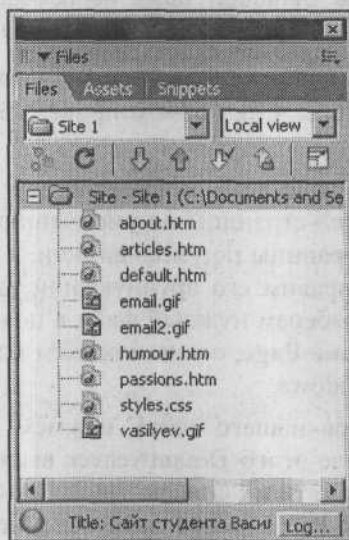


Рис. 4.8. Панель Files

А тот раскрывающийся список, что находится в инструментарии панели правее, позволяет выбрать копию сайта — локальную или удаленную, — содержимое которой нужно просмотреть в панели **Files**. Пункт **Local View**, выбранный по умолчанию, задает отображение списка файлов локальной копии, а пункт **Remote View** — файлов удаленной копии.

Ранее говорилось, что список файлов панели **Files** похож на аналогичный список Проводника Windows за тем исключением, что в нем отображаются и папки, и файлы. И так же, как Проводник, панель **Files** предоставляет возможности по управлению этими файлами и папками.

Так, чтобы удалить выделенный в списке файл или папку, нужно выбрать пункт **Delete** подменю **File** дополнительного меню или нажать клавишу . После этого Dreamweaver спросит, действительно ли мы хотим удалить этот файл (папку); после этого нужно нажать **Yes** или **No**, соответственно, для удаления или отказа от него.

Внимание!

Dreamweaver выполняет удаление файлов и папок с файлами в корзину Windows. Однако пустые папки удаляются бесследно.

Чтобы переименовать выделенный файл или папку, нужно щелкнуть по нему

Rename подменю **File** дополнительного меню. После этого на месте имени данного файла появится небольшое поле ввода, где будет подставлено его старое имя. Редактируем его или вводим новое, после чего нажимаем клавишу <Enter> для сохранения нового имени или клавишу <Esc> для отмены.

Если на других Web-страницах присутствуют гиперссылки, указывающие на переименованный файл (или файл, находящийся в переименованной папке), Dreamweaver предложит нам исправить их. На экране появится диалоговое окно **Update Files**, показанное на рис. 4.9. В списке **Update links in the following files?** находятся Web-страницы, на которых были найдены гиперссылки, указывающие на переименованный файл. Нажмем кнопку **Update**, чтобы обновить их, или **Don't Update**, чтобы отказаться от обновления; в последнем случае нам придется исправить интернет-адреса в гиперссылках вручную.

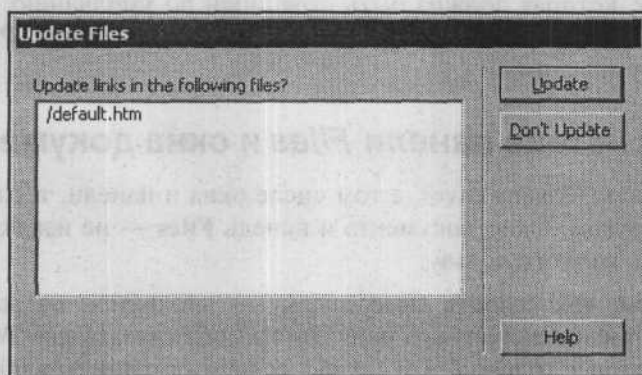


Рис. 4.9. Диалоговое окно Update Files

Если дважды щелкнуть по какому-либо файлу или выделить его и нажать клавишу <Enter>, Dreamweaver откроет его. Такой номер проходит с Web-страницами, а если на компьютере установлена какая-либо программа графического редактора (например, Macromedia Fireworks), — то и с графическими файлами.

С помощью панели **Files** можно также создать новую Web-страницу, выбрав пункт **New File** подменю **File** дополнительного меню или нажав комбинацию клавиш <Ctrl>+<Shift>+<N>. После этого в самом низу списка появится новый файл с именем `untitled.htm`, которое мы сразу же можем (и должны, собственно) изменить. Изменив его, нажимаем клавишу <Enter>, дважды щелкаем по ней мышью, Dreamweaver открывает новую пустую страницу, и мы начинаем с ней работать.

Новая папка создается аналогично, только нужно выбрать пункт **New Folder**

самом низу списка появится новая папка с именем `untitled`, которое мы должны будем изменить и нажать клавишу `<Enter>`. Потом мы можем перетащить мышью нужные файлы (и другие папки) в эту папку, а Dreamweaver предложит исправить гиперссылки, указывающие на них, выведя уже знакомое нам диалоговое окно **Update Files** (см. рис. 4.9). Если же во время перетаскивания файлов удерживать нажатой клавишу `<Ctrl>`, Dreamweaver вместо переноса скопирует эти файлы и папки.

Кроме того, любую Web-страницу мы можем просмотреть в Web-обозревателе. Для этого нужно выделить ее в списке файлов и нажать клавишу `<F12>`.


И еще. Предположим, что при регистрации своего сайта в Dreamweaver мы ошиблись в задании файла страницы по умолчанию (или ошибся сам Dreamweaver, а мы его не поправили). Чтобы исправить эту ошибку, выделим Web-страницу, которая должна быть страницей по умолчанию, и выберем в подменю **Site** дополнительного меню пункт **Set as Home Page** (сделать страницей по умолчанию).

Взаимодействие панели *Files* и окна документа

Все инструменты Dreamweaver, в том числе окна и панели, тесно взаимосвязаны друг с другом. Окно документа и панель **Files** — не исключение. И мы можем извлечь из этого пользу.

Мы уже знаем, как создать гиперссылку — проходили на *занятии 3*. Но Dreamweaver предоставляет нам еще два способа связать две Web-страницы воедино, доступные только в том случае, если эти страницы являются частью сайта, и мы этот сайт зарегистрировали. Сейчас мы их рассмотрим.

Нам понадобится какая-либо Web-страница, например, список увлечений `passions.htm`. По правилам Web-дизайна, на каждой странице сайта обязательно должна быть гиперссылка, указывающая на главную страницу, чтобы посетитель смог беспрепятственно вернуться к началу сайта. Давайте откроем страницу `passions.htm`, в самом ее верху введем текст "на главную страницу" и выделим его. Он и станет новой гиперссылкой. После этого проверим, видны ли на экране редактор свойств и панель **Files**, и, если нет, выведем их.

Так, текст пока еще не созданной гиперссылки у нас выделен... Посмотрим на редактор свойств, а именно на поле ввода **Link**, где указан интернет-адрес гиперссылки. Справа от него находится небольшой значок . Так вот, чтобы создать гиперссылку, достаточно перетащить его прямо на требуемый файл в списке панели **Files**. Так, если мы перетащим этот значок на файл `default.htm` в списке файлов и "бросим" его, выделенный нами текст на главную страницу превратится в гиперссылку. Это и есть первый способ создать гиперссылку

Другой способ также прост. Для создания гиперссылки мы можем перетащить нужный файл из списка панели **Files** прямо в поле ввода **Link** редактора свойств. И все!

Проверка Web-страниц

Ну, теперь-то уж можно публиковать сайт? Нет, еще рано. Терпение — осталось совсем чуть-чуть! Вот проверим корректность HTML-кода и гиперссылок — и опубликуем наше творение в Сети.

Проверка правильности HTML-кода

При создании Web-страниц (особенно если непосредственно править код HTML или переделывать чьи-то старые страницы) немудрено допустить ошибки. Такими ошибками могут быть, например, неправильный интернет-адрес гиперссылки или отсутствие названия страницы (тега <TITLE>). И если второе еще более-менее терпимо, то первое — нет.

Итак, проверим правильность HTML-кода. Закроем все открытые окна документов, чтобы они нам не мешали. Выберем пункт **Reports** в меню **Site** главного окна программы или одноименный пункт в подменю **Site** дополнительного меню панели **Files**. На экране появится диалоговое окно **Reports**, показанное на рис. 4.10.

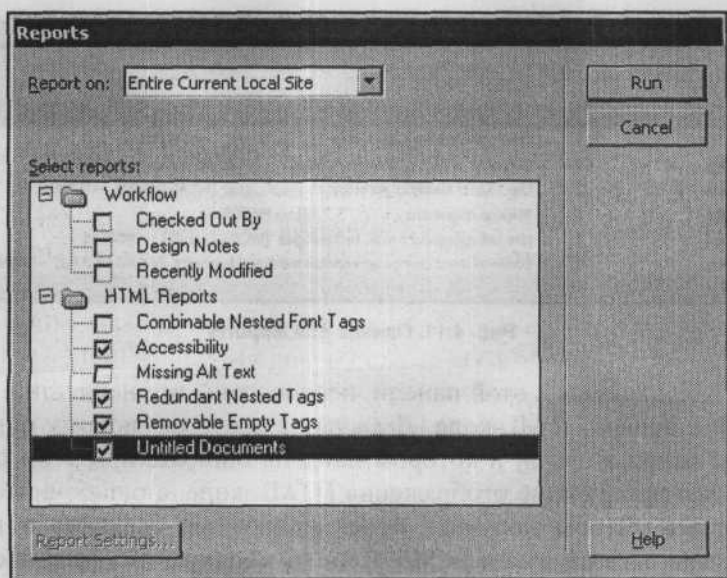


Рис. 4.10. Диалоговое окно Reports

Большую часть этого окна занимает состоящий из двух "ветвей" иерархический список **Select reports**, который позволяет выбрать данные, включаемые в отчет. Для нас представляет интерес вторая "ветвь" — **HTML Reports** — содержащая, в числе прочих, такие пункты:

- ◆ **Accessibility** включает проверку на корректность задания различных элементов страницы (для всех ли страниц задана кодировка и т. п.);
- ◆ **Redundant Nested Tags** включает поиск избыточных вложенных тегов (например, двух вложенных друг в друга тегов ``);
- ◆ **Removable Empty Tags** включает поиск *пустых* тегов (не имеющих содержимого), которые могут быть безболезненно удалены;
- ◆ **Untitled Documents** включает поиск Web-страниц без названия (без тега `<TITLE>`).

Каждый из этих пунктов представляет собой флажок, который мы можем установить или сбросить. Давайте установим их все.

Далее проверим, чтобы в раскрывающемся списке **Report on** был выбран пункт **Entire Current Local Site**, предписывающий Dreamweaver проверить все страницы сайта. (Выбор пункта **Current Document** запускает проверку Web-страницы, открытой в активном окне документа.) После этого нажмем кнопку **Run**. Через некоторое время Dreamweaver выведет панель **Site Reports**, показанную на рис. 4.11.

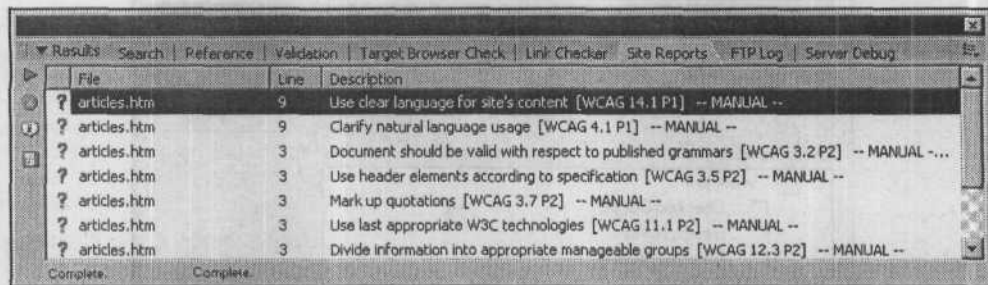


Рис. 4.11. Панель Site Reports

В списке, находящемся в этой панели, перечислены все найденные ошибки и недостатки в нашем HTML-коде. Дважды щелкнув по любому пункту этого списка, мы откроем файл, в котором найдена ошибка; при этом окно документа откроется в режиме отображения HTML-кода, а ошибочный фрагмент будет выделен. Чтобы получить более подробные сведения о найденной ошибке, нужно нажать в панели **Site Reports** кнопку с изображением восклицательного знака, после чего откроется уже знакомая нам панель **Reference** с описанием ошибки.

В принципе то, что нашел в нашем коде Dreamweaver, не очень страшные ошибки, скорее, Dreamweaver просто перестраховывается — а вдруг?.. Но если там встретится что-то действительно серьезное, лучше это исправить.

Проверка гиперссылок

Давайте предположим, что мы переименовали какую-то страницу нашего сайта и не исправили указывающие на нее гиперссылки, а Dreamweaver не сделал это за нас. Такие гиперссылки, указывающие на несуществующий файл, называются "оборванными" или "мертвыми".

Если посетитель нашего сайта щелкнет по оборванной гиперссылке, Web-сервер не сможет найти отсутствующую страницу и отправит Web-обозревателю сообщение об ошибке с кодом 404. Это и есть печально известная *ошибка 404*, с которой мы так часто сталкиваемся, когда путешествуем по Сети...

Конечно, Dreamweaver делает все, чтобы не допустить появления "оборванных" ссылок. Но, как говорится, и на старуху бывает проруха. Поэтому перед публикацией сайта стоит выполнить проверку гиперссылок.

Проверить гиперссылки можно как на отдельной странице, так и на всем сайте. Чтобы проверить гиперссылки на отдельной странице сайта, сначала откроем ее. После этого выберем пункт **Check Links** подменю **Check Page** меню **File** главного окна, одноименный пункт подменю **File** дополнительного меню панели **Files** или нажмем комбинацию клавиш <Shift>+<F8>. Если же мы хотим проверить ссылки на всех страницах сайта, нужно будет использовать пункт **Check Links Sitewide** подменю **Site** дополнительного меню панели **Files** или комбинацию клавиш <Ctrl>+<F8>.

В любом случае после этого на экране появится панель **Link Checker**, показанная на рис. 4.12.

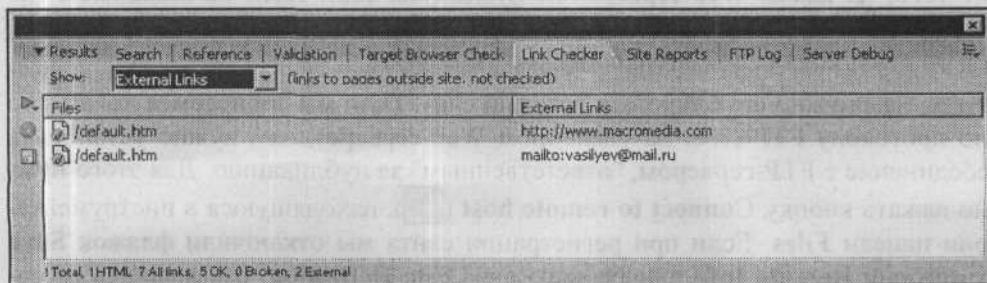


Рис. 4.12. Панель Link Checker

В верхней части этой панели находится раскрывающийся список **Show**, где можно выбрать, что будет показано в списке, занимающем панель.

В этом списке доступны три пункта:

- ◆ **Broken Links** — *"оборванные"* гиперссылки;
- ◆ **External Links** — *"внешние"* гиперссылки, т. е. ссылки, указывающие на другие сайты;
- ◆ **Orphaned Files** — *файлы-"сироты"*, т. е. файлы, на которые не указывает ни одна гиперссылка.


Большую же часть панели **Files** занимает список найденных ошибочных гиперссылок. Каждый пункт этого списка содержит адрес Web-страницы, на которой была найдена гиперссылка, и интернет-адрес этой ссылки. Щелкнув по любой позиции данного списка, мы откроем соответствующую Web-страницу в окне документа, причем Dreamweaver сам выделит ошибочную ссылку.

Давайте посмотрим, все ли у нас в порядке с гиперссылками. Так, оборванных ссылок нет, а внешних ссылка всего две: одна ссылается на сайт фирмы Macromedia, а другая — вообще адрес электронной почты. Файлов-"сирот" также нет. Значит, все в порядке.

Публикация сайта

Итак, настал тот желанный миг, когда наш сайт совсем готов! Мы создали все страницы, связали их друг с другом, исправили ошибки в HTML-коде и даже проверили на корректность интернет-адреса гиперссылок. Фактически мы сделали все, что очень часто забывают сделать даже довольно опытные Web-дизайнеры. Теперь наш сайт можно публиковать на Web-сервере.

Процесс публикации сайта на Web-сервере весьма прост (ну а что в Dreamweaver делается сложно?). Но он несколько различается в зависимости от того, на каком Web-сервере мы публикуем свой сайт: на локальном (по локальной сети) или удаленном (по протоколу FTP). Различается в деталях, но об этих деталях обязательно нужно упомянуть.

Итак, нам нужно опубликовать готовый сайт. Если мы собираемся делать это по протоколу FTP (т. е. на удаленном Web-сервере), нам нужно установить соединение с FTP-сервером, "ответственным" за публикацию. Для этого нужно нажать кнопку **Connect to remote host** () , находящуюся в инструментарии панели **Files**. Если при регистрации сайта мы отключили флажок **Save** категории **Remote Info** диалогового окна **Site Definition**, отказавшись от сохранения введенного нами пароля, Dreamweaver выведет небольшое диалоговое окно с полем ввода, где мы должны будем ввести пароль, и кнопками **OK** (соединение) и **Cancel** (отказ от соединения). Если же флажок **Save** был оставлен включенным, Dreamweaver соединится с FTP-сервером "молча".

Ну а в случае публикации сайта на локальном Web-сервере по сети нам вообще ничего не нужно делать. Локальная сеть, а тем более жесткий диск нашего собственного компьютера доступны всегда.

Простейший способ опубликовать наш сайт — это выделить его корневую папку ("корень" иерархического списка) в списке файлов локальной копии панели **Files** и нажать кнопку **Put File(s)** инструментария этой панели (🏠). Данная кнопка запускает процесс копирования файлов в корневую папку Web-сервера. Также мы можем выбрать пункт **Put** подменю **Site** дополнительного меню панели или нажать комбинацию клавиш <Ctrl>+<Shift>+<U>.

Так как мы выбрали корневую папку, т.е., фактически, весь сайт, Dreamweaver переспросит нас, действительно ли мы хотим скопировать весь сайт на сервер. В ответ нажмем кнопку **ОК**. Мгновение — и... наш сайт опубликован на сервере.

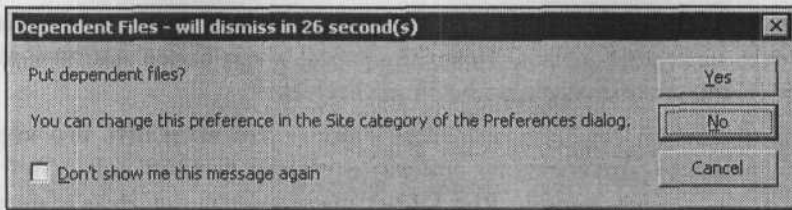
Давайте проверим, правильно ли все Dreamweaver сделал. Выберем в правом раскрывающемся списке инструментария панели **Files** пункт **Remote View**. Список этой панели тотчас покажет все файлы удаленной копии сайта. (Если этого не произошло, проверьте, выполнили ли вы подключение к FTP-серверу.) Чтобы продолжить работу с локальной копией, достаточно выбрать в этом же списке пункт **Local View**.

А теперь можно запустить Web-обозреватель, набрать в строке адреса интернет-адрес нашего сайта и увидеть наш сайт воочию.

Dreamweaver также позволяет нам опубликовать на сервере отдельные файлы сайта. Для этого достаточно выделить нужные файлы в списке панели **Files** и нажать все ту же кнопку **Put file(s)**. Если у нас какие-то Web-страницы открыты, изменены и не сохранены, Dreamweaver предложит сохранить их. В этом случае нужно нажать кнопку **Yes** для сохранения соответствующего файла, кнопку **No** — для отказа от сохранения или кнопку **Cancel** — для отказа от его публикации.

Также Dreamweaver может спросить нас, публиковать ли на сервере файлы, на которые ссылается данная страница, выведя на экран диалоговое окно **Dependent Files** (рис. 4.13). (Это вполне может случиться, когда мы начнем работать с графическими изображениями на занятии 5.) Если нажать кнопку **Yes**, эти файлы будут опубликованы, если **No** — будут опубликованы только те файлы, которые мы выделили в списке панели **Files**, если **Cancel** — публикация файлов на Web-сервере будет прервана.

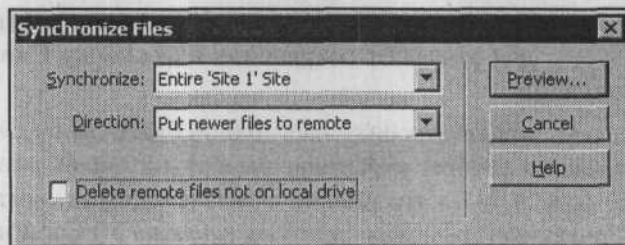
Да, все это просто. Если мы точно знаем, какие файлы хотим опубликовать. Но что делать, если мы забыли, какие файлы изменяли (обычная, кстати, ситуация...)? Публиковать заново весь сайт? А если его файлы занимают много места, а наш канал доступа в Интернет оставляет желать лучшего?

Рис. 4.13. Диалоговое окно **Dependent Files**

Дело в том, что операционная система Windows (как и многие другие операционные системы) хранит дату и время последнего изменения каждого файла. Сравнивая две даты, можно выяснить, какой файл новее. И если файл, принадлежащий локальной копии сайта, имеет более позднюю дату последнего изменения, чем тот же файл удаленной копии, это значит, что он был изменен в Dreamweaver, но еще не опубликован. И, следовательно, его нужно скопировать на сервер, "освежив" удаленную копию сайта, или, как говорят профессиональные Web-дизайнеры, поддержать ее актуальность.

Именно на таком принципе основан механизм *синхронизации* копий сайта. Dreamweaver проверяет даты разных копий файлов и принимает решение, какие из них нужно скопировать на сервер. Просто и надежно, если, конечно, встроенные часы и календарь нашего компьютера установлены правильно.

Чтобы запустить синхронизацию файлов, выберем пункт **Synchronize** подменю **Site** дополнительного меню панели **Files**. (Этот же пункт доступен в контекстном меню панели **Files**.) На экране появится диалоговое окно **Synchronize Files**, показанное на рис. 4.14.

Рис. 4.14. Диалоговое окно **Synchronize Files**

Раскрывающийся список **Synchronize** позволяет задать, какие файлы мы хотим синхронизировать. Пункт **Selected Local File Only** позволяет синхронизировать только выделенные в списке файлы. А пункт **Entire <название нашего сайта> Site** позволяет синхронизировать весь сайт целиком, так что давайте выберем именно его.

В раскрывающемся списке **Direction** нам будет нужно выбрать пункт **Put newer files to remote**, чтобы Dreamweaver скопировал новые файлы на сервер, заменив ими устаревшие файлы удаленной копии. Если в локальной копии сайта мы удалили какие-то файлы, то должны также будем включить флажок **Delete remote files not on local drive**, иначе Dreamweaver в удаленной копии их так и не удалит. (И что это он все перестраховывается?..)

Сам процесс синхронизации запускается нажатием кнопки **Preview**. После этого на экране появится диалоговое окно **Synchronize**, показанное на рис. 4.15. В нем мы сможем просмотреть список всех файлов, подлежащих синхронизации, и указать, какие файлы нужно синхронизировать, а какие — нет.

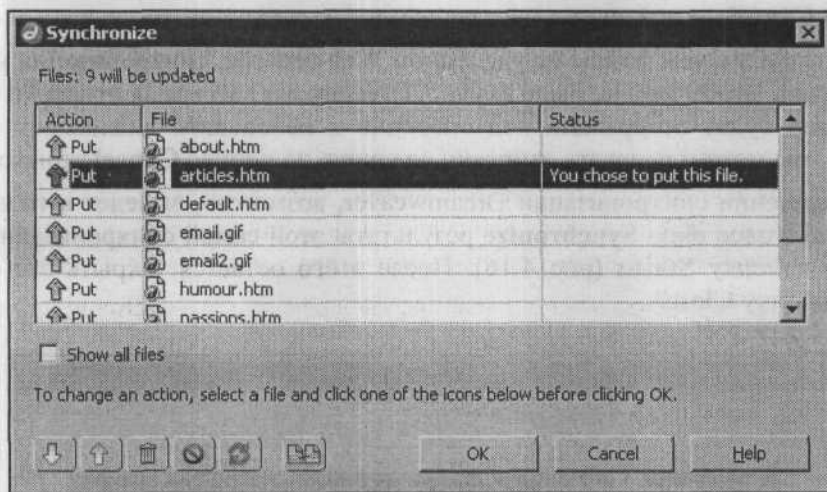


Рис. 4.15. Диалоговое окно Synchronize




Большую часть этого окна занимает собственно список синхронизируемых файлов. Он организован в виде таблицы с тремя колонками, которые мы сейчас рассмотрим.

Колонка **Action** содержит название действия, которое будет применено к файлу. В нашем случае таких действий может быть три:

- ◆ **Put** — копирование на сервер;
- ◆ **Delete** — удаление с сервера;
- ◆ **Ignore** — игнорирование.

Вдоль нижнего края окна находится набор кнопок, задающих действие над файлом.

Вот три кнопки из этого набора, которые будут нам полезны:

- ◆  — файл должен быть скопирован на сервер;
- ◆  — файл должен быть удален с сервера;
- ◆  — файл должен быть проигнорирован.

С остальными двумя колонками все ясно. Колонка **File** содержит имя файла, а колонка **Status** — текст, описывающий состояние синхронизации и показываемый после ее завершения.

Синхронизация запускается нажатием кнопки **OK**. Если какие-либо файлы в результате синхронизации должны быть удалены, Dreamweaver предупредит нас об этом; нажатие кнопки **OK** выполняет удаление этих файлов, а нажатие кнопки **Cancel** — отменяет.

Если мы публикуем файлы на удаленном Web-сервере, синхронизация может затянуться. Чтобы нам не было скучно, Dreamweaver во время отправки сайта на сервер будет отображать ход операции в небольшом диалоговом окне. Кстати, мы можем прервать отправку, щелкнув по кнопке **Cancel** этого окна.

По завершении синхронизации Dreamweaver, возможно, выведет в то же самое диалоговое окно **Synchronize** результаты этой самой синхронизации, заполнив колонку **Status** (рис. 4.16). После этого остается закрыть это окно, нажав кнопку **Close**.

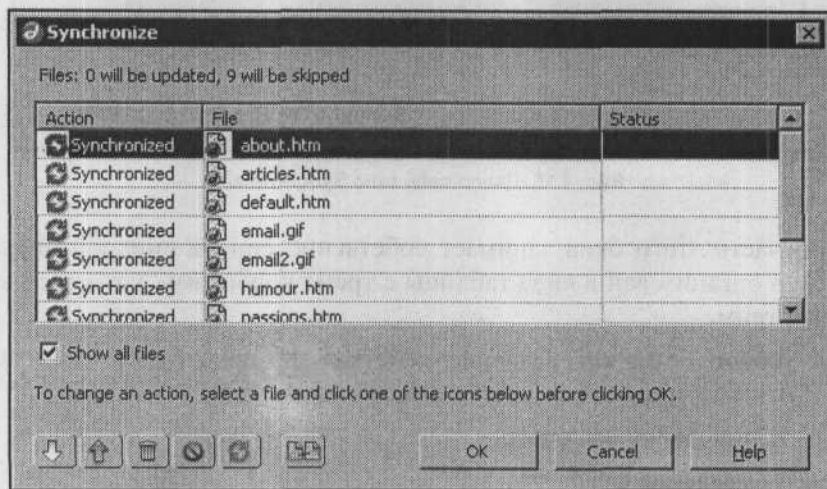



Рис. 4.16. Dreamweaver только что закончил синхронизацию

И еще одно важное замечание на тот случай, если мы публикуем сайт на удаленном Web-сервере. Закончив работу с удаленной копией сайта, нужно обя-

зательно разорвать соединение с FTP-сервером, через который мы публикуем свой сайт. Для этого достаточно щелкнуть кнопку **Disconnect from remote host**  инструментария панели **Files**.

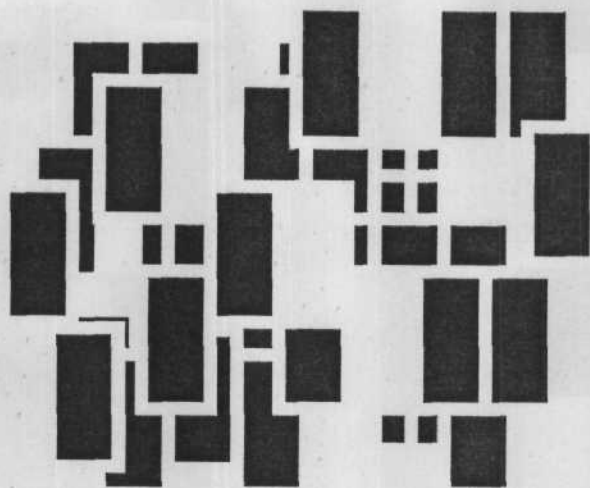
Внимание!

Нужно обязательно отключаться от FTP-сервера после окончания работы с удаленной копией сайта! На поддержание каждого соединения с клиентом FTP-сервер расходует память серверного компьютера, и может наступить такой момент, когда другому клиенту этих ресурсов может не хватить. Давайте уважать других клиентов!

На следующем занятии...

Что ж, наш первый настоящий Web-сайт готов. И не только готов, но и опубликован на Web-сервере. И теперь любой желающий может зайти на него и полюбоваться нашим творением.

Конечно, ничего особого на нашем сайте нет — только текст. Неплохо было бы "разбавить" его графикой. Этим мы и займемся на следующем занятии. И создадим страничку со сведениями о гипотетическом студенте Васильеве и его портретом.

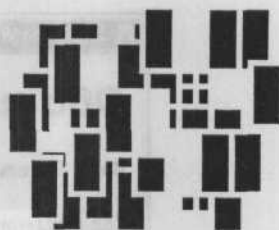


КУРС 2

Более сложный текстовый дизайн

- Занятие 5.** Работа с графикой
- Занятие 6.** Работа с таблицами
- Занятие 7.** Использование стилей CSS

ЗАНЯТИЕ 5



Работа с графикой

Что ж, наш сайт готов и даже опубликован в Сети. Возможно, он даже широко известен среди узкого круга друзей и однокашников студента Васильева. Будем ли мы останавливаться на достигнутом? Нет!

На занятии 3 мы научились работать с текстом. Сейчас же мы займемся графикой. Мы научимся помещать на Web-страницы графические изображения и использовать их для специальных целей, например, для создания изображений-гиперссылок и изображений-карт. А Dreamweaver нам в этом поможет.

Для экспериментов с графикой мы используем созданную нами ранее страницу со сведениями о создателе сайта `about.htm`. Давайте откроем ее в Dreamweaver и напишем текст, кратко рассказывающий о студенте Васильеве В. В. Не будем ничего особо выдумывать — вполне достаточно двух-трех абзацев. Созданная нами страница `about.htm` показана на рис. 5.1.

Сохраним эту страницу. Вообще, лучше сохранять Web-страницы (да и другие документы) почаще — так мы обезопасим себя от потери данных. И поговорим о том, каким же образом на Web-страницы помещаются графические изображения.

Графика на Web-страницах. Внедренные элементы

С чем мы имели дело ранее? С абзацами текста, отформатированными различными способами, иначе говоря, с *текстовыми элементами* Web-страниц. Все текстовые элементы создаются с помощью соответствующих тегов языка HTML, изученными нами ранее — `<P>`, `<H1>`, ``, `<EMP>`, `<A>` и пр., — и сохраняются в текстовых файлах с расширением `htm[1]`, т. е. в файлах собственно Web-страниц. Можно сказать, что большинство Web-страниц состоят из таких текстовых элементов.

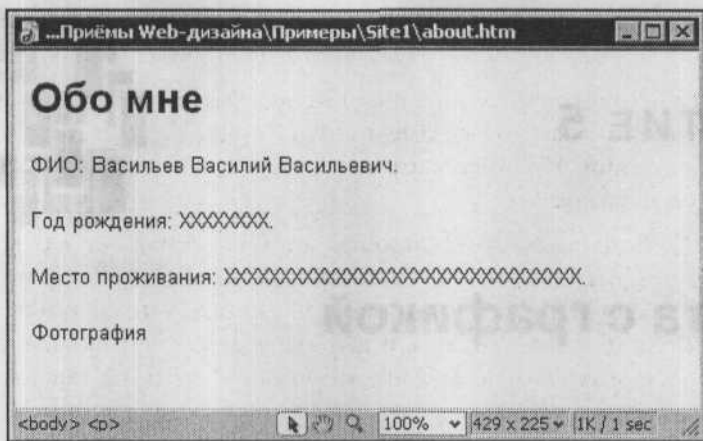


Рис. 5.1. Web-страница сведений об авторе сайта

Но графические изображения в виде текста сохранить невозможно — они имеют другую природу. Попробуйте как-нибудь открыть в текстовом редакторе, том же Блокноте, графическое изображение. Вы увидите только мешанину непонятных символов — текстовый редактор будет стараться вывести данные изображения в виде текста, но это у него так и не получится.

Вывод: графические изображения не удастся сохранить в составе Web-страницы. Что же делать?

А и не хранить их в самой Web-странице! Графические изображения, предназначенные для размещения на страницах, хранятся в отдельных файлах. А в коде HTML этих страниц с помощью особых тегов ставятся своего рода ссылки на эти файлы. Встретив такую ссылку, Web-обозреватель загружает нужный файл и выводит содержащееся в нем изображение в соответствующем месте Web-страницы.

Графические изображения и некоторые другие элементы страниц, также хранящиеся в отдельных файлах, называются *внедренными элементами* Web-страниц.

Форматов графических изображений на свете существует очень много. Но в Web-графике популярны только три из них: GIF, JPEG и PNG. Об этих форматах мы сейчас кратко поговорим.

Формат GIF (Graphic Interchange Format, формат обмена графикой) замечательно подходит для сохранения *штриховых* изображений, состоящих из набора линий разной длины, формы и цветов. Поэтому его используют для хранения элементов оформления страниц (линий, маркеров списков и т. п.), всяческих схем, чертежей, рисунков карандашом и т. п. Кроме того, формат GIF

нить в одном файле целый фильм (*анимированный GIF*), что незаменимо в рекламе.

Формат JPEG (Joint Pictures Encoding Group, группа кодирования неподвижных изображений), напротив, замечательно подходит для хранения *полутоновых* изображений. Поэтому картины и сканированные фотографии хранят только в этом формате.

Формат PNG (Portable Network Graphics, перемещаемая сетевая графика), как говорят его создатели, объединяет возможности GIF и JPEG, не "прихватывая" заодно с собой их недостатки. Надо сказать, что задумка получилась удачная: в PNG можно сохранять и штриховые, и полутоновые изображения без потери качества. Только вот что-то он пока никак не завоевывает заслуженную популярность...

На заметку

В Интернете, кроме трех перечисленных ранее, применяются также и другие форматы графики. В частности, очень популярен формат Shockwave/Flash, разработанный фирмой Macromedia и позволяющий создавать, кроме статичных изображений, также фильмы и даже целые программы. Но, в отличие от GIF, JPEG и PNG, эти форматы не поддерживаются Web-обозревателями напрямую и требуют дополнительных программ, а для помещения их на Web-страницы используются другие приемы.

Работа с графическими изображениями

Вот и настала пора дать студенту Васильеву "лицо" — поместить на страницу со сведениями об авторе сайта его портрет. Давайте поищем на жестком диске нашего компьютера какое-нибудь подходящее изображение, а если не найдем, то нарисуем сами (как поступил автор этой книги, хотя художник из него не очень). И поместим файл с портретом или рисунком в корневую папку локальной копии нашего сайта (Site1) — так нам самим будет проще.

Вставка графического изображения

Откроем страницу сведений об авторе сайта, если она еще не открыта или уже закрыта. Портрет мы вставим сразу же после слова *фотография*. Поместим текстовый курсор в конце этого слова и посмотрим на вкладку **Common** инструментария объектов — там находится кнопка **Image**. Нажмем на нее и в появившемся на экране меню (рис. 5.2.) выберем пункт **Image**. Также можно воспользоваться пунктом **Image** меню **Insert** или нажать <Ctrl>+<Shift>+<I>.

После этого на экране появится диалоговое окно **Select Image Source**, показанное на рис. 5.3.

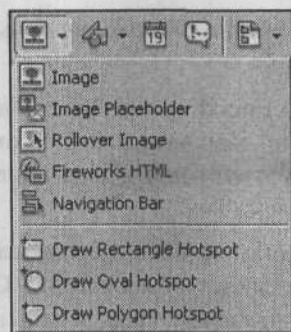


Рис. 5.2. Кнопка Image инструментария объектов с открытым меню

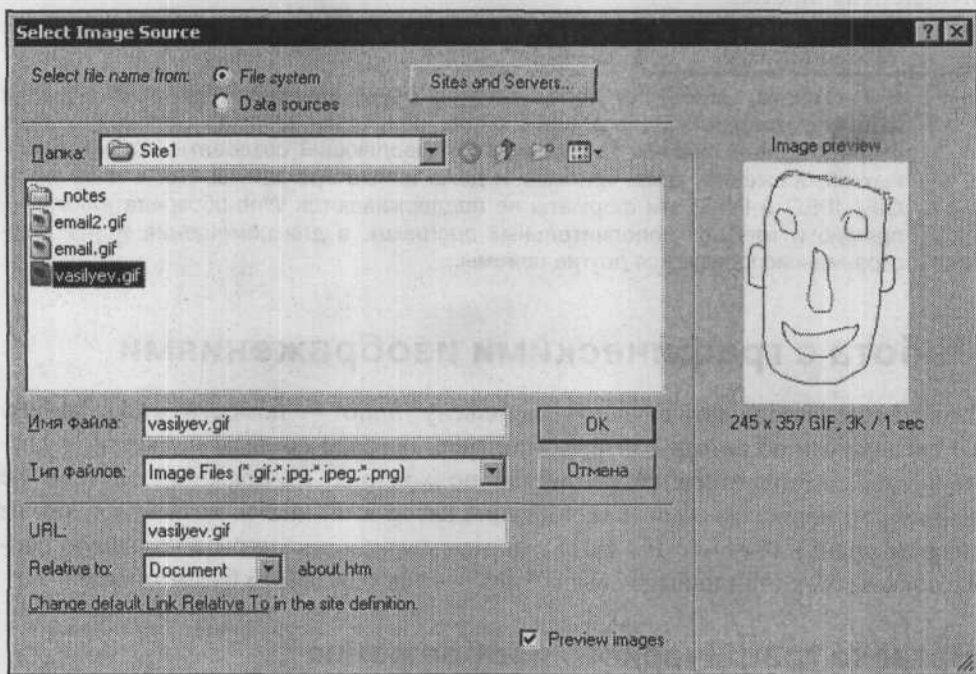


Рис. 5.3. Диалоговое окно Select Image Source

Раскрывающийся список папок и список файлов позволят нам выбрать нужную папку и файл. В поле ввода **Имя файла** появится имя выбранного файла (или мы можем сами ввести его туда). А раскрывающийся список **Тип файлов** позволит нам выбрать, какой тип файлов нам нужно найти. Все это знакомо нам по стандартным диалоговым окнам открытия и сохранения файлов Windows. Единственное отличие — справа находится панель предваритель-

сильевича. А если она кому-то настолько противна, что он хочет убрать ее с глаз долой, пусть просто отключит флажок **Preview images**.

Итак, мы выбрали файл, где находится портрет гипотетического автора сайта. Осталось нажать кнопку **OK**. Но Dreamweaver потребует от нас еще кое-какую информацию, выведя диалоговое окно **Image Tag Accessibility Attributes**, показанное на рис. 5.4.

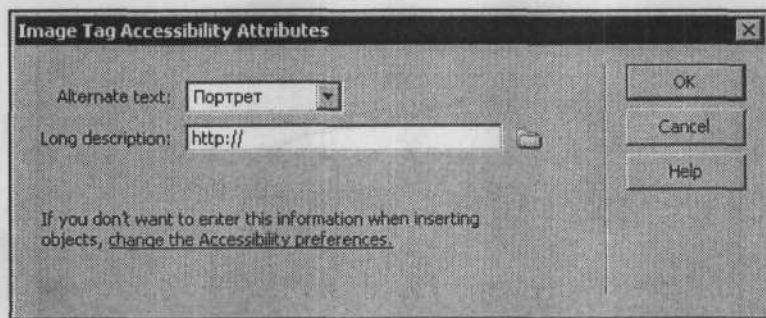


Рис. 5.4. Диалоговое окно Image Tag Accessibility Attributes

Комбинированный список **Alternate text** этого окна служит для задания так называемого *текста замены*. Это придумано для пользователей медленных каналов связи. После того как Web-обозреватель загрузит HTML-файл с Web-страницей, он вместо изображения, помещенного на ней, отобразит пустую рамку соответствующих размеров. Когда пользователь поместит курсор мыши над пустой рамкой рисунка, Web-обозреватель выведет небольшую подсказку, содержащую этот самый текст замены. Поэтому мы рекомендуем вам всегда пользоваться этой возможностью.

На заметку

Dreamweaver считает отсутствие текста замены ошибкой. Посмотрите на рис. 4.10 (там изображено диалоговое окно **Reports**, задающее параметры проверки HTML-кода на предмет ошибок). В этом окне имеется флажок **Missing Alt Text**, который мы не рассмотрели на занятии 4. Если его включить, Dreamweaver будет искать графические изображения с отсутствующим текстом замены.

Собственно в список **Alternate text** вводится *краткий текст замены*. Его ограничение — не более 50 символов. Если нам нужно выводить на экран более подробный текст замены, его можно сохранить в отдельный файл Web-страницы, а потом ввести его интернет-адрес в поле ввода **Long description**. Также можно щелкнуть кнопку в виде папки, расположенную правее этого поля, и выбрать нужный файл в диалоговом окне **Select File**, похожем на окно **Select Image File**.

Внимание!

Подробный текст замены поддерживается не всеми Web-обозревателями.

Закончив ввод данных, нажмем кнопку **OK**. Dreamweaver поместит графическое изображение в то место, где на данный момент находится текстовый курсор. И наша страница примет вид, представленный на рис. 5.5.

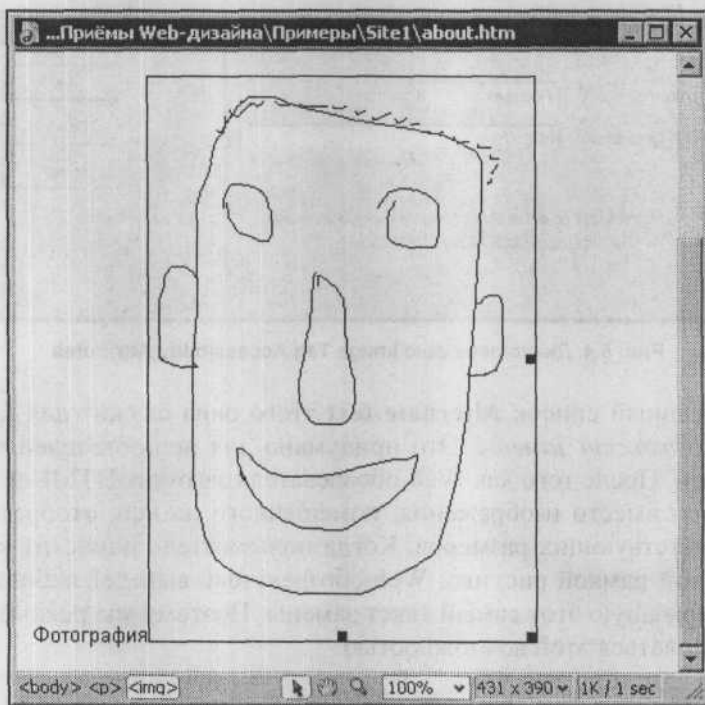


Рис. 5.5. Страница сведений об авторе с его портретом

Ну вот! Мало того, что этот портрет страшен, как семь смертных грехов, так он еще и занял почти все окно документа! Давайте его уменьшим, а то пользователь испугается и уйдет с сайта.

Давайте щелкнем мышью по только что вставленному изображению, чтобы его выделить. (Собственно, Dreamweaver автоматически выделит изображение сразу после его вставки.) После этого вокруг изображения появится тонкая черная рамка, на правой и нижней границе которой появятся небольшие черные квадратики. Это так называемые *маркеры изменения размера*. Мы можем "захватить" мышью любой маркер и перетащить его на новое место, изменив тем самым горизонтальный или вертикальный размер изображения соответственно. А если мы хотим, чтобы оба размера изменялись пропорцио-

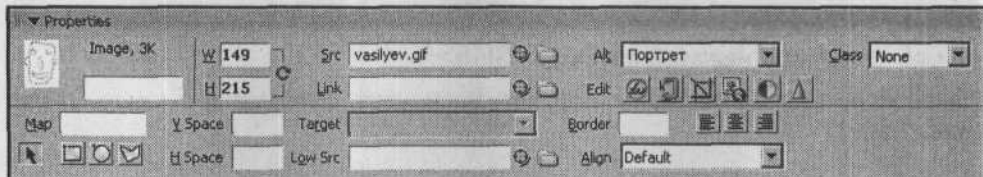


Рис. 5.7. Вид редактора свойств при выделенном изображении

Поля ввода **W** и **H** позволяют нам задать размеры изображения, введя его, соответственно, ширину и высоту вручную. Это может быть полезно, если выделенное изображение — часть оформления сайта; в остальных случаях удобнее задавать размеры изображения, перетаскивая мышью маркеры изменения размера.

Вообще-то, Dreamweaver при помещении на Web-страницу графического изображения сам помещает в эти поля ввода его изначальные ширину и высоту. Как мы уже знаем, Web-обозреватель сразу после загрузки страницы отображает еще не загруженные изображения в виде пустых рамок. Если размеры изображений были явно заданы, они будут сразу же применены к рамкам, и оформление страницы не нарушится. В противном случае Web-обозреватель отобразит рамки некоего размера по умолчанию, и при последующей загрузке изображений их размеры будут меняться, что вызовет перерисовку самой страницы. А это очень неприятно.

Поля ввода **V Space** и **H Space** задают, соответственно, вертикальное и горизонтальное расстояние от края изображения до обтекающего его текста. По умолчанию оба они равны нулю.

Поле ввода **Src** задает интернет-адрес файла, где хранится графическое изображение. Справа от него видны две кнопки. Нажав на правую из них (с изображением папки), мы откроем диалоговое окно **Select Image Source**, показанное на рис. 5.3. А левая кнопка (в виде мишени) уже знакома нам по *заплатке 4*.

Мы также можем изменить имя файла изображения, щелкнув по изображению правой кнопкой мыши и выбрав пункт **Source File** в контекстном меню, или просто дважды щелкнув мышью по изображению. После этого на экране появится все то же диалоговое окно **Select Image Source**.

Поле ввода **Low Src** аналогично полю **Src**, за тем исключением, что задает имя файла, где сохранено так называемое "черновое" изображение. "Черновое" изображение имеет меньший размер, как правило, за счет более низкого качества, и придумано, опять же, для владельцев низкоскоростных каналов связи. Web-обозреватель первым делом загрузит "черновик", т. к. он имеет значительно меньший размер, и выведет его на странице. А уже потом, пока

пользователь просматривает готовую страницу, постепенно загружается основное изображение и подменяет собой "черновик".

Рекомендуется изготавливать "черновик" только тогда, когда оригинальное изображение слишком велико, чтобы быстро загрузиться. В частности, это подойдет, если мы соберемся делать сайт с художественной графикой.

Dreamweaver предоставляет вам другую возможность задать имя файла "черновика". Для этого нужно воспользоваться пунктом **Low Source** в контекстном меню изображения и выбрать нужный файл в появившемся на экране диалоговом окне **Select Image Source**.


Поле ввода **Border** позволяет задать толщину рамки, отображаемой вокруг изображения, в пикселах. По умолчанию она равна нулю, т. е. рамки нет.

Комбинированный список **Alt** задает уже знакомый нам краткий текст замены.

А теперь обратимся к раскрывающемуся списку **Align**. Он позволяет нам задать выравнивание изображения, фактически — относительное местоположение его и текста, в котором оно находится. Список **Align** содержит следующие пункты:

- ◆ **Default** — расположение по умолчанию, обычно аналогично пункту **Baseline**;
- ◆ **Baseline** — низ изображения совпадает с *базовой линией текста* (воображаемой линией, на которой находится строка текста) строки, в которой оно находится;
- ◆ **Top** — верх изображения совпадает с верхом текста строки, в которой оно находится;
- ◆ **Middle** — середина изображения совпадает с базовой линией текста;
- ◆ **Bottom** — низ изображения совпадает с низом текста (обычно не то же самое, что **Baseline**);
- ◆ **TextTop** — верх изображения совпадает с верхом самого высокого символа текста (обычно не то же самое, что **Top**);
- ◆ **Absolute Middle** — середина изображения совпадает точно с *центральной линией текста* (линией, проходящей через центр строки);
- ◆ **Absolute Bottom** — низ изображения совпадает с низом самого нижнего символа текста;
- ◆ **Left** — изображение "прижимается" к левому краю страницы, а текст "обтекает" его справа;
- ◆ **Right** — изображение "прижимается" к правому краю страницы, а текст "обтекает" его слева.

В последних двух случаях изображение становится *плавающим*, т. е. не привязанным жестко к тексту. "Плавающее" изображение может быть смещено

Web-обозревателем влево или вправо, при этом текст, в который оно было вставлено, будет обтекать его. А в точке, где было вставлено плавающее изображение, Dreamweaver отображает специальный *маркер плавающего изображения* — . Этот маркер выводится только для удобства Web-дизайнера и только в Dreamweaver; Web-обозреватель его никак не отображает.

Внимание!

Скорее всего, изначально Dreamweaver не будет показывать маркеры плавающих изображений. Чтобы увидеть их, выберите пункт **Preferences** в меню **Edit**, переключитесь на категорию **Invisible Elements** диалогового окна **Preferences**, включите флажок **Anchor points for aligned elements** и нажмите кнопку **OK**.

Задать выравнивание изображения также можно в подменю **Align** контекстного меню. Это подменю содержит те же пункты, что и раскрывающийся список **Align** редактора свойств.

Итак, с редактором свойств мы разобрались. Теперь давайте зададим свойства портрета нашего героя. Расстояние от текста сделаем по 5 пикселей по горизонтали и вертикали, выравнивание — по левому краю, "альтернативный" текст — **Портрет**. И сохраним страницу.

Теперь опишем еще несколько возможностей, предлагаемых Dreamweaver для работы с изображениями. Нужно сразу оговориться, что они весьма невелики — если вы хотите что-то подправить, лучше воспользуйтесь специализированным графическим редактором, например, Macromedia Fireworks, который замечательно уживается в паре с Dreamweaver.

Чтобы вернуть исходные размеры изображения, мы можем воспользоваться кнопкой отмены (рис. 5.8), расположенной между полями ввода **W** и **H** редактора свойств и правее их, а также пунктом контекстного меню **Reset Size**. Это полезно, если мы сильно их исказили и хотим начать все сначала.

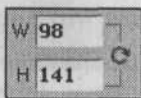



Рис. 5.8. Кнопка редактора свойств для установки исходных размеров изображения (правее полей **W** и **H** и между ними)

Кнопка **Edit**  редактора свойств и пункт **Edit With <имя программы>** контекстного меню позволят нам открыть выделенное изображение в программе, установленной в системе как программа для открытия данных файлов по умолчанию. То есть это та программа, в которой графические файлы будут открыты при двойном щелчке на них в окне Проводника. (У автора — это Microsoft Photo Editor, поставляющийся в составе Microsoft Office.)

Специальные изображения

Язык HTML и Dreamweaver позволяют создавать не только обычные изображения, но и имеющие особые свойства. Это изображения-гиперссылки, активные изображения и карты-изображения. Они часто используются на Web-страницах, поэтому нам нужно будет их рассмотреть.

Изображения-гиперссылки

На занятии 3 мы научились преобразовывать фрагменты текста в гиперссылки. Но, кроме текста, гиперссылкой можно сделать и графическое изображение. Такие *изображения-гиперссылки* часто встречаются на Web-страницах.

Сделать изображение-гиперссылку очень просто, так же просто, как и обычную, *текстовую гиперссылку*. Нужно только выделить необходимое изображение и ввести в поле ввода **Link** редактора свойств нужный адрес.

Давайте, чтобы проиллюстрировать ранее сказанное, вставим на нашу страницу со сведениями об авторе еще одно изображение. Пусть это будет значок "коммерческое эт", часто применяющийся в качестве символа электронной почты. Создадим его в любом графическом редакторе или найдем готовый. Назовем файл с этим изображением `email.gif` и сохраним в корневой папке нашего сайта (`Site1`).

Наше изображение-гиперссылка будет находиться между абзацем Место проживания и фотографией. Создадим там новый абзац и поместим в него изображение `email.gif`. Уменьшим это изображение так, чтобы оно не сильно отличалось от размера шрифта текста, и установим для него выравнивание **Absolute Middle** (одноименный пункт раскрывающегося списка **Align** в редакторе свойств). Результат показан на рис. 5.9.



Рис. 5.9. Значок "коммерческое эт" — символ электронной почты

Теперь преобразуем этот значок в почтовую гиперссылку. Для этого выделим его, наберем в поле ввода **Link** редактора свойств почтовый адрес нашего героя

mailto:vasilyev@mail.ru

Внешне изображение-гиперссылка ничем не отличается от обычного изображения. Однако если мы вызовем Web-обозреватель для предварительного просмотра Web-страницы нажатием клавиши <F12>, то при наведении курсора мыши на это изображение он изменит свою форму на "указующий перст". А при щелчке на этом изображении откроется почтовый клиент.

Если параметр **Border** изображения-гиперссылки имеет значение, отличное от нуля, то рамка, рисуемая вокруг изображения, будет иметь такой же цвет, каким по умолчанию Web-обозреватель выделяет гиперссылки. В нашем случае, такая рамка будет иметь светло-синий цвет. Данный эффект можно специально использовать, чтобы создать оригинальный дизайн Web-страницы, но обычно такое оформление выглядит очень коряво, и Web-дизайнеры этой возможностью не пользуются.

Ну и, конечно же, изображение-гиперссылка может указывать на обычную Web-страницу. На то она и гиперссылка!

Осталось сказать, что HTML-код, использующийся для создания изображения-гиперссылки, имеет вид:

```
<A HREF="<Интернет-адрес">"><IMG SRC="<Файл изображения">"></A>
```

т. е. изображение фактически является содержимым тега гиперссылки (<A>).

Активные изображения

Изображение-гиперссылка, в отличие от гиперссылки обычной, текстовой, при наведении на него курсора мыши никак не проявляет себя. Конечно, можно выставить параметр **Border** равным пяти или десяти, но вряд ли это будет удачной выдумкой. Вот сделать бы так, чтобы изображение менялось при наведении на него курсора мыши, чтобы пользователь сразу видел, что это гиперссылка...

Оказывается, это можно сделать. Такие изображения, реагирующие на действия пользователя, называются *активными*. И Dreamweaver позволяет делать их несколькими щелчками мыши. Единственное, о чем нам придется позаботиться, так это о втором изображении, которое будет появляться на месте исходного, когда пользователь наведет на него курсор мыши (так называемое *перекрывающее изображение*).

Несколько слов о том, как работают активные изображения. Dreamweaver встраивает в код Web-страницы небольшие программы — *Web-сценарии*, выполняемые Web-обозревателем (подробнее о Web-сценариях мы поговорим на *занятии 13*). Когда посетитель сайта наводит курсор мыши на активное изображение, выполняется сценарий, заменяющий исходное изображение

на перекрывающее. Когда же посетитель уводит курсор мыши с изображения, второй сценарий возвращает изначальное изображение на его законное место.

Давайте рассмотрим, как поместить на страницу активное изображение. Мы используем страницу сведений об авторе `about.htm`, куда мы уже вставили изображение-гиперссылку. Согласитесь, оно выглядит не очень эстетично. Давайте заменим его активным изображением. Создадим в графическом редакторе еще одно изображение значка "коммерческое эт", но теперь уже инвертированное, — оно и будет перекрывающим изображением. Сохраним его в файле `email2.gif` и также поместим его в корневую папку сайта.

На странице `about.htm` (она у нас должна быть еще открыта) удалим созданное ранее изображение-гиперссылку. Текстовый курсор оставим на том же месте.

Активное изображение создается с помощью пункта **Rollover Image** кнопки **Image** (см. рис. 5.2) вкладки **Common** инструментария объектов или пункта **Rollover Image** подменю **Image Objects** меню **Insert**. При этом на экране появляется диалоговое окно **Insert Rollover Image**, показанное на рис. 5.10.

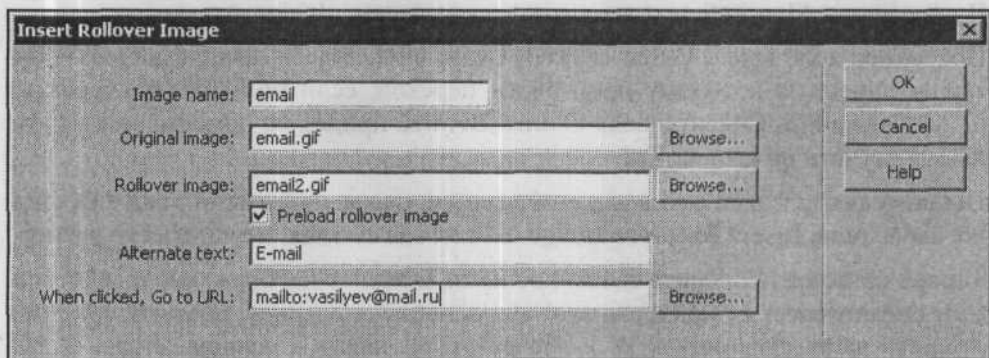


Рис. 5.10. Диалоговое окно **Insert Rollover Image**

В поле ввода **Image name** вводится уникальное имя создаваемого активного изображения. Имя должно содержать только латинские буквы, цифры и знаки подчеркивания, причем начинаться должно с буквы. Проверьте только, действительно ли это имя уникально.

В поле ввода **Original image** вводится имя файла изначального изображения. Можно также нажать кнопку **Browse** справа от этого поля ввода и выбрать нужный файл в появившемся на экране диалоговом окне **Select Image Source**.

В поле ввода **Rollover image** вводится имя файла перекрывающего изображения. Если у нас нет желания вводить его вручную, на помощь всегда придет

Флажок **Preload rollover image** заставляет Dreamweaver сгенерировать и вставить в HTML-код вашей страницы дополнительный Web-сценарий, который заставит Web-обозреватель заранее загрузить перекрывающее изображение и сохранить его в кэше. Этот флажок включен по умолчанию, и отключать его не стоит. Давайте разберемся, почему.

Предположим, при создании активного изображения мы не включили этот флажок. Тогда, если посетитель зайдет на эту страницу и "укажет" мышью на активное изображение, Web-обозреватель будет вынужден загрузить файл перекрывающего изображения. А на загрузку этого файла потребуется какое-то время, в течение которого посетитель вместо этого изображения будет созерцать пустую рамку. То еще зрелище...

Если же мы включим флажок, Web-обозреватель, повинаясь командам созданной Dreamweaver программы-сценария, загрузит перекрывающее изображение во время загрузки страницы и сохранит его в кэше. Когда же оно понадобится, Web-обозреватель позаимствует его прямо из кэша, что займет крошечную долю секунды. И никакие пустые рамки при этом просто не успеют появиться.

В поле ввода **Alternate text** вводится текст замены. Лучше его ввести.

Последнее поле ввода **When clicked, Go to URL** задает самое главное — интернет-адрес, по которому произойдет переход, если пользователь щелкнет на изображении. Спасительная кнопка **Browse** придет на помощь тем, кто не любит стучать по клавиатуре.

Dreamweaver создаст активное изображение после нажатия кнопки **OK** диалогового окна **Insert Rollover Image**. Потом мы сможем изменить его размер.

Теперь остается проверить созданное нами активное изображение в действии. Сам Dreamweaver не поддерживает выполнение сценариев, поэтому придется открыть нашу страницу в Web-обозревателе, нажав клавишу <F12>. Когда Web-обозреватель выведет нашу страницу, поместим курсор мыши над активным изображением, — и мы увидим, как оно изменится. Если же мы щелкнем по нему, откроется почтовый клиент — гиперссылка работает.

Карты-изображения

Изображение-гиперссылка — это простейший случай *графических гиперссылок*, применяемых сейчас в Web-дизайне. Более сложными случаями являются рассмотренное нами чуть раньше активное изображение и *карта-изображение* — сложная графическая гиперссылка, представляющая собой изображение, разбитое на части, причем каждая часть является отдельной гиперссылкой и указывает на свой интернет-адрес. Такие части называются

Для создания карты-изображения используются обычные графические изображения, созданные в любом графическом редакторе. А описание местонахождения, формы, размеров "горячих областей" и интернет-адресов, на которые "горячие области" указывают, хранится в HTML-коде страницы в особых тегах.

Мы сейчас не планируем использовать на своих Web-страницах карты-изображения. Но научиться работать с ними полезно. Поэтому давайте создадим небольшую тестовую Web-страничку с картой-изображением.

Создадим пустую Web-страничку и любое изображение для наших учебных нужд. Изображение сохраним в файле `map.gif`, а страничку — в файле `5.1.htm`. Помещать эти файлы в корневую папку нашего сайта не надо — ведь они не входят в его состав. Напоследок откроем в Dreamweaver страницу `5.1.htm` и поместим на нее изображение `map.gif` (рис. 5.11).

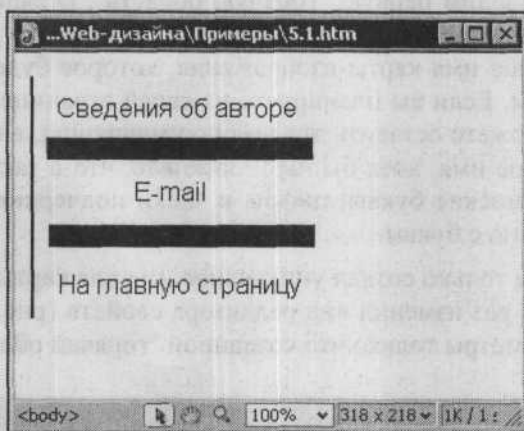


Рис. 5.11. Учебная Web-страница с картой-изображением

Разместить на изображении "горячие области" нам помогут несколько элементов управления, находящиеся в левом нижнем углу редактора свойств и пока еще нами не рассмотренные. Теперь пришла пора разобраться с ними (рис. 5.12).

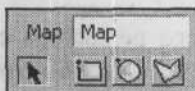


Рис. 5.12. Элементы управления редактора свойств, используемые для создания "горячих областей"

Прежде всего, рассмотрим четыре кнопки, расположенные вдоль нижней

расположенная правее и состоящая из трех кнопок, позволяет создать на изображении соответственно прямоугольную, круглую и многоугольную "горячую область". Единственная кнопка, расположенная левее, позволяет манипулировать уже созданными "горячими областями".

Так как же поместить на изображении "горячую область"? Очень просто. Давайте начнем с прямоугольной. Щелкнем кнопку с изображением прямоугольника, щелкнем по графическому изображению и, не отпуская левой кнопки мыши, протащим ее, пока не захватим всю область изображения, которую хотим сделать "горячей". При этом будет заметно, что за курсором мыши "тянется" синий прямоугольник. "Охватив" этим прямоугольником нужную область (в нашем случае, надпись Сведения об авторе), отпустим кнопку мыши, и созданная нами "горячая область" тотчас закрасится светло-синим.

Как только мы создадим первую "горячую область", Dreamweaver подставит в поле ввода **Map** редактора свойств (см. рис. 5.12) слово **Map**. Что оно значит? Это уникальное имя карты-изображения, которое будет использоваться Web-обозревателем. Если вы планируете на своей странице только одну карту-изображение, можете оставить это имя по умолчанию, в противном случае дайте ей уникальное имя, хотя бы **Map1**. Заметьте, что в таких именах допускаются только латинские буквы, цифры и знаки подчеркивания, причем начинаться имя должно с буквы.

Но Dreamweaver не только создал уникальное имя для карты-изображения, он еще и в очередной раз изменил вид редактора свойств (рис. 5.13). Теперь мы можем задать параметры только что созданной "горячей области".

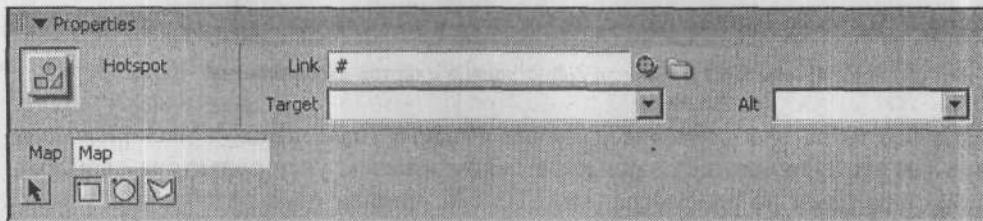


Рис. 5.13. Вид редактора свойств для выделенной "горячей области"

Все элементы управления нового редактора свойств нам уже знакомы. Это поле ввода **Link**, используемое для ввода интернет-адреса, раскрывающийся список **Target**, задающий цель гиперссылки, и комбинированный список **Alt**, где вводится текст замены.

Теперь обратим внимание на саму "горячую область". По углам прямоугольника, представляющего собой эту самую "горячую область", расположены

мера. Это значит, что "горячая область" в данный момент выделена. "Захватив" любой такой маркер мышью, мы можем изменить размеры "горячей области". А чтобы переместить ее на другое место, достаточно просто перетащить ее мышью.

Однако перед всеми этими операциями нужно проверить, что из кнопок, показанных на рис. 5.12, нажата та, на которой изображена стрелка. Если это не так, нажмите ее. Если же "горячая область" почему-то не выделена, щелкните по ней.

Теперь добавим на наше изображение круглую "горячую область". Для этого нажмем кнопку с изображением круга и точно так же "протащим" мышью по изображению, охватив синим кругом надпись E-mail. Dreamweaver поместит круглую область там, где мы ему указали, и активизирует ее, разместив по четырем сторонам круга маркеры изменения размеров.

Мы можем изменять размеры и местоположение круглой "горячей области" так же, как и прямоугольной. Редактор свойств имеет в этом случае такой же вид (см. рис. 5.13).

Многоугольная "горячая область" создается несколько сложнее. Нажав кнопку с изображением многоугольника, прежде всего, щелкните мышью в месте, где должна располагаться первая из *угловых точек* многоугольной области. После этого там появится синяя точка — начало нашего многоугольника. Затем щелкните мышью там, где должна быть вторая угловая точка, и Dreamweaver проведет между этими двумя точками линию. Далее вам останется щелкнуть в местах расположения остальных угловых точек многоугольника, а Dreamweaver сам проведет между ними линии, которые и образуют нужную нам многоугольную "горячую область". Впоследствии вы можете перетащить мышью любую точку этой области. Вот только вставить новую или удалить старую угловую точку вы уже не сможете — придется удалить созданную "горячую область" и создать новую.

Иногда случается так, что две "горячие области" перекрывают друг друга, т. е. имеют общий фрагмент. При этом по щелчку на такой "спорной территории" активизируется "горячая область", расположенная сверху, а сверху всегда появляется та область, что была создана последней. Но не всегда получается разместить "горячие области", как было предусмотрено по плану. Так вот, чтобы управлять их перекрытием, используются пункты **Bring To Front** и **Send To Back** контекстного меню. Первый переносит выделенную "горячую область" наверх, а второй — вниз. (Также можно воспользоваться пунктами **Bring To Front** и **Send To Back** подменю **Arrange** меню **Modify**.)

Итак, мы создали на нашем изображении три "горячие области": прямоугольную, круглую и многоугольную. Выделим их поочередно и зададим для них

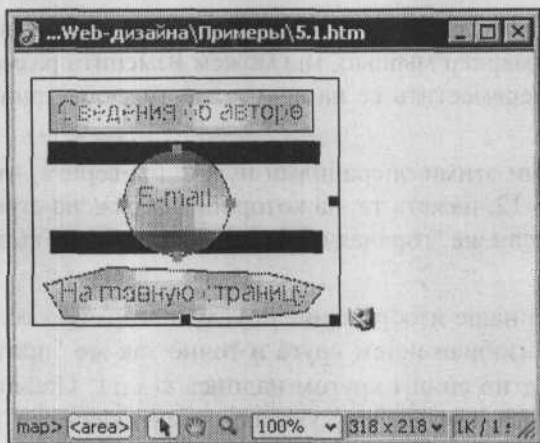


Рис. 5.14. Готовая карта-изображение

После этого можно вызвать Web-обозреватель для просмотра нашей страницы и испытать карту-изображение в действии. Сделайте так — она действительно работает!

Дополнительные возможности по работе с графикой

Иногда бывает нужно исправить само графическое изображение, которое мы уже поместили на Web-страницу. Обычно для этого все-таки приходится пользоваться графическим редактором, например, если мы собираемся что-то перерисовать. Однако, если нам нужно просто обрезать края изображения или преобразовать его в черно-белое, будет достаточно возможностей Dreamweaver.

Обрезка изображения — операция, которую Web-художники, подготавливающие уже готовые изображения к публикации в Сети, выполняют очень часто. В самом деле, если нужно выложить в Сеть фотографию кого-то, надо бы и отрезать случайно попавшие в кадр чужие руки, локти и физиономии. Другой случай: вырезать ключевой фрагмент из огромной бессодержательной фотографии. Во всех этих случаях Dreamweaver идет нам на помощь.

Чтобы выполнить обрезку изображения, нужно сначала его выделить, а потом нажать кнопку **Crop** (☒), находящуюся в редакторе свойств ниже раскрывающегося списка **Alt**. Dreamweaver выдаст небольшое предупреждение, говорящее, что выполняемая нами операция затронет сам файл графического изображения. Закроем его нажатием кнопки **OK** — ведь мы точно знаем, что делаем. не так ли?

Убедившись в нашей решительности, Dreamweaver попытается "вычислить" содержательную часть нашего изображения и окружит ее штриховым прямоугольником (рис. 5.15). Этот прямоугольник показывает границу, по которой будет выполнена обрезка, — *границу обрезки*. Если же Dreamweaver ошибся в своих "вычислениях", мы можем его поправить, перетащив мышью нужный маркер изменения размера (эти маркеры хорошо заметны на рис. 5.15). Мы также можем переместить мышью всю границу обрезки целиком.

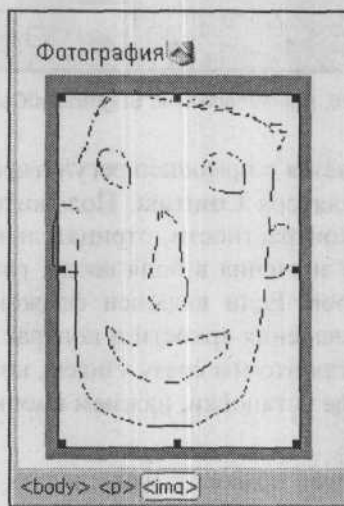


Рис. 5.15. Изображение, готовое к обрезке, и граница обрезки

Чтобы выполнить собственно обрезку изображения, нажмем клавишу `<Enter>`. Отказаться от обрезки можно, нажав клавишу `<Esc>`.

Еще одна часто выполняемая операция — *изменение размеров изображения*. Dreamweaver изменяет размеры изначального изображения так, чтобы оно вписалось в размеры, которые мы для него задали после помещения на Web-страницу. Запускается этот процесс после нажатия кнопки **Resample** (🔄), находящейся в редакторе свойств правее кнопки **Crop**. Снова надоедливое предупреждение, и — размеры изначального изображения, записанного в файле, изменились!

Внимание!

При сильном увеличении растрового изображения возможны потери в качестве. Поэтому не увеличивайте изображения без особой нужды.

Еще Dreamweaver может задать *яркость и контрастность* нашего изображения для чего служит кнопка **Brightness and Contrast** (👤).

свойств. После ее нажатия на экране появляется диалоговое окно **Brightness/Contrast** (рис. 5.16), в котором и выполняются нужные настройки.

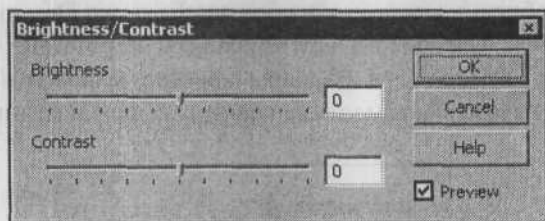


Рис. 5.16. Диалоговое окно **Brightness/Contrast**

Яркость изображения задается с помощью регулятора **Brightness**, а контрастность — с помощью регулятора **Contrast**. Положительные значения задают увеличение яркости или контрастности, отрицательные — уменьшение. Также можно ввести нужные значения в поля ввода, расположенные правее соответствующих регуляторов. Если включен флажок **Preview**, Dreamweaver будет показывать новые значения яркости и контрастности прямо на изображении в окне документа, так что мы сразу узнаем, когда нужно остановиться. Чтобы применить заданные установки, нажмем кнопку **OK**, чтобы отказаться от них — кнопку **Cancel**.

Последнее, что Dreamweaver позволит нам сделать с изображением, — это *увеличить его резкость*. Для этого служит кнопка **Sharpen** (🔍) редактора свойств. После ее нажатия на экране появляется диалоговое окно **Sharpen** (рис. 5.17) с регулятором и полем ввода **Sharpen**, а также флажком **Preview**. Здесь все просто — задаем нужную степень увеличения резкости с помощью регулятора **Sharpen** (или вводим его прямо в поле ввода) и нажимаем кнопку **OK**.

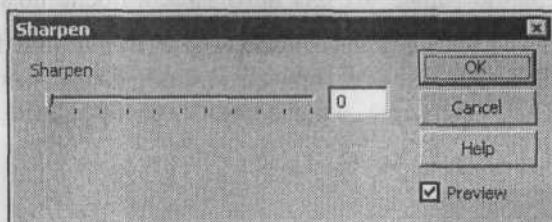


Рис. 5.17. Диалоговое окно **Sharpen**

На этом возможности Dreamweaver 8 по правке графики заканчиваются. Это же не графический пакет, в самом деле!..

Внимание!

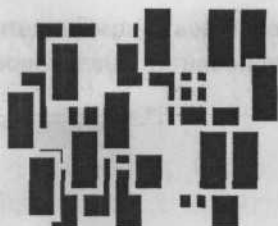
Описанные выше операции над графическими изображениями необратимы. Мы не сможем сделать откат после выполнения любой из них.

На следующем занятии...

На этом занятии мы узнали, как поместить на Web-страницу графическое изображение. Теперь наш однокашник Васильев обзавелся лицом, правда, не бог весть каким. Но для нас главное — научиться не рисованию портретов, а Web-дизайну.

А на следующем занятии мы поговорим о таблицах. В самом деле, таблица — лучший способ поместить максимальное количество данных на минимальной площади. Заодно мы доделаем остальные страницы нашего первого Web-сайта.

ЗАНЯТИЕ 6



Работа с таблицами

Таблицы, таблицы, таблицы... Никуда без них. Вся научная и техническая литература испещрена таблицами, а уж в справочниках и каталогах их столько, что рябит в глазах. Что за "тирания" таблиц обрушилась на наш мир?

На самом деле, у таблиц много достоинств. Например, они позволяют расположить множество данных на небольшой книжной, журнальной или Web-странице. Собранные в таблицу данные выглядят очень наглядно, найти в них нужные сведения ничего не стоит. Кроме того, таблицы — это красиво!


Красота требует жертв, и это занятие мы приносим в жертву таблицам. Мы научимся их создавать, править и форматировать. И Dreamweaver нам в помощь!..

Простые таблицы

Начнем мы с самых простых таблиц. Это те самые таблицы, что составляют большинство в справочниках и каталогах, — обычные наборы строк, столбцов и ячеек без всяких премудростей.

Учиться создавать таблицы мы будем на списке увлечений студента Васильева, за которого мы делаем сайт. Откроем в Dreamweaver страницу passions.htm (она должна быть пуста) и приступим к созданию нашей первой таблицы.

Создание таблиц

Создать таблицу проще всего, нажав кнопку **Table**  на вкладке **Common** инструментария объектов. Также можно выбрать пункт **Table** меню **Insert**

или нажать комбинацию клавиш <Ctrl>+<Alt>+<T>. В любом случае на экране появится диалоговое окно **Table**, показанное на рис. 6.1.

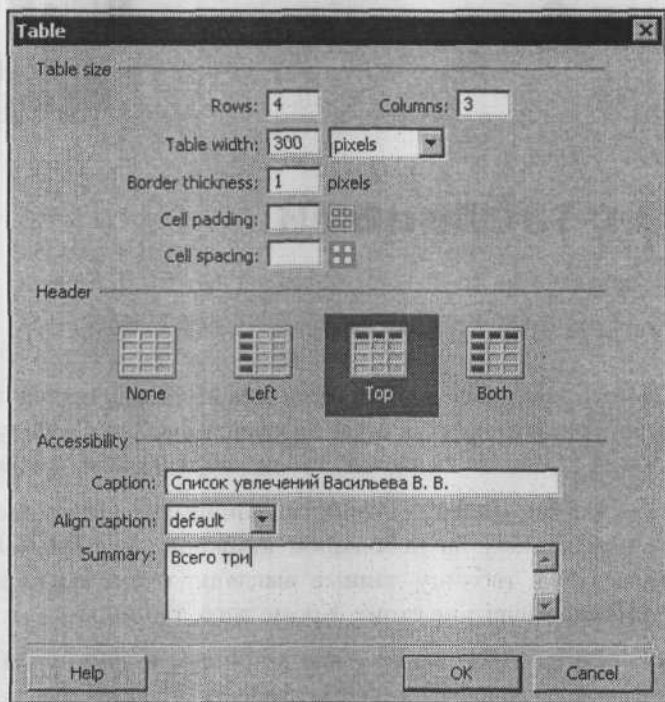


Рис. 6.1. Диалоговое окно Table

В полях ввода **Rows** и **Columns** вводится соответственно количество строк и столбцов создаваемой таблицы. Если мы ошибемся и введем большее или меньшее количество строк или столбцов, не беда — недостающие (лишние) строки и столбцы можно будет добавить (удалить) потом.

В поле ввода **Table width** задается ширина таблицы в пикселах или процентах от ширины Web-страницы. В раскрывающемся списке, расположенном справа от этого поля ввода, нужно будет выбрать соответственно пункт **pixels** или **percent**.

В поле ввода **Border thickness** задается толщина рамок в пикселах. Каждая таблица HTML может содержать видимую рамку. По умолчанию, если в это поле ничего не введено, рамки имеют толщину в один пиксел; вы также можете ввести 0, чтобы сделать рамку невидимой.

В поле ввода **Cell padding** задается отступ между границей ячейки таблицы и ее содержимым в пикселах. По умолчанию он равен одному пикселу.

Аналогично, поле ввода **Cell spacing** служит для задания отступа между границами отдельных ячеек. По умолчанию он равен 2 пикселям.

Набор переключателей **Header** позволит нам создать "шапку" и выделенный первый столбец таблицы. В этом наборе доступны переключатели **None** (нет ни "шапки", ни выделенного первого столбца), **Left** (выделенный первый столбец), **Top** ("шапка") и **Both** (и "шапка", и выделенный первый столбец). При том ячейки, составляющие "шапку" и выделенный столбец, будут оформлены как *ячейки заголовка*, а текст, который мы введем в них, будет автоматически выровнен по центру и выделен полужирным шрифтом.

В поле ввода **Caption** вводится *название таблицы*. Это название будет находиться над создаваемой таблицей.

Раскрывающийся список **Align caption** позволит нам задать местоположение и выравнивание названия (если, конечно, мы его ввели). Здесь доступны следующие пункты:

- ◆ **default** (по умолчанию) — выравнивание выполняет сам Web-обозреватель, обычно в этом случае название находится над таблицей и выравнивается по центру;
- ◆ **top** — название находится над таблицей и выравнивается по центру;
- ◆ **bottom** — название находится под таблицей и выравнивается по центру;
- ◆ **left** — название находится над таблицей и выравнивается по левому краю;
- ◆ **right** — название находится над таблицей и выравнивается по правому краю.

В области редактирования **Summary** вводится *примечание таблицы*. Это примечание не выводится Web-обозревателями на экран, но может быть использовано для каких-то других целей (например, его могут обрабатывать программы, читающие экранный текст). Задавать его не обязательно.

Итак, введем в диалоговом окне **Table** значения, представленные на рис. 6.1, и нажмем кнопку **OK**. У нас должно получиться то, что изображено на рис. 6.2.

Теперь поставим текстовый курсор в любую ячейку таблицы и наберем какой-нибудь текст. Повторим то же самое с остальными ячейками таблицы. У нас получится нечто, похожее на рис. 6.3.

Содержимое ячейки таблицы совсем не обязано быть текстовым. Мы можем поместить в ячейку графическое изображение, как показано на рис. 6.4.

Мы даже можем вставить в ячейку одной таблицы другую! (Кстати, именно этим мы займемся на *занятии 9*, когда будем говорить о табличном дизайне Web-страниц.) В этом случае ширина таблицы, задаваемая в поле ввода **Table**

width диалогового окна **Table** (см. рис. 6.1), считается относительно ширины ячейки, в которой находится таблица (т. е. родительского тега).

Внимание!

Ячейка таблицы обязательно должна иметь хоть какое-то содержимое, иначе Web-обозреватель может отобразить ее некорректно. Если же ячейка все-таки должна быть пустой, вставьте в нее неразрывный пробел (его код HTML — ` `), как это делает в подобных случаях сам Dreamweaver.

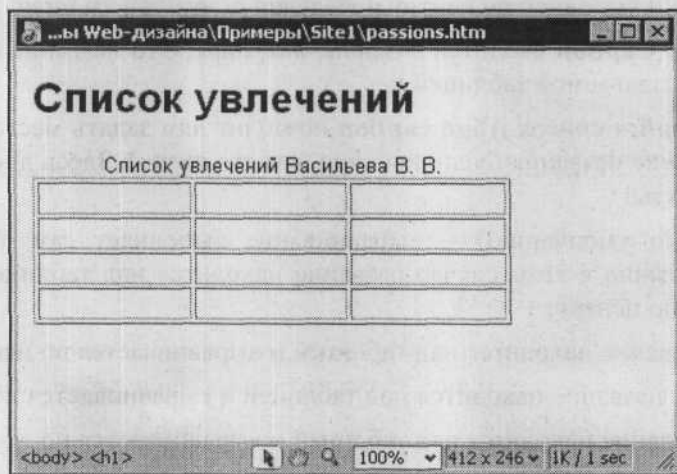


Рис. 6.2. Наша первая таблица

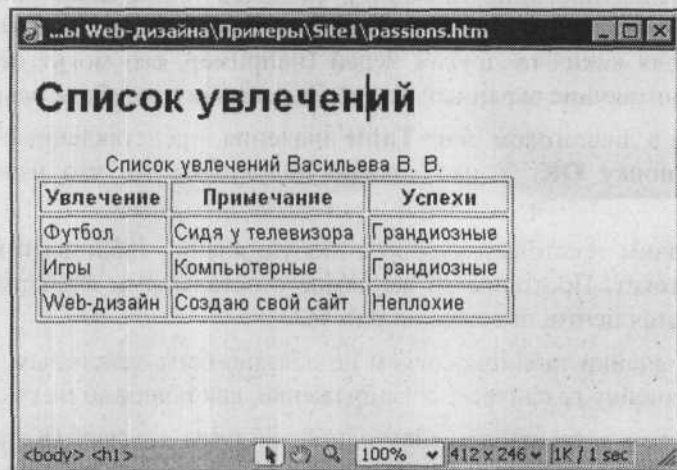


Рис. 6.3. Таблица с заполненными ячейками

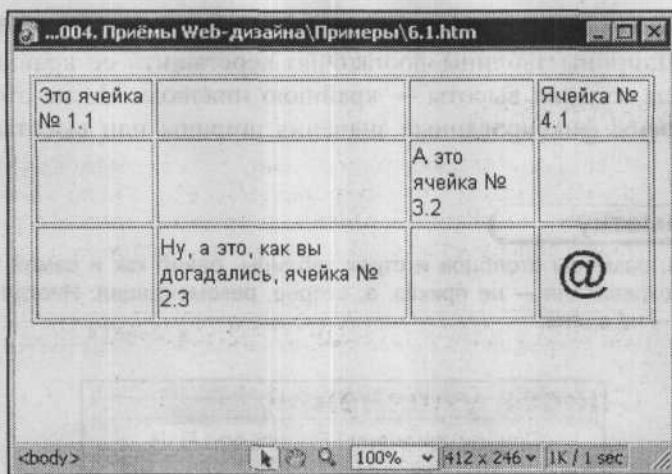


Рис. 6.4. Таблица, в одну из ячеек которой помещено графическое изображение

Работа с таблицами

Давайте попробуем изменить размер окна документа Dreamweaver, в котором открыта страница passions.htm. Заметим, что ширина таблицы при этом не меняется, так как мы создали *таблицу с фиксированной шириной*, заданной в пикселах. Если бы мы создали *таблицу с относительной шириной* ("резиную" таблицу), заданной в процентах от ширины Web-страницы, то таблица бы растягивалась и сжималась (как резиновая).

Также мы заметили, что при вводе текста в ячейки их ширина и высота изменяются, чтобы вместить текст полностью. Такие действия выполняют и Dreamweaver при создании страниц, и Web-обозреватели при их отображении; если столбцы и строки не имеют фиксированного размера (как в нашем случае), он всегда устанавливается программой, отображающей Web-страницу.

Поместим курсор на вертикальной границе между двумя столбцами. Мы увидим, что он примет форму двунаправленной стрелки, и мы сможем захватить мышью границу и переместить ее по горизонтали. Один из столбцов при этом станет шире, а другой — уже. Одновременно им будет присвоено фиксированное значение ширины; Web-обозреватель будет изменять ширину таких столбцов в самом крайнем случае, когда в ячейках, образуемых этими столбцами, не хватит места для их содержимого.

То же самое мы можем сделать и с горизонтальной границей между строками таблицы. В этом случае строкам, границу между которыми мы сдвинем, бу-

Ширина и высота самой таблицы изменяется точно таким же образом. Так, для задания ширины таблицы достаточно перетащить ее крайнюю правую границу, а для задания высоты — крайнюю нижнюю. После этого таблице будут присвоены фиксированные значения ширины или высоты соответственно.

На заметку

Вообще, размеры столбцов и строк таблицы, равно как и самой таблицы для Web-обозревателя — не приказ, а, скорее, рекомендация. Иногда это хорошо, иногда — не очень.

Список увлечений Васильева В. В.		
Увлечение	Примечание	Успехи
футбол	Сидя у телевизора	Грандиозные
Игры	Компьютерные	Грандиозные
Web-дизайн	Создаю свой сайт	Неплохие

Рис. 6.5. Таблица со строкой размеров

Теперь поставим текстовый курсор в любую ячейку таблицы и посмотрим на рис. 6.5. Что мы видим? Dreamweaver показывает ширину и самой таблицы, и всех ее ячеек в узкой светло-серой полосе, тянущейся вдоль нижнего (иногда — верхнего) края таблицы. Это так называемая *строка размеров*. Размеры ячеек и самой таблицы представлены в виде зеленых (на рисунке — темных) *размерных линий*, на которых обычно и проставляются значения ширины. Если же значения ширины нет, значит, она не задана (как у ячеек таблицы).

На заметку

Если строка размеров вам мешает, вы можете временно ее скрыть. Для этого отключите пункт-выключатель **Table Widths** в подменю **Visual Aids** кнопки **View Options** меню **View**. Чтобы вернуть строку размеров, снова включите этот пункт.

О строке размеров и ее использовании мы поговорим позже. Сейчас же давайте выясним, как можно добавить новые или удалить ненужные строки или столбцы.

Чтобы добавить в таблицу новую строку, нужно выбрать пункт **Insert Row** в меню **Table > Insert**.

комбинацию клавиш <Ctrl>+<M>. Новая строка появится выше строки, в ячейке которой находится текстовый курсор (*текущей строки*). Для добавления столбца нужно выбрать пункт **Insert Column** в подменю **Table** меню **Modify** или контекстного меню; также можно нажать <Ctrl>+<Shift>+<A>. Новый столбец будет создан правее столбца, в ячейке которого находится текстовый курсор (*текущего столбца*).

Для добавления в таблицу сразу нескольких строк или столбцов мы можем воспользоваться пунктом **Insert Rows or Columns** подменю **Table** меню **Modify** или контекстного меню. После его выбора на экране появится диалоговое окно **Insert Rows or Columns**, показанное на рис. 6.6.

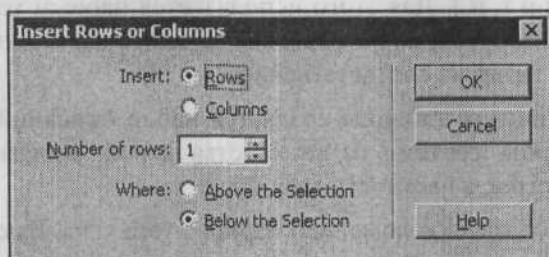


Рис. 6.6. Диалоговое окно **Insert Rows or Columns** (включен переключатель **Rows**)

С помощью переключателей группы **Insert** мы задаем, что именно нам нужно добавить в таблицу. Переключатель **Rows** задает добавление строк, а переключатель **Columns** — столбцов.

В случае если выбран переключатель **Rows**, в поле счетчика **Number of rows** задается количество добавляемых строк, а с помощью группы переключателей **Where** выбирается, где они будут добавлены. Переключатель **Above the Selection** добавит новые строки над текущей строкой, а переключатель **Below the Selection** — под ней.

Если же выбран переключатель **Columns**, в поле счетчика **Number of columns** задается количество добавляемых столбцов, а с помощью двух переключателей **Where** выбирается, где они будут добавлены. Переключатель **Before current Column** добавит новые столбцы левее текущего столбца, а переключатель **After current Column** — правее его.

Новые строки или столбцы будут созданы сразу после нажатия кнопки **OK**.

Ненужные строки и столбцы удаляются еще проще. Для удаления текущей строки нужно либо выбрать пункт **Delete Row** в подменю **Table** меню **Modify** или контекстного меню, либо нажать комбинацию клавиш <Ctrl>+<Shift>+<M>. Для удаления текущего столбца нужно либо выбрать пункт **Delete Column** в подменю **Table** меню **Modify** или контекстного меню, либо нажать

комбинацию клавиш <Ctrl>+<Shift>+<->. При этом столбец или строка будут удалены вместе со всем содержимым.

Исправить ошибку в тексте названия таблицы также очень просто. Просто ставим текстовый курсор в нужное место названия и вносим исправления.

Средства HTML, используемые для создания таблиц

Таблица средствами языка HTML формируется в четыре этапа. Рассмотрим их. На первом этапе создается сама таблица и задаются ее параметры (ширина, толщина границы и т. п.). Для этого используется парный тег <TABLE>, имеющий множество атрибутов, из которых мы выделим WIDTH и HEIGHT, задающие ширину и высоту таблицы соответственно.

На втором этапе внутри таблицы создается набор описаний строк таблицы. Собственно, таблица состоит из множества строк. Каждая строка может иметь свои параметры, в частности высоту.

Строка таблицы создается с помощью парного тега <TR>. Высота строки задается атрибутом HEIGHT. Тег <TR> может присутствовать только внутри тега <TABLE>.

На третьем этапе внутри каждой строки создаются описания отдельных ячеек. Каждая ячейка также может иметь параметры, например, ширину и высоту. Каждая строка таблицы, таким образом, состоит из набора ячеек.

Ячейка таблицы создается с помощью парного тега <TD>, ячейка заголовка — с помощью тега <TH>. Ширина ячейки задается атрибутом WIDTH, а высота — атрибутом HEIGHT. Тег <TD> может присутствовать только внутри тега <TR>.

Последний, четвертый, этап — это помещение в ячейки полезного содержания. Нужно запомнить, что это содержимое может находиться только в ячейках таблицы (теги <TD> и <TH>), но никак не в строках (<TR>) и не в самой таблице (<TABLE>). Если мы нарушим это правило, HTML-код таблицы может отобразиться Web-обозревателем неверно.

Если мы задали для таблицы название, то оно помещается внутри парного тега <CAPTION>. Этот тег может находиться только внутри тега <TABLE>.

Все это может показаться очень сложным, но на самом деле ничего сложного здесь нет — наоборот, все очень просто, если понять принцип. Давайте сами в этом убедимся. Ниже приведен пример HTML-кода некой таблицы — он послужит нам хорошей иллюстрацией к ранее сказанному.

```
<TABLE>  
<CAPTION>
```

```
    Название таблицы
</CAPTION>
<TR>
  <TD>
    Содержимое ячейки
  </TD>
  <TD>
    Содержимое ячейки
  </TD>
</TR>
<TR>
  <TD>
    Содержимое ячейки
  </TD>
  <TD>
    Содержимое ячейки
  </TD>
</TR>
</TABLE>
```

Видно, что HTML-код простой таблицы достаточно прозрачен, т. е. ясен для понимания, так что его, в принципе, можно набрать вручную в Блокноте. Но, если нам нужно создать сложную таблицу с большим количеством строк, с вложенными в ее ячейки другими таблицами, тогда могут возникнуть трудности. В таких случаях лучше пользоваться для создания таблиц (хотя бы начерно) Dreamweaver.

Форматирование таблиц

Ну вот, как работать с таблицами, мы разобрались. Теперь пора изучить средства, предлагаемые HTML для форматирования таблиц. Но сначала нужно выяснить, как выделить в окне документа Dreamweaver нужный элемент нашей таблицы.

Выделение элементов таблиц

Обратимся к Dreamweaver, а именно к страничке passions.htm нашего сайта. Поместим текстовый курсор в одну из ячеек таблицы и посмотрим на секцию тегов. Мы увидим там что-то, похожее на

```
<BODY><TABLE><TR><TD><P>
```

Тега <P> может и не быть (это содержимое ячейки), а на тег <BODY> мы отвлекаться не будем. Обратим внимание на уже знакомые нам теги, задающие отдельные части таблицы.

Щелкнем мышью по тегу <TD>. Ячейка, где стоит текстовый курсор, будет выделена толстой черной рамкой. Теперь мы можем производить какие-либо манипуляции с этой ячейкой, в частности задать значения ее параметров (как это сделать, мы выясним потом). Далее щелкнем по тегу <TR> в секции тегов. В этом случае будет выделена вся строка, в которой находится текущая ячейка. Ну и нажатие на тег <TABLE> выделяет всю таблицу.

Собственно, когда мы просто помещаем текстовый курсор в ячейку таблицы, она уже становится выделенной. Так что нам необязательно для выделения ячейки пользоваться секцией тегов.

Мы можем также выделить одновременно несколько ячеек. Для этого щелкнем мышью на ячейке, где начнется наше выделение, и, не отпуская левой кнопки мыши, протащим ее, пока все нужные ячейки не будут выделены.

Теперь рассмотрим другие способы выделить таблицу, строку или целый столбец сразу.

Чтобы выделить строку таблицы, поместим курсор мыши точно на левую границу самой левой ячейки этой строки. Курсор при этом примет вид небольшой черной стрелки, направленной вправо, а граница всей строки будет подсвечена красным цветом. Если мы щелкнем в этот момент левой кнопкой мыши, строка, напротив которой стоит курсор, будет выделена. А если мы нажмем левую кнопку мыши и, не отпуская ее, протащим мышью по вертикали, то сможем выделить сразу несколько строк.

Столбец или несколько столбцов можно выделить точно так же, только для этого курсор мыши помещается на верхнюю границу самой верхней строки таблицы. При этом он принимает вид черной стрелки, направленной вниз.

Подобным же образом можно выделить и всю таблицу. Поместим курсор на один из углов таблицы и добьемся того, чтобы Dreamweaver подсветил границу всей таблицы красным цветом. После этого щелкаем мышью — и вся таблица выделена.

Внимание!

Если мы задали для таблицы название, нужно будет учесть, что это название входит в состав таблицы. И для того, чтобы выделить таблицу, нам придется мысленно продолжить ее границу так, чтобы она охватила и название, и "нащупывать" ее курсором именно там.

Для обладателей неуклюжих рук и нечищенных мышек разработчики Dreamweaver предоставили так называемый *режим отображения увеличенных таблиц*. Чтобы включить его, либо выберем пункт **Expanded Tables Mode** подменю **Tables** из меню **View** или аналогичного подменю контекстного меню, либо просто нажмем клавишу <F6>. После этого на экране появится небольшое окно-предупреждение: закроем его, нажав кнопку **OK**. И наша таблица примет такой вид, как на рис. 6.7.

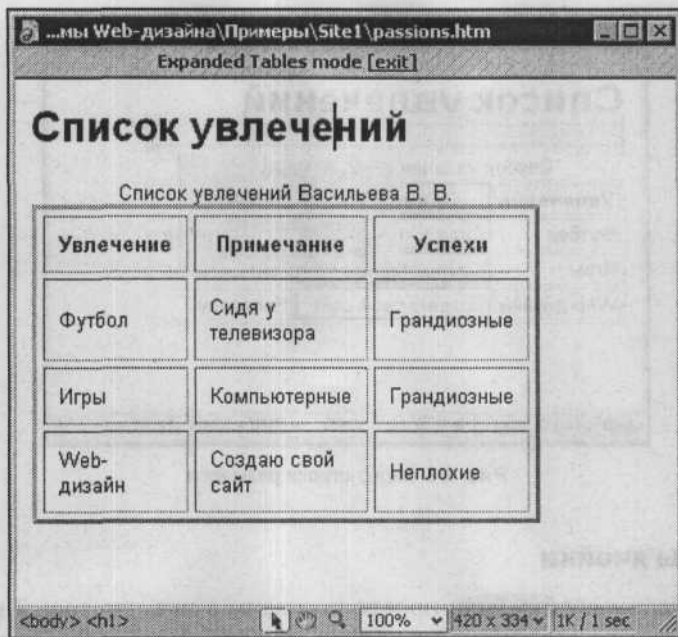


Рис. 6.7. Режим отображения увеличенных таблиц

Видно, что Dreamweaver увеличил расстояние между ячейками таблицы, чтобы нам стало проще позиционировать мышь на нужной границе. Чтобы вернуться в *режим отображения обычных таблиц*, либо выберем снова пункт **Expanded Tables Mode** подменю **Tables** из меню **View** или аналогичного подменю контекстного меню, либо нажмем клавишу <F6>.

Есть еще один, самый быстрый, способ выделить столбец и таблицу целиком (выделить таким образом строку, к сожалению, не получится). Для этого нужно воспользоваться *меню строки размеров* (рис. 6.8). Чтобы вызвать его, щелкнем мышью по значению ширины или, если оно отсутствует, по небольшой зеленой стрелке, направленной вниз, над (или под) нужным столбцом или таблицей. Для выделения столбца достаточно выбрать пункт **Select Column**, а для выделения таблицы — пункт **Select Table**.

Dreamweaver также позволяет нам быстро выделить всю таблицу. Для этого поставим текстовый курсор в любую ячейку таблицы и выберем пункт **Select Table** подменю **Table** меню **Modify** или контекстного меню.

Кстати, после выделения таблицы на ее границах мы увидим уже знакомые нам маркеры изменения размеров. С их помощью мы можем изменить размеры таблицы.

Что ж, выделять различные части таблицы мы научились. Пора выяснить, что мы с ними можем делать.

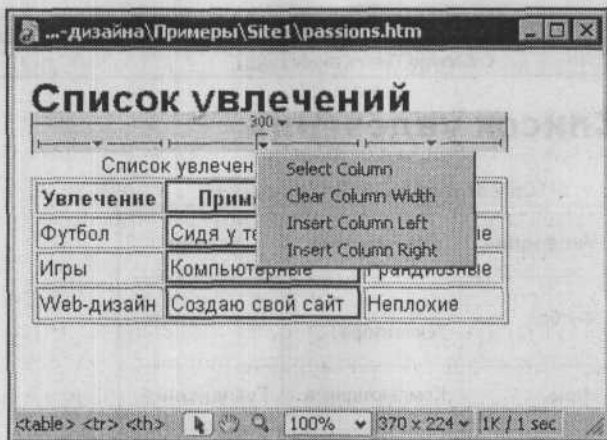


Рис. 6.8. Меню строки размеров

Параметры ячейки

Если мы выделим ячейку (или просто поставим в нее текстовый курсор), редактор свойств позволит нам задать ее параметры. Все необходимые для этого элементы управления будут показаны в нижней части редактора свойств (рис. 6.9).

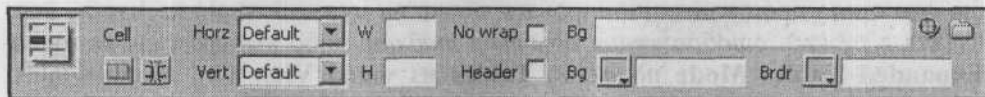


Рис. 6.9. Вид нижней части редактора свойств при выделении ячейке таблицы

Раскрывающийся список **Horz** позволяет задать горизонтальное выравнивание содержимого ячейки. Здесь доступны четыре пункта:

- ◆ **Default** — выравнивание по умолчанию, обычно по левому краю, в ячейках заголовка — по центру;
- ◆ **Left** — выравнивание по левому краю;
- ◆ **Center** — по центру;
- ◆ **Right** — по правому краю.

Аналогично раскрывающийся список **Vert** задает вертикальное выравнивание содержимого ячейки. Здесь доступно пять пунктов:

- ◆ **Default** — выравнивание по умолчанию, обычно посередине;
- ◆ **Top** — выравнивание по верху;
- ◆ **Middle** — посередине;

- ◆ **Bottom** — по низу;
- ◆ **Baseline** — по базовой линии.

В поля ввода **W** и **H** вводятся соответственно величины ширины и высоты ячейки. Мы можем задать их либо в пикселах, либо в процентах от ширины всей таблицы. Во втором случае нужно просто добавить после цифр знак процента. Чтобы Dreamweaver применил заданные значения к ячейке, нужно нажать клавишу <Enter>.

Если мы хотим удалить заданные нами ранее значения ширины или высоты, нам нужно будет очистить соответствующее поле ввода и нажать клавишу <Enter>. В этом случае управлять размерами ячейки будет Web-обозреватель, и он сделает все, чтобы содержимое ячеек гарантированно поместилось в них, а таблица заняла минимум места на экране. Для удаления значений ширины у всех ячеек столбца можно воспользоваться пунктом **Clear Column Width** меню строки размеров (см. рис. 6.8).

Флажок **No wrap** запрещает перенос строк внутри ячейки. В этом случае текст, содержащийся внутри ячейки, будет вытянут в одну строку, и ячейка растянется по ширине, чтобы вместить его. Если же этот флажок отключен, Web-обозреватель будет переносить текст в ячейке по строкам и растягивать ячейку по ширине только в самом крайнем случае, если в тексте встретится очень длинное слово.

Внимание!

Используйте флажок **No wrap** с большой осторожностью, иначе ваша таблица может оказаться столь широкой, что пользователю придется прокручивать ее по горизонтали. А слишком широкие таблицы, не помещающиеся в окне Web-обозревателя, — очень плохой стиль в Web-дизайне.

Флажок **Header** позволит превратить выделенную ячейку в ячейку заголовка таблицы. Используйте этот параметр только для ячеек первой строки таблицы, содержащей ее "шапку". Впрочем, Dreamweaver, как вы помните, позволяет сделать "шапку" и выделенный первый столбец уже при создании таблицы (собственно, мы уже воспользовались этой его возможностью, когда создавали нашу первую таблицу).

Поле ввода **Bg** служит для задания имени файла фонового изображения ячейки. Селектор цвета **Bg** (не путать с одноименным полем ввода!) позволяет задать цвет фона ячейки таблицы. Если ни то, ни другое не задано, будет использован фон по умолчанию (обычно — белый без всяких фоновых рисунков).

Селектор цвета **Brdr** позволяет задать цвет рамки ячейки.

Параметры строки

Для строк предлагается такой же набор параметров, как и для ячеек (см. рис. 6.9). Естественно, все эти параметры применяются не к отдельной ячейке, а ко всей строке. Мы не будем их подробно описывать, сделаем только несколько замечаний.

- ◆ Если ввести в поле ввода **W** какое-либо значение ширины и нажать клавишу <Enter>, данное значение будет применено ко всем ячейкам таблицы. Используйте эту особенность, если хотите создать таблицу с ячейками одинаковой ширины. Но если вы зададите некорректное значение (например, 50 % для таблицы из десяти столбцов), Web-обозреватель проигнорирует ваши установки и задаст ширину ячеек сам.
- ◆ Если включить флажок **No wrap** (запрет переноса строк), ваша таблица может расширяться до совершенно невообразимых пределов. Хорошо подумайте, прежде чем включить этот флажок.
- ◆ Если вы задали фоновое изображение (поле ввода **Bg**), цвет фона (селектор цвета **Bg**) или цвет рамки (селектор цвета **Brdr**), значения этих параметров будут применены сразу ко всей строке.

Здесь нужно заметить вот что. Дело в том, что язык HTML позволяет задавать фоновое изображение, цвет фона и цвет рамки как для всей строки, так и для отдельной ее ячейки. В таком случае параметры ячейки перекроют параметры строки. То есть, если мы зададим для строки серый фон, а для одной из ее ячеек — белый, эта ячейка будет действительно иметь белый фон. Это прекрасно видно на рис. 6.10.

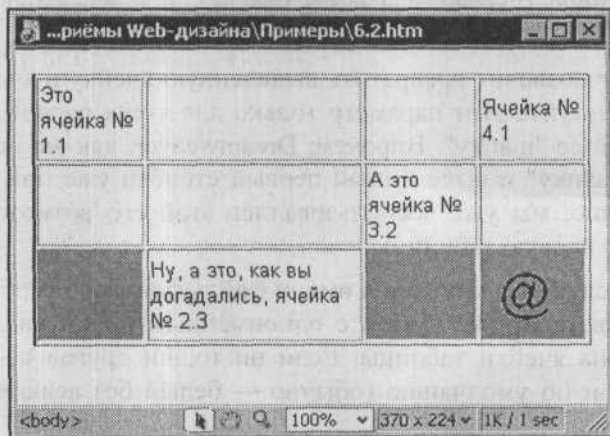


Рис. 6.10. Результат установки разных цветов фона для всей нижней строки и ячейки № 2.3

Параметры таблицы

Вид редактора свойств при выделенной таблице представлен на рис. 6.11. Набор параметров здесь сильно отличается от набора параметров ячеек и строк. И это не удивительно.

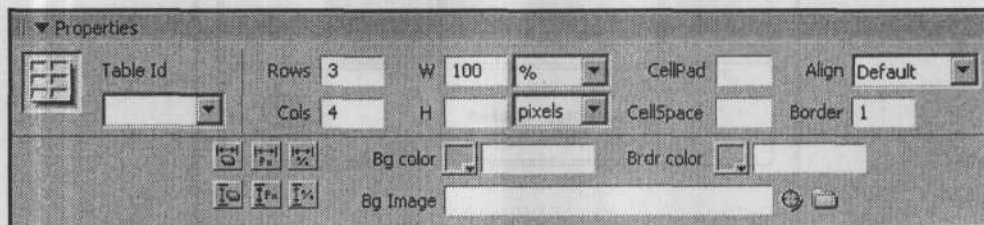


Рис. 6.11. Вид редактора свойств при выделенной таблице

Сначала поговорим о знакомом.

Поля ввода **Rows** и **Cols** служат для быстрого изменения количества соответственно строк и столбцов таблицы.

Внимание!

Если вы задали количество строк или столбцов меньше существующего, лишние строки или столбцы будут удалены без предупреждения.

Поля ввода **W** и **H** служат для задания соответственно ширины и высоты таблицы. Эти значения могут быть заданы как в пикселах, так и в процентах относительно ширины или высоты Web-страницы (или родителя — ячейки внешней таблицы). И не забудем выбрать в раскрывающихся списках, находящихся справа от этих полей ввода, пункт **pixels** (пиксели) или **%**, а после ввода значения ширины и высоты — нажать клавишу <Enter>, чтобы Dreamweaver сразу же применил их.

Если нам нужно удалить заданные для таблицы значения ширины и высоты, удалим их из этих полей ввода и нажмем клавишу <Enter>. Тогда размерами таблицы будет управлять Web-обозреватель.

Поле ввода **Bg Image** служит для задания имени файла фонового изображения для всей таблицы. Селектор цветов **Bg color** задает цвет фона таблицы. А селектор цветов **Brdr color** задает цвет рамки таблицы.

Опять же, перечисленные выше параметры таблицы перекрываются аналогичными параметрами строки, а те, в свою очередь, — параметрами ячейки. Посмотрим на рис. 6.12 — там для ячейки № 3.2 задан светло-серый фон, для ячейки № 4.1 — белая граница, для всей таблицы — белый фон и черная граница.

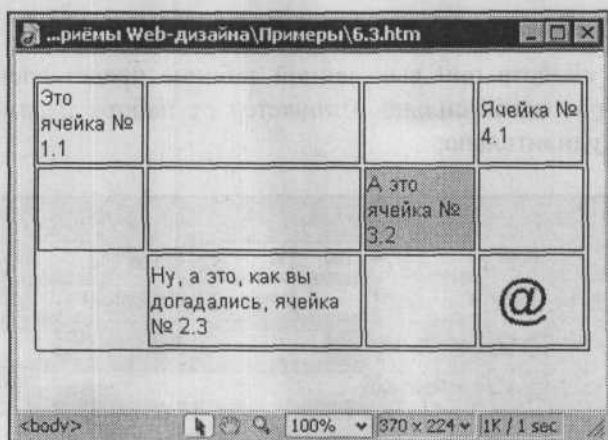


Рис. 6.12. Перекрытие параметров таблицы параметрами ячеек

Поле ввода **CellPad** служит для задания отступа между границей ячейки и ее содержимым в пикселах. Поле ввода **CellSpace** задает отступ между границами находящихся рядом ячеек и тоже в пикселах. В поле ввода **Border** устанавливается толщина рамки ячейки в пикселах. Все эти параметры знакомы нам по диалоговому окну **Table** (см. рис. 6.1).



Теперь рассмотрим параметры, с которыми мы еще не встречались, а если и встречались, то в приложении к другим элементам страницы.

Раскрывающийся список **Align**, как ясно из его названия, служит для задания горизонтального выравнивания таблицы. В нем доступны четыре пункта:



- ◆ **Default** — выравнивание по умолчанию, обычно влево;
- ◆ **Left** — выравнивание влево;
- ◆ **Center** — по центру;
- ◆ **Right** — вправо.



Заметим, что с помощью этого списка задается не выравнивание содержимого таблицы, а выравнивание самой таблицы. То есть будет ли таблица прижата к левому краю окна или центрирована в нем.

В левом нижнем углу редактора свойств есть группа из шести кнопок, управляющих значениями ширины и высоты ячеек и строк таблицы. Давайте выясним, что они делают.

Кнопка  удаляет все заданные для ячеек таблицы и самой таблицы значения ширины, а кнопка  — все заданные для ячеек и таблиц значения высоты. Ту же задачу выполняют пункты **Clear Cell Heights** (стереть значения

высоты ячеек) и **Clear Cell Widths** (стереть значения ширины ячеек) подменю **Table** меню **Modify**.

Кнопка  позволяет преобразовать относительные значения ширины ячеек (заданные в процентах) в фиксированные (заданные в пикселах). А кнопка  преобразовать относительные значения высоты строк (заданные в процентах) в фиксированные (заданные в пикселах). Аналогичного эффекта можно достичь, выбрав пункты **Convert Widths to Pixels** (преобразовать значения ширины в пиксели) или **Convert Heights to Percents** (преобразовать значения высоты в пиксели) подменю **Table** меню **Modify**.

Кнопка  выполняет обратное преобразование значений ширины ячеек — из фиксированных (пиксели) в относительные (проценты). А кнопка  прodelьвает то же самое со значениями высоты строк. Также можно воспользоваться пунктами **Convert Widths to Percent** (преобразовать значения ширины в проценты) или **Convert Heights to Percent** (преобразовать значения высоты в проценты) подменю **Table** меню **Modify**.

Объединение ячеек таблиц

А теперь, научившись работать с таблицами и их составными частями, поговорим об одной особенности языка HTML. Это так называемое *объединение ячеек* таблиц. Что это такое, проще всего объяснить на примере.

Предположим, мы создали Web-страницу с простой таблицей (рис. 6.13). Обычная таблица, какие мы уже умеем создавать в Dreamweaver — простейший набор строк и ячеек, без названия и даже без форматирования. Ячейки этой таблицы содержат их порядковые номера — так нам будет проще с ней работать.

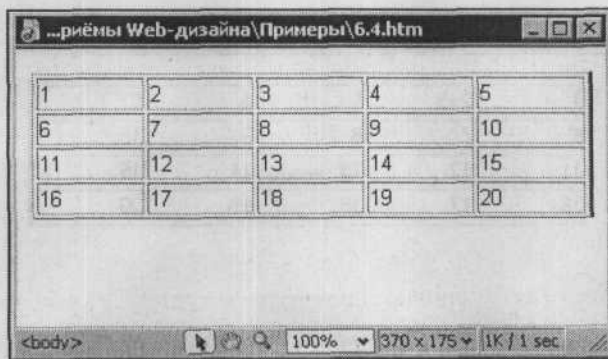


Рис. 6.13. Простая таблица


Все это очень просто и прекрасно нам знакомо. Теперь давайте рассмотрим более сложную таблицу (рис. 6.14).

1+6	2+3		4+5	
	7	8	9	10
11	12+13+14+15			
16	17	18	19	20

Рис. 6.14. Более сложная таблица

Здесь видно, что некоторые ячейки объединены в одну (об этом говорит знак "плюс" между их номерами). Такой прием и называется объединением ячеек. Причем ячейки могут быть объединены по горизонтали и по вертикали, а количество объединяемых ячеек не ограничено.

Но как это сделать на Web-странице? Сейчас мы это выясним. И для начала объединим ячейки с номерами 2 и 3.

Прежде чем объединять ячейки таблицы, их нужно выделить. Как это делается, мы уже знаем. После этого обратим внимание на редактор свойств. В его левом нижнем углу находится кнопка , с помощью которой и производится объединение ячеек. Нажмем ее. На рис. 6.15 показано, что получится в этом случае.

Вместо того чтобы нажимать эту кнопку, мы можем выбрать пункт **Merge Cells** подменю **Table** меню **Modify** или контекстного меню либо нажать комбинацию клавиш <Ctrl>+<Alt>+<M>.

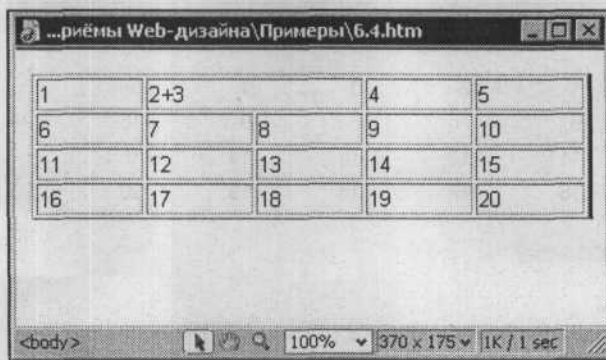




Рис. 6.15. Таблица, редактируемая после объединения ячеек 2 и 3

Теперь поместим текстовый курсор в ячейку 4. Мы объединим ее с ячейкой 5, но другим способом. Выберем пункт **Increase Column Span** подменю **Table** меню **Modify** или контекстного меню или нажмем комбинацию клавиш <Ctrl>+<Shift>+<J>. Ячейки 4 и 5 сольются в одну. Это еще один способ объединения ячеек по горизонтали, самый быстрый, но позволяющий объединить за один раз только две ячейки.

Остается объединить четыре ячейки — 12, 13, 14 и 15 — в одну. Выполним это любым из только что рассмотренных способов.

Итак, мы узнали, как выполняется объединение ячеек по горизонтали. Осталось рассмотреть, как ячейки таблицы объединяются по вертикали. В частности, нам нужно объединить ячейки 1 и 6. Опять же, мы можем выделить их и нажать кнопку объединения . Также мы можем поставить текстовый курсор в ячейку 1 и выбрать пункт **Increase Row Span** подменю **Table** меню **Modify** или контекстного меню. Так и сделаем.

Но что делать, если мы хотим разъединить объединенные ячейки? Dreamweaver и здесь идет нам на помощь. Достаточно поместить текстовый курсор в ячейку, получившуюся в результате объединения, и сделать следующее:

- ◆ если выполнялось объединение по горизонтали, выбрать пункт **Decrease Column Span** подменю **Table** меню **Modify** или контекстного меню либо нажать комбинацию клавиш <Ctrl>+<Shift>+<I>;
- ◆ если выполнялось объединение по вертикали, выбрать пункт **Decrease Row Span** подменю **Table** меню **Modify** или контекстного меню;
- ◆ и, наконец, универсальный способ — нажать кнопку , находящуюся в левом нижнем углу редактора свойств.

Объединение ячеек задается с помощью атрибутов COLSPAN и ROWSPAN тега <TD>. Атрибут COLSPAN задает объединение ячеек по горизонтали, а атрибут ROWSPAN — по вертикали. Количество объединяемых ячеек задается в качестве значения этих атрибутов.

Вообще, объединение ячеек — штука мудреная. Поэтому будет лучше, если мы рассмотрим HTML-код нашей таблицы. Вот он:

```
<TABLE WIDTH="100%" BORDER="1">
  <TR>
    <!-- Определяем ячейки первой строки -->
    <!-- Объединяем ячейки 1 и 6 по вертикали -->
    <TD ROWSPAN="2">1+6</TD>
    <!-- Объединяем ячейки 2 и 3 по горизонтали -->
    <TD COLSPAN="2">2+3</TD>
```

```

<!-- Объединяем ячейки 4 и 5 по горизонтали -->
  <TD COLSPAN="2">4+5</TD>
</TR>
<TR>
<!-- Определяем ячейки второй строки. Заметим, что здесь нет
определения ячейки 6, т. к. она уже объединена с первой
(см. определение ячеек первой строки). -->
  <TD>7</TD>
  <TD>8</TD>
  <TD>9</TD>
  <TD>10</TD>
</TR>
<TR>
<!-- Это третья строка. Здесь всего две ячейки: 11 и большая
ячейка, образованная объединением ячеек 12-15. -->
  <TD>11</TD>
  <TD COLSPAN="4">12+13+14+15</TD>
</TR>
<TR>
<!-- Последняя строка самая простая. Здесь вообще нет никаких
объединений -->
  <TD>16</TD>
  <TD>17</TD>
  <TD>18</TD>
  <TD>19</TD>
  <TD>20</TD>
</TR>
</TABLE>

```

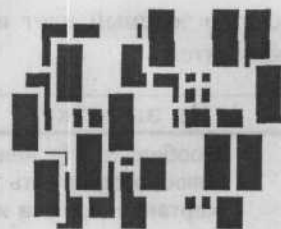
Вот и все об объединении ячеек. Да и о таблицах, пожалуй, тоже рассказывать больше нечего.

На следующем занятии...

Ох и таблицы в HTML!.. Мудреные. Но и мощные!

Но есть вещицы и помощнее. На следующем занятии мы будем говорить об одной из таких могучих вещиц. Это каскадные таблицы стилей. Они формально не относятся к языку HTML, но "работают", так сказать, в тесном контакте с ним. И они такое позволяют сделать со страницами... Ну да сами все узнаете.

ЗАНЯТИЕ 7



Использование стилей CSS

На занятиях 5 и 6 мы поговорили о том, как поместить на Web-страницу графическое изображение и таблицу. И практически закончили две страницы нашего сайта — сведения о создателе сайта, студенте-бездельнике Васильеве, за которого мы этот самый сайт делаем, и список его увлечений. Доделка остальных страниц сайта — не проблема, ведь мы уже знаем все, чтобы наполнить их содержимым. (Проблема, правда, есть — каким именно содержимым их наполнить, но этому посвящены другие книги...)

Ладно, допустим, что наш Web-сайт готов. И мы сидим у компьютера в глубокой задумчивости — что с ним делать? Можно, конечно, опубликовать его в Интернете (мы это уже делали на занятии 4). А можно еще немного над ним поработать.

Давайте посмотрим на страницы нашего сайта. Конечно, выглядят они неплохо, но как-то уж слишком серо. Мы ведь попутешествовали по Интернету, насмотрелись на потрясающе красивые сайты, с которых прямо-таки не хотелось уходить, и неужели теперь будем выкладывать в Сеть такое убожество! Нет, нет и еще раз нет!

Понятие о стилях CSS

Да, но есть ли в языке HTML какие-нибудь средства для оформления Web-страниц? Что-то позатейливее, чем разноцветный, выделенный полужирным шрифтом и курсивом текст, шесть уровней заголовка, списки и отступы.

И да, и нет. А почему — сейчас выясним.

Сам язык HTML содержит совсем немного средств для оформления текста. Это уже знакомые нам теги ``, `<EMP>`, `<BLOCKQUOTE>` и пр. Негусто, хотя

текст в зеленый цвет или выделить его другим шрифтом с их помощью не получится.

На заметку

Вообще-то это неверно. Стандарт HTML определяет несколько тегов, позволяющих оформить текст. Так, с помощью парного тега `` можно задать начертание шрифта и цвет текста. Но все эти теги, в том числе и ``, объявлены устаревшими, и применять их во вновь создаваемых Web-страницах не рекомендуется.

Стоп! Но мы ведь как-то выделяли цветом фрагмент текста нашей первой Web-страницы `default.htm`. И все получалось!!! Почему?!

А все дело в том, что для подобного форматирования текста использовался отнюдь не язык HTML. (Если мы откроем страницу `default.htm` и переключимся в режим отображения HTML-кода, то ясно это увидим.) Для форматирования текста, когда бывает недостаточно возможностей HTML, Dreamweaver использует каскадные таблицы стилей.

Каскадные таблицы стилей (Cascading Style Sheets, CSS) или просто *таблицы стилей* — это набор правил, описывающих форматирование текста и других элементов Web-страниц и хранящихся отдельно от него. Одно такое правило, описывающее форматирование какого-либо фрагмента или целой группы фрагментов страницы, называется *стилем*.

Для создания таблиц стилей используется особый язык, не похожий на HTML, который так и называется — CSS. В процессе изучения стилей мы с ним кратко познакомимся.

Таблицы стилей часто хранятся в отдельных файлах с расширением `css` (*внешние таблицы стилей*), но могут быть сохранены и в файле самой Web-страницы (*внутренние таблицы стилей*). Очевидно, что таблицы стилей, используемые всеми страницами сайта, лучше всего хранить в отдельном файле, т. е. делать внешними. Если же какие-то стили используются только в одной конкретной странице, их стоит сохранить во внутренней таблице стилей, прямо в файле использующей их Web-страницы.

Таблицы стилей — вещь очень мощная. С ее помощью мы можем проделывать такое, что не снилось HTML с его убогим набором тегов. Мы можем задавать для абзацев фоновый цвет, создавать отступы и рамки, даже располагать фрагменты текста на странице произвольно, как нам вздумается. (На этом, последнем свойстве таблиц стилей основан так называемый контейнерный Web-дизайн, о котором мы поговорим на *занятиях 11 и 12*.) Ну разве это не замечательно?

Кроме того, таблицы стилей имеют еще одно достоинство. Если таблица стилей, описывающая оформление всего сайта, хранится в отдельном файле, мы

сто исправив эту таблицу стилей — один-единственный файл! Согласитесь — это много проще и легче, чем перелопачивать все страницы сайта, которых может быть и десятки, и сотни...

Ладно, хватит восторгов! Пора браться за дело. Давайте создадим таблицу стилей и сохраним ее в отдельном файле `styles.css`. Эту таблицу и хранящиеся в ней стили мы используем, чтобы немного разукрасить скучный текст наших Web-страничек.

И начнем мы с заголовков страниц. В самом деле, почему они выровнены по левому краю! Неужели их нельзя отцентрировать!

Создание и использование стилей CSS

Откроем в Dreamweaver страницу `default.htm` — главную страницу нашего сайта. Поставим текстовый курсор на заголовок и с помощью редактора свойств зададим для него выравнивание по умолчанию. Для этого будет достаточно "отжать" все кнопки-переключатели задания выравнивания, описанные на занятии 3.

Вся работа со стилями протекает в панели **CSS Styles** Dreamweaver, показанной на рис. 7.1. Чтобы вызвать ее на экран, либо выберем в меню **Window** пункт **CSS Styles**, либо нажмем комбинацию клавиш `<Shift>+<F11>`. Сразу же после этого найдем в верхней части этой панели кнопку-переключатель **All** и включим ее.

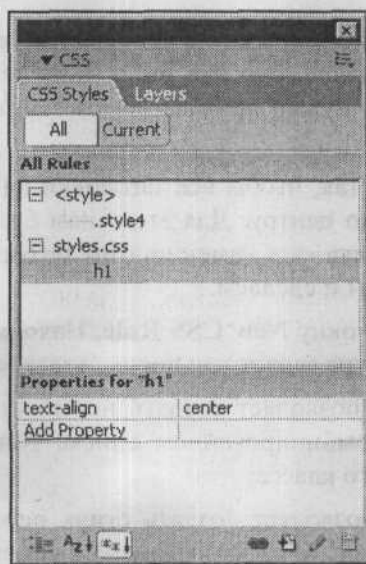


Рис. 7.1. Панель CSS Styles

Итак, нам нужно создать новый стиль, который заставит заголовок Web-страницы выравниваться по центру. Выберем пункт **New** контекстного или дополнительного меню панели **CSS Styles** или нажмем одноименную кнопку (☰), расположенную в правом нижнем углу этой же панели. На экране появится диалоговое окно **New CSS Rule**, показанное на рис. 7.2. Об этом окне нам нужно поговорить подробнее.

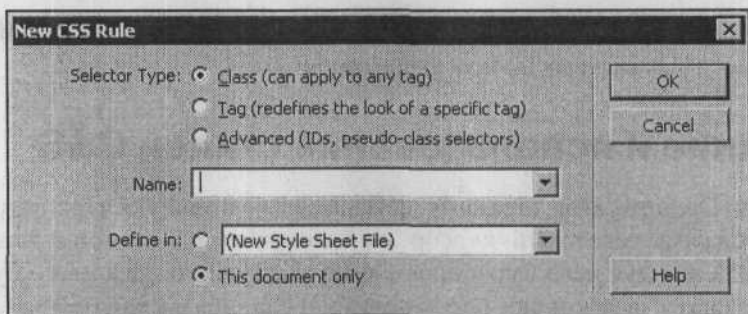


Рис. 7.2. Диалоговое окно **New CSS Rule**

Нужно сразу сказать, что стили CSS бывают разных типов. Одни стили могут быть привязаны к любому фрагменту страницы и носят название *стилевых классов*. Такие стили должны иметь имя, уникальное в пределах таблицы стилей. Это имя используется для привязки стилового класса к какому-либо тегу.

Другие же стили, наоборот, жестко привязываются к определенным тегам и называются *стилями переопределения тега*. Имя такого стиля совпадает с именем тега, к которому нужно привязать этот стиль, но не включает знаки `<` и `>`.

Итак, мы хотим сделать так, чтобы все заголовки на всех страницах нашего сайта были выровнены по центру. Для этого нам будет проще всего создать стиль переопределения тега `<h1>` (именно этим тегом выделены заголовки на наших страницах). Так мы и сделаем.

Вернемся к диалоговому окну **New CSS Rule**. Находящаяся в нем группа переключателей **Selector Type** задает тип вновь создаваемого стиля:

- ◆ переключатель **Class** позволяет создать новый стиливой класс. Если мы его включим, то в комбинированном списке **Name** сможем ввести имя создаваемого стилового класса;
- ◆ переключатель **Tag** позволяет создать стиль переопределения тега. Выбрав этот переключатель, в раскрывающемся списке **Tag** мы сможем отыскать нужный тег;

- ◆ переключатель **Advanced** позволяет создать более сложные стили, о которых мы поговорим потом.

Вторая группа переключателей **Define in** этого окна задает, в какую таблицу стилей будет помещен создаваемый стиль:

- ◆ если выбран верхний переключатель, стиль будет сохранен во внешней таблице стилей. Имя файла этой таблицы стилей выбирается в расположенном правее раскрывающемся списке. Если же внешняя таблица стилей пока не существует, следует выбрать пункт (**New Style Sheet File**), и Dreamweaver сам ее создаст. В последнем случае после нажатия кнопки **OK** на экране появится диалоговое окно сохранения файла, где мы сможем ввести имя файла создаваемой таблицы стилей;
- ◆ если выбран переключатель **This document only**, стиль будет помещен во внутреннюю таблицу стилей этой Web-страницы.

Как обычно, введя нужные данные, необходимо нажать кнопку **OK** или **Cancel**. Первая кнопка выполняет создание стиля, вторая — отменяет.

Мы хотим создать стиль переопределения тега <n1> и поместить определение этого стиля в пока еще не существующую внешнюю таблицу стилей styles.css. Сначала выберем переключатель **Tag**, затем — пункт **h1** в раскрывающемся списке **Tag**, далее — верхний переключатель в группе **Define in**, проследим, чтобы в раскрывающемся списке правее этого переключателя был выбран пункт (**New Style Sheet File**), и нажмем кнопку **OK**. В диалоговом окне сохранения файла выберем папку нашего сайта, введем имя файла создаваемой таблицы стилей styles.css и нажмем кнопку сохранения. Все, сам файл внешней таблицы стилей нами создан.

После создания таблицы стилей на экране появится диалоговое окно **CSS Rule Definition**. Оно очень велико и разделено на множество категорий, которые выбираются в списке **Category** и которые мы рассмотрим потом. Сейчас же нам нужно как можно быстрее получить новый стиль, так что не будем отвлекаться на излишние подробности.

Нам нужно задать выравнивание для нового стиля. Переключимся в категорию **Block** окна **CSS Rule Definition** и выберем в раскрывающемся списке **Text align** пункт **center**. И не забудем нажать кнопку **OK**.

После этого мы заметим три вещи:

- ◆ заголовок Web-страницы default.htm оказался выровненным по центру;
- ◆ Dreamweaver открыл еще одно окно документа, содержащее CSS-код созданной нами таблицы стилей;
- ◆ в иерархическом списке уже созданных стилей, занимающем верхнюю

(Если вы не видите этот список, проверьте, включена ли кнопка-переключатель **All**, находящаяся в верхней части панели, над всеми списками.) Они означают, соответственно, созданную нами внешней таблицей стилей и находящийся в ней единственный стиль переопределения тега.

Теперь страница `default.htm` щеголяет отцентрированным заголовком, но как же другие страницы? Давайте откроем одну из них, скажем, `passions.htm`. Что такое — ее заголовок выровнен по левому краю (т. е. задано выравнивание по умолчанию). Почему?

Да, мы создали внешнюю таблицу стилей. Но это только полдела. Нам еще нужно привязать ее к Web-странице, которая будет ее использовать. К странице `default.htm` привязывать ее не нужно — Dreamweaver сам сделал это за нас. Но вот привязку к другим страницам нашего сайта придется выполнять нам самим.

Сохраним только что созданную таблицу стилей, переключившись на окно документа, в котором она открыта, и нажав комбинацию клавиш `<Ctrl>+<S>`. После этого переключимся на окно со страницей `passions.htm` и посмотрим на панель **CSS Styles**. В нижней ее части имеется кнопка **Attach Style Sheet** (📎). Она-то нам и нужна. Нажмем ее, и на экране появится диалоговое окно **Attach External Style Sheet**, показанное на рис. 7.3. (Мы также можем воспользоваться пунктом **Attach Style Sheet** контекстного или дополнительного меню панели **CSS Styles**.)

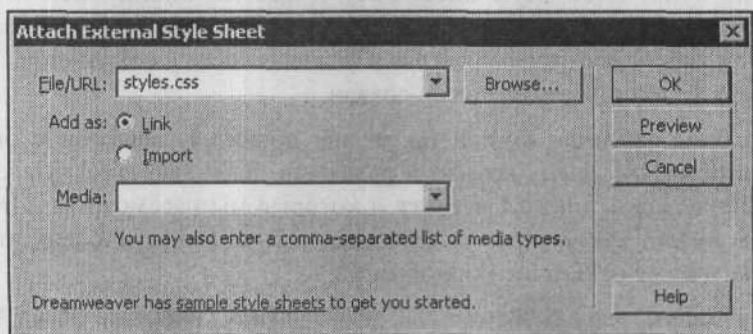


Рис. 7.3. Диалоговое окно **Attach External Style Sheet**

Нажмем кнопку **Browse** этого окна, в появившемся на экране диалоговом окне **Select File** выберем файл нашей таблицы стилей и нажмем кнопку открытия. После этого имя выбранного файла появится в поле ввода **File/URL**. Включим переключатель **Link**, чтобы привязать эту таблицу стилей к Web-странице, и нажмем кнопку **OK**. (Переключатель **Import** заставляет Dreamweaver поместить все определенные во внешней таблице стили во внутрен-

Замечательно! Как только Dreamweaver по нашему указанию привязал таблицу стилей к странице `passions.htm`, все определенные в этой таблице стили автоматически применяются к этой странице. И заголовок ее сам выравнивается по центру.

Теперь переключимся на страницу `default.htm` и еще поэкспериментируем со стилями. Давайте создадим внутреннюю таблицу стилей, содержащую единственный стиль, который будет окрашивать текст в зеленый цвет. Это будет стилевой класс, который мы сможем привязать к любому тегу.

Снова нажмем кнопку **New CSS Rule** панели **CSS Styles**. В диалоговом окне **New CSS Rule** включим переключатель **Class** и введем в комбинированном списке **Name** имя нашего стилевого класса — `.green`. Заметьте, что имя стилевого класса обязательно должно начинаться с точки! После этого включим переключатель **This document only** в группе **Define in** и нажмем кнопку **OK**.

Когда на экране появится диалоговое окно **CSS Rule Definition**, переключимся в категорию **Text** и выберем в селекторе цвета **Color** зеленый цвет. После этого нажмем кнопку **OK**.

Мы сразу же заметим, что в списке стилей панели **CSS Styles** появится еще одна "ветвь" `<style>` с пунктом `.green`. Ветвь `<style>` обозначает внутреннюю таблицу стилей, созданную в Web-странице, что открыта в активном окне документа. Ну, а что обозначает пункт `.green`, вряд ли стоит объяснять.

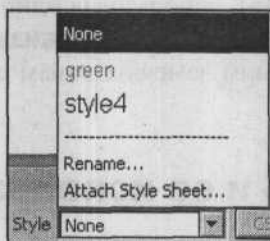


Рис. 7.4. Раскрывающийся список **Style** редактора свойств

Теперь нам нужно привязать только что созданный стиль к какому-либо тегу страницы `default.htm`. Давайте привяжем его к заголовку страницы и посмотрим, что получилось. Поставим на заголовок текстовый курсор и:

- ◆ либо выберем этот стиль в раскрывающемся списке **Style** редактора свойств (рис. 7.4);
- ◆ либо выберем нужный стиль в подменю **CSS Style** меню **Text** или контекстного меню;
- ◆ либо щелкнем правой кнопкой мыши по нужной кнопке секции тегов в строке статуса окна документа и выберем нужный стиль в подменю **Set**

- ◆ либо выделим в списке стилей панели **CSS Styles** соответствующий нужному стилю пункт и выберем пункт **Apply** контекстного или дополнительного меню этой панели.

Кстати, в раскрывающемся списке **Style** редактора свойств (см. рис. 7.4), как и в подменю **CSS Style** меню **Text** и контекстного меню, есть пункт **None**. Этот пункт убирает привязку любого стиля к данному тегу, что может нам потом пригодиться.

Вы уже, наверно, заметили, что в "ветви" `<style>` списка стилей панели **CSS Styles** есть пункт `.style4` (так он называется у автора книги; у вас его имя может отличаться). Выходит, внутренняя таблица стилей страницы `default.htm` содержит еще некий стилевой класс `style4`. Откуда он взялся (ведь мы его не создавали) и что делает? Неужели самодеятельность Dreamweaver?

Можно сказать — да. Помните, на *занятии 3* мы, когда набирали текст первой нашей Web-страницы — как раз `default.htm`, выделили первый ее абзац (не считая заголовка) увеличенным шрифтом? Так вот, чтобы это сделать, Dreamweaver сам создал стиль `style4`, поместил его во внутреннюю таблицу стилей и привязал его к тегу `<p>`, которым выделен первый абзац. Так что Dreamweaver сам использует стили CSS для форматирования Web-страниц, не спрашивая нашего разрешения.

Но давайте все-таки привяжем только что созданный стиль `green` к заголовку страницы `default.htm` (тег `<h1>`), используя один из описанных выше способов. Что мы получим? Правильно, как и ожидалось, заголовок позеленел. А мы столкнулись с еще одним замечательным свойством таблиц стилей — каскадностью.

Каскадность CSS и ее использование

Каскадность стилей CSS — это свойство разных стилей накладываться друг на друга и складываться друг с другом. После всех этих наложений и сложений результирующий стиль, примененный Web-обозревателем к какому-либо тегу, является комбинацией нескольких стилей, определенных в разных таблицах стилей.

Все это проще объяснить на примере. Ранее мы определили во внешней таблице стиль переопределения тега `<h1>`. После этого уже во внутренней таблице стилей мы создали стилевой класс `green`, окрашивающий текст в зеленый цвет, и привязали его к тегу `<h1>`. В результате, как мы уже заметили, заголовок первого уровня (содержимое тега `<h1>`) будет выровнен по центру и набран зеленым шрифтом. То есть Web-обозреватель (как и Dreamweaver) просто сложил два стиля, определенные в разных таблицах стилей.

Это и есть каскадность. Собственно, именно поэтому таблицы стилей и называются "каскадными".

А теперь рассмотрим пример посложнее. Предположим, что во внутренней таблице стилей мы создали еще один стиль переопределения все того же тега `<n1>` и задали для него выравнивание по правому краю. Возник, как говорят профессиональные Web-дизайнеры, *конфликт стилей* — два стиля определяют один и тот же параметр абзаца. Как справится Web-обозреватель с этим?

Чтобы разобраться со всей этой мешаниной стилей, Web-обозреватель следует нескольким очень простым правилам. Давайте их перечислим.

1. Стилиевой класс имеет приоритет перед стилем переопределения тега. То есть параметры, определенные в стилиевом классе, перекрывают такие же параметры, но определенные в стиле переопределения тега.
2. Стиль, определенный во внутренней таблице стилей, имеет приоритет перед стилем из внешней таблицы стилей.
3. Стиль, определенный в таблице стилей позже, имеет приоритет перед стилем, определенным в той же таблице стилей, но раньше. То есть "поздние" стили перекрывают "ранние".

Правил всего три, и они, на первый взгляд, сложны. Но это не так. Все их можно выразить одной фразой: "более частные и позже определенные стили имеют приоритет перед более общими и определенными раньше". А для любителей кратких и точных правил можно сказать проще: "своя рубашка ближе к телу".

Если кто не понял, давайте разберем эти правила на примере нашего сайта.

- ◆ Стиль переопределения тега `<n1>`, определенный во внешней таблице стилей, считается общим, т. к. он применяется сразу ко множеству фрагментов разных Web-страниц, поскольку переопределяет сам тег.
- ◆ Стиль переопределения тега `<n1>`, определенный во внутренней таблице стилей, считается более частным, т. к. он применяется только к конкретной Web-странице. Это, можно сказать, более "близкая к телу рубашка". Значит, он будет иметь приоритет над тем же стилем переопределения тега, но из внешней таблицы стилей.
- ◆ Стилиевой класс `green` из внутренней таблицы стилей считается самым частным из этих трех стилей. Поэтому он будет иметь приоритет над обоими стилями переопределения тега.

Вообще, конечно, лучше всего поэкспериментировать с разными стилями самим. Пусть это будет вашим домашним заданием.

Каскадность, вообще, отличная штука. Разобравшись с ее правилами, мы можем поместить более, скажем так, общие стили во внешнюю таблицу, а более конкретные, переопределяющие их, — во внутреннюю. Так мы сократим объем наших таблиц стилей до минимума.

Краткое введение в язык CSS

Ну что, пора посмотреть, как выглядит таблица стилей. Давайте переключимся на окно документа, в котором открыта внешняя таблица стилей, и посмотрим на нее. (Если окно с таблицей стиля уже закрыто, открыть его можно, просто дважды щелкнув по файлу с этой таблицей стиля, названному в списке панели **Files**.) Вот что мы увидим:

```
h1 {  
  text-align: center;  
}
```

Да, это и есть наша небольшая таблица стилей. Она содержит только один стиль переопределения тега `<h1>`. Давайте его рассмотрим.

По правилам языка CSS, каждый стиль должен состоять из имени и описания. *Имя* однозначно идентифицирует стиль и в нашем случае совпадает с названием тега. *Описание* стиля записывается после имени через пробел в фигурных скобках и содержит набор *атрибутов стиля* и их значений. Между атрибутом и его значением должен стоять знак двоеточия, а пары "атрибут-значение" отделяются друг от друга знаком точки с запятой. Описание нашего единственного стиля содержит единственный атрибут `text-align` (выравнивание текста в абзаце), которому присвоено значение `center` (по центру).

Привязка внешней таблицы стиля к Web-странице выполняется с помощью одинарного тега `<LINK>`:

```
<LINK HREF="<Имя таблицы стилей>" REL="stylesheet" TYPE="text/css">
```

Имя файла внешней таблицы стилей указывается в качестве значения атрибута `HREF` этого тега. Остальные атрибуты носят чисто служебный характер, и мы их рассматривать не будем. Сам же тег `<LINK>` помещается, опять же, в секцию заголовка Web-страницы.

Теперь давайте посмотрим на внутреннюю таблицу стилей, созданную в Web-странице `default.htm`. Выберем окно документа, в котором открыта эта страница, переключимся в режим редактирования HTML-кода и найдем в секции заголовка страницы (внутри тега `<HEAD>`) парный тег `<STYLE>`. Вот он:

```
<STYLE TYPE="text/css">
```

```
.green {
  color: #00FF00;
}
</STYLE>
```

Видно, что код CSS, образующий внутреннюю таблицу стилей, помещается в парный тег `<STYLE>`, находящийся в секции заголовка страницы. А в остальном все здесь уже нам знакомо. Атрибут `font-size` в определении стиля `style4` задает размер шрифта, которым набран текст. Его значение равно `larger` и задает увеличенный шрифт. А атрибут `color` в определении стиля `green` задает цвет текста.

Для привязки стилового класса к тегу служит особый атрибут `CLASS`, поддерживаемый практически всеми тегами. Имя стилового класса указывается в качестве значения этого атрибута:

```
<H1 CLASS="green">Сайт студента Васильева В. В.</H1>
```

Есть еще одна разновидность стилей CSS — так называемые *встроенные* или *внедренные* стили. Эти стили не имеют имени, а описание их задается прямо в нужном теге с помощью атрибута `STYLE`:

```
<H1 STYLE="color: #00FF00;">Сайт студента Васильева В. В.</H1>
```

Атрибут `STYLE` также поддерживается абсолютным большинством тегов. Понятно, что встроенные стили удобно использовать только для какого-то одного тега. Если же какой-то стиль должен быть применен сразу к нескольким тегам, выгоднее использовать стиль переопределения тега или стилового класса.

На заметку

Dreamweaver сам активно использует встроенные стили для создания свободно позиционируемых контейнеров, с которыми мы познакомимся на занятии 12.

Кстати, стили переопределения тегов и стилевые классы можно комбинировать друг с другом, порождая стили, которые так и называются — *гибридными*. Вот пара примеров гибридных стилей:

```
P.green { color: #FF0000;
          font-size: smaller; }
H1 STRONG ( font-size: larger; )
```

Первый стиль будет применен ко всем тегам `<P>`, имеющим значение атрибута `CLASS` равное `green`.

```
<P CLASS="green">Это зеленый курсив</P>
```

```
<H1 CLASS="green">А это не зеленый и не курсив </H1>
```


Второй стиль будет применен ко всем тегам ``, вложенным в тег `<H1>`.

```
<H1>Это <STRONG>большой и полужирный</STRONG> шрифт</H1>
```

```
<P>А это <STRONG>просто полужирный</STRONG> шрифт</P>
```

При создании гибридных стилей нам будет нужно включить в диалоговом окне **New CSS Rule** (см. рис. 7.2) переключатель **Advanced**. После этого мы беспрепятственно сможем ввести имя создаваемого стиля в комбинированный список **Name**.

И напоследок. В составе Dreamweaver поставляется весьма подробный справочник по CSS все того же издательства O'Reilly. Чтобы вывести его на экран, нужно вызвать панель **Reference**, выбрав пункт **Reference** в меню **Window** или нажав комбинацию клавиш `<Shift>+<F1>`. После этого в раскрывающемся списке **Book** этой панели выбираем пункт **O'REILLY CSS Reference** — и кладешь премудрости перед нами! Чтобы просмотреть справочную информацию по какому-либо атрибуту, достаточно выбрать его имя в раскрывающемся списке **Style**. Например, на рис. 7.5 представлена справочная информация по уже знакомому нам атрибуту `color`.

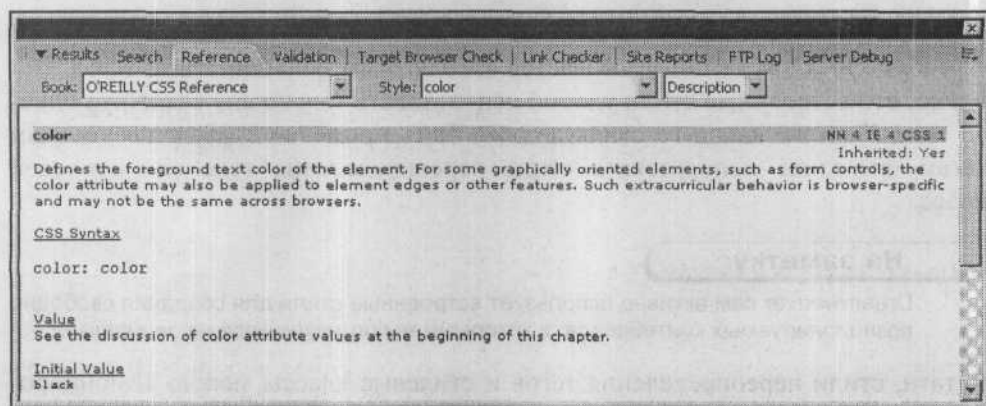


Рис. 7.5. Справка по атрибуту `color` в панели **Reference**

Возможности CSS по оформлению Web-страниц

А теперь мы рассмотрим все возможности языка CSS по оформлению Web-страниц. А также выясним, какие инструменты предлагает нам Dreamweaver для работы со стилями.

Параметры шрифта

Параметры шрифта, которым набран текст, задаются в категории **Type** диалогового окна **CSS Rule definition** (рис. 7.6).

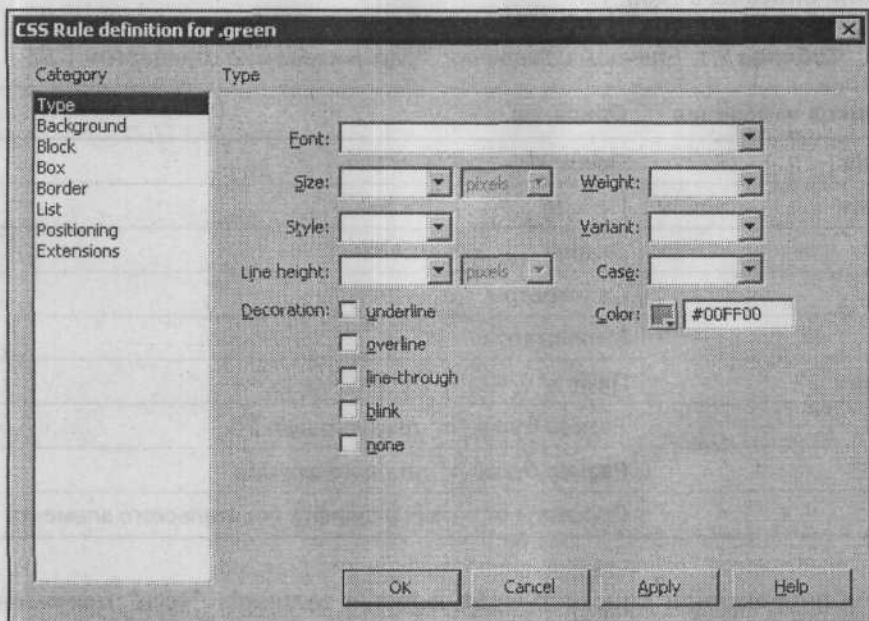


Рис. 7.6. Категория **Type** диалогового окна **CSS Rule definition**

Сам шрифт выбирается в комбинированном списке **Font**. Мы можем выбрать один из шрифтов либо ввести его имя вручную. При этом не забываем о правилах назначения шрифтов, описанных на *занятии 3*.

В комбинированном списке **Size** задается размер шрифта. Здесь присутствуют три вида пунктов:

- ◆ пункты — числовые значения. Задают размер шрифта в одной из поддерживаемых CSS единиц измерения (о них — чуть ниже);
- ◆ пункты **xx-small**, **x-small**, **small**, **medium**, **large**, **x-large** и **xx-large** задают одно из predetermined значений размера шрифта;
- ◆ пункты **larger** и **smaller** задают относительный размер шрифта. Пункт **larger** увеличивает шрифт на одну позицию в списке predetermined значений, а пункт **smaller** — уменьшает.

Ну и, конечно, мы можем просто ввести нужный размер шрифта вручную прямо в этот список.

В раскрывающемся списке, расположенном правее **Size**, выбирается единица измерения размера шрифта. Этот список доступен только тогда, когда в списке **Size** задано числовое значение. Всего здесь доступно девять пунктов, представляющих девять *единиц измерения*, поддерживаемых стандартом CSS. Все они описаны в табл. 7.1.

Таблица 7.1. Единицы измерения, поддерживаемые стандартом CSS

Единица измерения	Описание
pixels	Пиксели
points	Пункты
in	Дюймы
cm	Сантиметры
mm	Миллиметры
picas	Пики
ems	Размер буквы "m" текущего шрифта
exs	Размер буквы "x" текущего шрифта
%	Проценты от размера шрифта родительского элемента

В комбинированном списке **Weight** задается величина "веса" полужирного начертания шрифта. Мы можем ввести числовое значение "веса" от 100 до 900, либо выбрать одно из predeterminedных абсолютных или относительных значений. Как и в списке **Size**, здесь доступны пункты трех типов:

- ◆ пункты — числовые значения. Задают "вес" шрифта в числах от 100 до 900. Обычный шрифт имеет значение 400, полужирный — 700;
- ◆ пункты **normal** и **bold** задают соответственно обычный и полужирный шрифт;
- ◆ пункты **bolder** и **lighter** задают относительный "вес" шрифта. Пункт **bolder** увеличивает "вес" на одну позицию в списке predeterminedных значений, а пункт **lighter** — уменьшает.

В раскрывающемся списке **Style** мы можем выбрать "стиль" начертания шрифта. Здесь доступны три пункта: **normal** — обычный шрифт, **italic** — курсив и **oblique** — зависит от конкретной программы Web-обозревателя, но обычно тоже курсив.

В раскрывающемся списке **Variant** задается вид малых букв шрифта. Пункт **normal** задает нормальный их вид, а **small-caps** делает их похожими на уменьшенные заглавные буквы.

В комбинированном списке **Line height** задается вертикальный размер строки текста. Значение **normal** устанавливает размер по умолчанию, вычисляемый самим Web-обозревателем. Чтобы задать свой размер, введем его вручную и выберем единицу измерения. Раскрывающийся список, устанавливающий единицу измерения, находится правее. Если мы его откроем, то увидим, что он имеет новый пункт **multiple**, который задает умножение размера по умолчанию на введенное нами число.

В раскрывающемся списке **Case** устанавливается вид текста: будет ли он отображаться только большими или только маленькими буквами. Здесь доступны четыре пункта:

- ◆ **capitalize** — каждое слово текста отображается с большой буквы;
- ◆ **uppercase** — текст отображается только большими буквами;
- ◆ **lowercase** — только маленькими буквами;
- ◆ **none** — текст отображается как он набран, без всяческих преобразований (поведение по умолчанию).

Группа флажков **Decoration** задает дополнительное "украшение" текста: подчеркивание, надчеркивание или зачеркивание. Здесь доступно пять флажков:

- ◆ **underline** — подчеркивает текст (поведение по умолчанию для гиперссылок);
- ◆ **overline** — надчеркивает текст;
- ◆ **line-through** — зачеркивает текст;
- ◆ **blink** — заставляет текст мерцать (поддерживается только старыми версиями Navigator);
- ◆ **none** — убирает все эти "украшения" (поведение по умолчанию для остального текста).

Селектор цветов **Color** задает цвет текста.

Параметры фона

Параметры фона элемента страницы задаются в категории **Background** диалогового окна **CSS Rule definition** (рис. 7.7).

Селектор цветов **Background color** задает цвет фона. Значение по умолчанию зависит от программы Web-обозревателя.

В комбинированном списке **Background image** задается графическое изображение, которое будет использоваться в качестве графического фона страницы. Мы можем ввести имя нужного файла или выбрать пункт **none**, убирающий графический фон. Также мы можем щелкнуть кнопку **Browse** и выбрать нужный файл в диалоговом окне **Select File**.

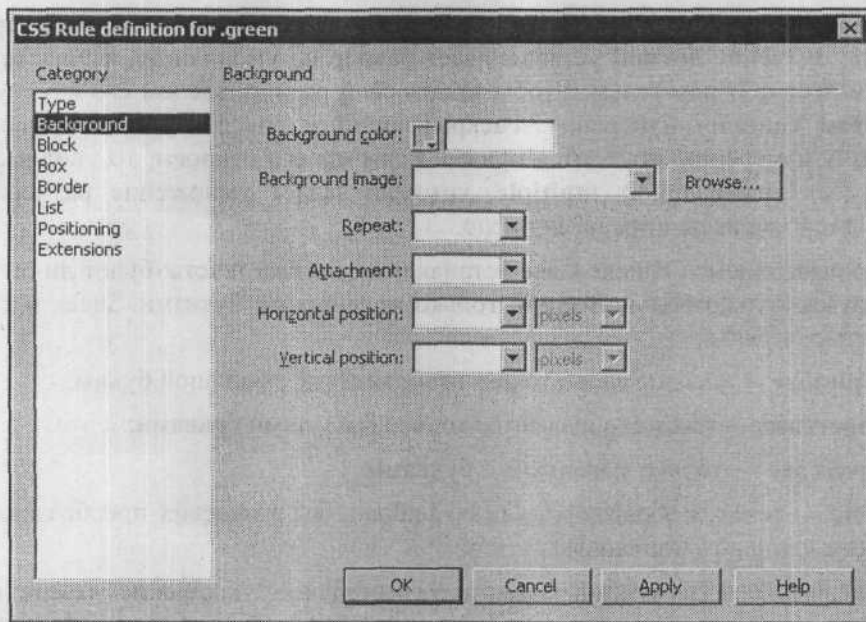


Рис. 7.7. Категория Background диалогового окна CSS Rule definition

Обычно графическое изображение, используемое в качестве фона, значительно меньше в размерах, чем сама страница. В этом случае Web-обозреватель многократно повторяет это изображение, выкладывая своего рода "мозаику", занимающую всю страницу. Управлять тем, как будет повторяться это изображение, можно с помощью раскрывающегося списка **Repeat**. Здесь доступны четыре пункта:

- ◆ **no-repeat** — отключает повторение. Изображение помещается на страницу всего один раз, в левом верхнем углу элемента;
- ◆ **repeat** — включает повторение изображения по горизонтали и вертикали (поведение по умолчанию);
- ◆ **repeat-x** — включает повторение изображения только по горизонтали;
- ◆ **repeat-y** — только по вертикали.

Когда мы прокручиваем содержимое окна Web-обозревателя, то вместе с содержимым Web-страницы прокручивается и графический фон (если он есть). Некоторые Web-обозреватели (в частности, Internet Explorer, начиная с версии 4.0) поддерживают одну забавную возможность: можно запретить прокрутку графического фона, зафиксировав его на месте. Это может придать странице некоторую оригинальность, однако при прокрутке страницы в окне Web-обозревателя отнимет больше ресурсов компьютера. Зафиксировать фон

можно с помощью раскрывающегося списка **Attachment**. Пункт **scroll** этого списка заставляет графический фон прокручиваться (поведение по умолчанию), а **fixed** — фиксирует его на месте.

Также некоторые Web-обозреватели позволяют нам поместить графический фон в нужное место страницы. Для этого предназначены раскрывающиеся списки **Horizontal position** и **Vertical position**. Мы можем ввести в них числовые значения соответственно горизонтальной и вертикальной координат графического фона. В раскрывающихся списках, расположенных правее, выбирается одна из единиц измерения, поддерживаемых CSS.

Кроме того, в раскрывающемся списке **Horizontal position** доступны три пункта:

- ◆ **left** — выравнивание графического фона по левому краю элемента страницы (поведение по умолчанию);
- ◆ **center** — выравнивание по центру;
- ◆ **right** — выравнивание по правому краю.

В раскрывающемся списке **Vertical position** доступны также три пункта:

- ◆ **top** — выравнивание графического фона по верху элемента страницы (поведение по умолчанию);
- ◆ **center** — выравнивание по центру;
- ◆ **bottom** — выравнивание по низу.

Очевидно, что задавать эти параметры нужно, предварительно задав соответствующие параметры других атрибутов стиля, перечисленных на вкладке **Background**. Вдобавок для достижения нужного результата наверняка придется поэкспериментировать.

Параметры абзаца

Параметры текстового абзаца задаются в категории **Block** диалогового окна **CSS Rule definition** (рис. 7.8). Здесь же задаются параметры, применяемые к нетекстовым элементам страницы, таким как графические изображения и таблицы.

В комбинированном списке **Word spacing** задается величина дополнительного отступа, помещаемого между словами. После ввода нужной величины нам надо будет выбрать единицу измерения в расположенном правее раскрывающемся списке. Пункт **normal** списка **Word spacing** убирает дополнительное расстояние между словами текста.

В комбинированном списке **Letter spacing** задается величина дополнительного отступа, помещаемого между символами. После ввода нужной величины

нам также нужно будет выбрать единицу измерения в расположенном правее раскрывающемся списке. А пункт **normal** списка **Letter spacing** убирает дополнительное расстояние между символами текста.

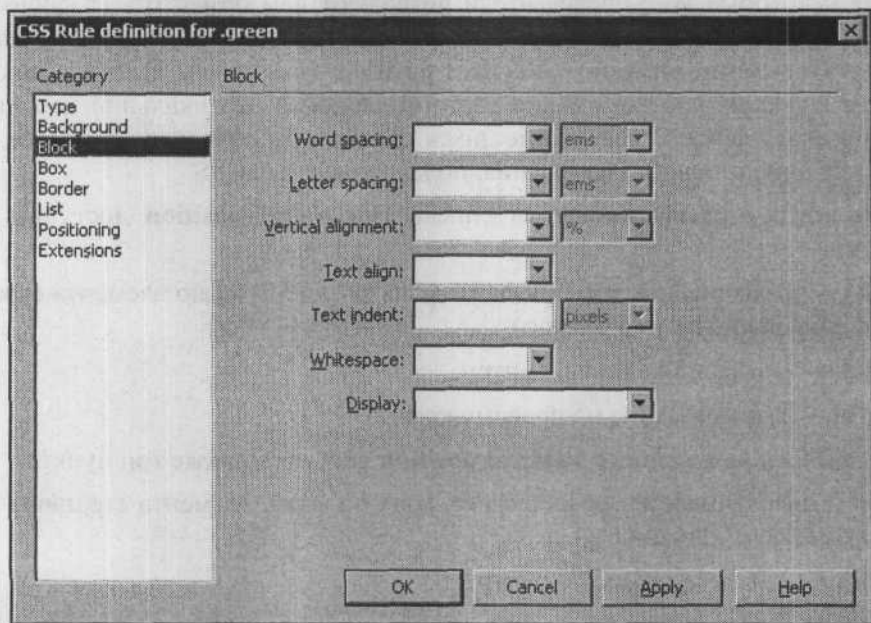


Рис. 7.8. Категория **Block** диалогового окна **CSS Rule definition**

Комбинированный список **Vertical alignment** задает вертикальное выравнивание элемента страницы относительно родителя. Здесь доступны следующие пункты:

- ◆ **baseline** — выравнивает базовую линию элемента страницы по базовой линии текста родителя (поведение по умолчанию);
- ◆ **sub** — помещает текст ниже остального текста, создавая нижний индекс;
- ◆ **super** — помещает текст выше остального текста, создавая верхний индекс;
- ◆ **top** — выравнивает элемент страницы по верхнему краю родительского элемента;
- ◆ **text-top** — выравнивает верхний край элемента страницы по верхнему краю текста родителя;
- ◆ **middle** — помещает середину элемента страницы на половину высоты шрифта выше базовой линии текста родителя;
- ◆ **bottom** — выравнивает элемент страницы по нижнему краю родительско-

◆ **text-top** — выравнивает нижний край элемента страницы по нижнему краю текста родителя.

Кроме того, мы можем ввести в этот комбинированный список процентное значение, задающее, насколько выше или ниже базовой линии родителя находится базовая линия элемента страницы. Опять же, скорее всего, здесь нам придется поэкспериментировать.

Раскрывающийся список **Text align** задает горизонтальное выравнивание текста в абзаце. Здесь доступны четыре пункта: **left** — выравнивание по левому краю (поведение по умолчанию), **center** — по центру, **right** — по правому краю, **justify** — по ширине.

В поле ввода **Text indent** задается величина отступа красной строки. Эта величина может быть как положительной, так и отрицательной (выступ). В раскрывающемся списке, расположенном правее, выбирается единица измерения.

Параметры размеров и размещения

Параметры, задающие размеры и размещение элемента страницы, устанавливаются в категории **Box** диалогового окна **CSS Rule definition** (рис. 7.9).

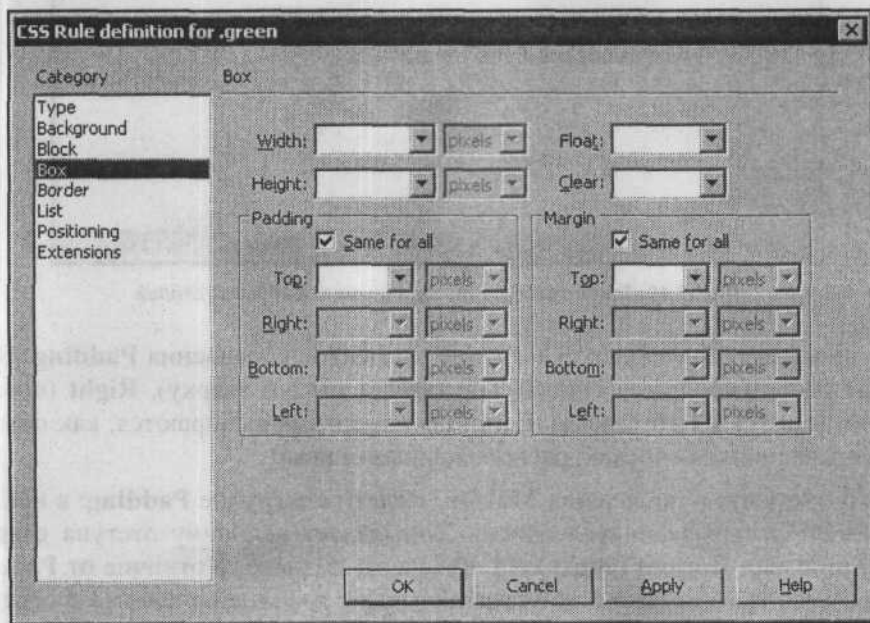


Рис. 7.9. Категория **Box** диалогового окна **CSS Rule definition**

Ширину и высоту элемента страницы мы можем задать в комбинированных списках **Width** и **Height**. Эти размеры могут быть установлены как абсолютные величины или процентное значение от ширины или высоты родителя (ячейки таблицы или самой Web-страницы). Единица измерения, как обычно, выбирается в раскрывающемся списке, расположенном справа от соответствующего комбинированного списка. Чтобы вернуть Web-обозревателю управление размерами элемента страницы, нужно выбрать пункт **auto** или вообще очистить соответствующий комбинированный список.

Раскрывающиеся списки **Float** и **Clear** мы рассмотрим на *занятиях 11 и 12*, когда будем говорить о контейнерном дизайне Web-страниц.

А теперь мы подходим к самому интересному. Дело в том, что, воспользовавшись особыми атрибутами стиля, мы можем задать отступы от воображаемой границы элемента до его содержимого (текста, изображения и пр.). Посмотрим, например, на рис. 7.10 — там для текста в абзаце заданы сантиметровые отступы со всех сторон.

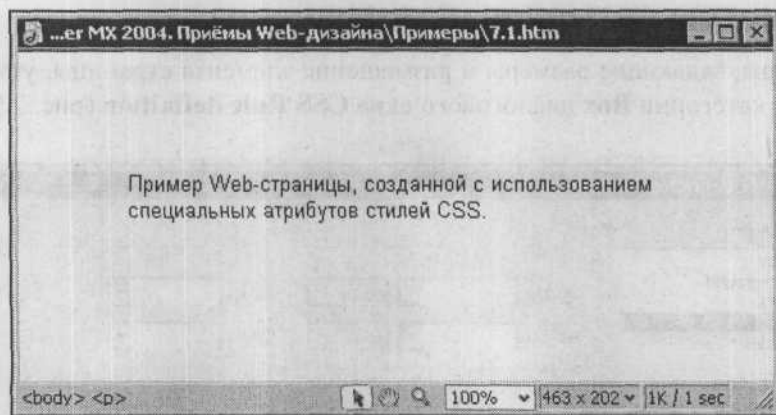


Рис. 7.10. Текст с отступами, заданными атрибутами стилей

Величины отступов задаются в группе элементов управления **Padding**. В нее входят раскрывающиеся списки **Top** (задает отступ сверху), **Right** (справа), **Bottom** (снизу) и **Left** (слева). Единицы измерения выбираются, как обычно, в раскрывающихся списках, расположенных справа.

Группа элементов управления **Margin** аналогична группе **Padding**: в нее также входят раскрывающиеся списки **Top** (задает величину отступа сверху), **Right** (справа), **Bottom** (снизу) и **Left** (слева). Однако, в отличие от **Padding**, списки этой группы задают расстояние между границей элемента и соседними элементами страницы. Единицы измерения выбираются также в раскрывающихся списках справа.

Обе эти группы формируют на странице свободное пространство. Но, если свободное пространство, созданное с помощью элементов группы **Padding**, принадлежит элементу страницы, к которому применяется стиль, то созданное с помощью группы **Margin** — не принадлежит ему, а точнее, принадлежит родителю (например, Web-странице). Здесь можно привести аналогию с ячейками таблицы: параметр **CellPad** служит для задания расстояния между границей ячейки и ее содержимым и аналогичен группе **Padding**, а параметр **CellSpace** задает расстояние между границами соседних ячеек и аналогичен группе **Margin**.

Если мы хотим задать одинаковые отступы со всех сторон, то можем включить флажок **Same for all**, находящийся в соответствующей группе элементов управления. После этого вводим нужную величину отступа в комбинированный список **Top** (остальные списки станут недоступными). Флажок **Same for all** включен по умолчанию, т. к. отступы для элемента страницы, как правило, задаются одинаковыми со всех сторон.

Параметры рамки

Параметры, задающие цвет и толщину рамки вокруг элемента страницы, задаются в категории **Border** диалогового окна **CSS Rule definition** (рис. 7.11).

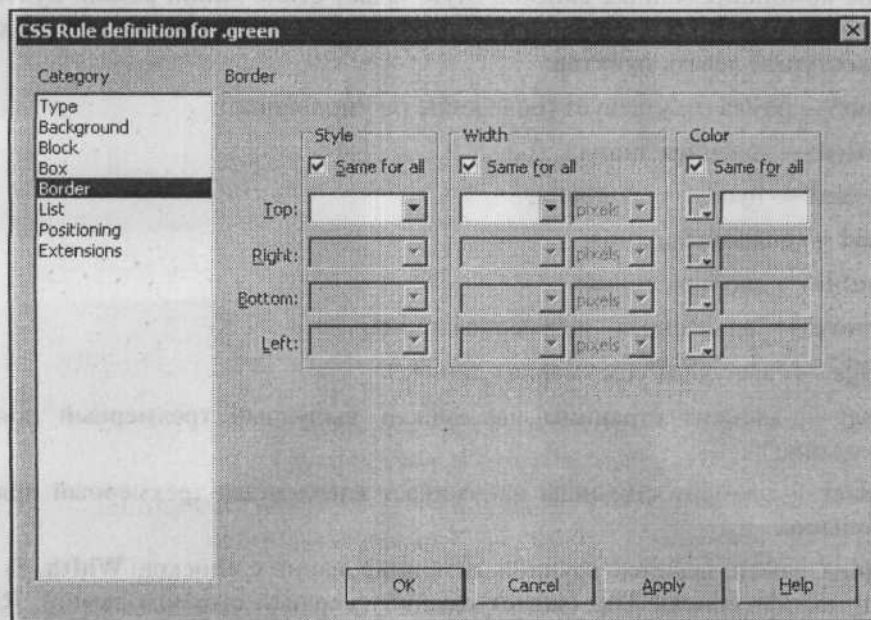


Рис. 7.11. Категория **Border** диалогового окна **CSS Rule definition**

Рамка рисуется по воображаемой границе элемента страницы (рис. 7.12). Мы можем задать отступы от границы до содержимого элемента с помощью атрибутов группы **Padding**, а отступы от границы до "соседей" этого элемента — **Margin**. Все это мы можем найти в категории **Box** диалогового окна **CSS Rule definition** (см. рис. 7.9).

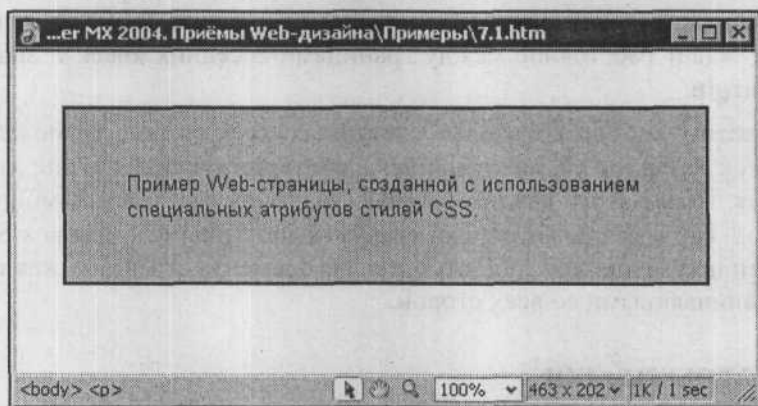


Рис. 7.12. Рамка вокруг абзаца (для удобства выбран светло-серый фон абзаца)

Группа комбинированных списков **Style** задает стиль линии рамки: будет ли она сплошная, точечная или вообще невидимая. Всего в каждом из этих списков доступны девять пунктов:

- ◆ **none** — рамка отсутствует (поведение по умолчанию);
- ◆ **dotted** — точечная линия;
- ◆ **dashed** — пунктирная линия;
- ◆ **solid** — сплошная линия;
- ◆ **double** — двойная линия;
- ◆ **groove** — "вдавленная" трехмерная линия;
- ◆ **ridge** — "выпуклая" трехмерная линия;
- ◆ **inset** — элемент страницы напоминает выпуклый трехмерный прямоугольник;
- ◆ **outset** — элемент страницы напоминает вдавленный трехмерный прямоугольник.

Толщина рамки задается группой комбинированных списков **Width**. В эту группу входят списки **Top** (задает толщину верхней стороны рамки), **Right** (правой), **Bottom** (нижней) и **Left** (левой). Мы можем ввести в любой из этих списков числовое значение и выбрать в расположенном правее раскрываю-

шемся списке единицу измерения, абсолютную или относительную. Кроме того, в этих комбинированных списках доступны три пункта, задающие predetermined значения толщины рамки:

- ◆ **thin** — тонкая рамка;
- ◆ **medium** — средняя;
- ◆ **thick** — толстая.

С помощью группы селекторов цветов **Color** задается цвет рамки. Если цвет не задан, рамка отображается тем цветом, который задан в качестве цвета текста для текущего элемента.

Если мы хотим задать одинаковые параметры для всех сторон рамки, то должны будем включить флажок **Same for all**, находящийся в соответствующей группе элементов управления. Этот флажок включен по умолчанию, т. е. обычно все стороны рамки делаются одинаковыми.

Пользуясь возможностями CSS по созданию рамок, мы можем творить чудеса. Например, что вы скажете о таблице, имеющей вместо обычной границы только горизонтальные линейки сверху и снизу? А ведь это делается элементарно — нужно только привязать к тегу `<TABLE>` соответствующий стиль.

Параметры маркеров списка

Параметры, устанавливающие вид маркеров списка, задаются в категории **List** диалогового окна **CSS Rule definition** (рис. 7.13).

В раскрывающемся списке **Type** задается вид маркера или нумерации позиций списка. Здесь доступны девять пунктов:

- ◆ **disc** — маркер в виде кружка с заполнением (поведение по умолчанию для маркированных списков);
- ◆ **circle** — маркер в виде окружности без заполнения;
- ◆ **square** — маркер в виде маленького квадратика с заполнением;
- ◆ **decimal** — нумерация арабскими цифрами (поведение по умолчанию для нумерованных списков);
- ◆ **lower-roman** — нумерация малыми римскими цифрами;
- ◆ **upper-roman** — нумерация большими римскими цифрами;
- ◆ **lower-alpha** — нумерация малыми латинскими буквами;
- ◆ **upper-alpha** — нумерация большими латинскими буквами;
- ◆ **none** — нет ни маркера, ни нумерации (поведение по умолчанию для обычных текстовых абзацев).

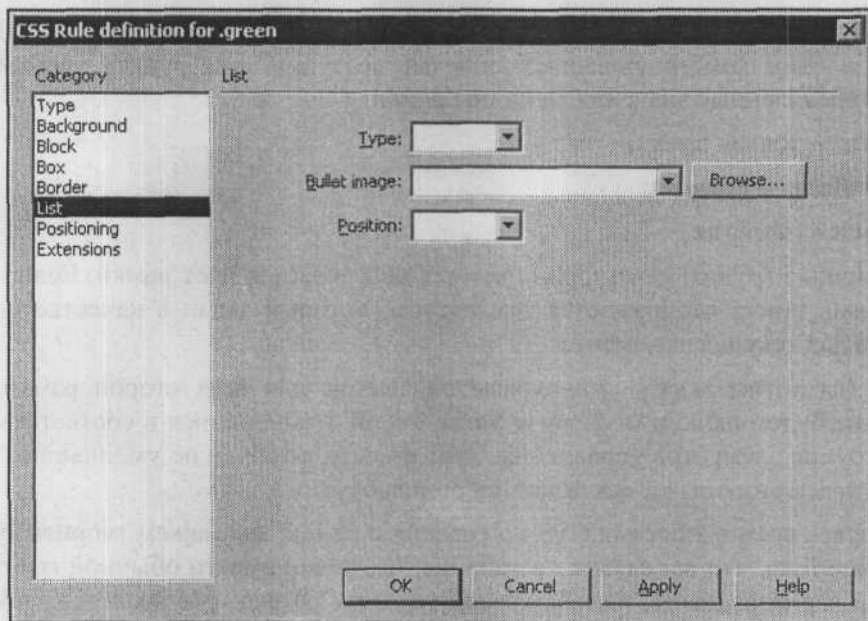


Рис. 7.13. Категория List диалогового окна CSS Rule definition

В поле ввода **Bullet image** вводится имя файла графического изображения, которое будет использоваться в качестве маркера. Можно также щелкнуть кнопку **Browse** и выбрать нужный файл в диалоговом окне **Select File**. Обратите внимание, что если задан графический маркер, установки атрибута **Type** теряют силу.

Раскрывающийся список **Position** задает местонахождение маркера или нумерации. Здесь доступны два пункта. Если выбран пункт **inside**, то маркер (нумерация) располагается внутри границ текстового абзаца, т. е. принадлежит ему; в этом случае текст пунктов списка будет выглядеть несколько компактнее. Если же выбран пункт **outside**, то маркер (нумерация) располагается за пределами границ абзаца; пункты таких списков лучше читаются. Значение по умолчанию — **outside**.

Управление стилями

Осталось рассмотреть, как в Dreamweaver осуществляется управление стилями, в частности изменение и удаление (как создавать новые стили, мы уже знаем).

Изменить нужный стиль проще простого. Выделяем его в списке панели **CSS Styles** и выбираем пункт **Edit** контекстного или дополнительного меню либо

нажимаем кнопку **Edit Style** (🔍) внизу панели **CSS Styles**. На экране появится диалоговое окно **CSS Rule Definition**, в котором мы сможем задать нужные изменения. После нажатия кнопки **OK** все они сохранятся и будут тотчас применены.

Удалить выделенный в списке панели **CSS Styles** стиль мы можем, воспользовавшись пунктом **Delete** контекстного или дополнительного меню. Но проще нажать кнопку **Delete CSS Style** (🗑️) в правом нижнем углу этой панели.

Внимание!

Стиль будет удален без всякого предупреждения. Все задаваемые стилем параметры форматирования пропадают. Однако атрибуты `CLASS` и их значения все же сохраняются, и если вы в дальнейшем создадите стиливой класс с таким же названием, он будет тотчас к ним применен.

Точно таким же образом мы можем удалить из списка панели **CSS Styles** саму таблицу стилей, внешнюю или внутреннюю. Если удалить внешнюю таблицу стилей, то из HTML-кода страницы будет удалена только ссылка на нее (тег `<LINK>`); сам файл внешней таблицы стилей останется на диске. Внутренняя таблица стилей также удаляется без предупреждения.

Кроме того, Dreamweaver предоставляет нам возможность создать копию выделенного стиля. Это может быть полезно, если мы хотим создать новый стиль, слегка переделав уже существующий. Для этого выберем нужный стиль и воспользуемся пунктом **Duplicate** контекстного или дополнительного меню панели **CSS Styles**. На экране появится диалоговое окно **CSS Rule Definition**, в котором мы сможем задать параметры нового стиля.

Особые случаи применения стилей CSS

Напоследок осталось поговорить о трех особых случаях применения стилей CSS. Это оформление гиперссылок, задание параметров самой Web-страницы и применение контейнеров для оформления больших фрагментов страниц. Сложного во всем этом нет ничего, но получается очень красиво.

Псевдостили гиперссылок

Псевдостили гиперссылок — это особые стили, позволяющие в достаточно широких пределах менять внешний вид гиперссылок на ваших страницах. Они не относятся ни к стиливым классам, ни к стилям переопределения тегов. Всего их четыре, и все они перечислены в табл. 7.2.

Таблица 7.2. Псевдостили гиперссылок

Псевдостиль	Описание
A:link	Обычная гиперссылка
A:active	Активная гиперссылка (та, на которой посетитель щелкает мышью в данный момент)
A:visited	Посещенная гиперссылка (указывающая на страницу, на которой посетитель уже побывал)
A:hover	Гиперссылка, на которую в данный момент указывает курсор мыши

Пользуясь псевдостилиями гиперссылок, мы можем задать для гиперссылок какое угодно оформление. Так, например, воспользовавшись категорией **Type** диалогового окна **CSS Rule Definition** (см. рис. 7.6), мы можем убрать у обычных гиперссылок подчеркивание (для этого достаточно включить флажок **none** в группе **Decoration**). Для гиперссылки же, на которую указывает курсор мыши, и для активной гиперссылки мы можем, наоборот, включить подчеркивание (за это "отвечает" флажок **underline** все в той же в группе **Decoration**). Ну и, разумеется, мы можем задать для разных гиперссылок разные цвета.

Как и в случае гибридных стилей, при создании псевдостилей гиперссылок нам будет нужно включить в диалоговом окне **New CSS Rule** (см. рис. 7.2) переключатель **Advanced**. После этого мы сможем просто выбрать имя нужного нам псевдостилия в комбинированном списке **Name**.

Вообще, псевдостили лучше всего задавать во внешней таблице стилей, поскольку все страницы сайта должны выглядеть одинаково. Тем более это касается гиперссылок. Представьте себе удивление и гнев посетителя, если на разных страницах одного и того же сайта гиперссылки отображаются разными цветами!

Контейнеры

Мы уже научились привязывать стилевые классы к определенным тегам. Мы знаем, как сделать фон у текстового абзаца или заключить его в рамку. И уверены, что все это нам обязательно пригодится.

Но что делать, если мы хотим заключить в одну рамку сразу несколько абзацев? Конечно, можно использовать таблицу (см. занятие 6) из одной строки и одного столбца, поместив в ее единственную ячейку нужные нам абзацы и задав для таблицы видимую рамку. Но таблица — это как-то не очень изящно, да и HTML-код, создающий таблицы, слишком громоздок. Нельзя ли сделать это с помощью стилей CSS?

Можно. Более того, никаких новых знаний о CSS нам для этого не понадобится. Для этого нам потребуется только создать необходимый стилевой класс, заключить нужные нам абзацы в контейнер и применить созданный стилевой класс прямо к этому контейнеру.

Что такое *контейнер*? Это особый тег HTML, служащий как бы "упаковкой" для других тегов. Содержимое контейнера ведет себя как единое целое, а значит, мы можем творить с этим единым целым все, что нам заблагорассудится.

В нашем случае понадобится так называемый *блочный контейнер*, или просто *блок*. Он ведет себя как отдельный абзац текста, т. е. отображается в отдельной строке и располагается на некотором расстоянии от других элементов страницы. А создается он при помощи парного тега `<DIV>`.

Давайте откроем страницу `about.htm`, содержащую сведения о студенте Васильеве, и попробуем заключить первые три абзаца в общую рамку. Для этого нам будет нужно поместить три тега `<P>`, задающие эти абзацы, вместе с их содержимым в один тег `<DIV>`. Проще всего это сделать, переключившись в режим отображения кода HTML. Итак, вставляем перед первым тегом `<P>` пустую строку и вводим в нее открывающий тег `<DIV>`. После этого вставляем пустую строку после последнего тега `</P>` и вводим туда закрывающий тег `</DIV>`. Все.

Далее создаем новый стилевой класс и задаем для него нужные параметры рамки. Ну и привязываем этот стилевой класс к созданному ранее тегу `<DIV>` — нашему блочному контейнеру. Проще всего это сделать, щелкнув правой кнопкой мыши по соответствующей кнопке секции тегов и выбрав стилевой класс в подменю **Set Class**. Результат показан на рис. 7.14.

Как мы помним, для привязки стилевого класса к тегу служит атрибут `class`. Этот атрибут поддерживается практически всеми тегами (по крайней мере, всеми видимыми), в том числе и тегом `<DIV>`.

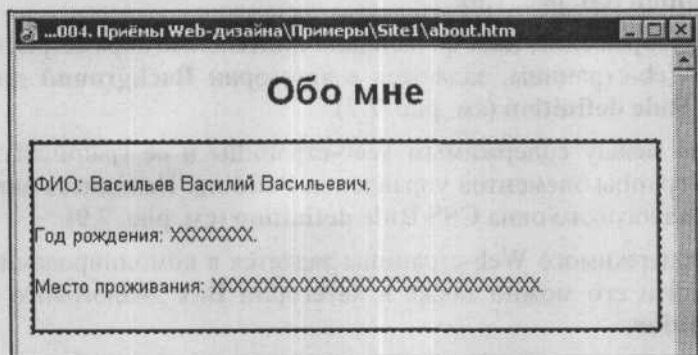


Рис. 7.14. Пример использования блочного контейнера для оформления сразу нескольких абзацев текста

Так, с блочными контейнерами все ясно. Теперь представим себе другую ситуацию: нам нужно выделить каким-то образом не весь абзац целиком, а часть его. Скажем, захотели мы выделить имя создателя сайта красным цветом. Каковы будут наши действия?

Специально для этого язык HTML предусматривает еще одну разновидность контейнеров — *встроенный контейнер*. Содержимое этого контейнера ведет себя как часть абзаца. А создается встроенный контейнер с помощью парного тега ``.

Итак, мы хотим выделить на странице `about.htm` имя нашего героя красным цветом. Создадим соответствующий стилевой класс и... Нет, для этого вовсе не надо переключаться в режим редактирования HTML-кода (хотя и можно). Достаточно выделить соответствующий фрагмент текста и задать для него этот стилевой класс, выбрав его в раскрывающемся списке **Style** редактора свойств (см. рис. 7.4). Dreamweaver сам поместит выделенный нами текст внутрь тега `` и задаст для него стиль с помощью атрибута `CLASS`.

Общие параметры Web-страницы

Сообразительный читатель сразу поймет, что с помощью стилей CSS мы можем задать некие общие параметры для самой Web-страницы. Это цвет текста и фона, цвета гиперссылок, расстояние между содержимым страницы и ее границей (фактически — рамкой окна Web-обозревателя), возможно — ширину содержимого (например, если страницы сайта создаются под определенное экранное разрешение) и пр. Как все это сделать? С помощью стиля переопределения тега `<BODY>`.

Итак, давайте перечислим все общие параметры Web-страницы и соотнесем их с подходящими инструментами Dreamweaver.

- ◆ Шрифт и цвет текста задается в категории **Type** диалогового окна **CSS Rule definition** (см. рис. 7.6).
- ◆ Фоновое изображение, цвет фона и дополнительные параметры, связанные с фоном Web-страницы, задаются в категории **Background** диалогового окна **CSS Rule definition** (см. рис. 7.7).
- ◆ Расстояние между содержимым Web-страницы и ее границей задается с помощью группы элементов управления **Padding**. Находится она в категории **Box** диалогового окна **CSS Rule definition** (см. рис. 7.9).
- ◆ Ширина содержимого Web-страницы задается в комбинированном списке **Width**. Найти его можно также в категории **Box** диалогового окна **CSS Rule definition**.
- ◆ Параметры гиперссылок задаются с помощью соответствующих псевдостилей, описанных выше.

- ◆ Параметры остальных элементов страницы (абзацев, заголовков, таблиц и пр.) задаются с помощью соответствующих им стилей переопределения тегов.

Поскольку все эти параметры должны применяться ко всем страницам сайта, их нужно будет поместить во внешнюю таблицу стилей. Эта таблица стилей будет использоваться всеми страницами сайта.

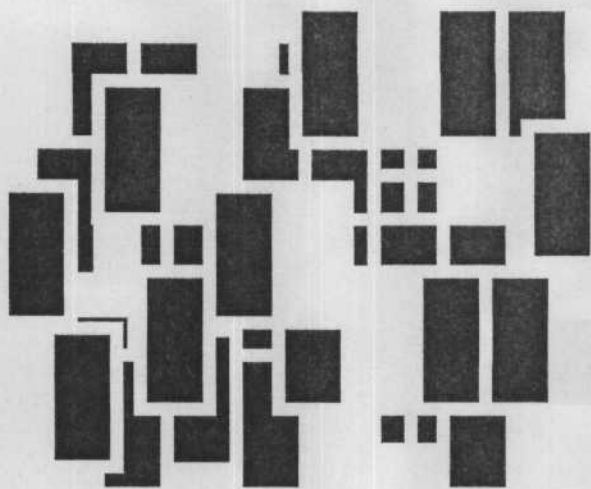
На заметку

Нужно сказать, что Dreamweaver предоставляет особое диалоговое окно **Page Properties**, с помощью которого можно задать общие параметры Web-страницы. Вызвать это окно можно, выбрав в меню **Modify** пункт **Page Properties** или нажав комбинацию клавиш **<Ctrl>+<J>**. Здесь можно задать все перечисленные выше общие параметры, а также название Web-страницы и ее кодировку. Но проблема в том, что стиль, созданный на основе этих параметров, Dreamweaver помещает во внутреннюю таблицу стилей, что почти всегда нежелательно.

На следующем занятии...

На этом разговор о стилях CSS подошел к концу. Многое, конечно, осталось за рамками этого занятия, но основные сведения все-таки мы получили.

Следующий курс занятий будет посвящен исключительно Web-дизайну. А именно более прогрессивным и в настоящее время самым популярным его способам: фреймовому и табличному. Так что постарайтесь не прогуливать следующие занятия!

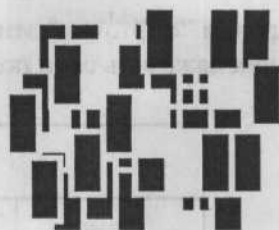


КУРС 3

Фреймовый и табличный дизайн

- Занятие 8.** Фреймовый дизайн
- Занятие 9.** Табличный дизайн
- Занятие 10.** Использование шаблонов

ЗАНЯТИЕ 8



Фреймовый дизайн

Первые два пройденных нами курса занятий были весьма плодотворными. Мы научились пользоваться пакетом Macromedia Dreamweaver 8 и создали свой первый Web-сайт. Фактически мы уже стали Web-дизайнерами.

Этот — третий — курс занятий опять посвящен исключительно Web-дизайну. А именно двум его разновидностям, которые сейчас наиболее популярны в Интернете: фреймовому и табличному дизайну. Простой же текстовый дизайн, который мы уже знаем по предыдущим занятиям и с которым мы уже, так сказать, познакомились на практике, сейчас считается таким же анахронизмом, как текстовые программы Web-обозревателей. (Да, на заре Интернета были такие!..) Так что пора познакомиться с современными веяниями!

И начнем мы с фреймового дизайна. Именно ему будет посвящена это занятие.

Введение во фреймы и наборы фреймов

Давайте откроем в Dreamweaver или Web-обозревателе любую страничку нашего сайта и посмотрим на нее. Что мы видим? И название сайта, и набор гиперссылок, указывающих на другие страницы, и основное содержимое, и сведения об авторских правах представляют собой единое целое. И все это (ну, почти все) дублируется на каждой странице сайта.

А теперь представим себе, что окно Web-обозревателя разбито на несколько меньших "форточек" и в каждой "форточке" загружается своя Web-страница. В одной — набор гиперссылок, в другой — заголовок сайта, в третьей — сведения об авторских правах, а в самой большой — основное содержимое страницы. Схематично все это можно представить так, как показано на рис. 8.1.

Причем "форточки" эти ведут себя как независимые окна Web-обозревателя: у них даже есть свои полосы прокрутки.

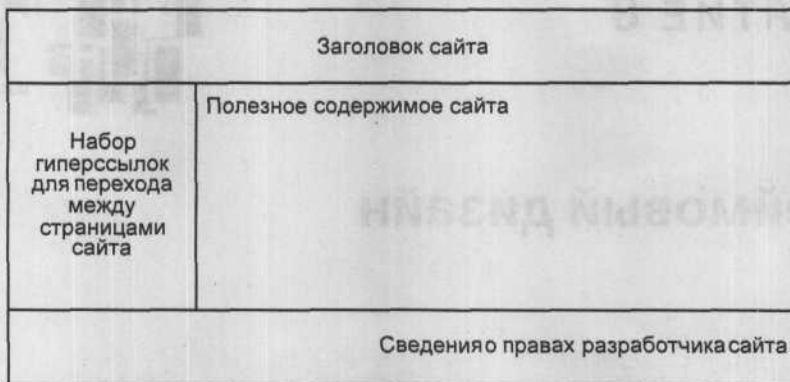


Рис. 8.1. Классический набор из четырех фреймов

Если же щелкнуть по гиперссылке, нужная страница загрузится непосредственно в соответствующую ей "форточку". Причем заполнена эта страница будет только основным содержимым. А набор гиперссылок, заголовок сайта и сведения об авторских правах так и останутся в своих "форточках" неизменными.

Это очень удобно. Во-первых, страницы загружаются значительно быстрее, т. к. на них размещено только основное содержимое, без набора гиперссылок, заголовка, сведений об авторских правах и прочей вспомогательной информации. Во-вторых, не нужно всю эту вспомогательную информацию дублировать на каждой странице сайта — достаточно создать специальные странички, содержащие ее части, а потом загрузить их в соответствующие "форточки". А если нам нужно изменить, скажем, заголовок сайта, мы просто меняем его на одной-единственной страничке вместо того, чтобы перелопачивать все страницы сайта.

Отлично! Теперь давайте назовем вещи своими именами. "Форточки", иначе говоря, части, на которые делится окно Web-обозревателя, — это и есть *фреймы* (frame, кадр). А набор таких частей назовем *набором фреймов* (frameset). Способ же разработки Web-сайтов с использованием фреймов называется *фреймовым дизайном*.

Как же формируется набор фреймов? С помощью специальной Web-страницы, в которой, собственно, и задаются параметры набора фреймов и самих фреймов. Там же указываются размеры фреймов и интернет-адреса страниц, которые будут в них загружены. Эта Web-страница представляет собой набор

ется страницей по умолчанию, т. е. загружается при наборе адреса сайта в строке адреса Web-обозревателя.

Нужно также помнить, что такая страница, кроме набора фреймов, не должна содержать больше ничего. Редчайшее исключение среди Web-страниц — она даже не содержит тега `<BODY>`! Собственно, а зачем он ей нужен — ведь во фреймах так или иначе отображается содержимое совсем других Web-страниц.

Вообще, фреймы — интересная и полезная вещь. Неудивительно, что фреймовый дизайн применяется уже достаточно давно. Причем настолько давно, что среди снобов от Web-дизайна он уже стал считаться дурным тоном. А зря!

Работа с фреймами в Dreamweaver

Работу с фреймами в Dreamweaver мы будем рассматривать на примере второй версии сайта студента Васильева. Сначала создадим на диске папку Site2, где будут храниться файлы локальной копии этого сайта. Далее зарегистрируем наш второй сайт в Dreamweaver под именем site 2; как это сделать, было описано на занятии 4.

А после этого можно создать новую Web-страницу. Эта страница будет содержать набор фреймов.

Создание фреймов

Конечно, приступить к созданию фреймов можно без всяких приготовлений. Но лучше все-таки открыть в подменю **Visual Aids** меню **View** и включить пункт-выключатель **Frame Borders**. После этого Dreamweaver станет отображать границы наших будущих фреймов, что очень поможет нам в дальнейшем.

Внимание!

После открытия Web-страницы, содержащей набор фреймов, вам придется каждый раз включать пункт-выключатель **Frame Borders** подменю **Visual Aids** меню **View**. К сожалению, Dreamweaver не запоминает этой установки.

Проще всего пару новых фреймов можно создать, воспользовавшись кнопкой **Frames** вкладки **Layout** инструментария объектов. На рис. 8.2 показана эта кнопка и меню, которое открывается при нажатии на нее. Также можно воспользоваться пунктами подменю **Frames**, подменю **HTML** меню **Insert**. В табл. 8.1 приведены описания всех пунктов меню кнопки **Frames** и соответствующих им пунктов подменю **Frames**.

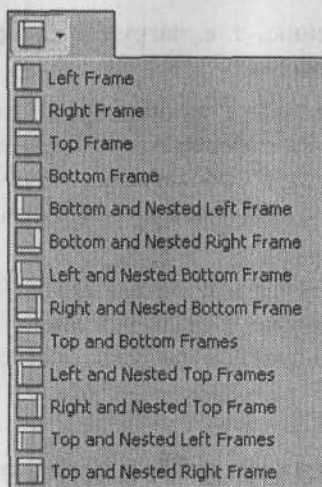


Рис. 8.2. Кнопка Frames инструментария объектов с открытым меню

Таблица 8.1. Пункты меню кнопки **Frames** и одноименного подменю

Название пункта кнопки Frames	Пункт подменю Frames	Описание
Left Frame	Left	Создает фрейм слева от <i>текущего</i> (того, в котором стоит текстовый курсор) в том же наборе
Right Frame	Right	Создает фрейм справа от текущего в том же наборе
Top Frame	Top	Создает фрейм выше текущего в том же наборе
Bottom Frame	Bottom	Создает фрейм ниже текущего в том же наборе
Bottom and Nested Left Frame	Bottom Nested Left	Создает фрейм ниже текущего в том же наборе, создает в текущем фрейме вложенный набор с двумя горизонтальными фреймами и делает текущим правый фрейм вложенного набора
Bottom and Nested Right Frame	Bottom Nested Right	Создает фрейм ниже текущего в том же наборе, создает в текущем фрейме вложенный набор с двумя горизонтальными фреймами и делает текущим левый фрейм вложенного набора

Таблица 8.1 (окончание)

Название пункта кнопки Frames	Пункт подменю Frames	Описание
Left and Nested Bottom Frame	Left Nested Bottom	Создает фрейм левее текущего в том же наборе, создает в текущем фрейме вложенный набор с двумя вертикальными фреймами и делает текущим верхний фрейм вложенного набора
Right and Nested Bottom Frame	Right Nested Bottom	Создает фрейм правее текущего в том же наборе, создает в текущем фрейме вложенный набор с двумя вертикальными фреймами и делает текущим верхний фрейм вложенного набора
Top and Bottom Frames	Top and Bottom	Создает два фрейма сверху и снизу от текущего
Left and Nested Top Frames	Left Nested Top	Создает фрейм левее текущего в том же наборе, создает в текущем фрейме вложенный набор с двумя вертикальными фреймами и делает текущим нижний фрейм вложенного набора
Right and Nested Top Frame	Right Nested Top	Создает фрейм правее текущего в том же наборе, создает в текущем фрейме вложенный набор с двумя вертикальными фреймами и делает текущим нижний фрейм вложенного набора
Top and Nested Left Frames	Top Nested Left	Создает фрейм выше текущего в том же наборе, создает в текущем фрейме вложенный набор с двумя горизонтальными фреймами и делает текущим правый фрейм вложенного набора
Top and Nested Right Frame	Top Nested Right	Создает фрейм выше текущего в том же наборе, создает в текущем фрейме вложенный набор с двумя горизонтальными фреймами и делает текущим левый фрейм вложенного набора

Итак, давайте приступим к созданию первого нашего набора фреймов.

Прежде всего, создадим верхний фрейм, в котором у нас будет помещаться заголовок сайта. Нажмем кнопку **Frames** и выберем в ее меню пункт **Top Frame**. На экране появится диалоговое окно **Frame Tag Accessibility Attributes**, показанное на рис. 8.3.

В этом диалоговом окне задаются *заголовки фреймов*. Эти заголовки никак не

граммами чтения с экрана. Можно задать их, а можно не задавать. Но, поскольку задавать заголовки фреймов стало хорошим тоном Web-дизайна, давайте все-таки их зададим.

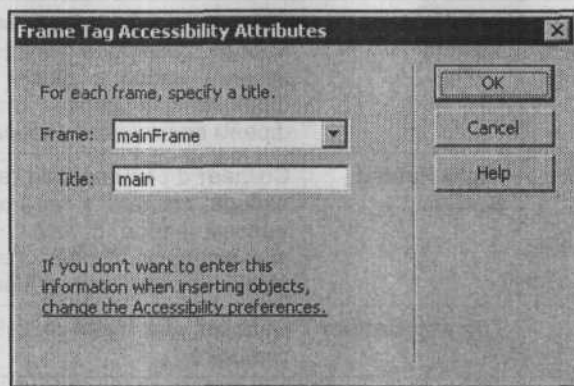


Рис. 8.3. Диалоговое окно **Frame Tag Accessibility Attributes**

В раскрывающемся списке **Frame** диалогового окна **Frame Tag Accessibility Attributes** выберем пункт **mainFrame** — это нижний, основной, фрейм. Введем в поле ввода **Title** заголовок этого фрейма — **Главный**. После этого выберем в том же списке **Frame** пункт **topFrame** (это верхний фрейм) и введем в поле ввода **Title** его заголовок — **Заголовок**. После нажатия кнопки **OK** Dreamweaver создаст два фрейма, расположенные горизонтально, друг над другом (рис. 8.4).

Теперь создадим еще один горизонтальный фрейм внизу страницы. Убедимся, что текстовый курсор стоит в нижнем фрейме, и выберем пункт **Bottom Frame** меню кнопки **Frames**. В списке **Frame** диалогового окна **Frame Tag Accessibility Attributes** выберем пункт **bottomFrame** (это только что созданный нижний фрейм) и введем в поле ввода **Title** его заголовок — пусть он будет **Права**. Нажмем кнопку **OK** и получим третий фрейм (рис. 8.5).

Теперь поместим курсор мыши на границу между верхним и средним фреймами. Мы увидим, что курсор примет вид двунаправленной стрелки, что во всех Windows-программах означает только одно: мы можем перемещать что-либо вверх-вниз. В данном случае мы можем перемещать границу между фреймами, изменяя их размеры. То же самое мы можем делать с границей между средним и нижним фреймами.

Изменим размеры фреймов так, чтобы средний фрейм занимал максимум места на экране. (В конце концов, главное в сайте — это основное содержимое?) Верхний и нижний фреймы сделаем совсем узкими. А если впоследст-

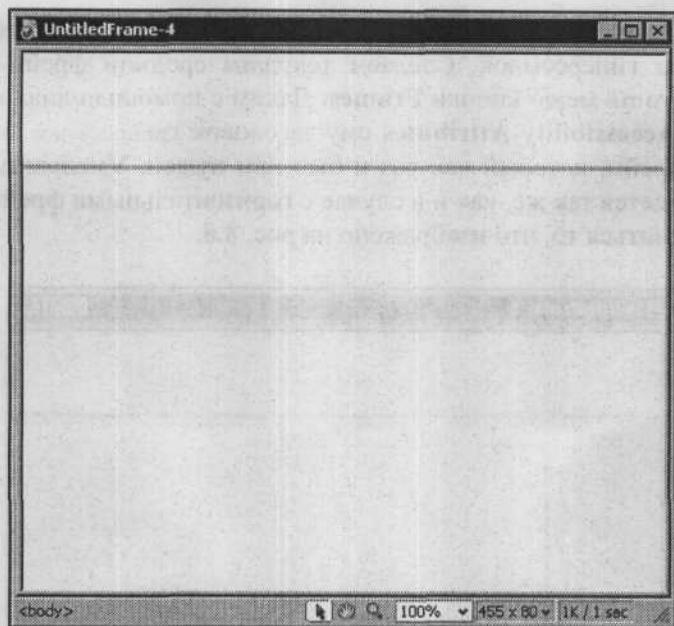


Рис. 8.4. Два горизонтальных фрейма

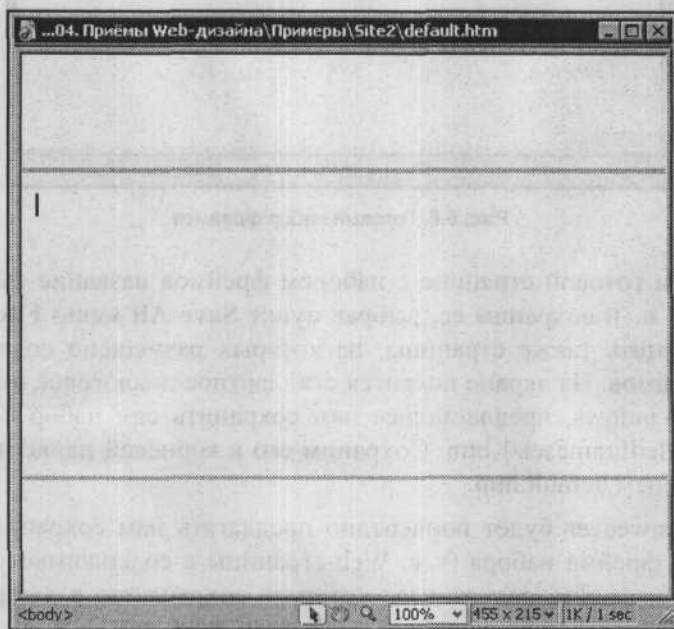


Рис. 8.5. Три горизонтальных фрейма

Осталось создать еще один фрейм — левый вертикальный, где будет помещаться набор гиперссылок. Сделаем текущим средний фрейм и выберем пункт **Left Frame** меню кнопки **Frames**. Дадим с помощью диалогового окна **Frame Tag Accessibility Attributes** ему заголовок **Гиперссылки**. Мы получим последний фрейм, который как раз и был нам нужен. Уменьшим его ширину — это делается так же, как и в случае с горизонтальными фреймами. У нас должно получиться то, что изображено на рис. 8.6.

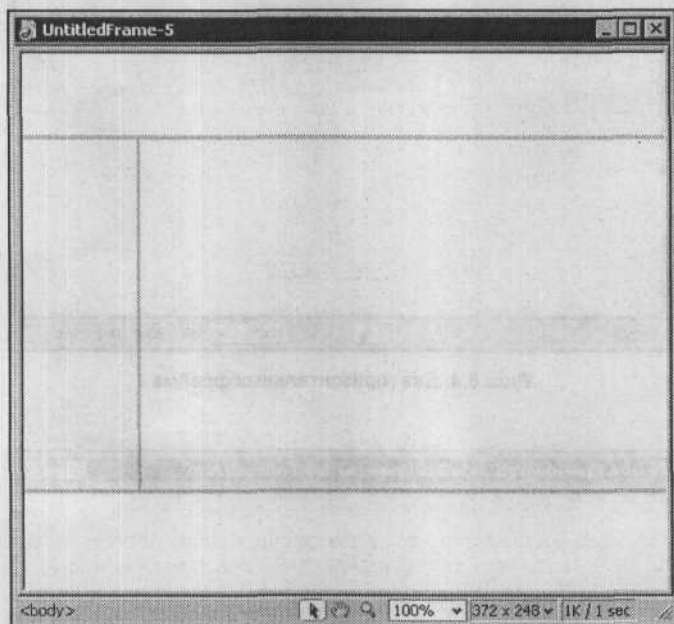


Рис. 8.6. Готовый набор фреймов

Теперь дадим готовой странице с набором фреймов название **Сайт студента Васильева В. В.** и сохраним ее, выбрав пункт **Save All** меню **File**, — это позволит сохранить также страницы, на которых размещено содержимое остальных фреймов. На экране появится стандартное диалоговое окно сохранения файла **Windows**, предлагающее нам сохранить сам набор фреймов под именем **UntitledFrameset-1.htm**. Сохраним его в корневой папке нового сайта **Site2** под именем **default.htm**.

Теперь **Dreamweaver** будет поочередно предлагать нам сохранить содержимое каждого фрейма набора (т. е. **Web-страницы** с содержимым этого фрейма). При этом фрейм, содержимое которого сохраняется в данный момент, выделяется толстой штриховой линией (рис. 8.7), так что будьте внимательны!

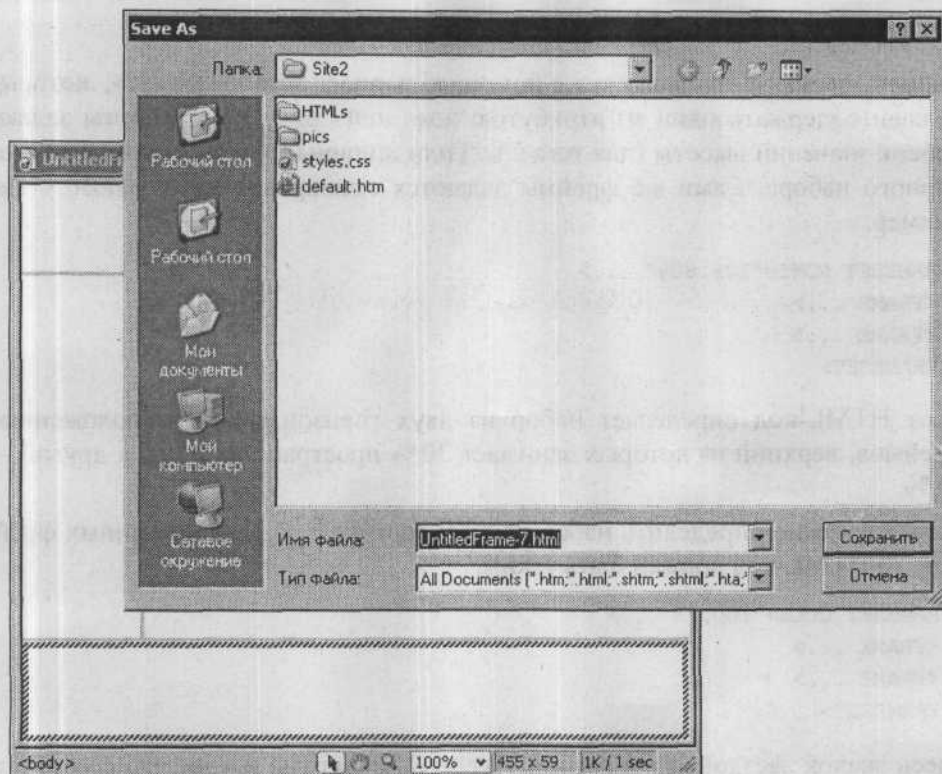


Рис. 8.7. Толстая штриховая линия выделяет фрейм, чье содержимое сохраняется в данный момент

Давайте сохраним все остальные страницы сайта в папке HTMLs, которую создадим в корневой папке Site2. Странице со сведениями об авторских правах дадим имя copyright.htm. Страницу с основным содержимым назовем main.htm, страницу со списком гиперссылок — nav.htm, а страницу с заголовком сайта — header.htm.

А теперь — другой способ создания фреймов. Самый быстрый, для лентяев и торопыг.

Посмотрим еще раз на рис. 8.6. И обратим внимание на толстую серую рамку, окружающую весь наш набор фреймов. Так вот, мы можем создавать новые фреймы, просто захватив эту рамку мышью и перетащив на то место, где должна находиться граница вновь создаваемых фреймов. И нам не нужна ни кнопка **Frames**, ни меню. Правда, диалоговое окно **Frame Tag Accessibility Attributes** при этом не появится, а значит, мы не сможем задать заголовки фреймов.

Формирование фреймов

Наборы фреймов описываются с помощью парного тега <FRAMESET>, который должен содержать один из атрибутов: ROWS или COLS. Эти атрибуты задают список значений высоты (для тега ROWS) или ширины (для COLS) всех фреймов данного набора. Сами же фреймы задаются одинарным тегом <FRAME>. Например:

```
<FRAMESET ROWS="20%,80%" ...>
  <FRAME ...>
  <FRAME ...>
</FRAMESET>
```

Этот HTML-код определяет набор из двух горизонтально расположенных фреймов, верхний из которых занимает 20 % пространства окна, а другой — 80 %.

Если же нужно определить набор из двух вертикально расположенных фреймов, то HTML-код должен быть таким:

```
<FRAMESET COLS="100,*" ...>
  <FRAME ...>
  <FRAME ...>
</FRAMESET>
```

Здесь значок звездочки (*) говорит Web-обозревателю о том, что соответствующий (правый) фрейм должен занять все оставшееся пространство набора (*"резиновый" фрейм*). Левый же фрейм будет иметь ширину 100 пикселей.

Теперь остановимся и немного подумаем. Из всего сказанного ранее выходит, что один набор фреймов может содержать только фреймы, расположенные по горизонтали, либо только фреймы, расположенные по вертикали. Но ведь в нашем наборе есть и горизонтальные, и вертикальные фреймы! В чем же дело?

А дело в том, что стандарт HTML предусматривает создание так называемых *вложенных наборов фреймов*. Один набор фреймов просто вкладывается в другой, как показано ниже:

```
<FRAMESET ROWS="20%,80%" ...>
  <FRAME ...>
  <FRAMESET COLS="100,*" ...>
    <FRAME ...>
    <FRAME ...>
  </FRAMESET>
</FRAMESET>
```

Осталось только сказать, что тег `<FRAME>` поддерживает некоторые атрибуты, которые стоит описать. Атрибут `SRC` задает интернет-адрес Web-страницы, которая будет загружена в соответствующий фрейм по умолчанию. А атрибут `TITLE` задает заголовок фрейма.

Тег `<FRAME>` поддерживает еще один атрибут — `NAME`, задающий уникальное имя фрейма. Эти имена пригодятся нам, когда мы будем создавать остальные Web-страницы нашего сайта. Dreamweaver при создании фреймов генерирует эти имена автоматически, но будет лучше потом изменить их на, что называется, "говорящие". Как задавать имена фреймов, мы выясним чуть позже.

Параметры фреймов и наборов фреймов

И наборы фреймов, и отдельные фреймы имеют определенный набор параметров. Давайте выясним, что это за параметры и чем они могут нам помочь.

Параметры наборов фреймов

Чтобы получить доступ к параметрам наборов фреймов и отдельных фреймов, нужно сначала выделить нужный фрейм или набор. Лучше всего это проделывать в панели **Frames** (рис. 8.8). Чтобы вывести ее на экран, выберем в меню **Window** пункт **Frames** или нажмем комбинацию клавиш `<Shift>+<F2>`.

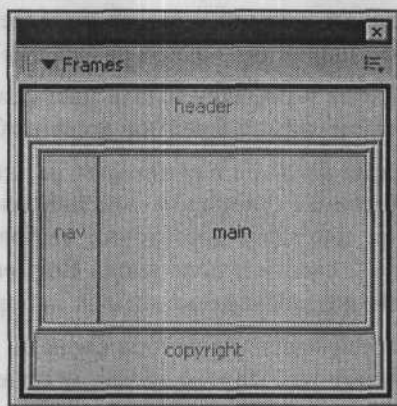


Рис. 8.8. Панель Frames

В этой панели схематично показана вся структура наборов фреймов. Фреймы изображены в виде серых прямоугольников, внутри которых написаны их имена. Надпись **(no name)** говорит о том, что у фрейма пока нет имени. Наборы же фреймов представлены в виде толстых темно-серых границ, окружающую структуру в них фреймы.

Мы можем щелкать мышью по границам наборов фреймов, чтобы выделить тот или иной набор. Также мы можем щелкать мышью и по самим фреймам, выделяя их. При этом редактор свойств отобразит параметры выделенного фрейма или набора фреймов.

Хорошо, выделять наборы фреймов мы научились. Давайте выделим какой-либо из них и посмотрим на редактор свойств. Мы увидим то, что показано на рис. 8.9.

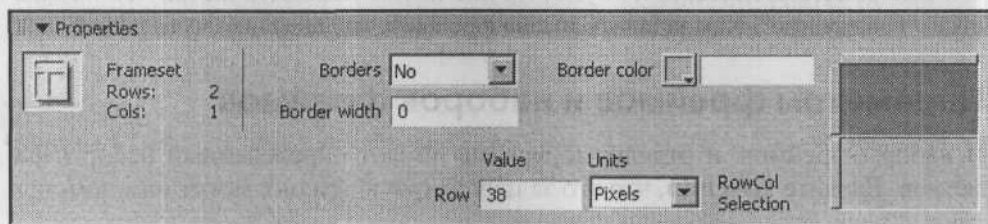


Рис. 8.9. Вид редактора свойств при выделении набора фреймов

Раскрывающийся список **Borders** позволяет задать наличие или отсутствие рамок между фреймами набора. Здесь доступны три пункта: **Yes** — рамки есть, **No** — рамок нет и **Default** — значение по умолчанию, зависящее от Web-обозревателя (как правило, рамки есть).

Если в списке **Borders** мы выберем пункт **Yes** или **Default**, то в поле ввода **Border width** сможем задать толщину рамок в пикселах. Мы также можем ввести в это поле нулевое значение — тогда рамки станут невидимыми.

Селектор цвета **Border color** позволяет задать цвет рамок. Разумеется, чтобы этот параметр имел действие, рамки фреймов должны быть видимыми.

Вообще, в настоящее время фреймы с видимыми рамками считаются дурным тоном. Сейчас в моде фреймы с невидимыми или очень тонкими рамками. Мы тоже сделаем такие, для чего достаточно выбрать в раскрывающемся списке **Borders** пункт **No** и ввести в поле ввода **Border width** ноль — на всякий случай. И сделаем это же для всех остальных наборов фреймов.

А пока обратимся к нижней части редактора свойств. Находящиеся там элементы управления позволят нам задать размеры отдельных фреймов выделенного набора.

В правой части редактора свойств мы видим схематичное изображение набора и содержащихся в нем фреймов. (Это похоже на уже знакомую нам панель **Frames**.) Мы можем выделить любой фрейм простым щелчком мыши; при этом выделенный фрейм будет окрашен темно-серым цветом.

В поле ввода **Value** устанавливается значение высоты или ширины фрейма. Это значение может быть задано в пикселах или процентах. Также мы можем

вести туда знак звездочки (*), обозначающий все остальное доступное пространство, оставшееся от других фреймов. Раскрывающийся список **Units** задает единицы измерения высоты или ширины фрейма. В нем доступны три пункта: **Pixels** — пиксели, **Percent** — проценты и **Relative** — все остальное пространство (аналогично вводу звездочки в поле **Value**).

Параметры фреймов

Теперь давайте выделим в панели **Frames** любой фрейм, скажем, самый верхний. Редактор свойств примет вид, приведенный на рис. 8.10.

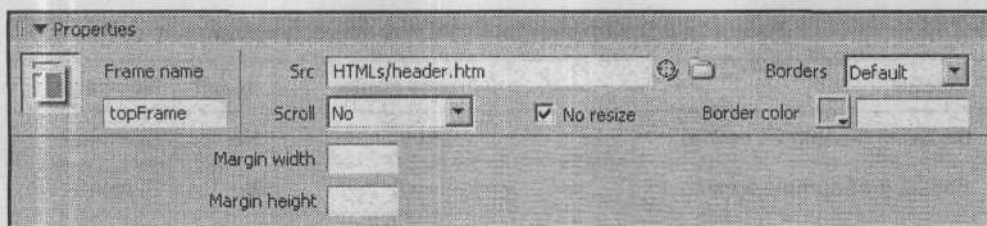


Рис. 8.10. Вид редактора свойств при выделении фрейма

В поле ввода **Frame name** вводится имя фрейма. Вот здесь-то мы и сможем его задать.

В поле ввода **Src** вводится интернет-адрес Web-страницы, отображаемой во фрейме. Обычно Dreamweaver сам заполняет это поле при создании и сохранении страницы набора фреймов.

Раскрывающийся список **Scroll** задает, будет ли фрейм содержать полосы прокрутки. Доступны четыре пункта:

- ◆ **Yes** — полосы прокрутки есть всегда;
- ◆ **No** — полос прокрутки нет даже тогда, когда содержимое фрейма не помещается в нем;
- ◆ **Auto** — полосы прокрутки появляются только тогда, когда в них появляется необходимость (содержимое фрейма в нем не помещается);
- ◆ **Default** — значение по умолчанию, зависящее от Web-обозревателя (как правило, аналогично **Auto**).

Включение флажка **No resize** запретит пользователю изменять размеры фреймов перетаскиванием их границ (*фиксированный фрейм*). Это полезно для фреймов, отображающих специальную информацию, таких как набор гиперссылок или заголовков сайта.

Раскрывающийся список **Borders** уже нам знаком. Однако его действие рас-

наличие или отсутствие рамок и у отдельного фрейма. Пункт **Default** этого меню позволит нам вернуться к параметрам набора фреймов.

В селекторе цвета **Border color** мы можем установить цвет рамок фрейма. А поля ввода **Margin width** и **Margin height** позволяют задать соответственно горизонтальный и вертикальный отступы между границами фрейма и его содержимым.

Давайте теперь поочередно выделим все четыре фрейма нашего набора и зададим для них имена, как указано в табл. 8.2. Остальные параметры оставим в положении по умолчанию.

Таблица 8.2. Параметры фреймов нашего набора

Фрейм	Frame name	Scroll	No resize
Верхний (название сайта)	header	No	Включен
Левый (набор гиперссылок)	nav	No	Включен
Основной (основное содержимое сайта)	main	Auto	Включен
Нижний (сведения об авторских правах)	copyright	No	Включен

Установив параметры наборов фреймов и самих фреймов, сохраним страницу, выбрав пункт **Save All** в меню **File**.

Создание Web-страниц для набора фреймов

Итак, набор фреймов создан. Осталось заполнить все его фреймы содержимым и создать остальные страницы нашего сайта: `about.htm`, `articles.htm`, `humour.htm` и `passions.htm`.

Заполнить фреймы набора содержимым очень просто. Dreamweaver фактически отображает во фрейме Web-страницу, которая будет в нем выводиться по умолчанию. Так что мы просто ставим в нужный фрейм текстовый курсор и вводим туда необходимый текст. Разумеется, мы его также форматируем, добавляем графику, если она нужна, и подключаем таблицу стилей. Как это делается, мы уже знаем.

Создать остальные страницы сайта также очень просто. Создаем новую страницу, вводим в нее текст и прочее содержимое и сохраняем в папке HTMLs, находящейся в корневой папке нашего нового сайта. Мы ведь договорились, что будем хранить Web-страницы там, не так ли?

Все! Набор фреймов заполнен содержимым, остальные страницы также созданы. Возникает проблема с гиперссылками, указывающими на

цы. (Как вы помните, набор этих гиперссылок должен находиться в левом фрейме с именем `nav`.) Какая проблема? А вот такая.

Давайте откроем готовую страницу с набором фреймов в Web-обозревателе и щелкнем по любой гиперссылке набора. О, ужас — страница, на которую ссылается гиперссылка, загрузится и займет все окно Web-обозревателя вместо того, чтобы "втиснуться" в отведенный ей фрейм. А нам ведь не это нужно!

Для того чтобы гиперссылки открывали соответствующую Web-страницу в нужном нам фрейме, надо правильно установить их цель. Возьмемся в Dreamweaver, щелкнем мышью на любой гиперссылке и откроем раскрывающийся список **Target**, расположенный в редакторе свойств. Помимо уже знакомых нам по занятию 3 пунктов, там будут перечислены имена всех созданных нами фреймов. Просто выбираем для каждой гиперссылки фрейм `main` (основное содержимое) — и все будет работать как надо.

Недостатки фреймов и их преодоление

Что ж, о фреймах мы узнали все. Осталось сказать, какие преимущества они дают Web-дизайнеру и какие недостатки несут в себе.

Преимущества фреймов налицо. Во-первых, не нужно дублировать название сайта, набор гиперссылок и сведения об авторских правах на всех страницах сайта — достаточно поместить их один раз в соответствующие фреймы, где они и останутся. Во-вторых, сами страницы уменьшатся в размерах и станут быстрее загружаться, чему будут рады пользователи, выходящие в Сеть по медленным каналам.

Недостатков же у фреймов два, по крайней мере, основных. И сейчас мы их рассмотрим.

Недостаток первый: нельзя задать название каждой странице сайта. Web-обозреватель все время будет отображать название страницы, содержащей набор фреймов. (Чтобы задать его, нужно выбрать в панели **Frames** самый внешний набор фреймов и ввести название в уже знакомое нам поле ввода, показанное на рис. 3.3 и находящееся в инструментарии документа.) Вряд ли этот недостаток будет для нас существенным.

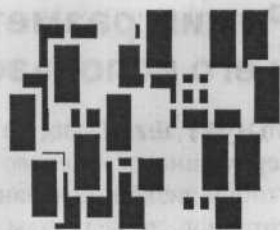
Недостаток второй: страницы сайтов, основанных на наборах фреймов, очень плохо обрабатываются поисковыми машинами. Согласитесь, это уже серьезно — наш сайт может и не появиться в таких известных поисковиках, как Google (<http://www.google.ru>), Яндекс (<http://www.yandex.ru>) и пр. И регистрировать наш сайт в поисковиках или каталогах нам придется самим, вручную.

На этом занятии, посвященное фреймам и фреймовому Web-дизайну, закончено.

На следующем занятии...

Разобравшись с фреймами и научившись использовать их для создания сайтов, двинемся дальше. Следующая остановка на нашем пути — табличный Web-дизайн. Мы будем создавать красивые, наполненные разнообразным содержимым странички, используя обычные таблицы, с которыми познакомились еще на *занятии 6*. В самом деле, почему бы и нет?

ЗАНЯТИЕ 9



Табличный дизайн

Web-дизайнерская мысль не стоит на месте. Казалось бы, еще совсем недавно в Интернете царили фреймы — и вот уже они практически сошли со сцены, уступив первое место таблицам. Сейчас в моде *табличный Web-дизайн*, в течение какой-то пары лет практически вытеснивший всех своих предшественников.

Рассказывать о табличном Web-дизайне особо нечего. Web-страница создается на основе большой таблицы (так называемой *таблицы разметки*), в ячейки (они называются, по аналогии с таблицей разметки, *ячейками разметки*) которой помещается и название сайта, и набор гиперссылок, и сведения об авторских правах и, разумеется, основное содержимое. Ничем эти таблицы не отличаются от тех, что мы создавали на *занятии 6*, кроме, может быть, чрезмерной сложности.

Чем привлекают таблицы Web-дизайнеров? Прежде всего, гибкостью. Web-страницы разнообразного внешнего вида с их помощью создавать гораздо проще, чем с помощью наборов фреймов. К тому же, таблицы предоставляют больше возможностей оформления, особенно если использовать стили CSS. Ради всего этого можно пренебречь некоторыми недостатками таблиц HTML, а именно громоздким HTML-кодом и долгой загрузкой.

Ну да ладно! Раз мы уже знаем о таблицах все (ну, почти все), давайте сразу же приступим к делу. Создадим новую, третью по счету, версию сайта студента Васильева. Для этого нам понадобится новая папка, которой мы дадим имя Site3; в этой папке мы создадим папки HTMLs и pics. И не забудем зарегистрировать новый, пока еще не существующий сайт site 3 в Dreamweaver.

Режим разметки страниц и его использование

Мы уже знаем, как создавать таблицы. (Забывчивые могут узнать об этом, вернувшись к *занятию 6*.) Казалось бы, ничего сложного в этом нет. Берем за основу дизайн, показанный на рис. 8.1, создаем таблицу из трех строк и двух столбцов, объединяем ячейки верхней и нижней строк и задаем размеры строк и столбцов. После этого нам остается поместить в ячейки готовой таблицы разметки содержимое и оформить его, как нам хочется. Все, первая страница нового сайта готова, а остальные не вызовут у нас никаких затруднений.

Да, никаких затруднений, если таблица разметки, на которой основан дизайн наших страниц, достаточно проста. А если в ней десяток строк и столько же столбцов, причем все разные? Вот в таких случаях нам очень поможет так называемый *режим разметки страниц*. В этом режиме мы можем просто нарисовать мышью нужные нам ячейки и таблицы разметки, а Dreamweaver сгенерирует для нас соответствующий HTML-код. А потом переключимся в обычный режим и заполним ячейки содержимым. Что может быть проще!

Режим разметки страниц

Раз уж Dreamweaver так хорош, что может создать для нас полноценную таблицу разметки, давайте усложним ему задачу. Добавим на главную страницу нашего нового сайта раздел новостей. После этого дизайн главной страницы будет выглядеть так, как показано на рис. 9.1.

Заголовок сайта		
Набор гиперссылок для перехода между страницами сайта	Полезное содержимое сайта	Новости сайта
Сведения о правах разработчика сайта		

Рис. 9.1. Классический табличный дизайн с колонкой новостей сайта

Создадим новую Web-страницу default.htm, сделаем ее страницей по умолчанию (как это сделать, мы увидим на следующем занятии).

мента. Зададим ее название: все тот же Сайт студента Васильева В. В. Теперь мы готовы.

Чтобы переключиться в режим разметки страниц, нажмем кнопку-переключатель **Layout**, расположенную на вкладке **Layout** инструментария объектов. Эта кнопка входит в группу кнопок переключения режимов отображения таблиц. Кнопка **Standard** выполняет переключение в обычный режим отображения таблиц, а кнопка **Extended** — в режим отображения увеличенных таблиц.

Для переключения в режим разметки также можно выбрать пункт **Layout Mode** в подменю **Table Mode** меню **View** или нажать комбинацию клавиш **<Ctrl>+<F6>**. Для обратного переключения выберем пункт **Standard Mode** в том же подменю или еще раз нажмем **<Ctrl>+<F6>**.

В любом случае после этого Dreamweaver выдаст увесистое предупреждение, описывающее, что мы можем делать в режиме разметки. Прочитаем его (или же не будем читать) и закроем, нажав кнопку **Done**. Чтобы Dreamweaver больше не надоедал нам этим предупреждением, перед его закрытием включим флажок **Don't show me this message again** — лучше всего так и сделать.

Самое время включить отображение *измерительных линеек*! Эти линейки, проградуированные в одной из единиц измерения, располагаются вдоль границ окна документа. Они позволят нам в дальнейшем более точно позиционировать фрагменты содержимого страницы. Чтобы включить их, выберем в подменю **Rulers** меню **View** пункт-выключатель **Show** или нажмем комбинацию клавиш **<Ctrl>+<Alt>+<R>**. Чтобы убрать линейки, отключим этот пункт. Пункты-переключатели **Pixels** (Пиксели), **Inches** (Дюймы) и **Centimeters** (Сантиметры) того же подменю **Rulers** меню **View** позволят нам выбрать единицу измерения.

Теперь обратим внимание на сами линейки. В месте их пересечения в левом верхнем углу находится *точка начала отсчета*, "ноль" системы координат, иначе говоря, точка с координатами (0, 0). По умолчанию она находится именно там — в левом верхнем углу. Но мы можем переместить ее в любое другое место; для этого достаточно "захватить" ее мышью и перетащить куда надо. Как только мы отпустим кнопку мыши, шкалы на линейках изменятся и станут отсчитывать все координаты от вновь заданного "нуля". Иногда это бывает полезно. А чтобы вернуть начало координат на свое законное место — в левый верхний угол окна, достаточно выбрать пункт **Reset Origin** подменю **Rulers** меню **View**.

У Dreamweaver есть еще одно средство помочь нам разместить фрагменты содержимого в нужном месте — *координатная сетка*. Чтобы отобразить ее,

жмем комбинацию клавиш <Ctrl>+<Alt>+<G>. Чтобы убрать сетку, нужно отключить этот пункт.

Можно сделать так, чтобы фрагменты содержимого, рисуемые мышью, "приклеивались" к линиям сетки. Это позволит нам более точно позиционировать их по делениям шкал. Чтобы включить "приклеивание", нужно включить пункт **Snap to Grid** подменю **Grid** меню **View** или нажать комбинацию клавиш <Ctrl>+<Alt>+<Shift>+<G>.

Если нас не устраивает шаг координатной сетки, мы можем его изменить. Для этого выберем пункт **Grid Settings** (Настройка сетки) подменю **Grid** меню **View**. На экране появится диалоговое окно **Grid Settings**, показанное на рис. 9.2.

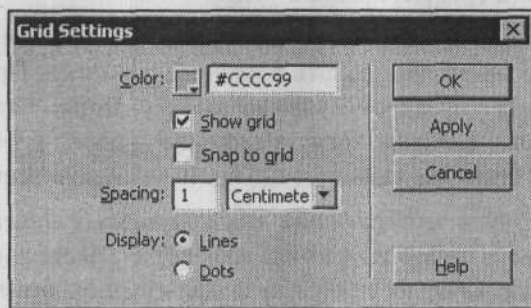


Рис. 9.2. Диалоговое окно **Grid Settings**

С помощью селектора цвета **Color** задается цвет линий сетки.

Флажок **Show grid** включает или отключает показ сетки. Он аналогичен пункту **Show grid** подменю **Grid** меню **View**.

Флажок **Snap to grid** включает или отключает "приклеивание" фрагментов содержания к линиям сетки. Он аналогичен пункту **Snap to grid** подменю **Grid** меню **View**.

Поле ввода **Spacing** задает шаг линий сетки — то, что нам больше всего нужно. Раскрывающийся список, расположенный справа от него, задает единицу измерения: **Pixels** (Пикселы), **Inches** (Дюймы) или **Centimeters** (Сантиметры).

Группа переключателей **Display** задает вид линий сетки. Переключатель **Lines** делает их такими, какими мы видим их сейчас, — сплошными линиями. Это поведение Dreamweaver по умолчанию. Переключатель **Dots** включает отображение точечных линий; в таком виде они меньше отвлекают внимание, однако хуже видны на экране.

Ну и, закончив настройку сетки, нужно нажать **OK** для сохранения заданных установок или **Cancel** для отмены от них. Мы можем также нажать **Apply**

чтобы применить заданные установки без закрытия диалогового окна **Grid Settings**.

Внешний вид окна документа в режиме разметки с включенными линейками и координатной сеткой с шагом в один сантиметр показан на рис. 9.3.

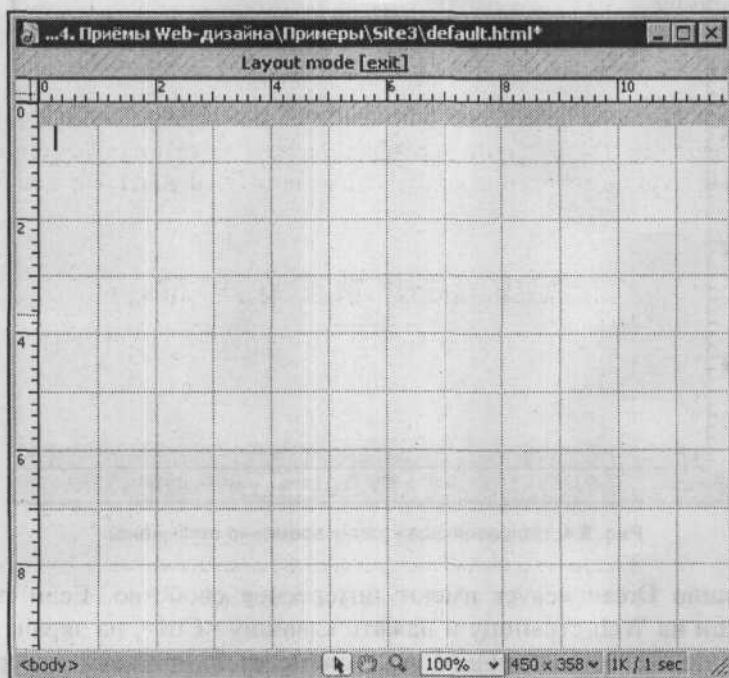


Рис. 9.3. Окно документа в режиме разметки с включенными линейками и сеткой

И еще один инструмент, припасенный Dreamweaver труженикам режима разметки, — *направляющие*. Это особые линии, сходные с линиями координатной сетки, но, в отличие от них, располагающиеся так, как нам нужно. Мы сами можем создавать направляющие и потом, когда они перестают быть нам полезными, удалять.

Прежде чем создать первую направляющую, нам будет нужно их включить. Проверим, включен ли пункт-выключатель **Show Guides** подменю **Guides** меню **View**, и, если нет, включим его. (Также можно нажать комбинацию клавиш <Ctrl>+<G>.) После этого помещаем курсор мыши на координатную линейку — горизонтальную или вертикальную, в зависимости от того, горизонтальную или вертикальную направляющую нам нужно создать. Далее нажимаем левую кнопку мыши и, не отпуская ее, протаскиваем мышью на нужное место страницы и отпускаем. Созданная нами таким образом направляющая показана на рис. 9.4

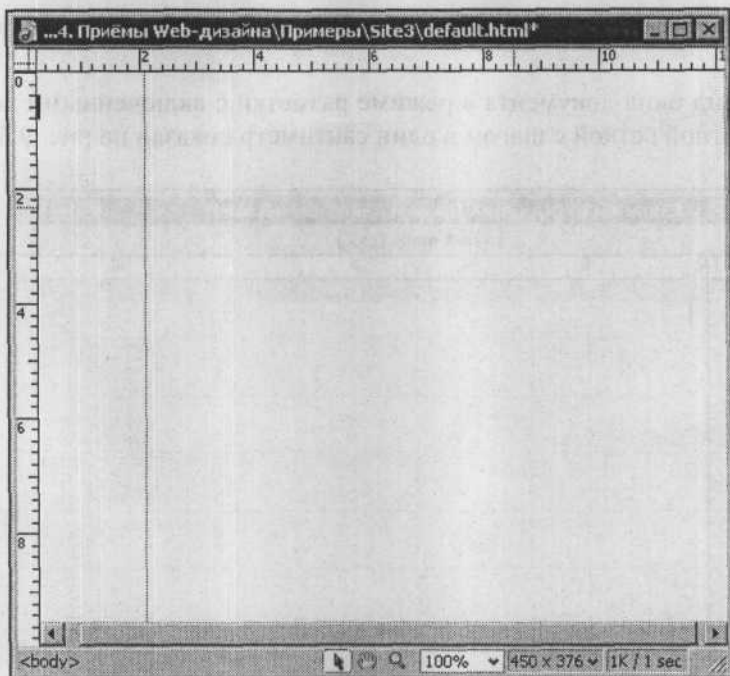


Рис. 9.4. Направляющая (сетка временно отключена)

Направляющие Dreamweaver имеют интересное свойство. Если поместить курсор мыши на Web-страницу и нажать клавишу <Ctrl>, на экране появятся размерные линии, показывающие расстояние от ближайшей к курсору направляющей до края окна (рис. 9.5). Как только мы отпустим клавишу <Ctrl>, размерные линии пропадут.

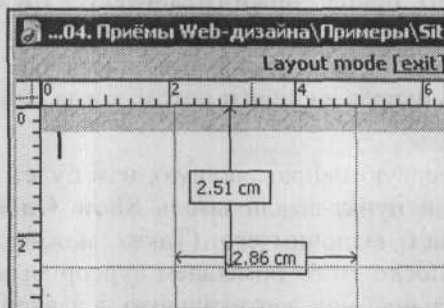


Рис. 9.5. Размерные линии

Мы можем переместить любую направляющую на другое место мышью. Чтобы удалить ненужную направляющую, достаточно перетащить ее на ли-

нейку. А чтобы удалить все направляющие, достаточно выбрать пункт **Clear Guides** подменю **Guides** меню **View**.

Если включить пункт-выключатель **Lock Guides** подменю **Guides** меню **View** или нажать комбинацию клавиш <Ctrl>+<Alt>+<S>, направляющие будут заблокированы, и их нельзя будет перетащить. Чтобы разблокировать направляющие, достаточно отключить этот пункт.

Можно включить "приклеивание" фрагментов содержимого страницы к направляющим, для чего достаточно включить пункт-выключатель **Snap to Guides** подменю **Guides** меню **View** или нажать <Ctrl>+<Shift>+<S>. А если включить пункт-выключатель **Guides Snap To Elements** подменю **Guides** меню **View** или нажать <Ctrl>+<Shift>+<E>, направляющие при их создании сами будут "приклеиваться" к фрагментам содержимого страницы.

Dreamweaver предоставляет возможность настроить некоторые параметры направляющих. Для этого достаточно выбрать пункт **Edit Guides** подменю **Guides** меню **View**. На экране появится диалоговое окно **Guides**, показанное на рис. 9.6.

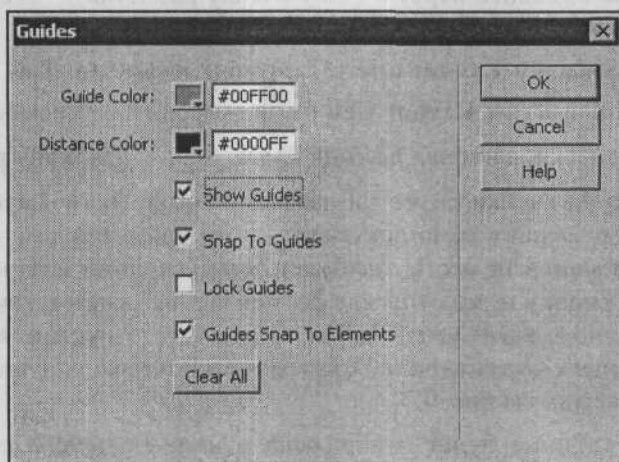


Рис. 9.6. Диалоговое окно **Guides**

С помощью селектора цвета **Guide Color** задается цвет направляющих, а с помощью селектора цвета **Distance Color** — цвет размерных линий.

Флажок **Show Guides** включает или отключает показ направляющих. Он аналогичен пункту **Show Guides** подменю **Guides** меню **View**.

Флажок **Snap To Guides** включает или отключает "приклеивание" фрагментов содержания к направляющим. Он аналогичен пункту **Snap To Guides** подменю **Guides** меню **View**.

Флажок **Lock Guides** включает или отключает блокировку направляющих. Он аналогичен пункту **Lock Guides** подменю **Guides** меню **View**.

Последний флажок — **Guides Snap To Elements** — включает или отключает "приклеивание" направляющих к фрагментам содержимого страницы. Он аналогичен пункту **Guides Snap To Elements** подменю **Guides** меню **View**.

Кнопка **Clear All** удаляет все направляющие.



Настроив направляющие, нужно нажать **OK** для сохранения заданных установок или **Cancel** для отказа от них. Мы можем также нажать **Apply**, чтобы применить заданные установки без закрытия диалогового окна **Guides**.

Внимание!

Направляющие не сохраняются вместе с Web-страницей.

Работа с таблицами и ячейками разметки

Ну, вот и все! Все подготовительные действия закончились. Пора браться за дело. Сейчас мы создадим первую нашу ячейку разметки — это будет ячейка основного содержимого.

Посмотрим на вкладку **Layout** инструментария объектов. Там расположены две кнопки: кнопка **Draw Layout Cell**  позволит нам поместить на страницу ячейку разметки, а кнопка **Layout Table**  — таблицу разметки.

Итак, для создания ячейки разметки нажмем кнопку **Draw Layout Cell** и переместим курсор мыши в окно документа — он примет форму крестика. Поставим курсор мыши в то место, где будет левый верхний угол нашей ячейки, нажмем левую кнопку и, не отпуская ее, протащим мышью в точку, где будет другой, противоположный угол ячейки, после чего отпустим кнопку. После этого Dreamweaver сформирует таблицу разметки и находящуюся в ней ячейку, которые показаны на рис. 9.7.

Получившаяся таблица будет отображена в виде большого серого прямоугольника с зеленой рамкой, а ячейка — в виде меньшего белого прямоугольника с синей рамкой, расположенного внутри серого. Кроме того, Dreamweaver ненавязчиво покажет нам структуру таблицы разметки — если вы присмотритесь, то заметите тонкие белые линии — границы других ячеек этой таблицы.

Теперь поместим курсор мыши где-либо внутри только что созданной ячейки и щелкнем. Граница этой ячейки будет подсвечена, а внутри нее появится текстовый курсор; это значит, что ячейка теперь выделена, и мы можем набирать в ней текст, размещать графические изображения, таблицы и все остальное, чем богат HTML.

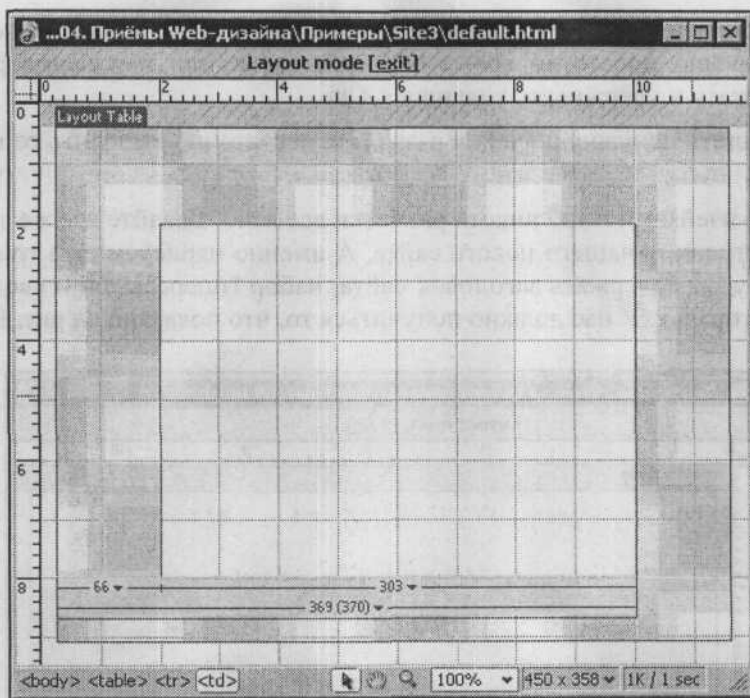


Рис. 9.7. Готовая ячейка основного содержимого и таблица разметки

Теперь поместим курсор мыши на границу ячейки так, чтобы она (граница) стала красной, и опять щелкнем. Ячейка будет выделена, и на ее границе появятся девять маркеров изменения размера. Пользуясь этими маркерами, мы можем изменять размеры ячейки. А чтобы переместить ячейку в другое место, достаточно просто "захватить" ее мышью за границу и перетащить куда нужно.

Для более точного перемещения ячеек разметки можно использовать клавиши-стрелки. При нажатии такой клавиши ячейка будет смещаться на один пиксел в соответствующем направлении. А если при нажатии клавиш-стрелок удерживать клавишу `<Shift>`, ячейка будет смещаться сразу на 10 пикселей.

Если же ячейку по каким-либо причинам нельзя перенести на это место или изменить ее размеры, Dreamweaver очень красиво вернет все на свои места. Это может случиться, например, если мы пытаемся наложить одну ячейку на другую. Таблицы HTML не позволяют накладывать ячейки друг на друга, и было бы странно, если бы они такое позволяли...

Заметим также, что при перетаскивании или изменении размеров ячейки ее граница будет "приклеиваться" к линиям сетки, если, конечно, мы включили

режим "приклеивание". Чтобы временно отключить этот режим, не забираясь в меню, нужно просто во время перетаскивания или изменения размеров ячейки нажать и удерживать клавишу <Alt>.

Чтобы удалить ненужную ячейку разметки, достаточно выделить ее и нажать .

Хорошо, с ячейками и таблицами разметки все ясно. Давайте все же закончим главную страницу нашего нового сайта. А именно нарисуем еще три ячейки, которые будут содержать заголовок сайта, набор гиперссылок и сведения об авторских правах. У нас должно получиться то, что показано на рис. 9.8.

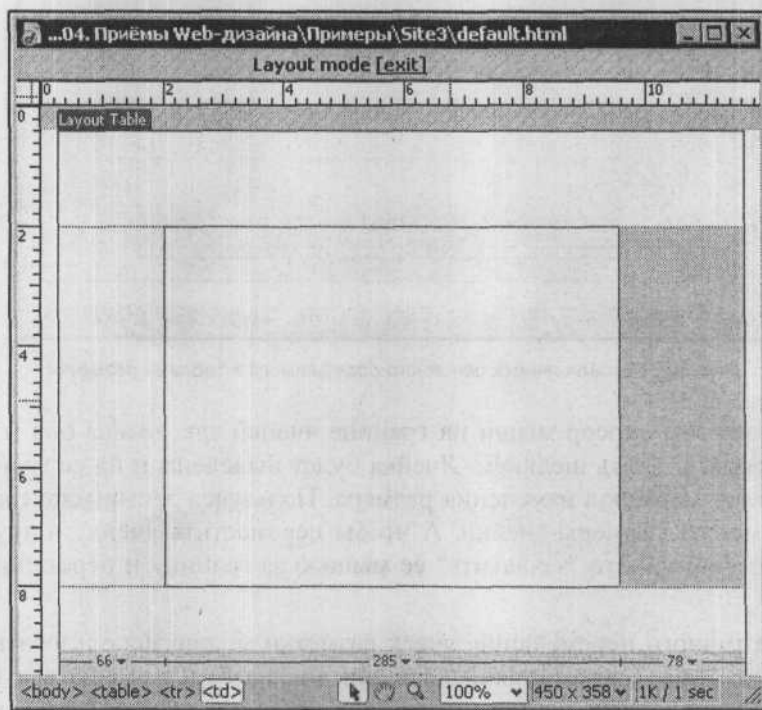


Рис. 9.8. Четыре ячейки разметки (сетка временно отключена)

Мы специально оставили свободное пространство в правой части получившейся таблицы разметки. Там мы поместим еще одну ячейку разметки, которая будет содержать новости сайта.

Но мы сделаем немного по-другому. Мы создадим на оставленном свободном месте таблицу разметки, а уже в ней поместим сам текст новостей. Зачем это нужно, мы узнаем на *занятии 10*. А заодно и попрактикуемся в создании таблиц разметки.

Создание таблицы разметки почти ничем не отличается от создания ячейки разметки за тем исключением, что щелкать нужно кнопку **Layout Table**. Рисуется таблица так же, как ячейка. Существуют, правда, несколько правил, которым нужно следовать при размещении на странице новой таблицы разметки.

- ◆ Если нарисованная таблица находится внутри другой, то она становится вложенной; Dreamweaver сам сформирует необходимую ячейку во внешней таблице.
- ◆ Если же нарисовать новую таблицу разметки вокруг уже существующих ячеек разметки, они становятся частью новой таблицы.
- ◆ Поместить таблицу разметки внутри ячейки разметки невозможно — это можно сделать только на свободном пространстве внутри внешней таблицы или вне ее.

На заметку

Вообще, сам Dreamweaver подскажет нам, можно ли на данном месте поместить ячейку или таблицу разметки. Если курсор мыши имеет вид крестика, то поместить на этом месте элемент разметки можно, если же он примет вид переречкнутого круга — нельзя.

Нарисуем на свободном месте таблицу разметки. Внутри нее создадим три ячейки:

- ◆ верхняя — узкая, под заголовок колонки новостей;
- ◆ центральная — самая большая, под собственно новости;
- ◆ нижняя — также узкая, под гиперссылку на страницу с архивом устаревших новостей ("старостей").

То, что должно у нас получиться, изображено на рис. 9.9.

А теперь давайте поближе присмотримся к обеим таблицам разметки — внешней и внутренней.

Первое — в левом верхнем углу таблицы разметки находится ее *заголовок*, имеющий вид небольшого зеленого ярлычка (на рис. 9.10 он серый). Если щелкнуть по нему, таблица будет выделена. Выделить таблицу можно так же, щелкнув по ее границе, но этот способ срабатывает только тогда, когда граница таблицы не совпадает с границей ячейки или другой таблицы. Так что лучше уж всегда щелкать по заголовку.

Вдоль верхнего и нижнего краев таблицы располагаются уже знакомые нам строки размеров. В ней обозначены значения ширины ячеек таблицы и самой таблицы. При работе со страницами, сделанными на основе табличного Web-дизайна, эти числа очень нам пригодятся.

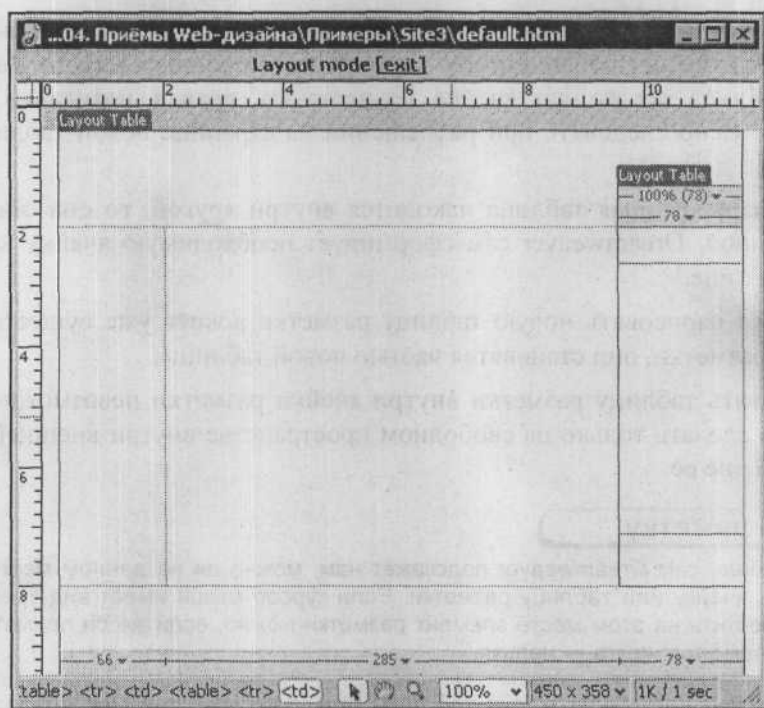


Рис. 9.9. Готовая разметка страницы (сетка временно отключена)

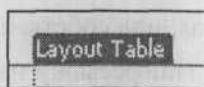


Рис. 9.10. Заголовок таблицы разметки

Может получиться так, что в строке размеров для какой-то ячейки будут отображаться два значения ширины, причем одно из них будет в скобках. Обычно это случается, когда в слишком узкую ячейку помещают слишком широкое содержимое, и Dreamweaver будет вынужден расширить эту ячейку. Так вот, первое значение в этом случае обозначает реальную ширину ячейки, а второе — ширину, заданную при разработке страницы. Если такое случится, лучше всего изменить ширину ячейки так, чтобы оба эти значения совпали; как это проще всего сделать, мы узнаем потом.

Мы можем делать с таблицами разметки все то же, что и с ячейками: изменять размеры, перемещать на другое место, удалять. При этом перемещать таблицу разметки проще всего за заголовок (см. рис. 9.10).

Форматирование таблиц и ячеек разметки

Когда мы находимся в режиме разметки страниц, Dreamweaver позволяет нам задать для таблиц и ячеек разметки довольно много параметров. Сейчас самое время о них поговорить.

Параметры ячеек разметки

Давайте выделим какую-либо ячейку разметки и посмотрим на редактор свойств. На рис. 9.11 показано то, что Dreamweaver нас приготовил.

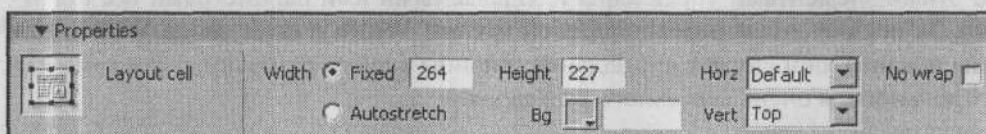


Рис. 9.11. Вид редактора свойств при выделении ячейки разметки

Группа переключателей **Width** позволит нам установить ширину ячейки. Переключатель **Fixed** и расположенное справа от него поле ввода **Width** задают фиксированную ширину ячейки. Такие ячейки очень часто используются для размещения служебных элементов: набора гиперссылок, тех же новостей сайта и пр.

А вот другой переключатель в этой группе — **Autostretch** — задает режим *саморастягивания* ячейки, т. е. ячейка будет занимать все пространство таблицы, оставшееся от других ячеек. Такие ячейки еще часто называются "*резиновыми*".

Внимание!

В таблице может быть только одна саморастягивающаяся ячейка.

В поле ввода **Height** задается высота ячейки.

Селектор цвета **Bg** задает цвет фона ячейки.

Раскрывающийся список **Horz** позволяет задать горизонтальное выравнивание содержимого ячейки. Здесь доступны четыре пункта:

- ◆ **Default** — выравнивание по умолчанию, обычно по левому краю;
- ◆ **Left** — выравнивание по левому краю;
- ◆ **Center** — по центру;
- ◆ **Right** — по правому краю.

Аналогично, раскрывающийся список **Vert** задает вертикальное выравнивание текста в ячейке.

Здесь доступно пять пунктов:

- ◆ **Default** — выравнивание по умолчанию, обычно посередине;
- ◆ **Top** — выравнивание по верху;
- ◆ **Middle** — посередине;
- ◆ **Bottom** — по низу;
- ◆ **Baseline** — по базовой линии.

Флажок **No wrap** запрещает перенос строк внутри ячейки.

С этими элементами управления и задаваемыми ими параметрами все понятно. За исключением переключателей группы **Width** и поля ввода **Width**. Как мы помним, они служат для задания ширины ячейки разметки, а это тема для отдельного, и очень серьезного разговора.

Задание ширины ячеек

В справке Dreamweaver рекомендуется задавать ширину ячеек только после заполнения их содержимым. По крайней мере, выставлять окончательные значения ширины лучше именно тогда: мы сможем сами увидеть, как будет выглядеть готовая страница. Но предварительно ширину можно выставить и до заполнения ячеек, что мы сейчас и сделаем.

Итак, ячейки могут иметь фиксированную ширину или быть саморастягивающимися. (К сожалению, в режиме разметки страниц Dreamweaver не позволяет сделать ячейку саморастягивающейся по высоте.) Только одна ячейка в таблице может быть саморастягивающейся; значения ширины остальных ячеек должны быть фиксированы. И это логично.

Давайте сделаем центральную ячейку внешней таблицы разметки, где будет отображаться основное содержимое, саморастягивающейся. Делается это очень просто: выделяем эту ячейку, включаем переключатель **Autostretch** в редакторе свойств и... Вот это фокус! Что Dreamweaver еще от нас нужно (рис. 9.12)?

Чтобы разобраться во всем этом, придется повести разговор издалека. Из *занятия 6* мы помним, что значение ширины (да и высоты) ячейки, проставленное в HTML-коде, зачастую является для Web-обозревателя не руководством к действию, а просто рекомендацией, которую можно и проигнорировать. И игнорируют! Да так, что из тщательно сделанной таблицы разметки получается бог весть что.

Значит, нужно каким-то образом отдать Web-обозревателю приказ: "держат ширину ячейки именно такой, какая указана в коде HTML, и не своевольничать". Лучший способ это сделать — вставить в ячейку графическое изображение, имеющее высоту в один пиксел и ширину, равную нужной нам шири-

не ячейки. Это изображение будет как бы "распирать" ячейку изнутри, не давая Web-обозревателю ее сузить. Оно, кстати, так и называется — *изображение-"распорка"*.

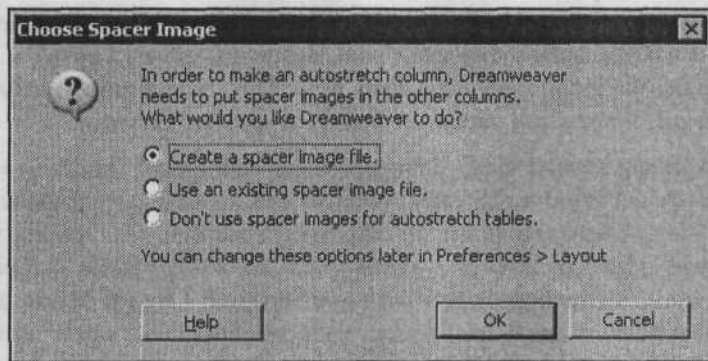


Рис. 9.12. Диалоговое окно **Choose Spacer Image**

Такое изображение-"распорка" помещается Dreamweaver в каждую ячейку таблицы разметки, как только мы делаем какую-то из них саморастягивающейся (в последнюю "распорка", разумеется, не помещается). Но сначала Dreamweaver должен узнать, какое именно изображение-"распорку" мы собираемся использовать. И, как только мы включаем переключатель **Autostretch**, на экране появляется диалоговое окно **Choose Spacer Image**, позволяющее выбрать изображение-"распорку" (см. рис. 9.12).

Мы можем выбрать один из трех переключателей:

- ◆ **Create a spacer image file** — изображение-"распорку" должен создать сам Dreamweaver. Этот переключатель следует выбирать, если мы не заготовили файл "распорки" самостоятельно (а так чаще всего и бывает);
- ◆ **Use an existing spacer image file** — использовать созданное ранее изображение-"распорку". После выбора этого переключателя и нажатия кнопки **OK** на экране появится диалоговое окно **Select File**, в котором мы сможем выбрать нужный файл;
- ◆ **Don't use spacer images for autostretch tables** — не использовать изображения-"распорки" вообще. Этот переключатель следует выбирать, если мы сами собираемся принять меры против самовольного изменения ширины ячеек Web-обозревателем, например, поместить туда содержимое, заполняющее ячейку целиком, или свое собственное изображение-"распорку". Если выбрать этот переключатель, Dreamweaver предупредит, что без принятия специальных мер таблица разметки может измениться до неузнаваемости.

Поскольку у нас нет файла изображения-"распорки", давайте прикажем Dreamweaver создать его. Выберем первый переключатель и нажмем **OK**. После этого на экране появится диалоговое окно **Save Spacer Image File As**, напоминающее стандартное диалоговое окно сохранения файла Windows. Проверим, выбран ли в раскрывающемся списке **Relative to** пункт **Site Root** (задание пути и имени файла относительно корневой папки сайта), и сохраним файл изображения-"распорки" в папке **pics**. По умолчанию он будет называться **spacer.gif**, но мы при желании можем изменить его имя.

После того как мы задали файл изображения-"распорки" (создали новый или выбрали уже существующий), Dreamweaver станет его использовать, больше нас об этом не спрашивая. Если мы захотим изменить имя файла изображения-"распорки" или создать его заново, то должны будем выбрать пункт **Preferences** меню **Edit**, переключиться в категорию **Layout Mode** диалогового окна **Preferences**, стереть имя файла изображения-"распорки" в поле ввода **Image file** и нажать кнопку **OK**. После этого Dreamweaver при создании саморастягивающейся ячейки снова выведет на экран диалоговое окно **Choose Spacer Image**.

Теперь обратим внимание на строку размеров внешней таблицы разметки. Ее части, относящиеся к крайним ячейкам (они имеют фиксированную ширину), показаны на рис. 9.13, — их вид означает, что внутрь такой ячейки помещено изображение-"распорка". А часть, относящаяся к саморастягивающейся ячейке, имеет вид, представленный на рис. 9.14.

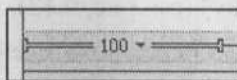


Рис. 9.13. Обозначение ячейки с фиксированной шириной и помещенной в нее "распоркой" в строке размеров

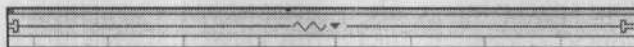


Рис. 9.14. Обозначение саморастягивающейся ячейки в строке размеров

Чтобы вернуть саморастягивающейся ячейке таблицы фиксированную ширину, достаточно включить переключатель **Fixed** в редакторе свойств и, если нужно, задать ее ширину в поле ввода **Width**.

Есть еще один способ преобразования ячейки из фиксированной в саморастягивающуюся и наоборот — воспользоваться меню строки размеров. Для этого служит пункт **Make Column Autostretch**. Чтобы вернуть саморастягивающейся ячейке фиксированную ширину, нужно выбрать пункт **Make Column Fixed Width**.

Меню строки размеров также содержит ряд пунктов, позволяющих нам произвести различные манипуляции с изображениями-"распорками".

- ◆ **Add Spacer Image** — принудительная вставка в ячейку изображения-"распорки".
- ◆ **Remove Spacer Image** — удаление изображения-"распорки", причем ширина ячейки остается фиксированной. В этом случае нам самим придется позаботиться о том, чтобы ширина ячейки не была изменена Web-обозревателем — самим вставить в нее изображение-"распорку" или иное содержимое, не дающее ячейке "съежиться".
- ◆ **Remove All Spacer Images** — удаление всех изображений-"распорок" во всех ячейках данной таблицы разметки. Значения ширины всех этих ячеек также остаются фиксированными.

Вот и все о задании ширины ячеек.

Параметры таблицы разметки

Давайте теперь посмотрим, что мы сможем сделать с таблицей разметки. Выделим любую из двух созданных нами таблиц разметки и взглянем на редактор свойств (рис. 9.15).

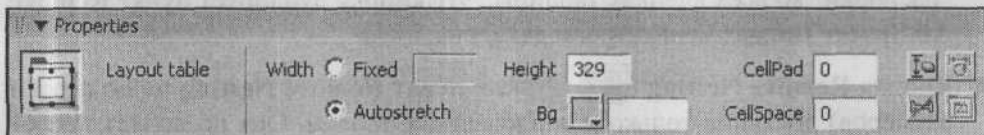


Рис. 9.15. Вид редактора свойств при выделении таблицы разметки

Здесь многое нам уже знакомо. Группа переключателей **Width**, поля ввода **Width** и **Height** выполняют точно такие же функции, что и их "коллеги", относящиеся к ячейкам разметки. Сразу понятно, таблицы разметки тоже могут быть саморастягивающимися (по умолчанию) или иметь фиксированную ширину.





Внимание!

Таблицы с фиксированной шириной не могут иметь саморастягивающиеся ячейки.

Селектор цвета **Bg** задает цвет фона таблицы.

Поля ввода **CellPad** и **CellSpace** задают отступы соответственно между границей ячейки и ее содержимым и между границами соседних ячеек. По умолчанию они равны нулю.

В правой части редактора свойств расположены четыре кнопки, позволяющие выполнить некоторые манипуляции с таблицей разметки.

- ◆ Кнопка **Clear Row Heights**  позволяет удалить значения высоты строк таблицы, проставленные в коде HTML. (Также можно воспользоваться пунктом **Clear Cell Heights** в меню строки размеров.) После этого каждая строка примет такую высоту, чтобы полностью вместить содержимое всех своих ячеек; фактически при этом таблица разметки будет иметь в окне Web-обозревателя минимальную возможную высоту.
- ◆ Кнопка **Make Cell Widths Consistent**  задает такие значения ширины ячеек таблицы, чтобы они вместили все свое содержимое. (Аналогично действует пункт **Make Cell Widths Consistent** меню строки размеров.) Это может помочь, если мы жестко задали ширину ячеек, но потом поместили в них слишком широкое содержимое. Dreamweaver при этом увеличит ширину ячеек, а в строке размеров покажет два значения ширины. Значение, показанное вне скобок, отображает реальную ширину ячейки, а значение в скобках — заданную нами. Нажатие кнопки **Make Cell Widths Consistent** поможет это исправить.
- ◆ Кнопка **Remove All Spacer Images**  удаляет все изображения-"распорки" во всех ячейках таблицы. Эту кнопку дублирует пункт **Remove All Spacer Images** меню строки размеров.
- ◆ Кнопка **Remove Nesting** , а также пункт **Remove Nesting** меню строки размеров) доступна только для вложенных таблиц. Она позволяет преобразовать вложенную таблицу в набор ячеек внешней таблицы; при этом содержимое ячеек не теряется. Иногда это позволяет уменьшить сложность HTML-кода, формирующего таблицу разметки, а значит, страница будет быстрее загружаться и выводиться Web-обозревателем.

Ну и, если в режиме разметки страниц Dreamweaver что-то не позволяет сделать, мы всегда можем переключиться в режим отображения обычных таблиц и подправить параметры таблиц разметки, их строк и ячеек там. Как это делается, мы изучали еще на *занятии 6*.

Создание страниц, использующих табличный Web-дизайн

Ну что ж, главную страницу, точнее, таблицу разметки мы закончили. Теперь нужно наполнить ее содержимым и создать остальные страницы нашего третьего по счету сайта.

С главной страницей проблем у нас быть не должно. Вводить и форматировать текст, помещать на Web-страницы графику и таблицы мы научились еще на курсах 1 и 2. Так что поговорим о создании остальных страниц сайта.

Конечно же, нам совсем не улыбается каждый раз заново рисовать таблицу разметки. Это уже не говоря о том, что допустить ошибку, скажем, сделать какую-то ячейку уже или не поставить в нее изображение-"распорку", проще простого. Как же быть? А вот как.

1. Открываем в Dreamweaver полностью законченную главную страницу сайта, если она еще не открыта.
2. Пересохраняем главную страницу под другим именем, например, about.htm (это, как мы помним, сведения об авторе). Для этого достаточно выбрать пункт **Save As** в меню **File** или нажать <Ctrl>+<Shift>+<S>.
3. Удаляем на странице about.htm колонку новостей сайта, т. е. вложенную таблицу разметки и ячейку внешней таблицы разметки, в которую она вложена. Колонка новостей уместна только на главной странице сайта, на остальных же она совершенно лишняя.
4. Вписываем на страницу about.htm новое содержимое (т. е. сведения об авторе — студенте Васильеве).
5. Сохраняем готовую страницу.
6. Повторяем пункты 2, 4 и 5 этого списка для всех страниц нашего сайта.

Неплохо придумано! Мы заимствуем общие элементы с другой страницы и создаем заново только ее содержимое. Это один из способов облегчить себе работу, за которые мы должны благодарить ленивых Web-дизайнеров.

Недостатки табличного дизайна и их преодоление

Теперь разбавим дегтем оставшийся мед. А именно поговорим о недостатках табличного Web-дизайна.

Вообще, добрая половина недостатков табличного Web-дизайна происходит от недостатков самих таблиц HTML. Давайте перечислим сначала их и сразу же наметим пути их преодоления, полного или частичного.

Недостаток таблиц (и, следовательно, Web-страниц, созданных на основе таблиц разметки) номер один — это медленная загрузка. HTML-код, описывающий таблицы, очень громоздок. Посмотрите на код, описывающий нашу таблицу разметки, — настоящее нагромождение тегов, в которых не сразу разберется даже опытный Web-дизайнер. А ведь наша таблица не очень-то и сложная!

Недостаток номер два — это очень медленная обработка таблиц некоторыми Web-обозревателями. Здесь особенно отличился популярнейший Microsoft Internet Explorer — пока он не загрузит всю таблицу со всем ее содержимым, он ее не выведет. А это значит, что посетитель сайта, сделанного на основе табличного Web-дизайна, будет вынужден очень долго ждать, прежде чем на экране появится хоть что-то. Другие Web-обозреватели — Opera, Mozilla, Firefox — отображают таблицы по мере их загрузки, по частям, и посетитель сайта, по крайней мере, не соскучится, ожидая загрузки очередной страницы.

Рекомендация по преодолению обоих этих недостатков очень проста — нужно тщательно планировать таблицы разметки. Чем они будут компактнее и проще, тем быстрее будет загружена страница, и тем быстрее ее выведет на экран Web-обозреватель. Это значит — как можно меньше строк и ячеек и как можно меньше вложенных таблиц. А если без какой-то таблицы можно обойтись, лучше без нее обойтись.

В частности, некоторые элементы оформления Web-страницы можно просто вынести за пределы таблицы разметки. Давайте посмотрим на Web-страницы нашего третьего сайта, что мы создавали на этом занятии. Не кажется ли вам, что заголовок сайта и сведения об авторских правах будут неплохо чувствовать себя вне таблицы? Попробуйте в качестве домашнего задания переделать сайт так, чтобы его заголовок и сведения об авторских правах находились за пределами таблицы разметки. Радикально это два описанных выше недостатка не устранил, но загружаться наши страницы будут побыстрее.

И еще. Нужно стремиться везде, где можно, задавать фиксированные значения ширины ячеек и самих таблиц разметки, причем не в процентах, а в пикселах. Прочитав в HTML-коде значение ширины в пикселах, Web-обозреватель будет знать, как ему вывести эту ячейку или таблицу на экран. Если же ширина ячейки или таблицы задана в процентах, либо ячейка или таблица вообще сделана саморастягивающейся, Web-обозревателю придется сначала загрузить все содержимое таблицы и вычислить его ширину, на что потребуется время, и немалое.

Недостаток третий можно назвать недостатком самого табличного Web-дизайна — большое количество повторяющихся элементов, присутствующих на всех страницах сайта. Посмотрите сами — на каждой странице нашего нового сайта Site 3 есть совершенно одинаковые заголовок сайта, набор гиперссылок и сведения об авторских правах. Именно из-за этого файлы Web-страниц, созданных на основе табличного Web-дизайна, так велики. И когда посетитель щелкает по гиперссылке, все это "хозяйство" каждый раз загружается заново.

Последний — четвертый — недостаток табличного Web-дизайна — большая

деле, если мы вдруг захотим добавить в набор новую гиперссылку или изменить заголовок сайта, нам придется делать это во всех страницах сайта. Хорошо, если их всего пять. А если пятьдесят?

Увы — третий недостаток табличного Web-дизайна непреодолим. Можно только, опять же, посоветовать не создавать слишком больших и сложных повторяющихся элементов. А вот четвертый недостаток вполне можно преодолеть. И в этом нам, как всегда, поможет великолепный Dreamweaver.

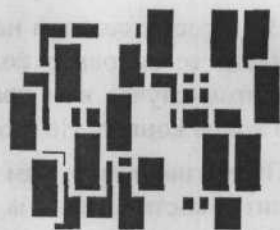
На следующем занятии...

О таблицы HTML! Что бы без вас делали Web-дизайнеры! До сих пор лепили бы простенькие текстовые странички или громоздили наборы фреймов. Но с вами, таблицы, их жизнь стала проще, а возможности — поистине безграничными.

Ну да, безграничными, — проворчит какой-нибудь особенно угрюмый читатель. А как же третий и четвертый недостатки табличного Web-дизайна, о которых нам только что разглагольствовал лектор? Куда они от нас денутся?

В том-то и дело, что никуда... По крайней мере, третий. А вот четвертый, как уже говорилось, вполне преодолим. И преодолеем мы его на следующем занятии с помощью особого инструмента Dreamweaver под названием шаблоны.

ЗАНЯТИЕ 10



Использование шаблонов

Вот мы и закончили наш третий Web-сайт. Все *занятие 9* мы занимались его созданием: рисовали ячейки разметки, наполняли их содержимым, подстраивали и подгоняли, чтобы страницы выглядели как следует. Но уж теперь-то все позади. До поры до времени...

Представим себе такую ситуацию. Мы вдруг решили добавить в наш сайт новую страницу. Чтобы это сделать, нам придется во всех страницах сайта дополнить набор гиперссылок, а для этого потребуется каждую из страниц открыть, исправить и потом сохранить. Конечно, наш сайт невелик по размеру, и мы выполним эту работу довольно быстро. А если бы он был большой, страниц с полсотни?!

Если бы мы работали не в среде Dreamweaver, вероятно, намучались бы. Но нам повезло. Dreamweaver поддерживает такое мощное средство, как шаблоны. Они уже упоминались на *занятии 9*. Сейчас же самое время рассмотреть их подробнее.

Введение в шаблоны

Шаблон — это своеобразный образец, "скелет" Web-страницы, содержащий общие для всех страниц элементы: заголовок сайта, набор гиперссылок, сведения об авторских правах, возможно, таблицу разметки и пр. Эти элементы остаются неизменными на всех страницах, созданных на основе данного шаблона, поэтому так и называются — *неизменяемыми элементами*.

Да, но каким же образом на странице помещается ее основное содержимое? Очень просто. Специально для этого в шаблоне создаются так называемые *изменяемые области*. Они-то как раз и предназначены для уникального содержимого страницы, которая будет создана на основе этого шаблона.

Все просто: создаем на основе шаблона страницу и вводим в изменяемые области ее основное содержимое. Неизменяемые же элементы Dreamweaver в этом случае нам править не дает, и правильно — они же неизменяемые, в конце концов. Но это не беда — мы всегда можем изменить сам шаблон.

Пусть, например, нам нужно исправить заголовок сайта. Поскольку он входит в состав шаблона, мы открываем шаблон, исправляем его и сохраняем. Dreamweaver тут же предлагает нам перенести сделанные в шаблоне изменения во все страницы, созданные на его основе. И переносит, да так аккуратно, что не затрагивает содержимое изменяемых областей! Фактически он за нас исправляет все повторяющиеся элементы на всех Web-страницах сайта.


Шаблоны сохраняются в особых файлах, имеющих расширение dwt, в папке Templates, которую Dreamweaver создает сам в корневой папке локальной копии сайта. Количество шаблонов, используемых в сайте, не ограничено, так что мы можем создавать одни страницы сайта на основе одного шаблона, а другие — на основе другого. А можем вообще ограничиться одним-единственным шаблоном, как, собственно, чаще всего и бывает.

Работа с шаблонами в Dreamweaver

Сейчас мы создадим шаблон для страниц нашего третьего сайта. Таким образом мы решим проблему создания множества страниц, основанных на таблицах и имеющих много повторяющихся элементов. Ну, и заодно попрактикуемся.

Создание шаблона

Есть два способа создать шаблон Dreamweaver. Во-первых, его можно создать "с нуля", а потом заполнить неизменяемым содержимым и изменяемыми областями. Во-вторых, существующую страницу можно превратить в шаблон, после чего удалить основное содержимое, оставив только общие для всех страниц сайта элементы, и создать изменяемые области. Поскольку у нас уже есть главная страница нашего сайта, давайте превратим ее в шаблон, т. е. пойдем вторым путем.

Итак, откроем главную страницу default.htm сайта site 3. После этого включим пункт-выключатель **Assets** меню **Window** или нажмем клавишу <F11>. На экране появится панель **Assets**, показанная на рис. 10.1. Вдоль левой стороны этой панели находится набор кнопок-переключателей; щелкнем по второй снизу кнопке , чтобы переключиться на список шаблонов.

Скажем пару слов об этой панели. Большую ее часть занимает список уже созданных шаблонов. Мы можем выбрать любой шаблон из этого списка и

произвести над ними какие-либо манипуляции. В принципе, все это нам уже знакомо — мы уже рассмотрели на панели со списками, подобные этой.

Чтобы создать новый шаблон на основе открытой в данный момент Web-страницы, достаточно выбрать в меню **File** пункт **Save as Template**. Также можно нажать кнопку **Templates** вкладки **Common** инструментария объектов и выбрать в появившемся меню (рис. 10.2) пункт **Make Template**. После этого на экране появится диалоговое окно **Save As Template**, показанное на рис. 10.3.

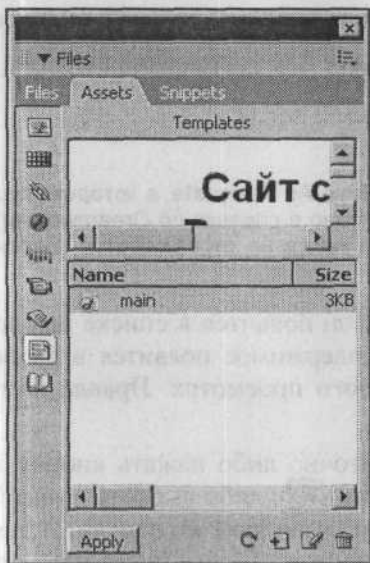


Рис. 10.1. Панель Assets

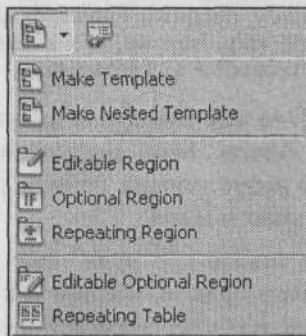


Рис. 10.2. Кнопка Templates инструментария объектов с открытым меню

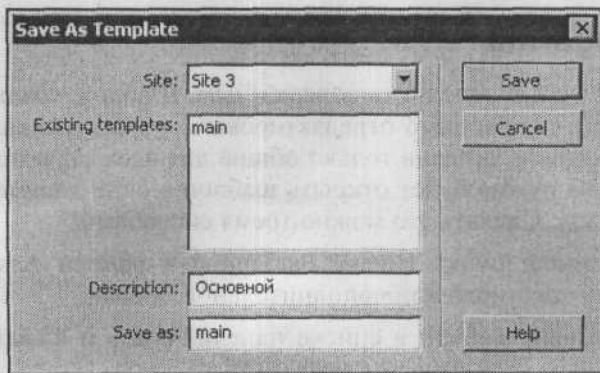


Рис. 10.3. Диалоговое окно Save As Template

В раскрывающемся списке **Site** выбирается сайт, который станет "собственником" шаблона. По умолчанию там выбран текущий сайт.

В списке **Existing templates** перечислены уже сохраненные в этом сайте шаблоны. Мы можем выбрать в этом списке любой шаблон и перезаписать его; в этом случае Dreamweaver переспросит нас, действительно ли мы хотим перезаписать уже существующий шаблон.

Само имя шаблона заносится в поле ввода **Save as**. Поскольку шаблоны сохраняются в виде файлов, их имена должны удовлетворять соглашениям по именованию файлов Windows. Давайте назовем наш новый шаблон **main**, поскольку это наш главный шаблон, на основе которого мы построим наш сайт.

Введя все нужные данные, нажмем кнопку **Save** для сохранения шаблона или **Cancel** для отказа от этого.

На заметку

Поле ввода **Description** диалогового окна **Save As Template**, в котором, судя по всему, вводится описание шаблона, не описано в справке по Dreamweaver. Более того, похоже, что введенное описание нигде не отображается. Вероятно, это ошибка в программе.

После того как мы создадим новый шаблон, он появится в списке шаблонов панели **Assets**. Если мы его выберем, его содержимое появится в расположенной выше списка панели предварительного просмотра. Правда, эта панель слишком мала...

Создать шаблон "с нуля" еще проще. Достаточно либо нажать кнопку **New Template** в правом нижнем углу панели **Assets** (📄), либо выбрать пункт **New Template** в контекстном или дополнительном меню этой же панели. В списке шаблонов появится новый пункт, вместо названия которого будет находиться поле ввода с текстом **Untitled**. Введем в него имя вновь создаваемого шаблона и нажмем клавишу **<Enter>**.

Правка шаблона

Пустой шаблон нужно наполнить содержимым. Шаблон, созданный на основе Web-страницы, необходимо отредактировать: убрать уникальное для этой страницы содержимое, оставив только общие для всех страниц сайта элементы. Для этого нам нужно будет открыть шаблон в окне документа, как обычную Web-страницу. Сделать это можно тремя способами:

- ♦ выделить нужный пункт списка шаблонов в панели **Assets** и выбрать пункт **Edit** контекстного или дополнительного меню;
- ♦ выделить нужный шаблон в списке панели **Assets** и нажать кнопку **Edit** в правом нижнем углу панели **Assets** (📄);

В любом случае на экране появится окно документа, в котором будет открыт выбранный нами шаблон. Внешне он ничем не будет отличаться от обычной Web-страницы.

Что мы можем сделать с шаблоном? Все, что угодно. Можно считать его обычной Web-страницей с некоторыми особенностями. (Эти особенности мы опишем далее.) Мы можем набирать текст, форматировать его, размещать изображения, таблицы, гиперссылки, переключаться в режим разметки страниц и создавать таблицы и ячейки разметки, создавать наборы фреймов, "чистить" HTML-код и т. д.

Но все-таки шаблон — не Web-страница и имеет некоторые особенности. Так, нам нужно будет поместить на нем изменяемые области, в которых впоследствии будет находиться основное содержимое страниц. (Более того, нам обязательно нужно будет это сделать, иначе зачем нам тогда этот шаблон.) Как это сделать, мы узнаем чуть позже.

А пока что откроем шаблон `main`, который мы создали на основе страницы `default.htm` (если он еще не открыт). Уберем текст основного содержимого и содержимое колонки новостей, оставим только заголовок сайта, набор гиперссылок и сведения об авторских правах. Проверим, все ли мы сделали правильно. И сохраним шаблон.

При сохранении шаблона, не содержащего ни одной изменяемой области, Dreamweaver предупредит нас об этом, выведя небольшое окно-предупреждение. Нажмем кнопку **ОК**, чтобы все-таки сохранить шаблон. Если мы не хотим, чтобы это окно появлялось в дальнейшем, перед тем, как нажать кнопки **ОК** или **Cancel**, включим флажок **Don't warn me again**.

Теперь нам нужно поместить в шаблон изменяемые области. Пока что изменяемая область будет всего одна — основное содержимое страницы. И располагаться она будет в самой большой ячейке нашей таблицы разметки. Колонку новостей мы пока оставим в покое.

Создание изменяемых областей

Как и в случае с самим шаблоном, мы можем создавать изменяемые области двумя способами. Во-первых, создать пустую изменяемую область на пустом месте страницы. Во-вторых, преобразовать фрагмент текущего содержимого страницы в изменяемую область. Первый способ лучше всего подходит, если мы создали шаблон "с нуля", а второй — если мы преобразовали в шаблон существующую страницу.

Создать в шаблоне пустую изменяемую область очень просто. Для этого сначала ставим текстовый курсор в то место, где мы хотим создать пустую изменяемую область, либо выставляем флажок

мы хотим в эту изменяемую область преобразовать. И решаем, чем воспользоваться:

- ◆ инструментарием объектов, нажав кнопку **Templates** на вкладке **Common** и выбрав в появившемся меню (см. рис. 10.2) пункт **Editable Region**;
- ◆ контекстным меню, выбрав в его подменю **Template** пункт **New Editable Region**;
- ◆ системным меню, выбрав в подменю **Template Objects** меню **Insert** пункт **Editable Region**;
- ◆ клавиатурой, что быстрее всего, — достаточно нажать <Ctrl>+<Alt>+<V>.

На экране после этого появится диалоговое окно **New Editable Region**, показанное на рис. 10.4.

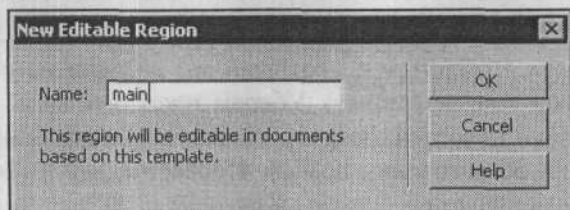


Рис. 10.4. Диалоговое окно **New Editable Region**

В единственном поле ввода **Name**, находящемся в этом окне, вводится уникальное имя вновь создаваемой изменяемой области. Каждая созданная нами в шаблоне изменяемая область должна иметь уникальное имя. Это имя может содержать любые символы, кроме букв русского алфавита, кавычек, апострофа и знаков "<", ">" и "&". Введя имя, нажмем кнопку **OK**, чтобы создать изменяемую область, или **Cancel** для отказа от этого.

Внимание!

Вы не можете дать изменяемой области имя `doctitle`. Почему, будет рассказано позже.



Рис. 10.5. Вновь созданная изменяемая область

Когда мы нажмем кнопку **OK**, увидим то, что показано на рис. 10.5. Так выглядит в окне документа Dreamweaver изменяемая область, в которую мы

большой голубой (на рисунке — серый) ярлычок — *заголовок*, содержащий имя изменяемой области; он похож на заголовок таблиц разметки. Щелкнув его, мы выделим изменяемую область. Ниже, в синей рамке, находится содержимое изменяемой области, в нашем случае — имя изменяемой области. (Если мы преобразуем в изменяемую область фрагмент страницы, рамка будет содержать этот самый фрагмент.) При выделении изменяемой области ее содержимое также будет выделено.

Если мы случайно поместили изменяемую область не в то место, куда хотели, это исправимо. Выделяем изменяемую область, щелкнув мышью по ее заголовку, "захватываем" ее мышью за содержимое и тащим на нужное место.

Если мы ошиблись в наборе имени изменяемой области, это тоже можно легко исправить. Выделяем нужную изменяемую область и смотрим на редактор свойств (рис. 10.6). В поле ввода **Name** задаем нужное имя и нажимаем клавишу <Enter>.

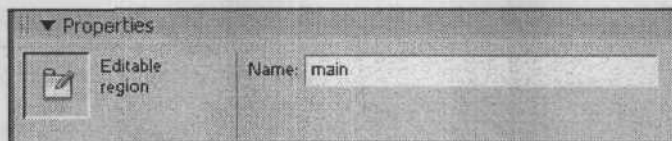


Рис. 10.6. Вид редактора свойств при выделенной изменяемой области шаблона

И, наконец, может случиться так, что мы захотим удалить изменяемую область. Для этого выделяем изменяемую область щелчком по заголовку и нажимаем клавишу . Нужно только иметь в виду, что после удаления изменяемой области ее содержимое остается в шаблоне. Так что, если нам нужно полностью удалить изменяемую область, мы будем должны удалить и ее содержимое.

А теперь — небольшой сюрприз. Дело в том, что Dreamweaver при создании любого шаблона автоматически создает небольшую изменяемую область *doctitle*. (Вот поэтому мы и не сможем дать изменяемой области имя *doctitle* — изменяемая область с таким именем уже существует). Данная изменяемая область включает в себя содержимое тега <TITLE>, иначе говоря, название Web-страницы.

Внимание!

Если вы удалите скрытую изменяемую область *doctitle*, то потеряете возможность менять названия Web-страниц, основанных на данном шаблоне. В этом случае все порожденные от этого шаблона страницы будут иметь одно название, совпадающее с названием, заданным в шаблоне.

Создание Web-страниц на основе шаблонов

Создать Web-страницу на основе шаблона можно двумя способами.

Первый способ заключается в том, чтобы использовать уже знакомый нам пункт **New** меню **File**. При этом на экране появится диалоговое окно **New Document**, показанное на рис. 3.1. Переключаемся на вкладку **Templates** этого окна — и видим то, что показано на рис. 10.7.

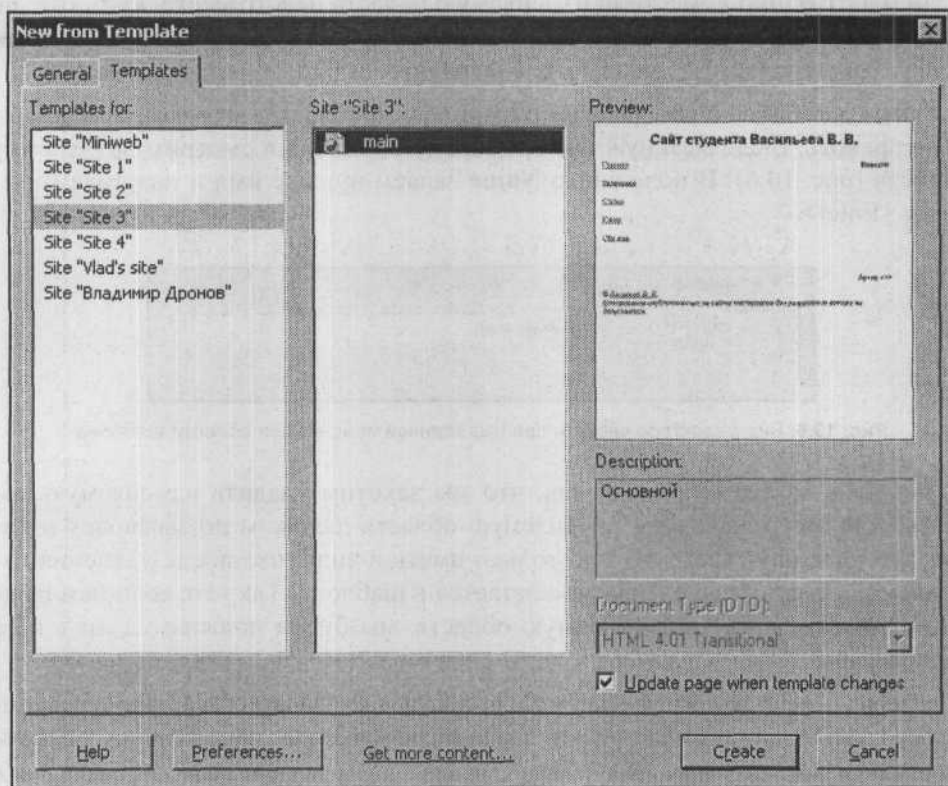


Рис. 10.7. Диалоговое окно **New Document** (вкладка **Templates**)

В списке **Templates for** выбирается сайт, из которого будет взят шаблон. Сам же шаблон выбирается в списке **Site <имя сайта>**. Если флажок **Update page when template changes** включен (а он включен по умолчанию), при изменении шаблона, на основе которого создается Web-страница, последняя будет соответственно изменена. И лучше этот флажок не отключать.

Чтобы создать новую страницу на основе выбранного нами шаблона (это, конечно же, шаблон **main** сайта **Site 3**), нужно нажать кнопку **Create**. Чтобы

Второй способ создания новой страницы на основе шаблона намного проще. Открываем панель **Assets**, переключаемся на список шаблонов, щелкнув по второй снизу кнопке этой панели, выделяем в списке нужный шаблон и выбираем в контекстном или дополнительном меню пункт **New from Template**.

Как бы то ни было, на экране появится окно документа, содержащее в себе новую Web-страницу (рис. 10.8). Созданная нами изменяемая область выделена синей рамкой. Кроме ее содержимого, ни один другой элемент страницы не может быть изменен; при наведении на него курсора мыши последний меняет форму на перечеркнутый круг. Мы даже не сможем выделить ничего из содержимого шаблона — только содержимое изменяемых областей.

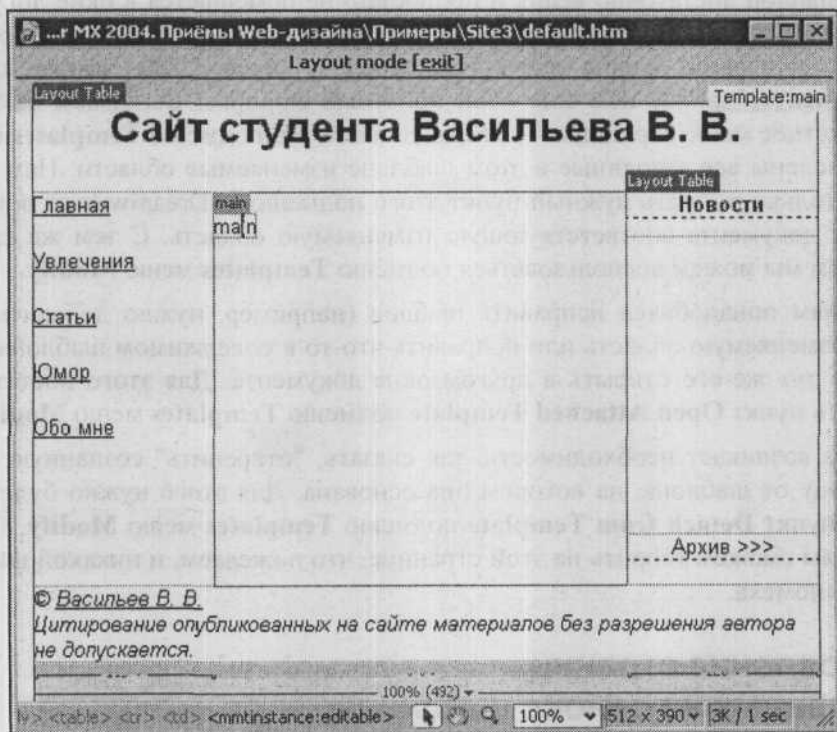


Рис. 10.8. Только что созданная Web-страница, основанная на шаблоне main

Более того, если мы переключимся в режим отображения HTML-кода, то и тогда не сможем исправить содержимое самого шаблона. HTML-код, принадлежащий шаблону, будет набран тускло-серым цветом — это предостережение для нас. Как видим, шаблон надежно защищен от редактирования.

В правом верхнем углу окна документа находится небольшой желтый (на рис. 10.8 — серый) значок, обозначающий имя шаблона, от которого "по-

рождена" эта Web-страница. Небольшое напоминание для забывчивых Web-дизайнеров...

Ну что ж, приступим к работе! Ставим текстовый курсор в изменяемую область, удаляем старое ее содержимое, оставшееся в "наследство" от шаблона, и вводим новое. Если же мы не хотим набирать это содержимое заново (у нас, в конце концов, есть готовая Web-страница, на основе которой мы создали этот шаблон), то можно воспользоваться методом, которые бывалые компьютерщики называют "копируй-и-вклеивай". Открываем старую Web-страницу, копируем ее основное содержимое в буфер обмена и вставляем ее в изменяемую область. Что может быть проще и быстрее!

Если шаблон достаточно велик и полностью не помещается в окне документа, для поиска нужной изменяемой области нам придется постоянно прокручивать содержимое окна вверх-вниз и влево-вправо. Для такого случая Dreamweaver припас нам еще один приятный сюрприз. Вызываем на экран контекстное меню страницы — в нижней части его подменю **Templates** будут перечислены все созданные в этом шаблоне изменяемые области. Нам останется только выбрать нужный пункт этого подменю, и Dreamweaver покажет в окне документа соответствующую изменяемую область. С тем же самым успехом мы можем воспользоваться подменю **Templates** меню **Modify**.

Если нам понадобится исправить шаблон (например, нужно добавить еще одну изменяемую область или исправить что-то в содержимом шаблона), мы можем тут же его открыть в другом окне документа. Для этого достаточно выбрать пункт **Open Attached Template** подменю **Templates** меню **Modify**.

Иногда возникает необходимость, так сказать, "открепить" созданную нами страницу от шаблона, на котором она основана. Для этого нужно будет выбрать пункт **Detach from Template** подменю **Templates** меню **Modify**. После этого мы сможем творить на этой странице, что пожелаем, и никакой шаблон нам не помеха.

Обновление страниц, созданных на основе шаблонов

Теперь предположим, что мы создали на основе шаблона несколько страниц и уже после этого вдруг заметили, что в содержимое шаблона вкралась ошибка. Мы вызываем шаблон на экран, правим его и сохраняем. Что произойдет в этом случае?

А произойдет вот что. Сначала Dreamweaver выведет на экран диалоговое окно **Update Template Files**, показанное на рис. 10.9. В списке, занимающем большую часть этого окна, перечислены все страницы, созданные на основе исправленного нами шаблона. Нам останется только нажать кнопку **Update**

чтобы Dreamweaver обновил страницы. После этого Dreamweaver выведет на экран еще одно диалоговое окно **Update Files**, содержащее сведения об обновленных страницах; закроем его щелчком по кнопке **Close**.

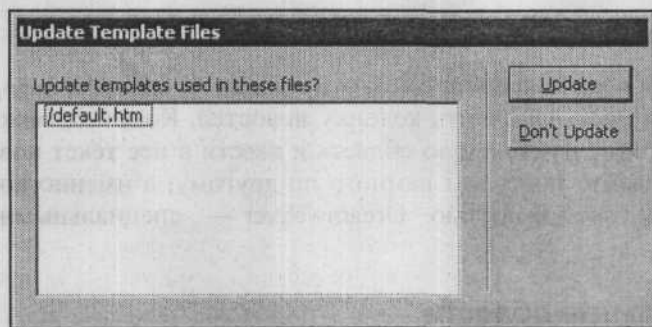


Рис. 10.9. Диалоговое окно Update Template Files

Управление шаблонами в списке панели Assets

А теперь немного отвлечемся от Web-дизайна и поговорим о панели **Assets** и предоставляемых ей средствах по управлению шаблонами.

Мы можем переименовать любой шаблон. Для этого выделим его в списке и либо щелкнем мышью по его имени, либо выберем пункт **Rename** в контекстном или дополнительном меню. Вместо имени данного шаблона появится поле ввода, где мы сможем ввести новое имя шаблона. После этого нажмем <Enter> для сохранения введенного имени или <Esc> для отказа от него.

Чтобы удалить ненужный шаблон, выделим его в списке и либо нажмем клавишу , либо выберем пункт **Delete** контекстного или дополнительного меню, либо нажмем кнопку **Delete** (🗑️), расположенную в правом нижнем углу панели **Assets**. Dreamweaver поинтересуется, действительно ли мы хотим удалить этот шаблон; нажмем кнопку **Да** (Yes) или **Нет** (No). После нажатия кнопки **Да** шаблон будет безвозвратно удален.

Внимание!

После удаления шаблона все основанные на нем страницы потеряют с ним связь. Кроме того, Dreamweaver удаляет шаблоны безвозвратно, минуя корзину.

Также мы можем скопировать какой-либо шаблон в другой сайт. Конечно, это можно сделать в Проводнике Windows или аналогичной программе управления файлами (например, FAR Commander), но зачем нужны сторонние программы, если есть Dreamweaver! Выделяем в списке панели **Assets**

нужный шаблон и щелкаем по нему правой кнопкой мыши, чтобы открыть контекстное меню. В подменю **Copy to Site** будут перечислены все зарегистрированные в Dreamweaver сайты — достаточно просто выбрать нужный.

Специальные области шаблонов

Что ж, главная страница сайта Site3, основанная на шаблоне `main`, почти готова. Осталось только заполнить колонку новостей. Конечно, можно создать в шаблоне еще одну изменяемую область и ввести в нее текст новостей — это просто. Но давайте поступим немного по-другому, а именно воспользуемся замечательной возможностью Dreamweaver — специальными областями шаблонов.

Необязательные области

Давайте еще раз посмотрим на главную страницу сайта `site 3`. Мы видим на ней колонку новостей, которую скоро начнем заполнять. Но дело в том, что на других страницах этого сайта колонки новостей нет. Что же делать?

Самый простой путь — сделать второй шаблон, не содержащий колонки новостей, и на его основе создавать остальные страницы. Но это не очень удобно: если нам понадобится что-то изменить, придется править два шаблона. А если таких шаблонов не два, а больше? В идеале было бы лучше сделать всего ОДИН шаблон, а при создании на его основе Web-страниц просто скрыть ненужные элементы. Возможно ли такое?

Возможно! Мы можем создать в шаблоне так называемые *необязательные области*, которые могут по нашему желанию либо присутствовать в Web-страницах, либо отсутствовать. И колонка новостей, а именно таблица, вмещающая в себе текст новостей, так и просится, чтобы ее преобразовали в необязательную область.

Откроем шаблон `main` и удалим изображение-"распорку" из третьей ячейки таблицы, где помещается колонка новостей. Это нужно для того, чтобы Web-обозреватель смог сузить эту ячейку, если в ней не будет содержимого, до нулевого размера. В противном же случае на каждой странице нашего сайта будет красоваться пустое пространство, оставшееся от колонки новостей, что, согласитесь, некрасиво.

Теперь можно приступить к созданию необязательной области. Выделим таблицу разметки, в которой помещается колонка новостей. Далее нажмем кнопку **Templates** на вкладке **Common** инструментария объектов и выберем в ее меню (см. рис. 10.2) пункт **Optional Region**. (Также можно выбрать пункт **Optional Region** подменю **Template Objects** меню **Insert** или пункт **New Optional Region** подменю **Templates** контекстного меню.) На экране появит-

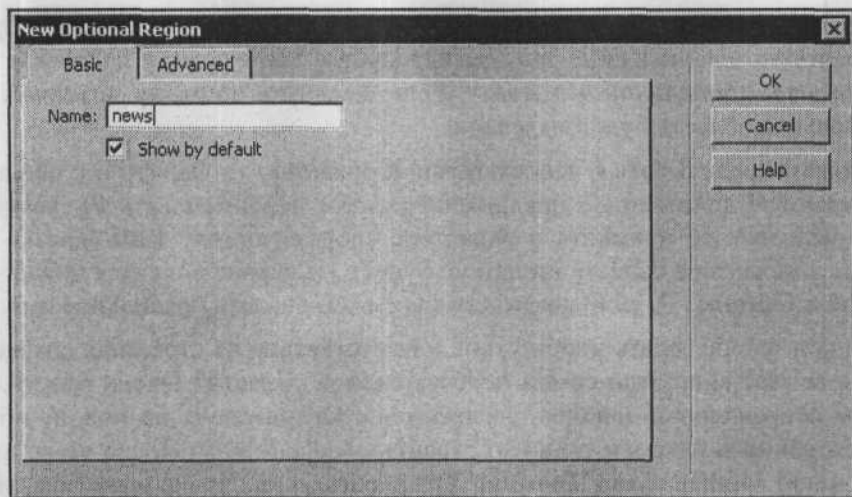


Рис. 10.10. Диалоговое окно New Optional Region

В поле ввода **Name**, что находится на вкладке **Basic** этого окна, занесем имя создаваемой необязательной области — *news*. (Имена необязательных областей должны удовлетворять тем же правилам, что и имена изменяемых областей.) Отключим флажок **Show by default**, т. к. мы не хотим, чтобы она отображалась по умолчанию. И нажмем кнопку **OK**. То, что у нас должно получиться, показано на рис. 10.11.



Видно, что созданная нами необязательная область окружена тонкой голубой границей и помечена в окне документа голубым заголовком с надписью вида **If** <имя необязательной области>. Если щелкнуть по этому заголовку, необязательная область будет выделена.

Мы можем проделывать с необязательной областью то же, что и с областью изменяемой. Единственное исключение: чтобы переименовать ее, нам придется выделить ее и нажать в редакторе свойств кнопку **Edit** (кроме этой кнопки, в редакторе свойств ничего не будет). На экране появится диалоговое окно **New Optional Region**, в котором мы сможем задать правильное имя.

Да, но как теперь задать присутствие или отсутствие на странице, созданной на основе шаблона, этой самой необязательной области? Очень просто. Сохраним исправленный шаблон, согласимся с Dreamweaver по поводу обновления страницы и откроем главную страницу (default.htm). После этого выберем в меню **Modify** пункт **Template Properties**, и на экране появится диалоговое окно **Template Properties** (рис. 10.12).

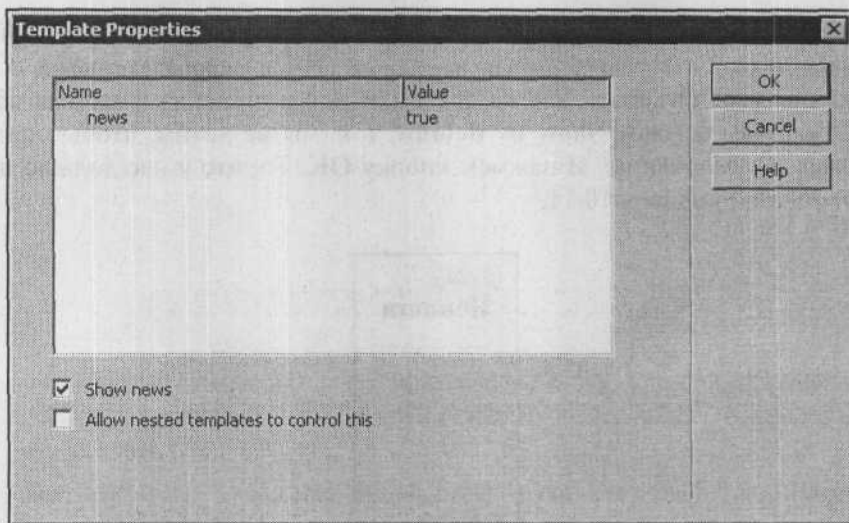


Рис. 10.12. Диалоговое окно **Template Properties**

Большую часть этого окна занимает список необязательных областей, созданных в шаблоне, на основе которого создана открытая страница. В колонке **Name** списка отображается имя необязательной области, а в колонке **Value** — значение *true* (отображается) или *false* (не отображается). Чтобы задать присутствие или отсутствие необязательной области, нам будет нужно сначала выделить ее в списке, а потом соответственно включить или отключить флажок **Show** <имя необязательной области> который находится под спи-

ском. Ну, и не забыть нажать потом кнопку **ОК**, чтобы Dreamweaver применил заданные нами установки.

Но на этом наши хлопоты не заканчиваются. Дело в том, что необязательная область Dreamweaver по своей природе не является изменяемой. Нам нужно поместить в нее изменяемую область, чтобы мы смогли задать содержимое колонки новостей. А можно поступить немного по-другому, а как — мы сейчас узнаем.

Повторяющиеся области

Кроме необязательных, Dreamweaver позволяет также создавать *повторяющиеся области*. Они так называются, потому что могут повторяться на странице подряд сколько угодно раз (иметь сколько угодно копий).

Давайте посмотрим на нашу многострадальную колонку новостей. Каждая новость в ней — это отдельный абзац текста. Колонка может содержать множество этих однотипных абзацев. Просто готовый кандидат в повторяющиеся области!

Снова откроем шаблон `main` и поместим текстовый курсор в ячейку новостей внутренней таблицы разметки. Нажмем кнопку **Templates** на вкладке **Common** инструментария объектов и выберем в ее меню (см. рис. 10.2) пункт **Repeating Region**. (Можно еще выбрать пункт **Repeating Region** подменю **Template Objects** меню **Insert** или пункт **New Repeating Region** подменю **Templates** контекстного меню.) На экране появится диалоговое окно **New Repeating Region**, показанное на рис. 10.13.

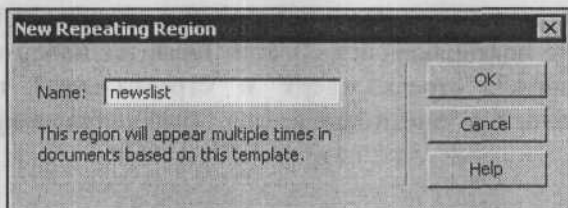


Рис. 10.13. Диалоговое окно **New Repeating Region**

Введем имя создаваемой повторяющейся области в единственном поле ввода этого окна. Назовем ее `newlist`. После этого нажмем кнопку **ОК**. Результат показан на рис. 10.14.

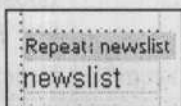


Рис. 10.14. Повторяющаяся область

Как и необязательная область, повторяющаяся область окружена тонкой голубой границей и помечена в окне документа голубым заголовком. Она отличается только надписью на заголовке, имеющей вид **Repeat: <имя повторяющейся области>**. Мы можем выделять, перемещать, переименовывать и удалять повторяющиеся области так же, как изменяемые.

Повторяющаяся область Dreamweaver также не является изменяемой. Поэтому нам нужно поместить внутрь нее еще и изменяемую область. (Таким образом, получится троекратная вложенность "необязательная область — повторяющаяся область — изменяемая область". Сложно, конечно, зато очень гибко.) Поставим текстовый курсор где-нибудь внутри содержимого повторяющейся области и создадим изменяемую область с именем `newsitem` (рис. 10.15). Вот теперь все готово к вводу в колонку новостей.

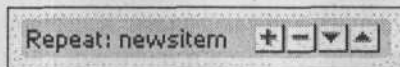


Рис. 10.15. Изменяемая область

Снова сохраним исправленный шаблон, согласимся с Dreamweaver по поводу обновления страниц и откроем главную страницу. Проверим, задали ли мы отображение колонки новостей. И введем в пока что единственную копию повторяющейся области текст какой-либо новости.

Так, одна новость у нас есть. Но как добавить остальные? Очень просто!

Для работы с копиями повторяющихся областей Dreamweaver предоставляет набор особых пунктов меню. Все они приведены в табл. 10.1 и 10.2. (Кроме того, на заголовке повторяющейся области появится набор из четырех кнопок: "плюс", "минус", "стрелка вверх" и "стрелка вниз"; только надежды на эти кнопки мало, т. к. они почти всегда скрываются за краем окна документа.)

Таблица 10.1. Пункты подменю **Templates** меню **Modify**

Пункт меню	Кнопка	Назначение
New Entry After Selection	"Плюс"	Добавляет новую копию ниже (правее) выделенной
New Entry Before Selection		Добавляет новую копию выше (левее) выделенной
Move Entry Up	"Стрелка вверх"	Перемещает выделенную копию выше
Move Entry Down	"Стрелка вниз"	Перемещает выделенную копию ниже

Таблица 10.2. Пункты подменю *Repeating Entries* меню *Edit*

Пункт меню	Кнопка	Назначение
Cut repeating entry		Вырезает выделенную копию в буфер обмена
Copy repeating entry		Копирует выделенную копию в буфер обмена
Delete repeating entry	"Минус"	Удаляет выделенную копию

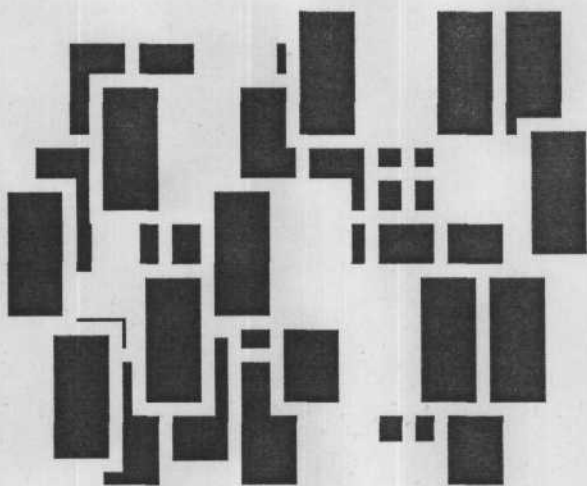
Как вы уже заметили, в табл. 10.2 не указан пункт **Paste repeating entry**, который осуществлял бы вставку копии из буфера обмена. Это выполняется, как обычно, вызовом пункта **Paste** меню **Edit** или нажатием комбинации клавиш <Ctrl>+<V>.

Собственно, это все, что автор собирался рассказать о шаблонах Dreamweaver. До следующего занятия!

На следующем занятии...

Что ж, с фреймовым и табличным Web-дизайном мы разобрались. Даже сделали два сайта: один — на основе фреймов, другой — на основе таблиц разметки. Да, не заслужил лентяй Васильев такой чести!..

А мы продолжим изучение различных принципов Web-дизайна. И на следующем курсе познакомимся с последним пискom в мире интернет-технологий — контейнерным дизайном. Посторонитесь, фреймы и таблицы!



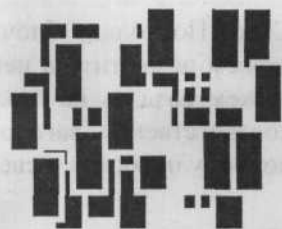
КУРС 4

Контейнерный Web-дизайн

Занятие 11. Плавающие контейнеры

Занятие 12. Свободно позиционируемые контейнеры

ЗАНЯТИЕ 11



Плавающие контейнеры

Уже несколько лет прошло с тех пор, как прогремела в Интернете "табличная революция". Большинство сайтов моментально перешли на табличный Web-дизайн, Web-страницы стали пестрее и объемнее, а их HTML-код — совершенно неудобоваримым. И хотя народу (т. е. посетителям) это нравилось, Web-дизайнеры призадумались — а нет ли в природе чего-нибудь такого же гибкого, как таблицы, но менее громоздкого.

Думали они, думали и наконец придумали. Точнее сказать, решение долго лежало у них прямо под носом. Они же просто обратили на него внимание.

Давайте обратимся к материалу *занятия 7*, где говорилось о стилях CSS. Помимо собственно стилей, мы узнали там еще кое о чем. А именно о контейнерах HTML.

Как мы уже знаем, контейнер — это особый тег HTML, служащий как бы "упаковкой" для других тегов. Содержимое контейнера ведет себя как единое целое, а значит, мы можем творить с этим единым целым все, что нам заблагорассудится. Вот практически дословно все то, что было сказано насчет контейнеров на *занятии 7*.

Опять же, согласно материалам *занятия 7*, контейнеры бывают блочные (блоки) и встроенные. Блочный контейнер ведет себя как отдельный абзац текста, т. е. отображается в отдельной строке и располагается на некотором расстоянии от других элементов страницы. Встроенный же контейнер ведет себя как часть абзаца или, в более широком смысле, блочного контейнера.


Блочный контейнер создается при помощи парного тега `<DIV>`. Встроенный контейнер создается с помощью парного тега ``. Оба этих тега поддерживают атрибут `CLASS`, предназначенный для задания стиля, — это важно. Вот, собственно, и все, что мы пока знаем о контейнерах.

Стоп! Поскольку блочный контейнер может содержать все что угодно, мы можем поместить в него целый фрагмент нашей Web-страницы. То есть мы можем создать на странице пять блочных контейнеров и поместить в них, соответственно, заголовок сайта, набор гиперссылок, основное содержимое, колонку новостей и сведения об авторских правах. А что, если попробовать?

Простейший случай контейнерного Web-дизайна

Сказано — сделано!

Для экспериментов с контейнерами мы создадим новый сайт, который будет помещаться в папке Site4. Зарегистрируем его в Dreamweaver под именем Site 4 и создадим новую пустую Web-страницу, которая станет главной страницей нашего нового сайта. И сохраним ее под именем default.htm.

Теперь нам нужно создать на странице первый блочный контейнер, т. е. тег <div>. Другой Web-дизайнер сейчас бы переключился в режим отображения HTML-кода и ввел этот тег вручную, но мы — пользователи неисчерпаемого Dreamweaver — поступим по-другому. Переключимся на вкладку **Layout** инструментария объектов и нажмем кнопку **Insert Div Tag** . Также можно выбрать в подменю **Layout Objects** меню **Insert** пункт **Div Tag**. На экране появится диалоговое окно **Insert Div Tag** (рис. 11.1).

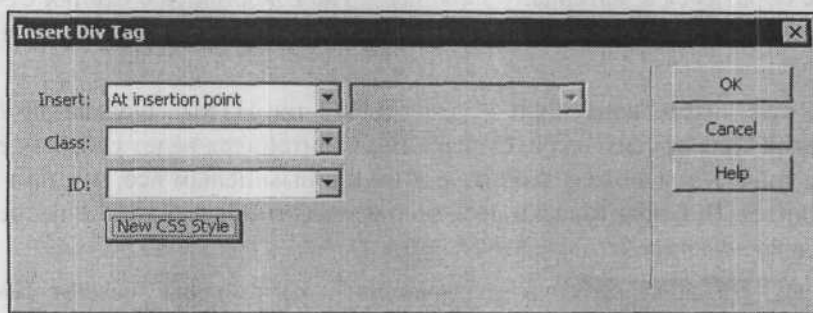


Рис. 11.1. Диалоговое окно Insert Div Tag

В раскрывающемся списке **Class** здесь выбирается стилевой класс, который нужно привязать к создаваемому блочному контейнеру. Поскольку к создаваемой нами странице пока что не привязана ни одна таблица стилей, мы оставим этот список пустым.

Раскрывающийся список **Insert** задает, где будет помещен контейнер. Здесь доступны три пункта:

- ◆ **At insertion point** — в том месте, где стоит текстовый курсор;
- ◆ **After start of tag** — сразу после открывающего тега, выбранного в раскрывающемся списке справа;
- ◆ **Before end of tag** — сразу перед закрывающим тегом, выбранным в раскрывающемся списке справа.

Выберем в списке **Insert** пункт **At insertion point** (впрочем, он выбран по умолчанию). И нажмем кнопку **OK**. Сразу же после этого Dreamweaver создаст блочный контейнер в самом начале страницы и поместит в него текст *Content for New Div Tag Goes Here* (содержимое нового тега `<DIV>` находится здесь). Этот текст показывает, где мы должны будем создать содержимое созданного контейнера.

Теперь поместим во вновь созданный контейнер заголовок сайта. Удалим созданный Dreamweaver текст и введем в него текст заголовка. Отформатируем его как заголовок HTML первого уровня и выровняем его по центру.

Очередь за набором гиперссылок. Опять нажмем кнопку **Insert Div Tag** на вкладке **Layout** инструментария объектов. В диалоговом окне **Insert Div Tag** в раскрывающемся списке **Insert** выберем пункт **Before end of tag**, а в раскрывающемся списке справа — пункт `<body>`. После нажатия кнопки **OK** Dreamweaver создаст новый контейнер перед закрывающим тегом `</BODY>`, т. е. в конце страницы, после первого созданного нами контейнера. Поаимствуем со страницы более старых наших сайтов и вставим в новый контейнер набор гиперссылок.

Точно таким же образом создадим остальные три контейнера, которые будут содержать основное содержимое, колонку новостей и сведения об авторских правах. И сохраним готовую страницу.

Давайте теперь посмотрим на нее — рис. 11.2. Да, честно говоря, не впечатляет. Похоже на обычный текстовый Web-дизайн, которым мы занимались на занятии 3. И это все, на что способны контейнеры?!

На заметку

Обратим внимание, что Dreamweaver выделяет контейнеры тонкими пунктирными рамками, чтобы помочь нам.

Да, контейнеры сами по себе ничего особенного не дают. Вы сами видите, что текст (или иное содержимое), помещенный в контейнер, ведет себя как обычный текстовый абзац, созданный с помощью хорошо нам знакомого тега `<P>`. А у текстовых абзацев правило очень простое — следовать друг за другом сверху вниз в том порядке, в каком они определены в коде HTML.

Чтобы раскрыть всю мощь контейнеров, нам придется кое-что к ним добавить. А именно стили CSS.

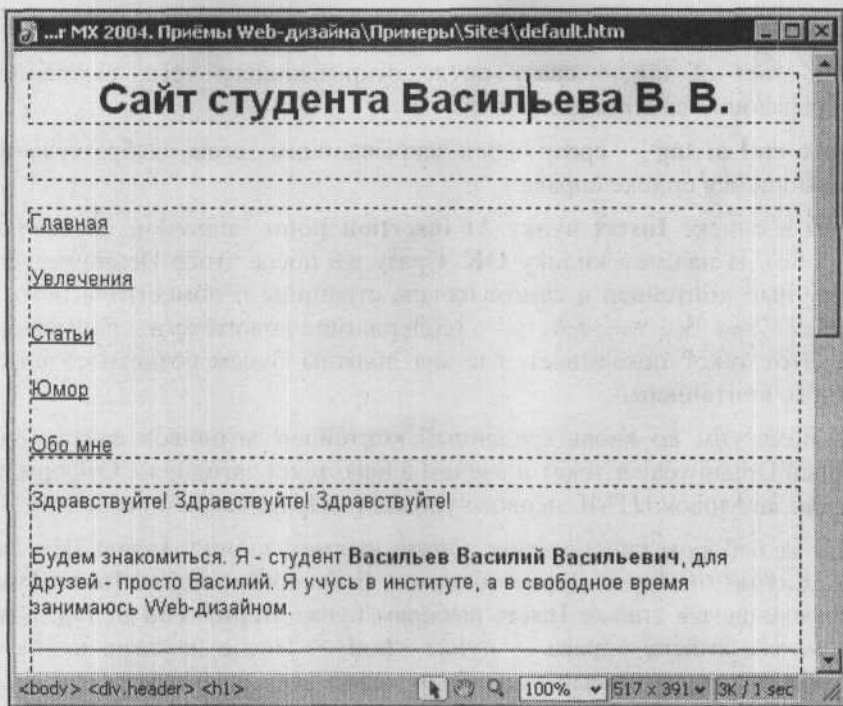


Рис. 11.2. Первоначальная страница, созданная на основе контейнеров

Стили CSS, предназначенные для управления контейнерами

Да-да, те самые каскадные стили CSS, изученные нами на *занятии 7*. Тогда мы рассмотрели практически все предоставляемые ими возможности, и их вполне хватило бы нам на первых порах. Но для работы с контейнерами нам придется пополнить наши знания о стилях.

Итак, первые два атрибута стилей, которые нам обязательно пригодятся, — это **Width** и **Height**. Конечно, это названия не самих атрибутов, а комбинированных списков, присутствующих в категории **Box** диалогового окна **CSS Rule Definition** (см. рис. 7.9). Мы уже встречались с ними раньше, но это было достаточно давно, так что давайте вспомним, зачем они нужны.

Комбинированные списки **Width** и **Height** категории **Box** диалогового окна **CSS Rule Definition** позволяют задать соответственно ширину и высоту элемента страницы. Эти размеры могут быть установлены как абсолютные величины (в пикселах) или процентное значение от ширины или высоты родите-

ля. Единица измерения CSS, в которой задаются эти величины, выбирается в раскрывающемся списке, расположенном правее соответствующего комбинированного списка.

Особый пункт **auto**, присутствующий в этих списках, возвращает управление размерами элемента страницы Web-обозревателю. Вместо выбора этого пункта можно просто очистить соответствующий комбинированный список.

Что это значит? Это значит, что мы можем задать ширину и высоту любого контейнера почти таким же образом, как ячейки таблицы. На практике, правда, практически всегда задается только ширина, т. к. именно она играет роль. Высоту же Web-обозреватель задаст сам, и ровно такую, чтобы вместить в контейнер все его содержимое.

Так, ширину контейнеров мы задать можем. Осталось сделать так, чтобы эти контейнеры могли вести себя подобно ячейкам таблицы и в остальном, т. е. выстраиваться по горизонтали, прижимаясь друг к другу "боками". Есть ли в CSS какие-то средства для этого?

Давайте обратимся все к той же категории **Box** диалогового окна **CSS Rule Definition** (см. рис. 7.9). Мы еще не рассмотрели раскрывающиеся списки **Float** и **Clear**. Зачем они нужны?

Раскрывающийся список **Float** позволяет задать такое поведение контейнеров (и не только контейнеров, а и любых элементов страницы), при котором они выстраиваются не друг за другом сверху вниз, а последовательно, один за другим сдвигаются к краю страницы, выстраиваясь по горизонтали, как колонки таблицы. При этом:

- ◆ пункт **left** заставляет контейнеры сдвинуться к левому краю страницы и выстроиться по горизонтали;
- ◆ пункт **right** заставляет контейнеры сдвинуться к правому краю страницы и также выстроиться по горизонтали;
- ◆ пункт **none** включает обычное, заданное по умолчанию поведение контейнеров (т. е. они ведут себя как обычные текстовые абзацы — выстраиваются по вертикали сверху вниз).

Давайте разберемся, как действует список **Float**, на примере — так будет понятнее. Предположим, мы создали стиль, для которого в этом списке был выбран пункт **left**, и привязали этот стиль к контейнерам, содержащим набор гиперссылок, основное содержимое и колонку новостей. Тогда:

- ◆ набор гиперссылок будет сдвинут к левой границе Web-страницы и прижмется к ней вплотную;
- ◆ основное содержимое также будет сдвинуто к левой границе Web-страницы, прижмется к набору гиперссылок и окажется справа от него;

- ◆ колонка новостей будет сдвинута к левой границе Web-страницы, прижмется к правой границе основного содержимого и окажется справа от него.

Все это схематично показано на рис. 11.3.

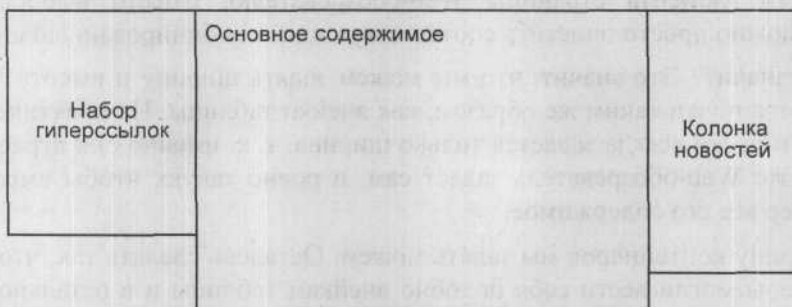


Рис. 11.3. Схема расположения контейнеров, прижимающихся к левой границе страницы

Здесь еще нужно сказать вот что. Web-обозреватель при расположении таких контейнеров учитывает размеры своего окна. Так, если мы уменьшим размеры окна, в котором отображаются показанные на рис. 11.3 контейнеры, так, что для самого правого контейнера (колонка новостей) не останется места, Web-обозреватель перенесет его вниз (рис. 11.4).

Осталось сказать только, что контейнеры, ведущие себя подобным образом, называются *плавающими*. В противоположность им, контейнеры, ведущие себя как обычные абзацы, именуют *фиксированными*.

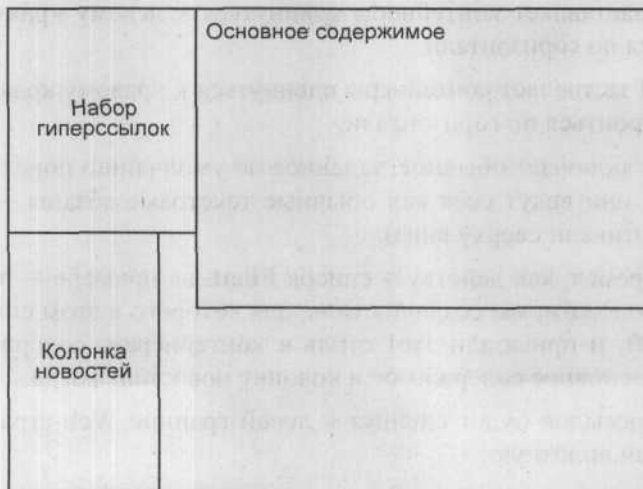


Рис. 11.4. Так расположатся контейнеры, показанные на рис. 11.3, если уменьшить размеры окна Web-обозревателя

Кратко подытожим все, что узнали. Пользуясь раскрывающимся списком **Float**, мы можем выстраивать контейнеры в структуры, похожие на показанную на рис. 11.3. Это значит, что мы можем создавать Web-дизайн, похожий на табличный, но без использования таблиц. Так ведь это замечательно!

Да, замечательно, но только до тех пор, пока мы не попытаемся добавить к показанному на рис. 11.3 дизайну сведения об авторских правах. Контейнер, содержащий их, должен быть в самом низу, ниже всех остальных контейнеров. Но как это сделать? Если мы просто зададим для него параметр **Float**, равный **left**, этот контейнер будет вести себя так же, как остальные три контейнера, т. е. сдвигаться к левому краю окна и выстраиваться в одну линию с остальными контейнерами. А если для него вообще не задавать параметр **Float**, поведение контейнера будет непредсказуемым. Но нам-то это не нужно!

И тут на первый план выходит до этого скромно прятавшийся в тени раскрывающийся список **Clear**. Он позволит нам задать такое поведение, когда какой-то контейнер в любом случае располагается ниже остальных. Для задания такого поведения нам предлагаются целых четыре пункта:

- ◆ **left** — контейнер должен находиться ниже всех плавающих контейнеров, параметр **Float** которых выставлен в **left**;
- ◆ **right** — контейнер должен находиться ниже всех плавающих контейнеров, параметр **Float** которых выставлен в **right**;
- ◆ **both** — контейнер должен находиться ниже всех плавающих контейнеров, параметр **Float** которых выставлен в **left** или **right**;
- ◆ **none** — поведение контейнера по умолчанию (т. е. он может располагаться на одной линии с остальными плавающими контейнерами).

Вот теперь мы знаем, что делать. Нужно всего лишь выставить для контейнера со сведениями об авторских правах параметр **Clear** в **left** или **both**. После этого он разместится ниже трех плавающих контейнеров, показанных на рис. 11.3.

На практике, однако, поступают немного по-другому — создают небольшой плавающий контейнер, содержащий один только неразрывный пробел. Для этого контейнера параметр **Clear** выставляют в **both**. Далее помещают этот контейнер перед тем контейнером, который нужно перенести на следующую линию, так что он начинает играть роль своего рода разделителя контейнеров. Его, кстати, так и называют — *разделитель*.

Разумеется, при создании контейнерного Web-дизайна используют и другие параметры стилей. Так, очень часто используют отступы — для создания отступов между контейнерами, рамок — ну, это понятно, зачем. О параметрах

Пример контейнерного Web-дизайна

Ну что, проверим полученные нами знания на деле! Откроем страничку `default.htm` нашего нового сайта, если она еще не открыта, и выведем на экран панель **CSS Styles** (проще всего это сделать, нажав `<Ctrl>+<Shift>+<S>`). Эта панель очень пригодится нам в дальнейшем.

Первым делом нам нужно рассчитать ширину содержимого нашей странички. Пусть она будет равна 540 пикселям — такая страница гарантированно поместится на экране монитора при разрешении 640×480 пикселей. Пока еще многие владельцы старых мониторов устанавливают на своих компьютерах именно такое экранное разрешение. (100 пикселей зарезервировано на границы окна Web-обозревателя и пр.)

На заметку

В настоящее время лучше всего ориентироваться на экранное разрешение 800×600 пикселей. Тогда ширина страницы должна быть 700—750 пикселей.

Теперь можно задать стиль для самого верхнего контейнера — заголовка сайта. Создаем стилевой класс под именем `header`, помещаем его во внешнюю таблицу стилей `styles.css` и задаем для него такие параметры:

- ◆ **Width** — 540 пикселей;
- ◆ **Float** — `left`.

Привязываем готовый стилевой класс к верхнему контейнеру. Проще всего это сделать, поставив текстовый курсор на текст заголовка сайта, щелкнув правой кнопкой мыши по кнопке `<div>` секции тегов, что в строке статуса окна документа, и выбрав в подменю **Set Class** контекстного меню соответствующий пункт.

Чтобы перенести все последующие контейнеры на новую линию, нам понадобится создать разделитель. Когда мы создавали первоначальную страницу, мы совсем упустили это из виду, но ведь мы тогда ничего и не знали о плавающих контейнерах. Так что сейчас нам придется исправить HTML-код вручную.

Сначала создадим стилевой класс для разделителя. Назовем его `spacer` и дадим ему всего один параметр — другие ему не нужны:

- ◆ **Clear** — `both` (на всякий случай).

После этого переключаемся в режим отображения HTML-кода и находим фрагмент, описывающий контейнеры заголовка сайта и набора гиперссылок. Вот он (возможно, у вас этот фрагмент будет слегка отличаться):

```
<DIV CLASS="header">
```

```

</DIV>
<DIV>
  <P><A HREF="main.htm" TARGET="main">Главная</A></P>
  <P><A HREF="passions.htm" TARGET="main">Увлечения</A></P>
  <P><A HREF="articles.htm" TARGET="main">Статьи</A></P>
  <P><A HREF="humour.htm" TARGET="main">Юмор</A></P>
  <P><A HREF="about.htm" TARGET="main">Обо мне</A></P>
</DIV>

```

Вставляем в этот фрагмент код, описывающий разделитель, после чего он станет таким (вставленный нами код выделен полужирным шрифтом):

```

<DIV CLASS="header">
  <H1 ALIGN="CENTER">Сайт студента Васильева В. В.</H1>
</DIV>
<DIV CLASS="spacer">&nbsp;</DIV>
<DIV>
  <P><A HREF="main.htm">Главная</A></P>
  <P><A HREF="passions.htm">Увлечения</A></P>
  <P><A HREF="articles.htm">Статьи</A></P>
  <P><A HREF="humour.htm">Юмор</A></P>
  <P><A HREF="about.htm">Обо мне</A></P>
</DIV>

```

Вот и все. Теперь можно переключиться в режим отображения страницы.

Следующий в очереди — набор гиперссылок. Стилиевой класс, который мы привяжем к содержащему его контейнеру, будет называться `nav` и иметь такие параметры:

- ◆ **Width** — 80 пикселей (этого должно хватить);
- ◆ **Float** — `left`.

Привязываем этот стилевой класс к контейнеру с набором гиперссылок, любуемся на результат и переходим к контейнеру с основным содержимым. Для него мы создадим стилевой класс `main` с параметрами:

- ◆ **Width** — 360 пикселей;
- ◆ **Float** — `left`.

Привязываем его к контейнеру с основным содержимым и не мешкая приступаем к контейнеру с колонкой новостей. Стилиевой класс для нее будет называться `news` и иметь такие же параметры, как и стилевой класс `nav`, а именно:

- ◆ **Width** — 100 пикселей;
- ◆ **Float** — `left`.

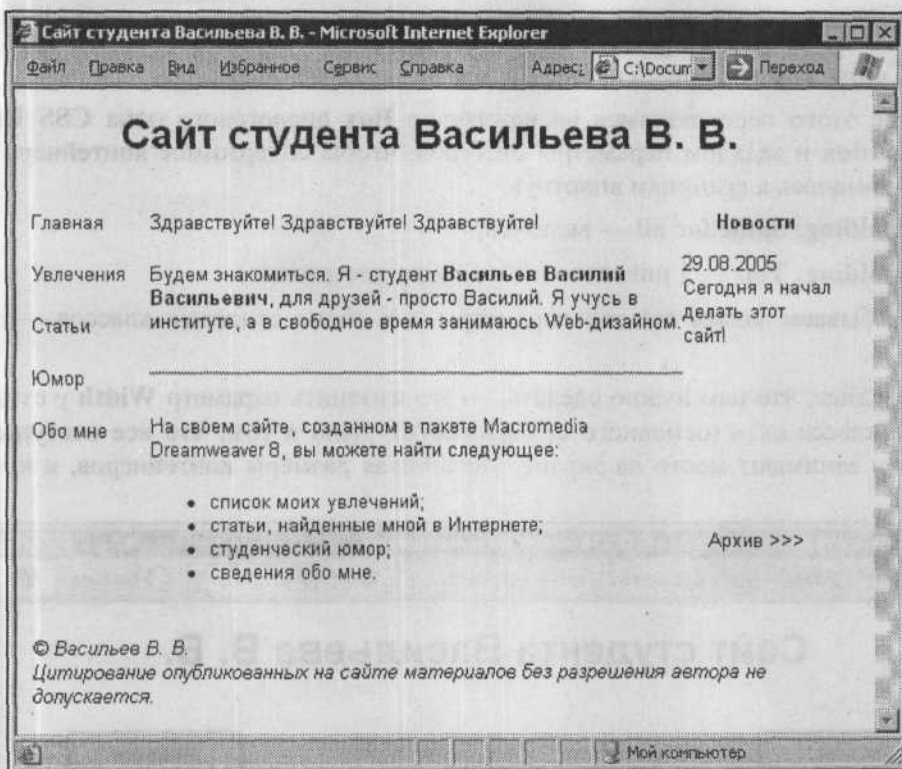


Рис. 11.5. Готовая страница, построенная с использованием плавающих контейнеров, в Web-обозревателе

Начнем с отступов между основным содержимым и границей контейнера, в котором оно находится. В стилевой классе `main` зададим такие параметры:

- ◆ **Padding: Same for all** — включен;
- ◆ **Padding: Top** — 5 пикселей (этого будет достаточно).

Теперь основное содержимое отступает от остальных контейнеров на 5 пикселей. Стало заметно лучше, не находите?

Рамки мы зададим для набора гиперссылок и колонки новостей, т. е. для стилевых классов `nav` и `news`. Для этого нам будет нужно задать такие параметры в категории **Border** диалогового окна **CSS Rule Definition**:

- ◆ **Style: Same for all** — включен;
- ◆ **Style: Top** — **solid** (сплошная линия);
- ◆ **Width: Same for all** — включен;
- ◆ **Width: Top** — **thin** (тонкая линия; по умолчанию ее толщина равна 2 пик-

◆ **Color: Same for all** — включен;

◆ **Color: Top** — черный цвет.

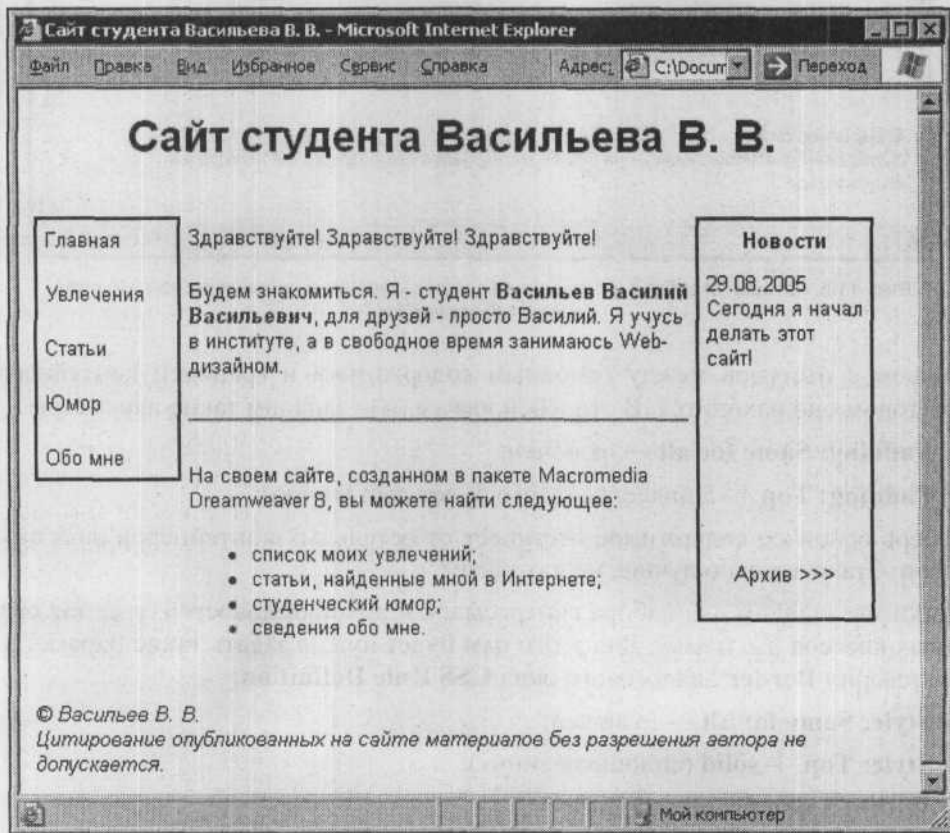
После этого переключимся на категорию **Box** диалогового окна **CSS Rule Definition** и зададим параметры отступов, чтобы содержимое контейнера не прижималось к границам вплотную:

◆ **Padding: Same for all** — включен;

◆ **Padding: Top** — 5 пикселей (этого будет достаточно).

Не забываем задать все эти параметры для обоих стилевых классов — `nav` и `news`.

Последнее, что нам нужно сделать, — это изменить параметр **Width** у стилевого класса `main` (основного содержимого). Дело в том, что все отступы и рамки занимают место на экране, увеличивая размеры контейнеров, к кото-



рым они применены. Если сложить величины всех отступов трех контейнеров (5 пикселей * 2 стороны * 3 контейнера = 30 пикселей) и толщину рамок (2 пикселя * 2 стороны * 2 контейнера = 8 пикселей), получим 38 пикселей — именно на столько мы должны уменьшить ширину контейнера с основным содержимым. Новая ширина основного содержимого будет равна $360 - 38 = 322$ пикселя.

Закончив работу, выводим готовую страницу в Web-обозревателе и смотрим на получившийся результат — рис. 11.6.

Контейнеры набора гиперссылок и колонки новостей получились разной высоты. Но все равно выглядит неплохо, даже стильно!..

Здесь мы столкнулись с одним из недостатков контейнеров. Мы не можем задать для набора плавающих контейнеров, выстроившихся в одну линию, одинаковую высоту. А для ячеек таблицы — пожалуйста!

И еще один недостаток контейнеров прямо бьет в глаза. Не догадались, какой?

Пример реализации саморастягивающихся контейнеров

Правильно! Мы не можем сделать один из контейнеров саморастягивающимся. Мы должны жестко задавать ширину каждого контейнера либо в пикселях, либо в процентах.

Если мы зададим ширину каждого контейнера в пикселях, полная ширина всей линии контейнеров будет неизменна. Поэтому либо в окне Web-обозревателя справа останется свободное место, что нравится не всем, либо посетителю придется прокручивать содержимое окна по горизонтали, что не нравится никому. Если же мы зададим ширину контейнеров в процентах, то все контейнеры будут растягиваться и сжиматься при изменении ширины окна Web-обозревателя, что выглядит очень некрасиво. Ну, а если мы зададим ширину одних контейнеров в пикселях, а ширину других — в процентах, результаты будут непредсказуемыми.

Было бы очень неплохо, если бы ширина всех контейнеров, кроме одного (а именно основного содержимого), была задана жестко, а оставшийся один контейнер сам подгонял свою ширину под размер окна Web-обозревателя. Но как это сделать? Неужели никак?

Вообще-то, есть один способ. Но довольно мудреный и крайне неочевидный. И вдобавок нам часто придется править HTML-код вручную.

Давайте переименуем созданную нами ранее страницу default.htm и табли-

styles_old.css соответственно. Лучше всего это сделать прямо в среде Dreamweaver. Когда мы переименуем таблицу стилей, Dreamweaver выведет на экран диалоговое окно **Update Files**, похожее на уже знакомое нам окно **Update Template Files** (см. рис. 10.9). Нажмем кнопку **Update**, чтобы выполнить обновление самой страницы, к которой эта таблица стилей привязана.

После этого создадим новую страницу и сохраним ее под именем default.htm. Это и будет новая главная страница нашего сайта, имеющая в своем составе саморастягивающийся контейнер.

Поместим на страницу наш первый контейнер, который будет содержать заголовки сайта, введем в него текст заголовка и отформатируем его. Создадим для него стилевой класс под именем *header*, поместим его во внешнюю таблицу стилей styles.css и зададим для него такие параметры:

- ◆ **Width** — 100 % (поскольку мы делаем Web-дизайн, который будет подстраиваться под ширину окна Web-обозревателя);
- ◆ **Float** — left.

Привяжем готовый стилевой класс к заголовку сайта. После этого создадим второй контейнер, который станет разделителем. (Поскольку мы создаем эту страницу "с нуля", нам нет нужды править HTML-код вручную.) Стилиевой класс, который мы к нему привяжем, будет называться *spacer* и содержать параметр:

- ◆ **Clear** — both.

По сравнению со старой страницей здесь ничего не изменилось. Разделители — штука, можно сказать, унифицированная, она не меняется практически никогда.

Теперь создаем очередной контейнер и помещаем в него набор гиперссылок. Стилиевой класс этого контейнера — *nav* — будет таким же, как у старой страницы:

- ◆ **Width** — 80 пикселей;
- ◆ **Float** — left;
- ◆ **Style: Same for all** — включен;
- ◆ **Style: Top** — solid (сплошная линия);
- ◆ **Width: Same for all** — включен;
- ◆ **Width: Top** — thin (тонкая линия);
- ◆ **Color: Same for all** — включен;
- ◆ **Color: Top** — черный цвет;
- ◆ **Padding: Same for all** — включен;

Здесь мы сразу включили в стиль рамки и отступы, чтобы потом не делать никаких исправлений.

На очереди — контейнер с основным содержимым. Его стилевой класс `main` будет иметь несколько другие параметры:

- ◆ **Width** — 100 % (это обязательно!);
- ◆ **Float** — **right** (да, мы заставляем этот контейнер сдвинуться к правому — противоположному краю страницы);
- ◆ **Margin: Same for all** — выключен;
- ◆ **Margin: Left** — -104;
- ◆ **Padding: Same for all** — включен;
- ◆ **Padding: Top** — 5 пикселей.

Последние два параметра задают отступы в 5 пикселей, что нам знакомо еще по старой странице. Оставим их в покое и обратим внимание на первые четыре параметра.

Итак, первым же делом мы задаем ширину контейнера, который собираемся сделать саморастягивающимся, в 100 %. Понятно, что этот контейнер будет растягиваться на всю ширину окна Web-обозревателя. Также понятно, что он будет сдвинут к правому краю окна (см. второй параметр).

Третьим и четвертым параметрами мы задаем отступ слева — о ужас! — в -104 пиксела. Во-первых, откуда взялась величина 104? А это просто сумма ширины контейнера набора гиперссылок (80 пикселей), четырех отступов (5 пикселей * 4 стороны = 20 пикселей) и двух значений толщины рамки (2 пиксела * 2 стороны = 4 пиксела). Но, во-вторых, почему минус?

Дело в том, что Web-обозреватель, встретив отрицательное значение отступа, посчитает его свободным пространством и постарается впихнуть туда предыдущий контейнер (или следующий, если отрицательный отступ задан справа). Причем это срабатывает только тогда, когда отступ задан с помощью параметра **Margin** (расстояние между воображаемой границей элемента страницы и соседним элементом).

Давайте сами посмотрим, что у нас получилось. Сохраним страницу и таблицу стилей и вызовем окно Web-обозревателя, нажав клавишу <F12>. (Dreamweaver обрабатывает отрицательные отступы некорректно и покажет нам совсем не то, что Web-обозреватель.) Мы увидим то, что показано на рис. 11.7.

Более того — если мы изменим размеры окна Web-обозревателя, созданный нами саморастягивающийся контейнер тоже изменит свою ширину. А ширина контейнера с набором гиперссылок останется неизменной. А ведь это то, что нам нужно!

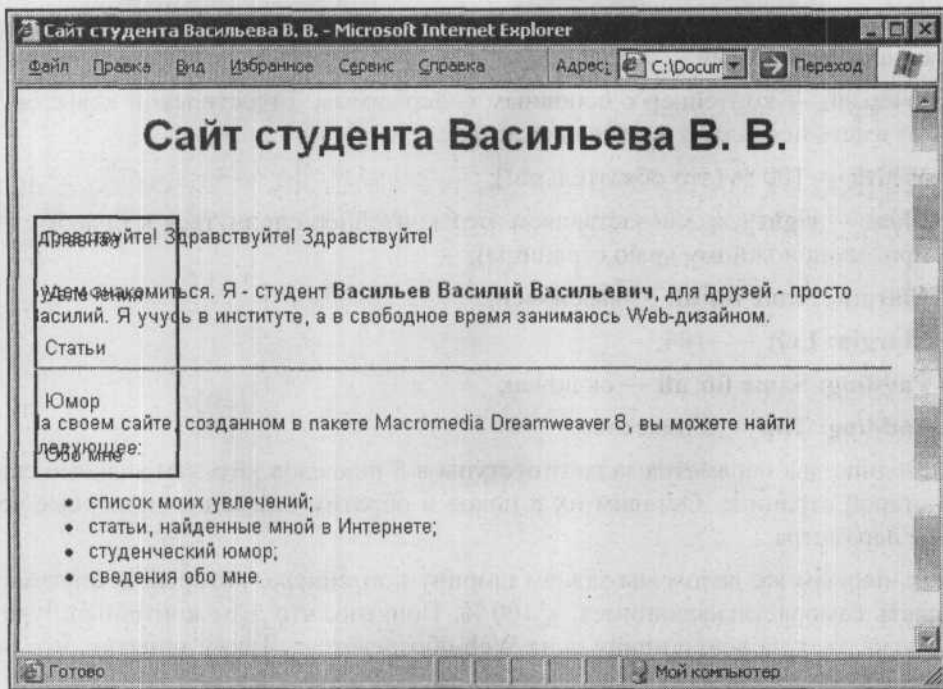


Рис. 11.7. Два плавающих контейнера, созданные с использованием отрицательного отступа слева

Одна только беда — эти контейнеры накладываются друг на друга. Надо как-то их "развести". Проще всего сделать это, поместив контейнер с основным содержимым в другой контейнер и задав для него значение **Margin: Left** равным 104, т. е. таким же, только положительным. Переключаемся в режим отображения HTML-кода, ищем код, определяющий основное содержимое, и вставляем код, создающий новый контейнер. Вот что мы должны получить после этого (вставленный нами код выделен полужирным шрифтом):

```
<DIV CLASS="main">
<DIV CLASS="main1">
  <P>Здравствуйте! Здравствуйте! Здравствуйте!</P>
  . . .
  <LI><A HREF="about.htm">сведения обо мне</A>.</LI>
</UL>
</DIV>
</DIV>
```

Видно, что мы привязали к новому контейнеру стилевой класс `main1`, который пока не существует. Создадим его и зададим для него единственный пара-

◆ **Margin: Same for all** — выключен;

◆ **Margin: Left** — 104.

Вот теперь все в порядке — рис. 11.8.

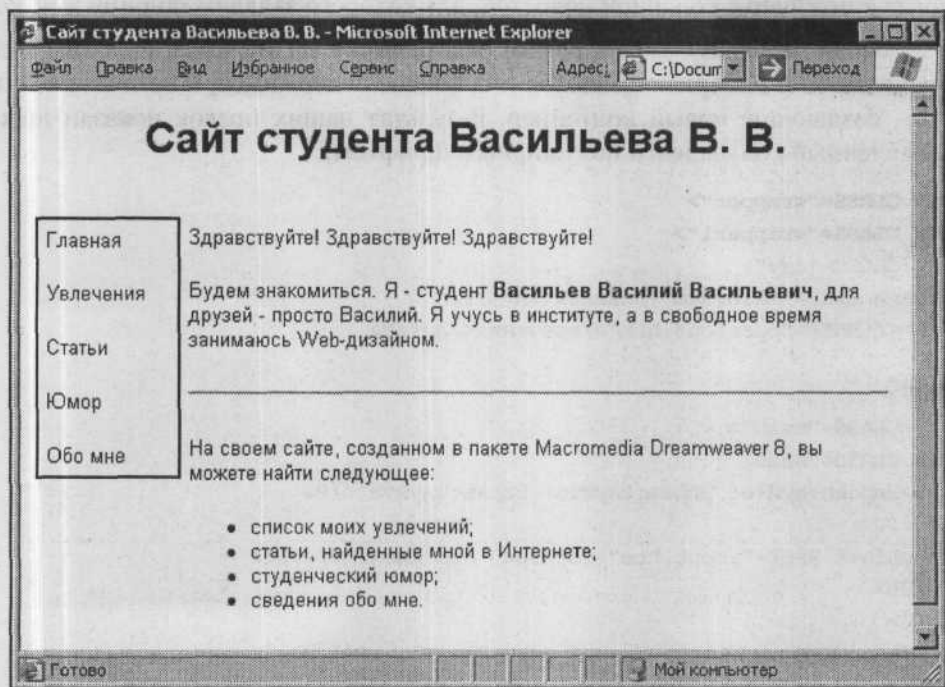


Рис. 11.8. Два плавающих контейнера, созданные с использованием отрицательного отступа слева, после устранения их наложения друг на друга

Следующим нашим шагом будет создание контейнера с колонкой новостей. И здесь мы столкнемся с новой проблемой. Дело в том, что, в одной линии плавающих контейнеров, содержащей саморастягивающийся контейнер, может нормально существовать только один контейнер с жестко заданной шириной. То есть, если мы создали в одной линии один саморастягивающийся контейнер и один контейнер с жестко заданной шириной (неважно — в пикселах или процентах), то третий контейнер с заданной шириной мы добавить в эту линию не сможем.

На заметку

В одной линии плавающих контейнеров, содержащей саморастягивающийся контейнер, мы можем создать сколько угодно контейнеров с заданной шириной — стандарты HTML и CSS это допускают. Проблема в другом: Web-обозреватели будут обрабатывать эту конструкцию как им заблагорассудится, и уж

Что же нам делать? Выход довольно прост: мы поместим только что созданные контейнеры набора гиперссылок (с жестко заданной шириной) и основного содержимого (саморастягивающийся) в другой контейнер, который тоже сделаем саморастягивающимся. А уже к этому-то контейнеру мы приделаем справа контейнер с колонкой новостей, для которого зададим ширину жестко.

Итак, снова переключаемся в режим отображения HTML-кода, находим код, определяющий набор гиперссылок и основное содержимое, и вставляем теги `<DIV>`, создающие новый контейнер. Результат наших правок показан ниже (добавленный код выделен полужирным шрифтом):

```
<DIV CLASS="wrapper">
<DIV CLASS="wrapper1">
<DIV CLASS="nav">
  <P><A HREF="main.htm">Главная</A></P>
  <P><A HREF="passions.htm">Увлечения</A></P>
  . . .
</DIV>
<DIV CLASS="main">
<DIV CLASS="main1">
  <P>Здравствуйте! Здравствуйте! Здравствуйте!</P>
  . . .
  <LI><A HREF="about.htm">сведения обо мне</A>.</LI>
</UL>
</DIV>
</DIV>
</DIV>
</DIV>
```

Мы поместили контейнеры с набором гиперссылок и основным содержимым не в один контейнер, а последовательно в два. Эта конструкция нужна нам для того, чтобы потом устранить наложение основного содержимого на колонку новостей (об этом мы говорили выше). К этим контейнерам мы привязали стилевые классы `wrapper` и `wrapper1`. Стилиевой класс `wrapper` будет иметь параметры:

- ◆ **Width** — 100 % (поскольку он саморастягивающийся);
- ◆ **Float** — `left` (он должен находиться левее колонки новостей);
- ◆ **Margin: Same for all** — выключен;
- ◆ **Margin: Right** — `-114`.

Все это нам уже знакомо по контейнеру с основным содержимым. Только теперь мы прижимаем его не к правому краю окна Web-обозревателя, а к левому и соответственно оставляем свободное место для нового контейнера

справа. А величина 114 — это сумма ширины контейнера колонки новостей (100 пикселей), двух отступов, заданных в третьем контейнере (5 пикселей * 2 стороны = 10 пикселей) и двух значений толщины рамки, заданных у него же (2 пикселя * 2 стороны = 4 пикселя).

Что касается стилевого класса `wrapper1`, все уже, наверное, догадались, каков будет его единственный параметр:

- ◆ **Margin: Same for all** — выключен;
- ◆ **Margin: Right** — 114.

Не отвлекаясь от просмотра получившегося — нечего пока там смотреть, — приступаем к созданию третьего контейнера — с колонкой новостей. Привязываем к нему стилевой класс `nav`, чьи параметры будут такими:

- ◆ **Width** — 100 пикселей;
- ◆ **Float** — **right**;
- ◆ **Style: Same for all** — включен;
- ◆ **Style: Top** — **solid** (сплошная линия);
- ◆ **Width: Same for all** — включен;
- ◆ **Width: Top** — **thin** (тонкая линия);
- ◆ **Color: Same for all** — включен;
- ◆ **Color: Top** — черный цвет;
- ◆ **Padding: Same for all** — включен;
- ◆ **Padding: Top** — 5 пикселей.

Вот теперь можно вызвать Web-обозреватель и полюбоваться на наше творение — рис. 11.9.

Осталось создать еще один разделитель и последний контейнер, содержащий сведения об авторских правах. С разделителем проблем у нас не будет — он создается так же, как первый. А вот стилевой класс `copyright`, привязанный к контейнеру сведений об авторских правах, будет иметь такие параметры:

- ◆ **Width** — 100 % (он также должен занимать всю ширину окна Web-обозревателя);
- ◆ **Float** — **left**.

Вот и все. Можно снова вызвать Web-обозреватель, поизменять размер его окна и посмотреть, что из этого получится. А получиться у нас должно то, что показано на рис. 11.6.

Свершилось! Мы сделали Web-дизайн, основанный на контейнерах, который подстраивается под размеры окна Web-обозревателя. Это значит, что мы

вполне можем обойтись без громоздких и запутанных таблиц. Да здравствуют контейнеры!

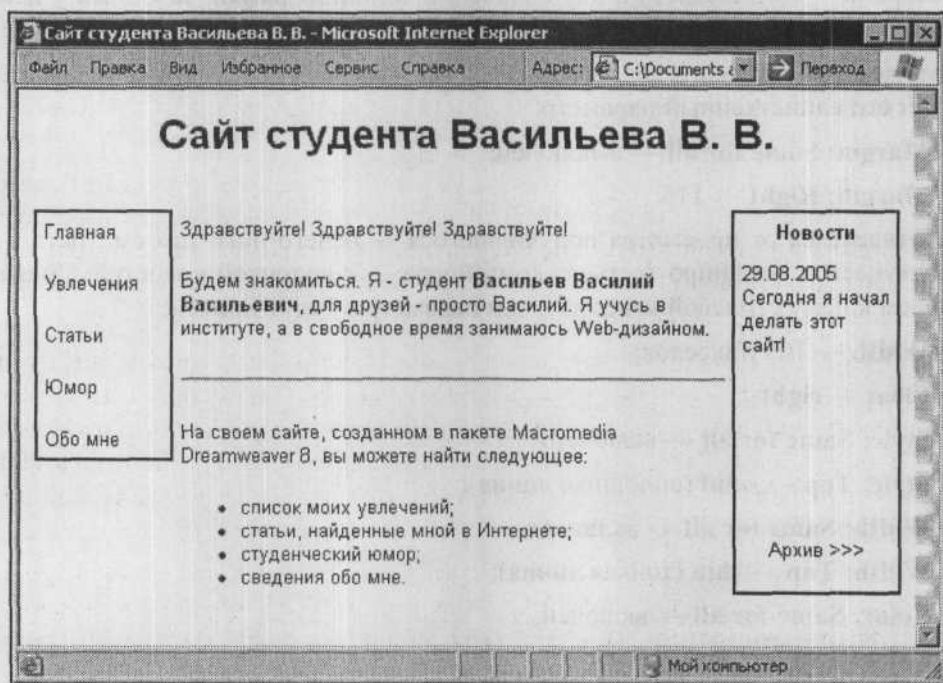


Рис. 11.9. Три плавающих контейнера

Недостатки контейнерного Web-дизайна

На десерт — опять ложка отборного дегтя. Поговорим о недостатках контейнеров и самого контейнерного Web-дизайна и выясним, можно ли их избежать.

С первым недостатком контейнеров мы столкнулись, когда создавали самую первую страницу, основанную на контейнерном Web-дизайне. Мы не могли задать одинаковую высоту для всех плавающих контейнеров в одной линии — все они имели разную высоту. Это хорошо видно на рис. 11.6.

Устранить этот недостаток можно тремя способами.

1. Жестко задать высоту всех контейнеров в линии. Для этого нужно воспользоваться комбинированным списком **Height** категории **Box** диалогового окна **CSS Rule Definition** (см. рис. 7.9). Только нужно учитывать, что это придется делать на всех Web-страницах сайта, что зачастую трудно.

2. Избавиться от видимых границ контейнеров и, вообще, от всего того, что "выдает" разность в их высоте. Так, если мы удалим со своих страниц, созданных на этой лекции, видимые границы, посетители сайта и не узнают, что контейнеры имеют разную высоту. Но тогда надо оформлять контейнеры как-то иначе, ведь без границ они выглядят очень плохо.
3. В конце концов, избавиться от контейнеров и вернуться к таблицам. Тоже выход из положения...

Второй недостаток мы сполна ощутили, когда экспериментировали с саморастягивающимися контейнерами. Чтобы сделать основное содержимое сайта "резиновым", нам пришлось заметно усложнить код HTML нашей страницы. Вдобавок мы использовали для этого отрицательные отступы — прием, мягко говоря, неочевидный.

Решить этот недостаток можно единственным способом — использовать на страницах только контейнеры с жестко заданной шириной. Такой Web-дизайн выглядит не так уж и плохо; его используют очень многие известные сайты. Сам автор этой книги на своем сайте (<http://vgi.volsu.ru:8000/~Vladimir.Dronov/>) использовал именно контейнеры с жестко заданной шириной, и получилось, на его взгляд, неплохо.

Да и такой ли уж это серьезный недостаток... HTML-код, создающий таблицы, много объемнее и сложнее, и приемы для создания таблиц используются тоже зачастую малоочевидные. Так ведь таблицами пользуются, и пользуются чаще, чем контейнерами.

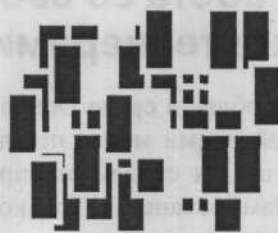
Так или иначе, выбор за вами. Решайте сами — пользоваться или нет. Дело автора — только рассказать, чем можно пользоваться.

На следующем занятии...

На этом все о плавающих контейнерах и Web-дизайне, на них основанном. Компактные, гибкие, они вполне могут потеснить таблицы. И теснят!

Но контейнеры могут не только покорно "плавать" от одного края окна Web-обозревателя к другому. Есть у них еще одно замечательное свойство, реализуемое все теми же неисчерпаемыми стилями CSS. О нем-то мы и поговорим на следующем занятии.

ЗАНЯТИЕ 12



Свободно позиционируемые контейнеры

Да, полезная штука эти контейнеры! Могут располагаться на странице в линию, смещаться к нужному нам краю, иметь заданную нами ширину или быть саморастягивающимися, "резиновыми". Могут вмещать в себя все что угодно. И вдобавок не так громоздки, как таблицы. Поэтому и используются в Web-дизайне.

На *занятии 11* мы как раз познакомились с контейнерным Web-дизайном. И даже сделали две Web-страницы, основанные на контейнерах. На этом занятии мы продолжим их изучение.

Контейнеры неотделимы от стилей CSS. Все их замечательные свойства — смещение к краю страницы, фиксированные ширина и высота или, наоборот, автоматическая подгонка под размер страницы, отступы, границы — суть возможности, предоставляемые стилями. Без стилей контейнеры — ничто, просто незатейливые текстовые абзацы.

Именно стили CSS дают контейнерам еще одну замечательную возможность. А именно — точное задание местонахождения контейнеров на странице. Это значит, что контейнеры могут располагаться абсолютно где угодно на странице, там, где мы их создадим. Такие контейнеры называются *свободно позиционируемыми*, или просто *свободными*.


Давайте выясним, что это за свободно позиционируемые контейнеры, на примере. Создадим третью версию главной страницы нашего многострадального сайта Site 4, уже на основе свободных контейнеров. Обе старые страницы, вместе с привязанными к ним таблицами стилей, уберем куда-нибудь или переименуем. Создадим новую страницу, которую сохраним под все тем же именем — `default.htm` — и откроем ее

Работа со свободно позиционируемыми контейнерами

Вообще, в среде Dreamweaver работать со свободно позиционируемыми контейнерами много проще, чем с плавающими. Нам не придется ни создавать для них стили, ни править HTML-код, чтобы внести какие-то изменения. Нам останется только двигать мышью — все остальное сделает за нас Dreamweaver.

Создание свободно позиционируемых контейнеров

В работе со свободно позиционируемыми контейнерами нам могут помочь описанные на *занятии 9* измерительные линейки, координатная сетка и направляющие. Поэтому стоит включить их прямо сейчас.

Первым свободным контейнером, который мы создадим, станет заголовок сайта. Чтобы его создать, проще всего нажать кнопку **Draw Layer** , расположенную на вкладке **Layout** инструментария объектов. После этого поместим курсор мыши в окно документа, и он примет вид крестика — это значит, что теперь мы можем нарисовать свободный контейнер, как на *занятии 9* рисовали ячейки и таблицы разметки. Щелкнем мышью там, где должен быть верхний левый угол создаваемого контейнера, и проведем мышью, не отпуская ее левой кнопки. Мы увидим, что между курсором мыши и точкой, где щелкнули ранее, тянется прямоугольник, обозначающий границы вновь создаваемого элемента. После того как он примет желаемые размеры, отпустим левую кнопку мыши. Все, первый наш свободно позиционируемый контейнер создан (рис. 12.1).

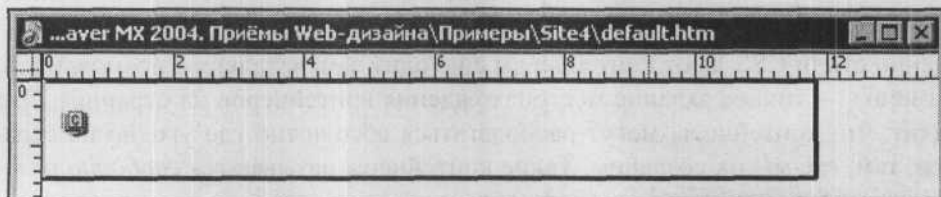


Рис. 12.1. Свободно позиционируемый контейнер в окне документа Dreamweaver (координатная сетка временно отключена)

Второй способ создания свободного контейнера — выбор пункта **Layer** подменю **Layout Object** меню **Insert**. В этом случае нам не придется рисовать новый контейнер в окне — он появится сразу же. Нам останется только изменить его размеры и местоположение (как это сделать, мы узнаем чуть позже).


Посмотрим на рис. 12.1. Изображенный на нем свободный контейнер окружает толстая синяя рамка — это значит, что контейнер выделен. Текстовый курсор находится внутри него, поэтому мы сразу же можем ввести текст заголовка сайта. Чтобы выйти из свободного контейнера, достаточно щелкнуть мышью где-либо вне его. А чтобы снова выделить свободный контейнер, нужно, соответственно, щелкнуть мышью по нему (но не по его границе).

Теперь щелкнем где-нибудь на границе свободного контейнера. Также можно щелкнуть внутри свободного контейнера, удерживая нажатой клавишу <Shift>. Результат показан на рис. 12.2. В этом случае говорят о *выделении границы* свободного контейнера.



Рис. 12.2. Свободно позиционируемый контейнер после щелчка на его границе

Видно, что в данном случае свободно позиционируемый контейнер предлагает нам полный набор маркеров изменения размера, с помощью которых мы можем задать его размеры. Кроме того, каждый свободный контейнер снабжен небольшим ярлычком (на рис. 12.2 не виден, т. к. контейнер примыкает к самому краю окна), называемым *захватом*. "Ухватившись" за него мышью, мы можем перемещать свободный контейнер. Точно такого же результата можно достичь, "ухватившись" мышью за саму границу контейнера, но специальный "захват", согласитесь, удобнее. Щелкнув по нему, мы также сможем выделить границу свободного контейнера. А для удаления выделенного свободного контейнера достаточно нажать клавишу .

Кроме всего прочего, в окне документа заметен некий значок . Это *значок свободного контейнера*, показывающий, где он должен был бы находиться, если бы не был свободным.

Внимание!

Скорее всего, изначально Dreamweaver не будет показывать значки свободно позиционируемых контейнеров. Чтобы увидеть их, выберите пункт **Preferences** в меню **Edit**, переключитесь на категорию **Invisible Elements**, включите флажок **Anchor points for layers** и нажмите кнопку **OK**.

Мы можем создавать свободно позиционируемые контейнеры, вложенные в другие свободно позиционируемые контейнеры. Собственно, в этом нет ни-

чего странного — еще на *занятии 9* мы выяснили, что контейнеры могут содержать все что угодно, в том числе и другие контейнеры. Пример вложенных друг в друга свободных контейнеров показан на рис. 12.3. Создаются они точно так же, как было описано выше.

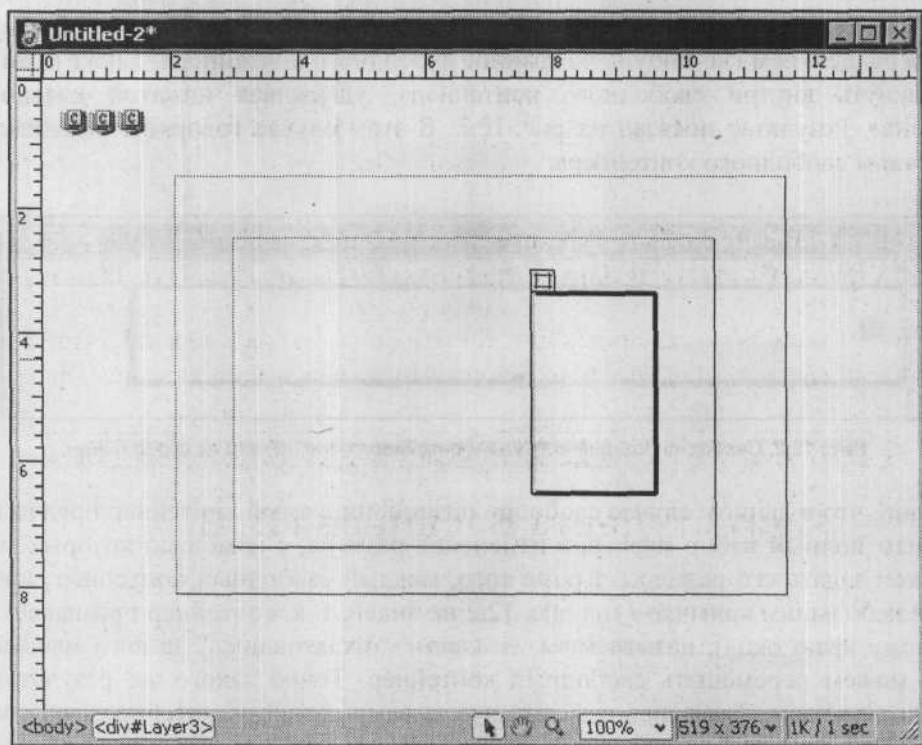


Рис. 12.3. Три свободно позиционируемых контейнера, вложенные друг в друга

У свободных контейнеров есть замечательное свойство — они могут перекрывать друг друга. При этом содержимое контейнеров, находящихся ниже, будет "просвечивать" сквозь пустые места в содержимом контейнеров верхних. Это, кстати, можно использовать в Web-дизайне для создания, например, теней, "отбрасываемых" контейнерами: контейнер, имеющий черный фон, подкладывается под нужный с небольшим смещением.

Здесь необходимо сказать о так называемом *z-индексе*. Это своего рода номер свободного контейнера, если считать с самого "низа", порядок, в котором они перекрывают друг друга. Самый нижний свободный контейнер имеет номер 0, тот, что находится над ним, — 1 и т. д. Понятно, что контейнеры с большим *z-индексом* перекрывают контейнеры с меньшим *z-индексом*.

На заметку

Z-индекс может принимать и отрицательные значения. Это может быть полезно, если нужно быстро поместить какой-то свободный контейнер ниже всех остальных.

В Dreamweaver контейнеры, нарисованные позже, находятся выше контейнеров, нарисованных раньше. Но мы можем поменять порядок их перекрытия друг другом (фактически — увеличить или уменьшить z-индекс на единицу). Для этого достаточно выделить нужный свободный контейнер и в подменю **Arrange** меню **Modify** выбрать пункт **Bring To Front**, чтобы увеличить z-индекс контейнера на единицу, или пункт **Send To Back**, чтобы его уменьшить.

В том же самом подменю **Arrange** меню **Modify** находится пункт-выключатель **Prevent Layer Overlaps**. Если он включен, Dreamweaver не позволит нам "наложить" один свободно позиционируемый контейнер на другой, т. е. мы не сможем ни переместить его, ни изменить его размеры так, чтобы он перекрыл другие свободные контейнеры. Поскольку в Web-дизайне перекры-

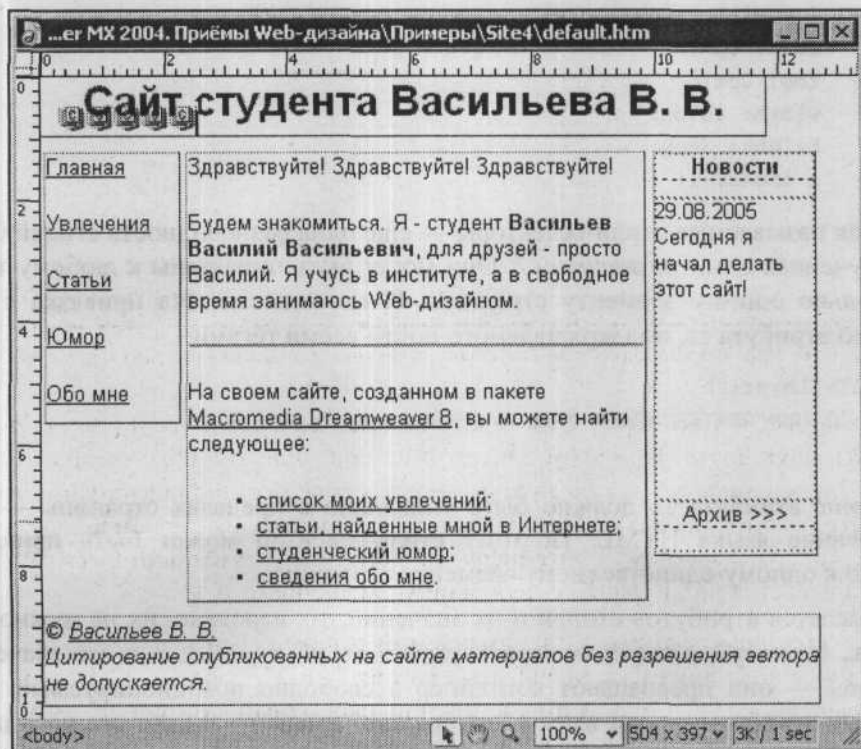


Рис. 12.4. Web-страница, созданная на основе свободно позиционируемых контейнеров

вающиеся свободные контейнеры используются нечасто, лучше всегда держать этот пункт-выключатель включенным.

Давайте создадим таким же образом еще четыре свободных контейнера: набор гиперссылок, основное содержимое, колонку новостей и сведения об авторских правах. После этого заполним их содержимым и разместим на странице. У нас должно получиться нечто, похожее на рис. 12.4.

Средства HTML и CSS для создания свободно позиционируемых контейнеров

А теперь переключимся в режим отображения HTML-кода и посмотрим, что натворил Dreamweaver. Видно, что свободные контейнеры, так же, как плавающие и фиксированные, создаются с помощью тегов `<DIV>`. Также Dreamweaver создал внутреннюю таблицу стилей и поместил в нее несколько весьма странных стилей — по одному на каждый свободный контейнер. Вот один из этих стилей:

```
#Layer1 { position: absolute;
  left: 0px;
  top: 0px;
  width: 447px;
  height: 36px;
  z-index: 1; }
```

Это так называемые стили-селекторы — еще одна разновидность стилей CSS, не изученная нами на *занятии 7*. Они могут быть привязаны к любому тегу, но только одному элементу страницы. И выполняется эта привязка с помощью атрибута `ID`, поддерживаемого почти всеми тегами:

```
<DIV ID="Layer1">
  <H1 ALIGN="CENTER">Сайт студента Васильева В. В.</H1>
</DIV>
```

Значение атрибута `ID` должно быть уникально в пределах страницы — это требование языка HTML. Поэтому стиль-селектор может быть привязан только к одному-единственному элементу страницы.

Что касается атрибутов стилей и их значений, то, вероятно, их не нужно пояснять. Пожалуй, следует отметить только атрибут `position` и его значение `absolute` — они превращают контейнер в свободно позиционируемый. Остальные атрибуты — они имеют "говорящие" имена — задают его координаты, размеры и `z`-индекс.

Параметры свободно позиционируемых контейнеров

Набор параметров свободных контейнеров, которые мы можем задать в Dreamweaver, весьма обширен. Если мы выделим границу свободного контейнера, редактор свойств их покажет (рис. 12.5).

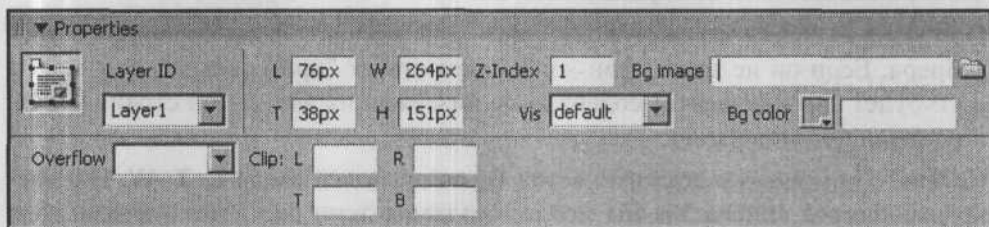


Рис. 12.5. Вид редактора свойств при выделенном свободном контейнере

В комбинированном списке **Layer ID** вводится уникальное имя свободного контейнера. Dreamweaver по умолчанию подставляет туда автоматически сгенерированное имя вида *Layer<Порядковый номер>*, которое мы можем при желании заменить на более вразумительное.

На заметку

Имя свободного контейнера используется Dreamweaver для именования стиля-селектора, привязанного к этому контейнеру.

В поля ввода **L** и **T** заносятся соответственно горизонтальная и вертикальная координаты верхнего левого угла свободно позиционируемого контейнера, а в поля ввода **W** и **H** — его ширина и высота. Эти величины также могут быть заданы в любой из единиц измерений, поддерживаемых CSS.

В поле ввода **Z-Index** задается уже известный нам z-индекс.

С помощью раскрывающегося списка **Vis** мы можем задать, будет ли свободный контейнер отображаться на странице (*видимость* контейнера). Здесь доступны четыре пункта:

- ◆ **inherit** — заставляет элемент "наследовать" видимость от родителя (например, другого контейнера, в который он вложен), т. е. если родитель видим, видим и сам этот контейнер, и наоборот;
- ◆ **visible** — делает контейнер видимым;
- ◆ **hidden** — делает контейнер невидимым (скрытым);
- ◆ **default** — как правило, действует аналогично **inherit**

Подобную функцию выполняет подменю **Visibility** контекстного меню свободного контейнера. Оно содержит те же четыре пункта, что и раскрывающийся список **Vis**.

В поле ввода **Bg image** заносится имя файла графического изображения, которое будет использовано в качестве фона контейнера. Также можно щелкнуть значок папки, расположенный справа от поля ввода, и выбрать нужный файл в появившемся на экране диалоговом окне **Select File**.

С помощью селектора цветов **Bg color** задается цвет фона свободного контейнера. Если он не задан, фон контейнера станет "прозрачным", т. е. сквозь него будет "просвечивать" содержимое низлежащих элементов страницы. Он и не задан по умолчанию.

Кстати, с помощью селектора цветов **Bg color**, полей ввода **L**, **T**, **W**, **H** и раскрывающегося списка **Vis** мы можем создавать тени под контейнерами. Для этого делаем вот что:

- ◆ создаем новый свободный контейнер;
- ◆ задаем для него черный или темно-серый цвет фона;
- ◆ задаем для него те же самые ширину и высоту, что и у контейнера, под которым нужно создать тень;
- ◆ помещаем точно под нужным контейнером, задав меньший на единицу z-индекс;
- ◆ задаем для него координаты, смещенные, скажем, на 10 пикселей относительно координат контейнера, под которым нужно создать тень. Все, тень готова!

С помощью раскрывающегося списка **Overflow** задается поведение свободного контейнера при его *переполнении*, т. е. в случае, если его содержимое в нем не помещается. Здесь доступны четыре пункта:

- ◆ **visible** — заставляет свободный контейнер растянуться по вертикали, чтобы вместить в себя все содержимое;
- ◆ **hidden** — просто скрывает ("обрезает") ту часть содержимого свободного контейнера, которая в нем не помещается;
- ◆ **scroll** — заставляет свободный контейнер отобразить полосы прокрутки независимо от того, помещается ли в нем все содержимое или нет;
- ◆ **auto** — заставляет свободный контейнер отобразить полосы прокрутки только тогда, когда его содержимое в нем не помещается;
- ◆ пункт *<пустая строка>* аналогичен **visible** (значение по умолчанию).

Группа полей ввода **Clip** позволяет задать координаты *видимой области* свободного контейнера. Изначально свободный

димую область, иначе говоря, фрагмент содержимого, который виден в окне Web-обозревателя. Содержимое, не входящее в эту область, скрывается. Задание видимой области может понадобиться для создания каких-либо спецэффектов на Web-странице.

В группу полей ввода **Clip** входят четыре поля:

- ◆ **L** — горизонтальная координата верхнего левого угла;
- ◆ **T** — вертикальная координата верхнего левого угла;
- ◆ **R** — горизонтальная координата нижнего правого угла;
- ◆ **B** — вертикальная координата нижнего правого угла.

Работа с группой свободно позиционируемых контейнеров

Dreamweaver, кроме всего прочего, позволяет нам выделить одновременно несколько свободно позиционируемых контейнеров и произвести над ними различные манипуляции. В этом случае группа выделенных контейнеров ведет себя как единое целое.

Чтобы выделить сразу несколько свободных контейнеров, мы должны сделать вот что. Сначала выделяем один из них, щелкнув на нем мышью (выделен сам контейнер или его граница — значения не имеет). После этого щелкаем на границах всех остальных контейнеров, удерживая нажатой клавишу <Shift>. Все контейнеры, по границам которых мы щелкнули, будут выделены. Заметим при этом, что маркеры изменения размеров всех выделенных таким образом контейнеров, кроме того, на котором мы щелкнули в последнюю очередь, будут иметь вид белых, а не черных, квадратов (рис. 12.6).

Прежде всего, мы можем перемещать выделенные таким образом свободные контейнеры, причем будет перемещаться сразу вся группа. Делается это все так же — мышью.

Кроме того, мы можем изменить размеры выделенных контейнеров, точнее, уравнивать их ширину или высоту. Для этого достаточно в подменю **Arrange** меню **Modify** выбрать пункт **Make Same Width** (одинаковая ширина — комбинация клавиш <Ctrl>+<Shift>+<7>) или **Make Same Height** (одинаковая высота — комбинация клавиш <Ctrl>+<Shift>+<9>).

Для группы свободно позиционируемых контейнеров также доступна возможность выравнивания по соответствующей стороне крайнего с этой стороны контейнера, входящего в эту группу. Например, если задано выравнивание по верху, все контейнеры выравниваются по верхней стороне самого верхнего контейнера.

А задать выравнивание мы можем, используя следующие пункты подменю **Arrange** меню **Modify**:

- ◆ **Align Left** (комбинация клавиш <Ctrl>+<Shift>+<1>) — выравнивание по левому краю;
- ◆ **Align Right** (комбинация клавиш <Ctrl>+<Shift>+<3>) — по правой стороне;
- ◆ **Align Top** (комбинация клавиш <Ctrl>+<Shift>+<4>) — по верхней стороне;
- ◆ **Align Bottom** (комбинация клавиш <Ctrl>+<Shift>+<6>) — по нижней стороне.

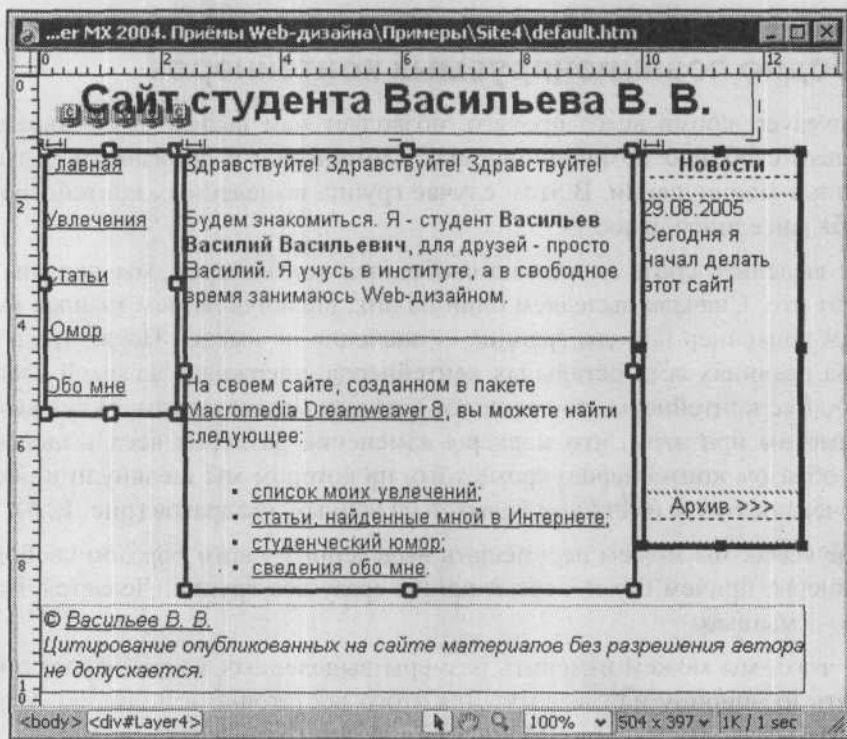


Рис. 12.6. Группа выделенных свободно позиционируемых контейнеров

И наконец, выделив группу контейнеров, мы можем задавать некоторые параметры сразу для всех контейнеров, входящих в нее. А именно координаты и размеры (поля ввода **L**, **T**, **W** и **H** редактора свойств), видимость (раскрывающийся список **Vis**), фоновое изображение (поле ввода **Bg image**) и цвет фона (поле ввода **Bg color**).

Использование панели *Layers*

На нашей Web-странице мы создали довольно много свободно позиционируемых контейнеров. Поэтому для работы с ними нам пригодится специально для этого предназначенная панель **Layers** (рис. 12.7). Она появляется на экране после включения пункта-выключателя **Layers** в меню **Window** или нажатия клавиши <F2>.

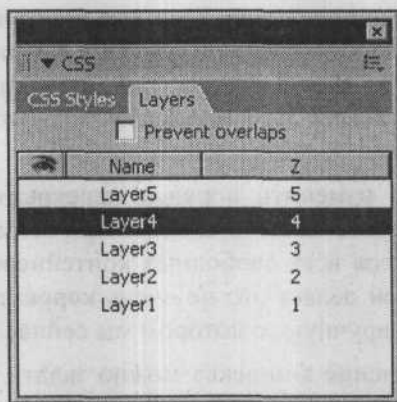


Рис. 12.7. Панель **Layers**

Большую часть этой панели занимает список уже созданных на странице свободно позиционируемых контейнеров. Он имеет вид таблицы, состоящей из трех колонок: "глаз" (видимость), **Name** (имя свободного контейнера) и **Z** (z-индекс). Мы можем менять размеры двух последних колонок, перетаскивая мышью границу между ними. И, разумеется, любой из элементов, перечисленных в списке, можно выделить и произвести над ним разные действия.

В верхней части панели находится флажок **Prevent overlaps**. Если он включен, Dreamweaver следит, чтобы свободные контейнеры не перекрывали друг друга. Этот флажок аналогичен пункту **Prevent Layer Overlaps**, находящемуся в подменю **Arrange** меню **Modify**, о котором мы уже говорили.

Пункты списка свободных контейнеров выделяются так же, как пункты любого другого списка. Но, кроме того, мы можем выделить сразу несколько контейнеров. Для этого сначала выделим первый контейнер обычным щелчком мыши, после чего щелкнем на остальных контейнерах, удерживая нажатой клавишу <Shift>.

Как только мы выделим какие-либо из контейнеров в списке панели **Layers**, они тотчас же отображаются в окне документа как выделенные. Это очень удобно — сразу видно, какие именно свободные контейнеры мы выделили. И, наоборот, если мы выделим один или несколько контейнеров в окне документа, они сразу же отображаются в панели **Layers** как выделенные.

Мы можем переименовать любой из свободных контейнеров, перечисленных в списке. Делается это уже знакомым нам способом: сначала — выделение соответствующего пункта в списке, далее — щелчок мышью по его имени в колонке **Name**. (Возможно, щелкнуть придется дважды.) Вместо имени свободного контейнера появляется поле ввода, в котором мы вводим новое имя. И не забываем после этого нажать клавишу <Enter> для сохранения введенного имени или клавишу <Esc> для отказа от него.

Теперь обратим внимание на колонку **Z** списка контейнеров. Дело в том, что свободные контейнеры отображаются в списке в том порядке, в котором они перекрывают друг друга. Иначе говоря, они отсортированы по z-индексу. И Dreamweaver предоставляет нам несколько возможностей по изменению данного порядка.

Самый простой способ изменить порядок перекрытия — это перетащить нужный контейнер на новое место в списке. При этом Dreamweaver сам изменит значения z-индексов всех свободных контейнеров, расположенных на странице. Увы, иногда он делает это не очень корректно; в таком случае не обойтись без изменения вручную, о котором мы сейчас и расскажем.

При желании новое значение z-индекса можно задать и вручную. Для этого выделяем нужный контейнер в списке и щелкаем по значению его z-индекса в колонке **Z**. Там появится поле ввода, в котором мы сможем ввести новое значение, после чего нужно нажать клавишу <Enter> для его сохранения или клавишу <Esc> для его отмены. Как правило, значение z-индекса приходится задавать не для одного, а сразу для нескольких контейнеров — его "соседей" в списке.

Также панель **Layers** предлагает нам самый простой способ изменить видимость свободного контейнера. Для этого используется колонка с изображением глаза, самая левая в списке. Просто выделяем нужный контейнер и щелкаем по нему в колонке "глаз". При этом содержимое колонки будет последовательно меняться:

- ◆ изображение закрытого глаза — контейнер невидим;
- ◆ изображение открытого глаза — контейнер видим;
- ◆ пустая колонка — контейнер наследует видимость у родителя.

Но самая интересная возможность, предлагаемая панелью **Layers**, — это "перенос" одних контейнеров в другие, в результате которого перенесенные контейнеры превращаются во вложенные. И выполняется все это мышью.

Итак, выделяем в списке свободный контейнер, который мы собираемся "вложить" в другой. Далее находим в списке контейнер, в который мы хотим "вложить" выбранный и который станет для него родителем. Но не выделяем этот контейнер, а перетаскиваем на него выделенный ранее контейнер...

той клавишу <Ctrl>; при этом контейнер, на который мы перетаскиваем выделенный, будет окружен тонкой синей рамкой. Осталось только "отпустить" перетаскиваемый контейнер и, возможно, исправить его z-индекс.

На рис. 12.8 изображен результат этого действия. Видно, что при наличии вложенных свободно позиционируемых контейнеров в списке создается "дерево", "растущее" из контейнера-родителя. Такие "деревья" могут быть очень "развесистыми", если мы создаем достаточно сложный дизайн.

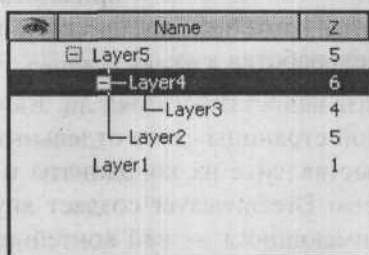


Рис. 12.8. Вложенные свободно позиционируемые контейнеры в списке панели Layers

Мы можем упорядочивать вложенные контейнеры, изменяя их z-индекс, независимо от невложенных контейнеров. Если же мы хотим "вынести" вложенные контейнеры из родителя, превратив их в невложенные, достаточно просто перетащить их в нужную позицию в списке. И, возможно, снова изменить их z-индекс.

Недостатки свободно позиционируемых контейнеров и их преодоление

Вообще, свободно позиционируемые контейнеры — очень хорошая вещь, если мы создаем достаточно сложный дизайн наших Web-страниц. У них нет ограничений фреймов, таблиц и плавающих контейнеров. Они могут находиться где угодно на странице и всегда остаются там, куда мы их поместим.

Но свободно позиционируемые контейнеры — это статичность. Они ни за что не сдвинутся с места, как не изменяй размеры окна Web-обозревателя. Их координаты и размеры задаются один-единственный раз — при создании страницы — и потом уже не меняются.

Что это значит? Это значит, во-первых, что мы не можем заставить свободно позиционируемый контейнер расширяться и сужаться, подстраиваясь под ширину окна Web-обозревателя. (Их плавающего "коллегу" мы, хоть и мудреным образом, но заставили это делать на занятии 11.) Создатели стандартов HTML и CSS просто не подумали, что Web-дизайнером может такое пригодиться. Вот она,

пресловутая пропасть между теорией и практикой, о которой знает каждый студент, когда сдает лабораторные...

Но шутки в сторону — это действительно серьезно! Во-вторых, помните, как на том же *занятии 11* мы поместили плавающий контейнер со сведениями об авторских правах под остальными контейнерами. Причем, сколько эти вышележащие контейнеры не растягивались или сжимались, сведения об авторских правах все равно оставались под ними. Со свободными контейнерами такой номер не проходит — мы не можем приказать какому-то контейнеру: "стой здесь и, если вот этот контейнер растянется в высоту, сдвинься вниз". Все та же досаднейшая недоработка в языке CSS...

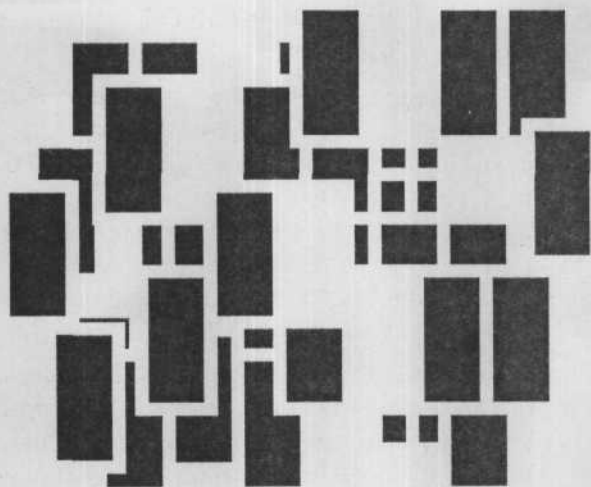
Можно ли преодолеть этот недостаток? Вряд ли. Единственный дельный совет: создавайте для каждой страницы сайта отдельный набор свободных контейнеров и тщательно выставляйте их координаты и размеры. Конечно, для каждой страницы при этом Dreamweaver создаст внутреннюю таблицу стилей, описывающую все имеющиеся на ней контейнеры, но с этим придется мириться.

И вообще, по личному мнению автора, лучше использовать старые добрые фреймы, таблицы и плавающие контейнеры. Свободные же контейнеры (это тоже мнение автора) практически никогда себя не оправдывают. Так что оставим их в покое.

На следующем занятии...

Уф! Вот мы и закончили с Web-дизайном — со всеми его видами. Остались в прошлом суровые фреймы, громоздкие таблицы и гибкие контейнеры. Отдохнем от них...

Впереди у нас — всего два занятия. На них мы изучим всякие полезные мелочи, которые могут пригодиться Web-дизайнеру. Это, во-первых, Web-сценарии, а во-вторых, различные полезные фокусы со стилями CSS. Мелочи, одним словом, но приятные.



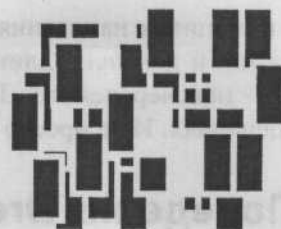
КУРС 5

Последние штрихи

Занятие 13. Использование Web-сценариев

Занятие 14. Маленькие хитрости

ЗАНЯТИЕ 13



Использование Web-сценариев

Итак, с Web-дизайном мы покончили. Изучили все, что нужно для создания современных Web-страниц и Web-сайтов, и попрактиковались в их создании. Многие из вас уже, наверно, продолжают свою практику, набираясь опыта и новых знаний. Что ж, знания и опыт никогда не в тягость — так пусть их будет как можно больше.

А мы заканчиваем изучение основных принципов Web-дизайна и работы с пакетом Macromedia Dreamweaver 8. Осталось совсем чуть-чуть.

Давайте внимательно посмотрим на любую из созданных нами Web-страниц. В чем ее недостаток? Правильно — она статична. Мы не можем сделать так, чтобы при наведении курсора мыши на гиперссылку в строке статуса отображался произвольный текст. Мы не можем скрыть или показать какой-либо элемент страницы в ответ на щелчок по изображению. Страница неизменна — как только Web-обозреватель ее загрузит и выведет на экран, она застывает в неподвижности.

Почему? А потому, что языки HTML и CSS не предусматривают никаких средств, чтобы описать поведения элемента страницы. Они описывают только ее содержимое и внешний вид.

Разумеется, Web-дизайнеров такое положение дел не устраивало. Поэтому появился особый язык программирования *JavaScript*, с помощью которого пишутся так называемые *Web-сценарии*. (Для краткости их называют просто — *сценарии*.) Это небольшие программы, которые помещаются прямо в HTML-код страницы и выполняются Web-обозревателем в ответ на наступление какого-либо *события*: щелчка мышью, помещения курсора мыши на элемент страницы, окончания загрузки страницы и др. А уж с помощью этих сценариев можно много чего натворить!..

Вы спросите: разве нам придется изучать еще один язык — JavaScript? Нет.

о принципах написания Web-сценариев — это тема для отдельного курса занятий, и довольно длинного. Мы обойдемся тем, что используем возможности неисчерпаемого Dreamweaver, который, разумеется, знает о Web-сценариях. И не просто знает, а очень хорошо в них разбирается...

Поведения Dreamweaver и их использование

Разработчики Dreamweaver многое сделали для того, чтобы облегчить жизнь начинающих Web-дизайнеров — мы уже знаем это. Коснулась их забота и начинающих Web-программистов. Так, они предусмотрели в Dreamweaver заранее определенный набор Web-сценариев, которые можно вставить в страницу буквально несколькими движениями мыши. Эти сценарии называются *поведениями* (behaviors). Среди поведений есть такие, которые помещают в строку статуса произвольный текст, скрывают или, наоборот, показывают свободно позиционируемый контейнер, меняют изображение на Web-странице и многое другое.

Поведения Dreamweaver при создании привязываются к какому-либо элементу страницы (изображению, свободному контейнеру, гиперссылке и др.) или же самой странице. Также поведение обязательно должно быть привязано к какому-либо событию, в ответ на которое выполняется соответствующий ему сценарий. Например, если поведение привязано к изображению и событию "окончание загрузки", то оно выполнится сразу после окончания загрузки файла этого изображения. К одному элементу страницы могут быть привязаны несколько поведений, каждое — к своему событию.

Сейчас мы рассмотрим все поведения Dreamweaver, которые будут полезны нам на первых порах. И, разумеется, поговорим о том, когда стоит их применять.

Работа с поведением в панели *Behaviors*

Вся работа с поведением Dreamweaver протекает в специальной панели **Behaviors** (рис. 13.1). Чтобы вызвать ее на экран, нужно включить пункт-выключатель **Behaviors** меню **Window** или нажать комбинацию клавиш <Shift>+<F4>.

Большую часть этой панели занимает список уже созданных поведений, привязанных к выделенному в окне документов элементу страницы. Наименование тега элемента, к которому привязаны отображаемые в списке поведения, находится в заголовке группы панелей. На рис. 13.1 это тег <BODY>, т. е. сама страница

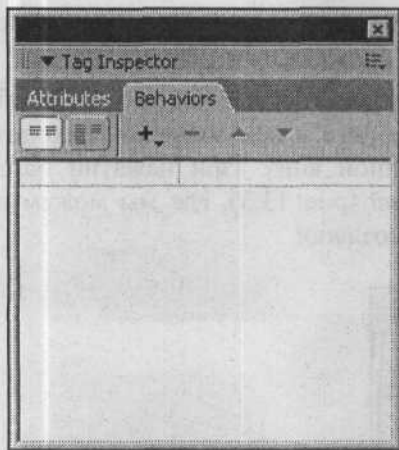


Рис. 13.1. Панель Behaviors

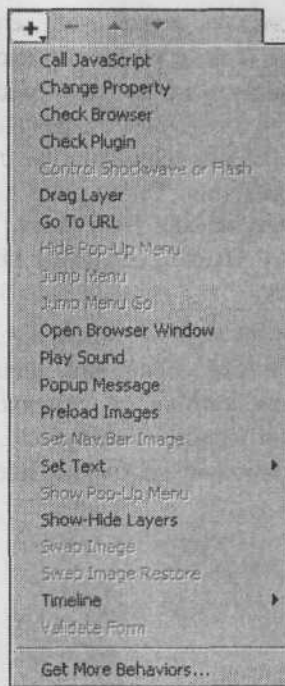


Рис. 13.2. Меню поведений панели Behaviors

Список поведений представляет собой таблицу из двух колонок: **Events** (событие) и **Actions** (поведение, выполняющееся в ответ на это событие). Мы можем выбрать любое поведение в списке и произвести над ним какие-либо действия.

Над списком поведений находятся две кнопки-переключателя. Левая кнопка, включенная по умолчанию, задает режим, при котором в списке показываются только уже созданные поведения. Если же включить правую кнопку, то в списке будут отображены все события, поддерживаемые выделенным элементом страницы, вне зависимости от того, привязаны ли к ним какие-то поведения или нет. При этом у событий, к которым не были привязаны поведения, правая колонка списка будет пуста.

В верхней части панели **Behaviors** находятся четыре кнопки: "плюс", "минус", "стрелка вверх" и "стрелка вниз". Давайте рассмотрим их по очереди.

Кнопка со знаком "плюс" позволит нам создать новое поведение. При ее нажатии на экране появляется довольно большое *меню поведений* (рис. 13.2), в котором можно выбрать нужное поведение. Некоторые из пунктов настоящего меню могут быть недоступными; обычно это означает, что к выделен-

Чтобы создать новое поведение, надо сначала выделить нужный элемент страницы. Также можно привязать поведение к самой странице, для чего следует щелкнуть по пустому месту на ней или выбрать тег `<BODY>` в секции тегов. После этого уже можно выбрать соответствующий пункт в меню поведения. На экране появится диалоговое окно задания параметров выбранного нами поведения. Такое окно для каждого поведения будет свое; мы рассмотрим их все далее в этой главе. Задаем нужные параметры и нажимаем кнопку **ОК**. После этого созданное поведение появится в списке поведений панели **Behaviors**.

Следующий шаг — привязка созданного поведения к событию. В столбце **Events** списка поведений, правее отображаемого в нем значения, находится небольшая кнопка со стрелкой, направленной вниз. При нажатии на эту кнопку на экране появляется *меню событий* (рис. 13.3), где мы можем выбрать нужное нам событие. Все, поведение создано!

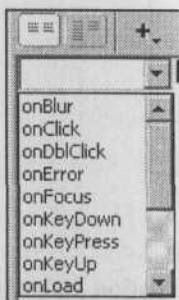


Рис. 13.3. Меню событий панели Behaviors

Dreamweaver "знает" довольно много событий, к которым можно привязать поведение. Многие из этих событий перечислены в табл. 13.1; полный же их список интересующиеся могут найти в справочной системе Dreamweaver.

Таблица 13.1. События, поддерживаемые Dreamweaver

Название	Описание
onAbort	Наступает, когда пользователь прерывает загрузку Web-страницы или изображения, например, нажав кнопку Стоп
onBlur	Наступает, когда окно Web-обозревателя, в котором открыта страница, перестает быть активным
onClick	Наступает при щелчке мыши по элементу страницы или по самой странице
onDbClick	Наступает при двойном щелчке мыши по элементу страницы или по самой странице

Таблица 13.1 (окончание)

Название	Описание
onError	Наступает при возникновении ошибки при загрузке страницы или графического изображения
onFocus	Наступает, когда окно Web-обозревателя, в котором открыта страница, становится активным
onHelp	Наступает, когда пользователь вызывает справочную систему Web-обозревателя
onKeyDown	Наступает при нажатии клавиши клавиатуры
onKeyPress	Наступает при нажатии и отпуске клавиши клавиатуры
onKeyUp	Наступает при отпуске клавиши клавиатуры
onLoad	Наступает после окончания загрузки страницы или графического изображения
onMouseDown	Наступает при нажатии кнопки мыши
onMouseMove	Периодически наступает при перемещении курсора мыши над элементом страницы или над самой страницей
onMouseOut	Наступает при "уходе" курсора мыши с элемента страницы
onMouseOver	Наступает, когда курсор мыши "заходит" на элемент страницы
onMouseUp	Наступает при отпуске кнопки мыши
onMove	Наступает при перемещении окна Web-обозревателя, в котором открыта страница
onResize	Наступает при изменении размеров окна Web-обозревателя, в котором открыта страница
onScroll	Наступает при прокрутке посетителем содержимого страницы или свободного контейнера

Вернемся к набору кнопок, что находится в верхней части панели **Behaviors**. Кнопка со знаком "минус" позволит нам удалить выделенное в списке поведение. С таким же успехом мы можем нажать клавишу .

Если к одному и тому же событию привязано сразу несколько поведений, то они выполняются в порядке следования в списке панели **Behaviors**. С помощью кнопок "стрелка вверх" и "стрелка вниз" мы можем изменить этот порядок; первая кнопка перемещает выделенное в списке поведение на позицию вверх, а вторая — вниз.

Ну, а изменить параметры выбранного в списке поведения проще простого.

экране появится диалоговое окно задания параметров соответствующего поведения. Закончив правку параметров, нажимаем кнопку **ОК** для сохранения сделанных изменений или **Cancel** для отказа от них.

Поведения, поддерживаемые Dreamweaver

Теперь настало время поближе познакомиться с поведением, предлагаемыми нам Dreamweaver, и всеми их параметрами. Название поведения в терминологии самого Dreamweaver приведено в скобках, так что мы сразу сможем найти его в меню поведений (см. рис. 13.2).

Нужно еще раз предупредить, что здесь перечислены не все поведения, поддерживаемые Dreamweaver, а только те, что пригодятся нам как начинающим Web-дизайнерам на первых порах. Полностью все поведения Dreamweaver описаны в его справке.

Перенаправление на разные Web-страницы в зависимости от версии Web-обозревателя (*Check Browser*)

Создатели программ Web-обозревателей всегда стараются делать их в соответствии со стандартами. (В самом деле, что это за Web-обозреватель, если он не понимает, скажем, таблиц или фреймов!) Стараются, но, к сожалению, у них не всегда выходит. И частенько бывает так, что какой-то Web-обозреватель отображает какие-то элементы страниц не так, как его "коллеги". Причем иной раз настолько не так, что тщательно разработанный Web-дизайн страниц превращается бог знает во что. Эта проблема известна как *несовместимость Web-обозревателей*.

Решают эту проблему обычно так. Создаются две версии сайта: одна — для обычных, строго следующих стандартам Web-обозревателей, другая — для "страдающих" несовместимостью. На страницу по умолчанию помещают особое поведение (в общем случае — Web-сценарий), которое проверяет, каким Web-обозревателем пользуется посетитель, и автоматически открывает главную страницу соответствующей версии сайта. Говорят, что такое поведение выполняет *перенаправление* на другую Web-страницу.

Dreamweaver создает такое поведение при выборе пункта **Check Browser** меню поведений. Это поведение следует привязывать к самой странице (т. е. к тегу <BODY>) и событию onLoad (окончание загрузки). В других случаях оно просто не будет работать, как нужно.

Для задания параметров этого поведения служит диалоговое окно **Check Browser**, показанное на рис. 13.4. Давайте посмотрим, что предлагает нам Dreamweaver.

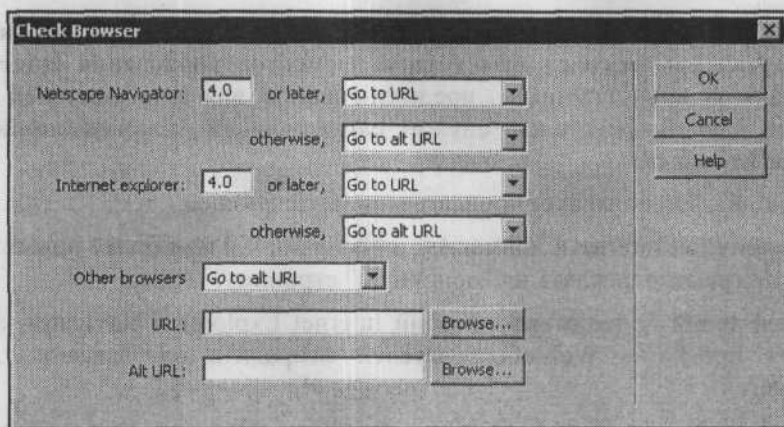


Рис. 13.4. Диалоговое окно Check Browser

А предлагает он нам три возможности:

- ◆ остаться на текущей странице;
- ◆ перейти на первую ("основную" в терминологии Dreamweaver) страницу;
- ◆ перейти на вторую ("альтернативную" в терминологии Dreamweaver) страницу.

Интернет-адрес "основной" страницы задается в поле ввода **URL**. Интернет-адрес "альтернативной" страницы задается в поле ввода **Alt URL**. Также мы можем нажать кнопку **Browse**, находящуюся правее нужного поля ввода, и выбрать необходимый файл в диалоговом окне **Select File**.

Возможно, выбор всего из трех страниц покажется кому-то небогатым. Однако если мы собираемся для каждой версии каждого Web-обозревателя делать свою версию каждой Web-страницы, этого вполне хватит.

Группа элементов управления **Netscape Navigator** позволяет задать, на какие страницы будет произведено перенаправление, если посетитель сайта пользуется Navigator. Сама версия Web-обозревателя задается в небольшом поле ввода, называемом так же — **Netscape Navigator**. Если версия Web-обозревателя окажется такой же, как мы ввели, или более поздней, осуществляется переход на страницу, заданную в раскрывающемся списке **or later**. В противном случае переход будет выполнен на страницу, заданную в раскрывающемся списке **otherwise**.

Раскрывающиеся списки **or later** и **otherwise** имеют по три пункта:

- ◆ **Stay on this Page** — оставаться на текущей странице;
- ◆ **Go to URL** — перейти на "основную" страницу;
- ◆ **Go to Alt URL** — перейти на "альтернативную"

Точно так же задаются страницы для пользователей различных версий Internet Explorer. Для этого служит группа элементов управления **Internet explorer**. Для задания страницы, предназначенной для пользователей других программ Web-обозревателей, служит единственный раскрывающийся список **Other browsers**.

По умолчанию Dreamweaver предлагает нам следующее:

- ◆ пользователей Internet Explorer 4.0 и Navigator 4.0 или более новых версий этих программ переслать на "основную" страницу;
- ◆ пользователей более старых версий Internet Explorer и Navigator, а также других программ Web-обозревателей отправить на "альтернативную" страницу.

Как правило, этих установок бывает достаточно. Нам остается только задать интернет-адреса "основной" и "альтернативной" Web-страниц. И, разумеется, нажать кнопку **OK**.

На заметку

Пользователь Web-обозревателя может в его настройках отключить обработку Web-сценариев, и созданное нами поведение не будет работать. Поэтому нужно предусмотреть запасной вариант — поместить на страницу, содержащую это поведение, набор гиперссылок, указывающий на все созданные нами версии сайта.

Перенаправление на другую Web-страницу (*Go To URL*)

Очень часто бывает так, что сайт "переезжает" с одного Web-сервера на другой. При этом практически всегда меняется его интернет-адрес. И возникает проблема: как известить всех посетителей этого сайта об изменении его адреса.

Часто поступают так. По старому адресу помещают страницу по умолчанию, содержащую особое поведение, которое сразу же перенаправляет посетителя на новый адрес. Таким образом, посетитель, даже если попытается воспользоваться старым адресом, все равно попадет на сайт.

Dreamweaver создает такое поведение после выбора пункта **Go To URL** меню поведений. Привязывать его следует также к самой странице (т. е. к тегу `<BODY>`) и событию `onLoad` (окончание загрузки).

Параметры этого поведения настраиваются в диалоговом окне **Go To URL**, показанном на рис. 13.5.

Сам интернет-адрес задается в поле ввода **URL**. Также можно нажать кнопку **Browse** и выбрать нужную страницу в диалоговом окне **Select File**.

В списке **Open in** задается фрейм текущей страницы, где должна быть загруз-

страница не содержит набора фреймов, в этом списке будет присутствовать единственный пункт **Main window** (все окно), как в нашем случае.

Задав требуемые параметры, нужно нажать кнопку **OK**.

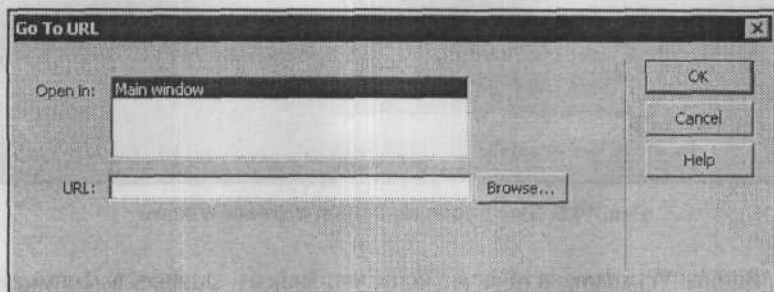


Рис. 13.5. Диалоговое окно **Go To URL**

Кстати, это поведение можно использовать, чтобы одновременно обновить содержимое двух или более фреймов. Для этого нужно привязать к событию `onClick` (щелчок мышью) гиперссылки нужное количество поведений `Go to URL` и задать для них в диалоговом окне **Go To URL** соответствующие параметры: имя фрейма и страницу.

Открытие нового окна Web-обозревателя (*Open Browser Window*)

Это поведение открывает другую Web-страницу, но не в том же, а в другом окне Web-обозревателя. Оно может быть привязано к самой странице, гиперссылке, изображению и "отвечать" на самые разные события: `onLoad`, `onClick` и пр.

Внимание!

Окна Web-обозревателя, открывающиеся при загрузке какой-либо страницы (*всплывающие окна*), — очень плохой тон в Web-дизайне. Они не привлекут к сайту новых посетителей, а скорее, отпугнут. Кроме того, очень многие пользователи Интернета в настоящее время используют особые программы — блокираторы всплывающих окон. Так что лучше от создания этих самых всплывающих окон воздержаться.

Для создания этого поведения нужно выбрать в меню поведений пункт **Open Browser Window**. На экране появится диалоговое окно **Open Browser Window**, показанное на рис. 13.6.

В поле ввода **URL to display** задается интернет-адрес страницы, которая будет загружена в новом окне. Также можно нажать кнопку **Browse** и выбрать

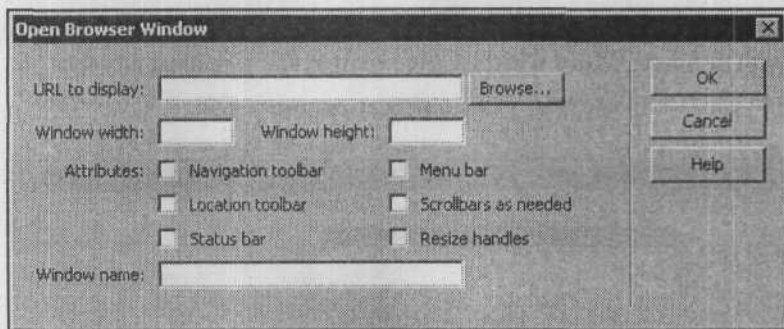


Рис. 13.6. Диалоговое окно **Open Browser Window**

В полях ввода **Window width** и **Window height** задаются соответственно ширина и высота нового окна. Задать координаты окна мы — увы! — не сможем.

С помощью группы флажков **Attributes** задаются дополнительные параметры нового окна:

- ◆ **Navigation toolbar** — включает или отключает наличие у нового окна инструментария навигации с кнопками **Вперед** (Forward), **Назад** (Back), **Остановить** (Stop), **Обновить** (Reload) и пр.;
- ◆ **Location toolbar** — включает или отключает наличие у нового окна инструментария с полем ввода интернет-адреса (строки адреса);
- ◆ **Status bar** — включает или отключает наличие у нового окна строки статуса;
- ◆ **Menu bar** — включает или отключает наличие у нового окна системного меню;
- ◆ **Scrollbars as needed** — разрешает или запрещает появление у нового окна полос прокрутки, если его содержимое в нем не помещается;
- ◆ **Resize handles** — разрешает или запрещает пользователю изменять размеры нового окна.

При этом нужно иметь в виду, что если мы не зададим ни размеров окна, ни одного из дополнительных параметров, то новое окно будет иметь случайные размеры, которые можно будет изменять, оба инструментария, строку статуса, меню, полосы прокрутки. Если же мы зададим размеры или хотя бы один дополнительный параметр, то все остальные параметры будут отключены, и нам придется включать их самим.

Кнопка **OK** диалогового окна сохраняет сделанные нами установки, а кнопка **Cancel** — отменяет их.

Воспроизведение звукового файла (*Play Sound*)

В свете последних тенденций к наполнению Интернета мультимедийной информацией Dreamweaver предоставляет нам возможность создать поведения *Play Sound*. Это поведение в ответ на какое-либо событие (например, окончание загрузки страницы или щелчок по гиперссылке) воспроизводит звуковой файл.

Внимание!

Звуки, воспроизводящиеся после загрузки страницы, — очень плохой тон в Web-дизайне. Мало того, что звуковые файлы, как правило, очень велики и поэтому долго загружаются, так они еще и не несут никакой информационной нагрузки. Так что без них лучше обойтись.

Чтобы создать поведение *Play Sound*, нужно выбрать пункт **Play Sound** меню поведений. После этого на экране появится диалоговое окно **Play Sound** (рис. 13.7).

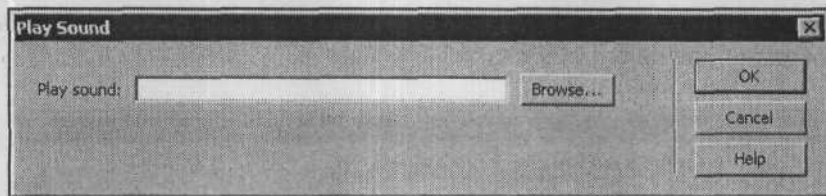


Рис. 13.7. Диалоговое окно *Play Sound*

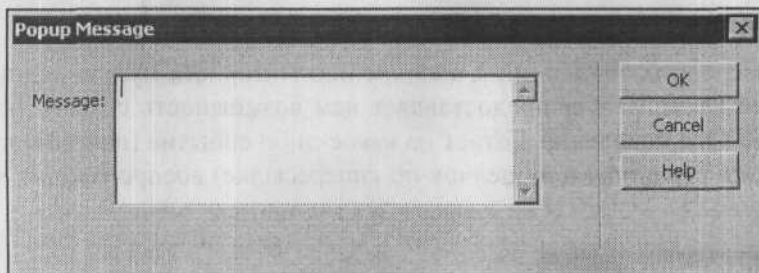
Это диалоговое окно содержит одно-единственное поле ввода **Play sound**, предназначенное для ввода имени звукового файла, который требуется воспроизвести. Также можно нажать кнопку **Browse** и выбрать нужный файл в диалоговом окне **Select File**. После этого следует нажать кнопку **OK**.

Вывод окна-предупреждения (*Popup Message*)

Окно-предупреждение — это небольшое окно с текстом предупреждения, изображением восклицательного знака и кнопкой **OK**. Оно может быть использовано, например, чтобы сообщить о чем-то важном посетителю, зашедшему на сайт. (Хотя, по мнению автора, лучше поместить текст предупреждения прямо на Web-страницу и как-то его выделить...)

Чтобы создать поведение *Popup Message*, выводящее окно-предупреждение, нужно выбрать пункт **Popup Message** меню поведений. На экране после этого появится диалоговое окно **Popup Message** (рис. 13.8).

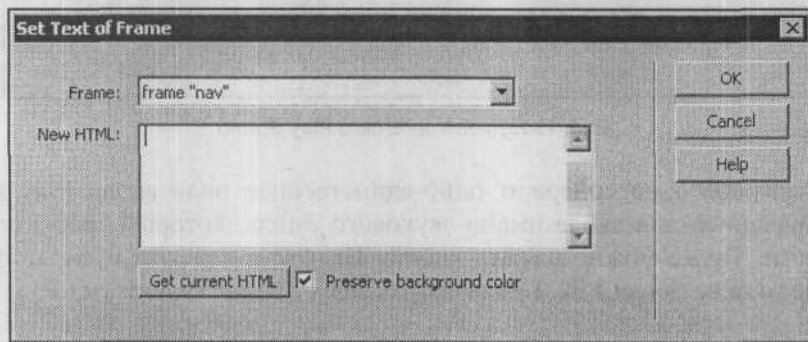
Вводим текст предупреждения в область редактирования **Message** и нажимаем **OK**.

Рис. 13.8. Диалоговое окно **Popup Message**

Задание нового содержимого фрейма (*Set Text of Frame*)

Dreamweaver позволяет задать новое содержимое для какого-либо из присутствующих на странице фреймов. За это "отвечает" поведение *Set Text of Frame*.

Для создания этого поведения выберем пункт **Set Text of Frame** в подменю **Set Text** меню поведений. На экране появится диалоговое окно **Set Text of Frame**, показанное на рис. 13.9.

Рис. 13.9. Диалоговое окно **Set Text of Frame**

В раскрывающемся списке **Frame** выбираем фрейм, в который нужно поместить новое содержимое. Само же содержимое в виде HTML-кода вводим в область редактирования **New HTML**. Мы также можем нажать кнопку **Get current HTML**, чтобы скопировать в эту область редактирования текущее содержимое фрейма, после чего внести в него необходимые изменения. Если включить флажок **Preserve background color**, Web-сценарий, обновляющий содержимое фрейма, сохранит цветовые настройки текста и фона; в противном случае эти настройки пропадут.

Задав нужные настройки, нажимаем кнопку **OK**.

Задание нового содержимого свободно позиционируемого контейнера (*Set Text of Layer*)

Dreamweaver позволяет задать новое содержимое для любого из присутствующих на странице свободно позиционируемых контейнеров. За это "отвечает" поведение *Set Text of Layer*.

Для создания этого поведения следует выбрать пункт **Set Text of Layer** в подменю **Set Text** меню поведений. После этого на экране появляется диалоговое окно **Set Text of Layer**, показанное на рис. 13.10.

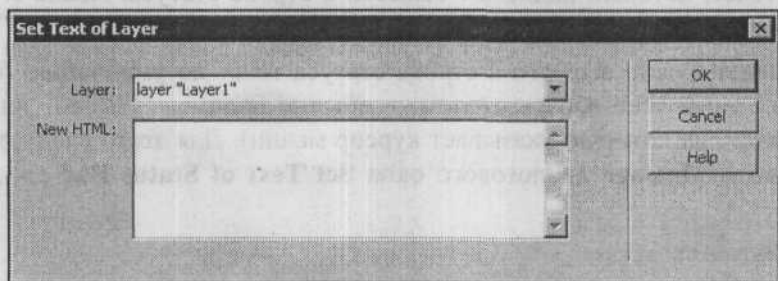


Рис. 13.10. Диалоговое окно *Set Text of Layer*

В раскрывающемся списке **Layer** выбирается свободный контейнер, в который нам нужно поместить новое содержимое. Само содержимое в виде HTML-кода вводится в область редактирования **New HTML**. После задания всех нужных параметров следует нажать кнопку **OK**.

Вывод произвольного текста в строке статуса окна Web-обозревателя (*Set Text of Status Bar*)

Обычно Web-обозреватель выводит в строке статуса своего окна интернет-адрес гиперссылки, на которую указывает курсор мыши. Но мы можем создать поведение *Set Text of Status Bar*, которое выводит туда произвольный текст. Это может быть текст подсказки по той же гиперссылке или краткое описание страницы. Иногда в строке статуса творятся совершенные чудеса: текст "ползет" справа налево либо "вырастает" слева направо, — но это требует более сложного программирования.

Внимание!

Посторонний текст в строке статуса окна Web-обозревателя — очень плохой тон Web-дизайна. Поэтому не стоит этим злоупотреблять.

Для создания этого поведения нужно выбрать пункт **Set Text of Status Bar** в подменю **Set Text** меню поведений. На экране появится диалоговое окно **Set Text of Status Bar**, показанное на рис. 13.11.

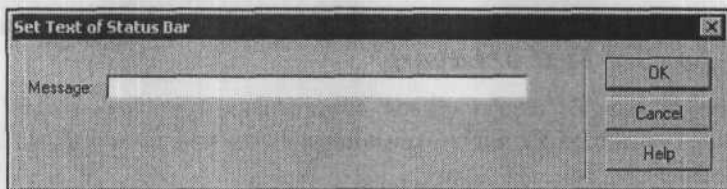


Рис. 13.11. Диалоговое окно **Set Text of Status Bar**

Это диалоговое окно содержит единственное поле ввода **Message**. В него заносится текст, который надлежит вывести в строке статуса. После его ввода нужно нажать кнопку **OK**.

Часто бывает нужно вывести в строке статуса текст по умолчанию (он формируется самим Web-обозревателем — обычно надпись "Готово" или адрес гиперссылки, на которую указывает курсор мыши). Для этого следует ввести в поле ввода **Message** диалогового окна **Set Text of Status Bar** следующий текст:

```
{window.defaultStatus}
```

Этот Web-сценарий на языке JavaScript обозначает содержимое строки статуса, выводимое Web-обозревателем по умолчанию. Имейте в виду: его нужно ввести точно так, как показано выше, без всяких изменений.

Показ и скрытие свободно позиционируемых контейнеров (*Show-Hide Layers*)

На *занятии 12* мы узнали, что свободные контейнеры могут быть видны на экране, а могут быть и скрыты до поры до времени. Видимостью свободных контейнеров можно управлять с помощью поведения *Show-Hide Layers*. Это можно использовать, например, для вывода подсказок по гиперссылкам при наведении на них курсора мыши (событие *onMouseOver*).

Для создания этого поведения следует выбрать в меню поведений пункт **Show-Hide Layers**. На экране появится диалоговое окно **Show-Hide Layers** (рис. 13.12).

Все созданные на странице свободно позиционируемые контейнеры перечислены в списке **Named layers**. Мы можем выделить в этом списке любой из них и задать для него выполняемое действие: показ или скрытие.

Для задания действия служат три кнопки, расположенные под списком:

- ◆ **Show** — делает свободный контейнер видимым (при этом в соответствующей строке списка правее его имени появляется надпись (**show**));
- ◆ **Hide** — скрывает свободный контейнер (налпись (**hide**));

- ◆ **Default** — делает свободный контейнер видимым, если видим его родитель, и наоборот (надпись (**default**)).

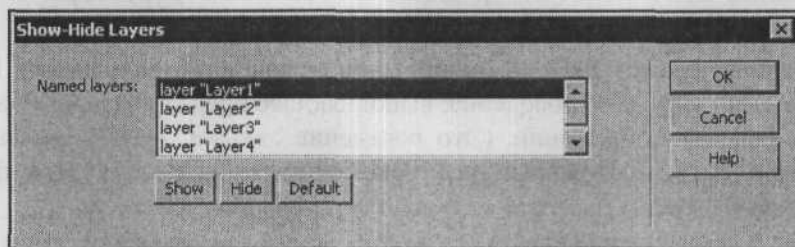


Рис. 13.12. Диалоговое окно Show-Hide Layers

Здесь все просто: выделяем в списке свободный контейнер и нажимаем нужную кнопку. Чтобы отменить ошибочно выбранное действие, нужно нажать ту же кнопку еще раз. По завершению следует нажать кнопку **OK**, чтобы сохранить сделанные установки.

Смена изображения (*Swap Image*)

Особое поведение *Swap Image* занимается тем, что в ответ на событие заменяет изображение, помещенное на страницу, другим. Понятно, что это поведение меняет значение атрибута `SRC` соответствующего тега ``.

Создать это поведение можно выбором в меню поведений пункта **Swap Image**. На экране после этого будет выведено диалоговое окно **Swap Image**, показанное на рис. 13.13.

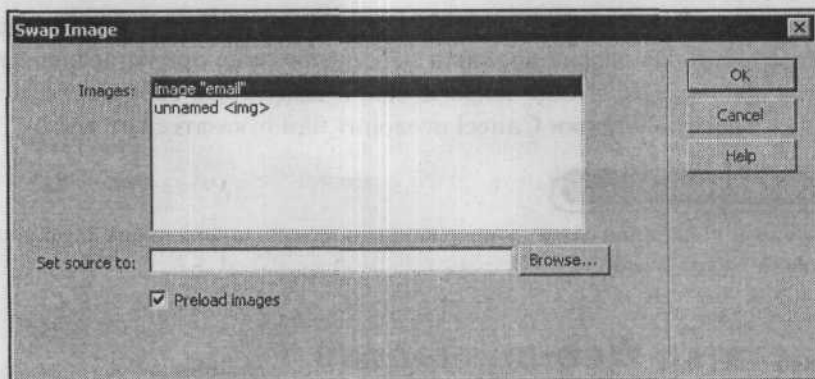


Рис. 13.13. Диалоговое окно Swap Image

Все помещенные на страницу графические изображения перечислены в спи-

Set source to вводим имя файла нового изображения. Также мы можем нажать кнопку **Browse** и выбрать нужный файл в появившемся на экране диалоговом окне **Select File**. Повторяем эти действия для всех изображений, которые нужно изменить, если их несколько.

Если включен флажок **Preload images** (а он включен по умолчанию), Dreamweaver создаст еще одно поведение, выполняющее предварительную загрузку нужных файлов изображений. (Это поведение **Preload Images**, описанное в справочной системе Dreamweaver.) Рекомендуется оставлять этот флажок включенным.

Если мы привяжем поведение **Swap Image** к событию **onMouseOver** (наведение курсора мыши на элемент страницы), то Dreamweaver автоматически создаст поведение **Swap Image Restore**, восстанавливающее все изначальные изображения, и привяжет его к событию **onMouseOut** (увод курсора мыши с элемента страницы). То есть при наведении курсора мыши изначальное изображение будет заменено другим, а при уводе — восстановлено. Если нам это не нужно, мы должны будем отключить флажок **Restore images on onMouseOut**.

Закончив работу, нужно нажать кнопку **OK**.

Восстановление всех изначальных изображений (*Swap Image Restore*)

Как уже говорилось ранее, поведение **Swap Image Restore** восстанавливает все изначальные изображения, измененные поведением **Swap Image**. Обычно поведение **Swap Image Restore** создается самим Dreamweaver, но иногда его требуется создать вручную.

Для создания этого поведения нужно выбрать пункт **Swap Image Restore** в меню поведений. На экране появится небольшое окно-предупреждение; нужно нажать кнопку **OK**, чтобы закрыть его, и выполнить собственно создание поведения. Нажатие кнопки **Cancel** позволит нам отказаться от этого.

Внимание!

Создать поведение **Swap Image Restore** можно только после создания поведения **Swap Image**.

Недостатки Web-сценариев и их преодоление

И закончим разговор о Web-сценариях мы традиционным разбором недостатков. А недостатков у сценариев всего два.

О первом их недостатке мы уже как-то говорили. Дело в том, что пользователь Web-обозревателя может отключить обработку сценариев — каждая программа Web-обозревателя предоставляет такую возможность. Это может быть сделано, например, из соображений безопасности — сейчас существует вполне реальная возможность получить с помощью Web-сценария какой-нибудь вирус или иную вредоносную программу. Поэтому многие пользователи Интернета относятся к сценариям настороженно.

Преодолеть этот недостаток можно двумя путями. К сожалению, если пользователь отключил в Web-обозревателе обработку сценариев, мы никаким образом не сможем включить ее вновь, ни с помощью HTML, ни с помощью CSS, ни с помощью JavaScript. Поэтому оба предлагаемых пути есть, по сути, компромисс.

1. Не следует делать Web-сценарии необходимой частью Web-страниц. Очень часто сайты имеют систему навигации, основанную на сценариях, например, всплывающее меню с гиперссылками. Так вот, делать так не стоит, ведь если Web-обозреватель не сможет выполнить эти сценарии, посетитель не уйдет дальше главной страницы такого сайта и, в конечном счете, покинет сайт и больше на него не вернется.
2. Если уж без сценариев не обойтись, нужно на главной странице сайта поместить какое-либо предупреждение, гласящее, что обработка сценариев в Web-обозревателе должна быть включена. Но, опять же, некоторые посетители могут этого не сделать из опасения по поводу безопасности.

Вывод напрашивается очень простой. Web-сценарии — вещь, конечно, хорошая, но если без них можно обойтись, то лучше так и сделать. В самом деле, так ли уж нужны, скажем, всплывающие окна (особенно если учесть, насколько они раздражают)?

Второй недостаток сценариев в настоящее время не очень актуален. На обработку их расходуются ресурсы компьютера клиента, и зачастую немалые. Хотя современные компьютеры достаточно мощны и могут справиться практически с любыми сценариями, за исключением, может быть, совсем уж сложных. Но часто ли вы будете писать сложные сценарии?..

На заметку

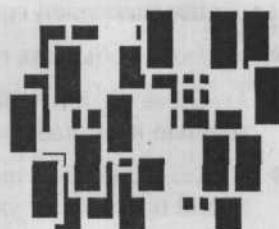
Надо сказать, что сейчас появляются сайты, чей дизайн построен на большом количестве очень сложных Web-сценариев. В первую очередь, это, конечно, почтовая служба поискового портала Google — Google.Mail (<http://mail.google.com>). Такую комбинацию сложных сценариев и специальных программ, работающих совместно с Web-сервером, называют технологией Ajax.

На следующем занятии...

Вот мы и закончили с Web-сценариями. Точнее, с поведением Dreamweaver — чтобы научиться писать сценарии на языке JavaScript, нужна еще одна книга, и немаленькая. Поведения же — самый простой путь добавить нашим страницам "жизни", не залезая в книги.

Последнее занятие будет посвящено исключительно полезным советам, маленьким хитростям, которые будут полезны любому Web-дизайнеру: и начинающему, и опытному. В основном, они касаются стилей CSS, чьи возможности, как иной раз кажется, просто безграничны.

ЗАНЯТИЕ 14



Маленькие хитрости

Последнее занятие будет чем-то очень похожим на рубрику журнала "Наука и жизнь", известную под названием "Маленькие хитрости". Это просто сборник полезных советов, которые весьма пригодятся любому Web-дизайнеру.

Подбор цветов для сайта

Какой набор цветов предпочесть для сайта? Вопрос...

Правила хорошего тона предписывают выбирать для оформления сайта как можно меньше цветов. Иначе от сайта у посетителя будет рябить в глазах, он не сможет воспринять содержимого, ради которого мы создали сайт, и, в конце концов, уйдет подобра-поздорову. А нам это совершенно не нужно.

Автор этой книги обычно пользуется для подбора цветов отличным электронным справочником Владимира Голованова и Андрея Якушева "Набор палитр для Web-дизайна". Найти его можно по адресам <http://www.tarusa.ru/~golovan> и <http://yakushev.da.ru>. Также можно просто позаимствовать цветное оформление понравившегося сайта, благо сам дизайн никакими авторскими правами не охраняется (в отличие от отдельных его элементов).

Оформление гиперссылок

Автору часто задают вопрос: "Можно ли как-то изменить вид гиперссылок, например, окрасить их не в синий, а в зеленый цвет?" Разумеется, можно!

Давайте вернемся к занятию 7. Помните, как мы рассматривали там псевдостили гиперссылок? Вот именно с их помощью мы и дадим нашим гипер-

Всего псевдостилей гиперссылок четыре:

- ◆ `A:link` — обычная гиперссылка;
- ◆ `A:active` — активная гиперссылка (та, на которой посетитель щелкает мышью в данный момент);
- ◆ `A:visited` — посещенная гиперссылка (указывающая на страницу, на которой посетитель уже побывал);
- ◆ `A:hover` — гиперссылка, на которую в данный момент указывает курсор мыши.

Этих четырех псевдостилей нам хватит с лихвой. Если же нам будет нужно выделить какие-то гиперссылки, мы можем применить к ним гибридный стиль, написанный таким образом:

```
A.nav:link
```

То есть обозначение псевдостилия (`:link`, `:active` и т. д.) следует за именем гибридного стиля. Приведенный выше стиль будет применен ко всем обычным гиперссылкам, атрибут `CLASS` тега `<A>` которых будет иметь значение `nav`.

При оформлении гиперссылок обычно придерживаются трех правил, которые перечислены ниже.

- ◆ Обычные гиперссылки по оформлению должны как-то отличаться от обычного текста. Как правило, их выделяют цветом или подчеркиванием — это, можно сказать, классика. (Вспомним — практически все Web-обозреватели выделяют гиперссылки синим цветом и подчеркиванием.) Гиперссылки, входящие в набор для перехода на другие страницы сайта, можно выделить немного по-другому, но именно немного.
- ◆ Активная гиперссылка и гиперссылка, на которую указывает курсор мыши, должны быть выделены более ярко, нежели обычные. Как правило, для этого используют красный цвет и, возможно, подчеркивание.
- ◆ Посещенные гиперссылки выделяют более тускло, чем обычные, и уж тем более активные, и те, на которые указывает курсор мыши. Посетитель сайта сразу должен видеть, что на страницах, на которые указывают эти гиперссылки, он уже побывал. Web-обозреватели обычно выделяют их темно-красным цветом.

В принципе, ничто не мешает нам задать для гиперссылок, скажем, другой фон или рамку. Нужно только подбирать такое оформление гиперссылок, чтобы оно не резало глаза и не выбивалось из общего оформления сайта.

Оформление таблиц

Можно ли создать рамку у самой таблицы, но не у ее ячеек? Конечно! Стили CSS позволяют это сделать.

Чтобы задать видимую рамку у таблицы, нужно создать стиль переопределения тега `<TABLE>` и задать для него в категории **Border** диалогового окна **CSS Rule Definition** (см. рис. 7.11) такие параметры:

- ◆ **Style: Same for all** — включен;
- ◆ **Style: Top** — **solid** (сплошная линия);
- ◆ **Width: Same for all** — включен;
- ◆ **Width: Top** — **thin** (тонкая линия);
- ◆ **Color: Same for all** — включен;
- ◆ **Color: Top** — черный цвет.

Это, как мы помним из *занятия 7*, параметры рамки. Далее переключаемся в режим отображения HTML-кода и у всех таблиц (в смысле, тегов `<TABLE>`) убираем атрибут `BORDER`, задающий толщину рамки. Это нужно, чтобы внести однозначность в параметры таблицы — пусть на нее действует только стиль CSS, а атрибут HTML пусть не мешает.

После этого можно переключиться в режим отображения страницы и посмотреть на результат. Но для чистоты эксперимента лучше просмотреть ее в Web-обозревателе (рис. 14.1).

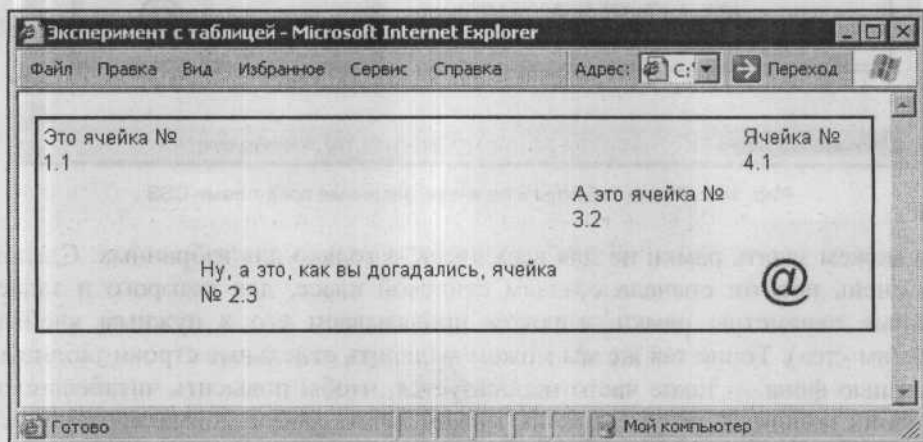


Рис. 14.1. Таблица с рамкой, заданной средствами CSS

Таким же образом можно создать рамки для ячеек этой таблицы. Давайте сделаем их серыми, для чего создадим стиль переопределения тега `<TD>` с такими параметрами:

- ◆ **Style: Same for all** — включен;
- ◆ **Style: Top** — **solid** (сплошная линия);

- ◆ **Width: Same for all** — включен;
- ◆ **Width: Top — thin** (тонкая линия);
- ◆ **Color: Same for all** — включен;
- ◆ **Color: Top** — серый цвет.

В общем, то же самое, что и у предыдущего стиля, переопределявшего тег `<TABLE>`, только цвет у рамки серый.

В теге `<TD>` за цвет рамки "отвечает" атрибут `BORDERCOLOR`. Переключаемся в режим отображения HTML-кода и удаляем этот атрибут из всех тегов `<TD>`. В результате мы получим то, что показано на рис. 14.2 (опять же, в Web-обозревателе).

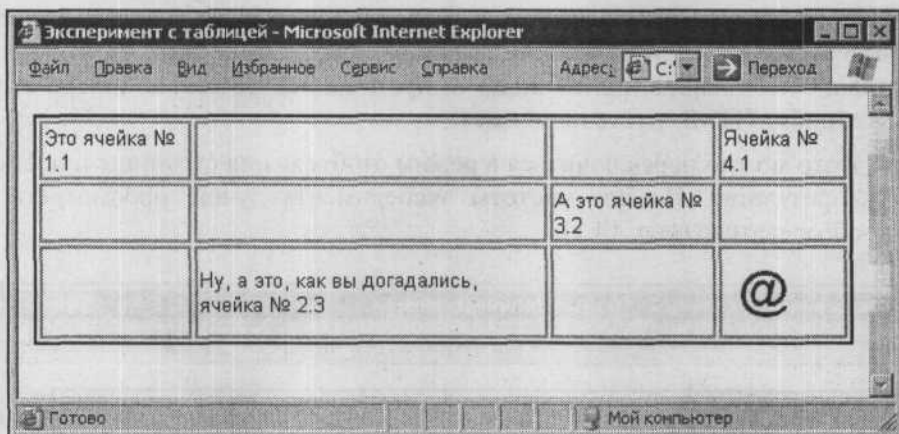


Рис. 14.2. Рамки таблицы и ее ячеек, заданные средствами CSS

Мы можем задать рамки не для всех ячеек, а только для избранных. Сделать это очень просто: сначала создаем стилевой класс, для которого и задаем нужные параметры рамки, а потом привязываем его к нужным ячейкам (к тегам `<TD>`). Точно так же мы можем выделить отдельные строки таблицы с помощью фона — такое часто используется, чтобы повисить читабельность больших таблиц. И, разумеется, мы можем задать любое форматирование для строки заголовка таблицы — для этого достаточно только создать стиль переопределения тега `<TH>`.

Оформление полос прокрутки

Все версии Microsoft Internet Explorer, начиная с 5.5, имеют одну очень интересную особенность: они могут менять оформление полос прокрутки своего окна, а также свободно позиционируемых контейнеров, для которых пере-

метр **Overflow** имеет значение **scroll** или **auto** (см. рис. 12.5). (О параметрах свободных контейнеров рассказывалось на занятии 12.) Делается это с помощью особых атрибутов CSS, которые мы сейчас рассмотрим.

К сожалению, Dreamweaver в этом случае нам не помощник. Нам придется внести изменения в таблицу стилей вручную, переключившись в режим отображения HTML-кода. И просмотреть полученный результат мы сможем только в Web-обозревателе.

Итак, атрибуты CSS, "ответственные" за задание внешнего вида полос прокрутки... Их довольно много, и все они перечислены в табл. 14.1.

Таблица 14.1. Атрибуты CSS, позволяющие изменять внешний вид полос прокрутки

Атрибут CSS	Описание
<code>scrollbar-3d-light-color</code>	Цвет "светлого" контура трехмерности движка и кнопок прокрутки
<code>scrollbar-arrow-color</code>	Цвет стрелок на кнопках прокрутки
<code>scrollbar-base-color</code>	Общий фон
<code>scrollbar-dark-shadow-color</code>	Цвет "темного" контура трехмерности движка и кнопок прокрутки
<code>scrollbar-face-color</code>	Цвет поверхности движка и кнопок прокрутки, а также пикселей "дорожки", по которой перемещается движок
<code>scrollbar-highlight-color</code>	Цвет "подсвеченной" стороны движка и кнопок прокрутки, а также альтернативных пикселей "дорожки", по которой перемещается движок
<code>scrollbar-shadow-color</code>	Цвет "затененной" стороны движка и кнопок прокрутки
<code>scrollbar-track-color</code>	Цвет всех пикселей "дорожки", по которой перемещается движок прокрутки. Имеет приоритет перед атрибутами <code>scrollbar-face-color</code> и <code>scrollbar-highlight-color</code>

Видно, что все эти атрибуты затрагивают цвета какого-либо элемента полосы прокрутки. Схематично это показано на рис. 14.3.

Давайте попробуем изменить полосы прокрутки Web-обозревателя. Создадим пустую Web-страницу и введем в ней какой-либо текст. После этого переключимся в режим отображения HTML-кода, найдем секцию заголовка страницы (тег `<HEAD>`) и поместим в нее внутреннюю таблицу стилей, код которой

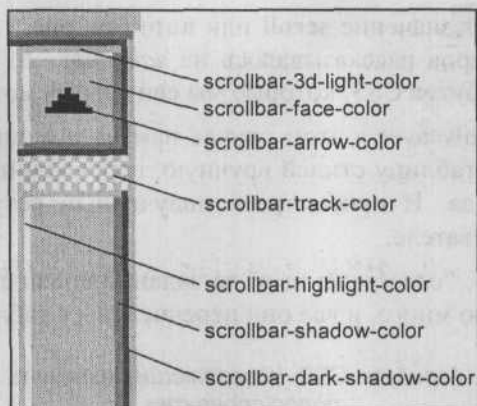


Рис. 14.3. Полоса прокрутки и атрибуты CSS, затрагивающие ее отдельные части

```
<STYLE>
BODY { scrollbar-3d-light-color: #000000;
      scrollbar-arrow-color: #000000;
      scrollbar-base-color: #000000;
      scrollbar-dark-shadow-color: #000000;
      scrollbar-face-color: #FFFFFF;
      scrollbar-highlight-color: #FFFFFF;
      scrollbar-shadow-color: #FFFFFF;
      scrollbar-track-color: #CCCCCC; }
</STYLE>
```

После этого сохраним страницу и откроем ее в Web-обозревателе. То, что мы увидим, показано на рис. 14.4.

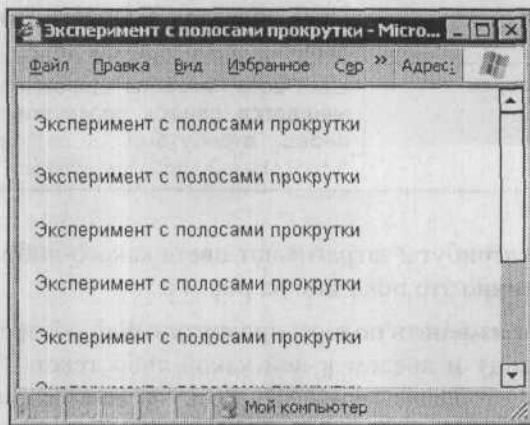


Рис. 14.4. Окно Web-обозревателя с измененными полосами прокрутки

Внимание!

Все описанные в табл. 14.1 атрибуты не входят в стандарт языка CSS и поддерживаются только Internet Explorer, начиная с версии 5.5. Но даже их поддержка более поздними версиями Internet Explorer оставляет желать лучшего. Так, Internet Explorer 6.0 не всегда меняет вид своих полос прокрутки, а если и меняет, то зачастую непредсказуемо.

Добавление иконки к пункту *Избранное*

Еще одна забавная возможность Microsoft Internet Explorer — это задание иконки для пункта меню **Избранное**. Меню **Избранное** Internet Explorer содержит добавляемые пользователем пункты — гиперссылки на понравившиеся ему сайты, снабженные именами.

Самый простой способ привязать к сайту иконку, которая будет потом помещена в пункт **Избранное** — это поместить ее в корневую папку в виде файла favicon.ico. Internet Explorer сам запросит этот файл, если пользователь вздумает сохранить сайт в **Избранное**.

Другой способ посложнее. Зато файл иконки может находиться где угодно, в смысле, в какой угодно папке, и иметь любое имя. Только в секции заголовка Web-страницы нужно поместить такой тег:

```
<LINK REL="SHORTCUT ICON" HREF="<имя файла иконки>">
```

Этот тег указывает Internet Explorer, в каком файле находится иконка, которую нужно будет поместить в **Избранное**.

Создание врезки к статье

Очень часто в обычных "бумажных" журналах применяются *врезки* — небольшие блоки текста, помещенные слева или справа от какого-либо абзаца и поясняющие или дополняющие его. Нет никаких причин отказываться от врезок и при создании документов электронных, т. е. Web-страниц.

Чтобы создать врезку, нужно перед абзацем, к которому эта врезка относится, поместить блочный контейнер, создаваемый парным тегом <DIV>. (О блочных контейнерах мы узнали на занятиях 7 и 11.) В этот блочный контейнер затем помещается содержимое врезки.

Следующий шаг — создание стилевого класса для готового контейнера. Ему нужно дать всего два параметра — они обязательны:

- ◆ **Float** — **right**, если врезка должна находиться справа от абзаца, и **left**, если слева;
- ◆ **Width** — значение ширины врезки в пикселях или процентах.

Оба этих параметра доступны в категории **Box** диалогового окна **CSS Rule Definition** (см. рис. 7.9).

Также можно задать для врезки фон, отступы и видимую границу, чтобы выделить ее относительно остального текста. Очень часто так и делают — врезка должна быть заметна.

Напоследок остается привязать только что созданный стилевой класс к контейнеру с содержимым врезки — и дело сделано.

Создание полосы навигации

На *занятии 5*, посвященном работе с изображениями, мы рассматривали так называемые активные изображения. Это изображения, которые при наведении курсора мыши меняются, а при щелчке по ним — выполняют переход на другую Web-страницу. Весьма полезная вещь для создания набора гиперссылок, не находите?

Так вот, Dreamweaver имеет встроенные средства для создания такого набора гиперссылок, представляющего собой множество активных изображений, указывающих на различные страницы сайта. Такой набор называется *полосой навигации*.

Чтобы создать в Dreamweaver полосу навигации, нам понадобится набор графических изображений в формате GIF, и немаленький. В общем, на каждое активное изображение, входящее в полосу навигации, нам понадобятся целых четыре изображения:

- ◆ отображаемое в обычном состоянии;
- ◆ отображаемое, когда пользователь помещает над ним курсор мыши (перекрывающее изображение);
- ◆ отображаемое в "нажатом" состоянии (когда страница, на которую указывает эта гиперссылка, загружена);
- ◆ отображаемое в "нажатом" состоянии, когда пользователь помещает над ним курсор мыши.

Эти изображения можно создать в любом графическом редакторе, поддерживающем формат GIF: в том же Paint, поставляемом в составе Windows, Adobe Photoshop, Macromedia Fireworks. Все эти изображения должны быть одинаковых размеров.

Затем можно приступить к созданию полосы навигации. Для этого сначала следует поставить текстовый курсор туда, где она должна находиться, например, в соответствующий фрейм или ячейку таблицы разметки. После этого в меню кнопки **Images** (см. рис. 5.2) вкладки **Common** инструментария

объектов выбираем пункт **Navigation Bar**. Также можно выбрать пункт **Navigation Bar** подменю **Image Objects** меню **Insert**. На экране появится диалоговое окно **Insert Navigation Bar**, показанное на рис. 14.5.

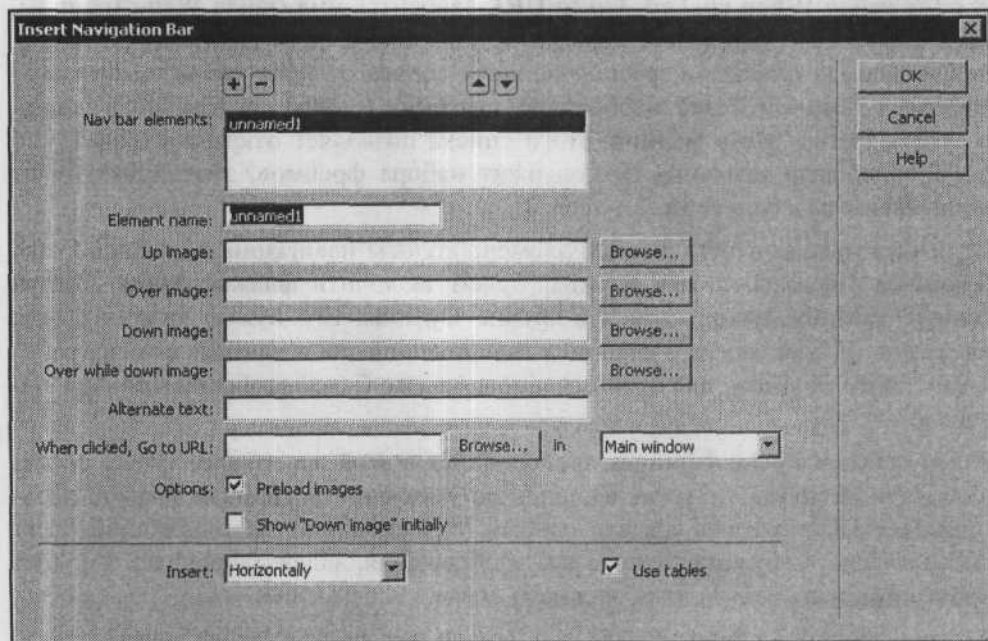


Рис. 14.5. Диалоговое окно **Insert Navigation Bar**

В списке **Nav bar elements** перечислены все элементы полосы навигации, которые мы уже создали. (Изначально там находится один элемент, созданный для нас Dreamweaver.) Мы можем выбрать любой из этих элементов и посмотреть либо изменить его параметры.

В поле ввода **Element name** вводится имя элемента. Каждый элемент полосы навигации должен иметь уникальное в пределах этой полосы имя. Лучше всего давать элементам "говорящие" имена, например `Humour` или `About`.

В поле ввода **Up image** заносится имя файла изображения, отображаемого в обычном случае. Также можно щелкнуть кнопку **Browse**, расположенную справа от этого поля ввода, и выбрать нужный файл в диалоговом окне **Select Image Source**, похожем на уже знакомое нам окно **Select File**.

Точно так же в поле ввода **Over image** заносится имя файла изображения, отображаемого, когда пользователь помещает над гиперссылкой курсор мыши, в поле **Down image** — имя файла изображения "нажатой" гиперссылки, а в поле **Over while down image** — имя файла изображения "нажатой" гипер-

ссылки, над которой пользователь поместил курсор мыши. Справа от каждого поля ввода находится спасительная кнопка **Browse**.

В поле ввода **Alternate text** заносится текст замены. Рекомендуется его ввести.

В поле ввода **When clicked, Go to URL** заносится имя файла Web-страницы, на которую осуществляется переход при щелчке на данном элементе. В раскрывающемся списке **in**, расположенном справа от поля ввода, выбирается фрейм, в котором будет отображена страница (одним словом, цель гиперссылки). Пункт **Main window** этого списка позволяет открыть страницу во всем окне; если страница не содержит набора фреймов, этот пункт будет единственным в списке **in**.

Если нам нужно, чтобы данный элемент полосы навигации изначально отображался "нажатым", мы должны будем включить флажок **Show "Down image" initially**, находящийся в группе **Options**. Его нужно включить для элемента, обозначающего страницу, над которой мы в данный момент работаем; таким образом, для этой страницы он уже будет отображаться как "нажатый".

А вот флажок **Preload images**, находящийся в этой же группе, лучше всегда включать (впрочем, он и так включен по умолчанию). Данный флажок предписывает Dreamweaver создать особый Web-сценарий, заставляющий Web-обозреватель загружать заранее все изображения, задействованные в полосе навигации, и сохранять их на жестком диске в своем кэше.

Раскрывающийся список **Insert** позволит задать расположение полосы навигации: горизонтальное (пункт **Horizontally**) или вертикальное (пункт **Vertically**). Флажок **Use tables** заставит Dreamweaver создать полосу навигации на основе таблиц HTML. Этот флажок по умолчанию включен, и отключать его не стоит.

Осталось рассказать о кнопках, расположенных над списком **Nav bar elements**. Кнопка "плюс" добавляет новый элемент в полосу навигации, а кнопка "минус" удаляет выбранный в списке элемент. Кнопка "стрелка вверх" перемещает выбранный в списке элемент на строку выше, а кнопка "стрелка вниз" — на строку ниже.

После ввода всех данных нужно нажать кнопку **OK**. И Dreamweaver создаст для нас замечательную полосу навигации.

Dreamweaver вообще замечательная программа...

Что дальше?

А дальше — все! Последнее занятие закончилась. Автор только скажет вам несколько слов напоследок.

Заключение

Вот и закончилась книга по Web-дизайну в среде Dreamweaver. Четырнадцать занятий, в течение которых мы учились Web-дизайну.

Мы узнали многое. Познакомились с современными интернет-технологиями и основными принципами Web-дизайна. Разобрались с основными возможностями замечательного пакета Macromedia Dreamweaver 8 и научились их использовать. Выяснили, что такое Web-дизайн текстовый, фреймовый, табличный и контейнерный, в чем сильные и слабые стороны этих четырех видов Web-дизайна, и когда стоит их применять. Мы создали целых четыре Web-сайта и изрядное количество Web-страниц, чтобы попробовать на деле свои знания. И наконец, мы продрались сквозь дебри языков HTML и CSS и умудрились запомнить множество новых терминов.

Но вот завершилась последнее занятие, и автор книги прощается с вами. Больше он не будет пичкать вас знаниями, разбавляя их своим незатейливым юморком. Считайте, что вы окончили курс обучения и получили диплом.

Конечно, вам не рассказали обо всех возможностях Dreamweaver. На самом деле, их так много, что им посвящают толстенные книги, в которых, опять же, многое остается неопианным. Возьмите ту же самую Microsoft Windows — сколько чудес она преподносит даже опытным пользователям! (Только не об ошибках! В других операционных системах, той же перехваленной Linux, ошибок не меньше, если не больше.) Дома у автора лежит толстенная тысячестраничная книга по Windows 2000, но, даже прочитав ее от корки до корки, он бы не сказал, что разобрался в ней досконально. И так обстоит дело с любым объемным программным продуктом.

Dreamweaver — мощный и сложный программный пакет. Все его возможности описать в одной, да еще такой тонкой книге просто нельзя.

- ◆ Автор не описал возможности серверного программирования, предлагаемые Dreamweaver. Серверное программирование — создание программ, работающих совместно с Web-сервером — это целый мир, который не опишешь в одной, даже толстой книге.
- ◆ Автор не описал мультимедийные возможности Dreamweaver, а именно, размещение на Web-страницах фильмов Macromedia Shockwave/Flash, аудио- и видеофайлов. А эти возможности весьма богаты.
- ◆ Автор не описал процесс создания на страницах анимированных элементов.
- ◆ Автор не описал специальные средства Dreamweaver: механизм поиска и замены, активы, библиотеки и пр.
- ◆ Автор совершенно не уделил внимания языку написания Web-сценариев JavaScript и принципам написания этих самых сценариев.
- ◆ Автор даже толком не рассказал о языках HTML и CSS, а ограничился только начальными сведениями о них.

Но, к счастью, все это описано в справке, входящей в комплект Dreamweaver. Правда, она полностью английская, но опытный компьютерщик обязан знать этот язык.

Кроме того, существует Интернет, всемирная сеть, в которой можно найти все что угодно. В табл. 3.1 приведен список интернет-ресурсов, посвященных Web-дизайну, HTML, CSS, другим интернет-технологиям и, конечно, самому Dreamweaver.

Таблица 3.1. Список интернет-ресурсов по теме книги

Интернет-адрес ресурса	Описание
Зарубежные ресурсы	
http://www.macromedia.com/	Официальный сайт фирмы Macromedia, разработчика Dreamweaver. Посвящен как самому Dreamweaver, так и другим продуктам этой фирмы (Fireworks, Flash, Shockwave и др.). Здесь можно найти документацию, расширения, примеры, советы, обновления и пр.
http://www.w3c.org/	Сайт комитета W3C, занимающегося интернет-стандартами. Здесь можно найти описания всех современных интернет-технологий, так сказать, из первых рук
http://webreview.com/	Огромное количество документации по HTML, CSS, JavaScript и Web-обозревателям

Таблица 3.1 (окончание)

Интернет-адрес ресурса	Описание
Отечественные ресурсы	
http://www.webmascon.com/	Электронный интернет-журнал, посвященный интернет-технологиям. Все статьи на русском языке
http://www.citforum.ru/	"Город" электронной документации по компьютерным и интернет-технологиям. Все по-русски. В числе прочего, есть и переводы стандартов с сайта комитета W3C
http://html.krsk.ru/	Этот сайт называется "Рецепты HTML". Здесь можно найти различные описания, статьи и полезные советы
http://subscribe.ru/	Сервер почтовых рассылок. Поищите рассылки, связанные с интернет-технологиями, — их там очень много. Есть и рассылка, посвященная Dreamweaver

Ну, вот и все. Книга закончилась, и автор прощается с вами. До новых встреч!

Владимир Дронов

Электронная почта: vlad@vgi.volsu.ru

Web-сайт: <http://vgi.volsu.ru:8000/~Vladimir.Dronov>

Предметный указатель

A

- Ajax 293
- ARPANET 16
- Assets 220
- Attach External Style Sheet 154

B

- Behaviors 278—280
- Brightness/Contrast 126

C

- Check Browser 282
- Choose Home Page 91
- Choose Spacer Image 211
- CSS 150, 158, 266
- CSS Rule Definition 153, 161—173, 242, 243
- CSS Styles 151

D

- Dependent Files 99
- DNS 24

F

- Files 91, 94
- Frame Tag Accessibility Attributes 185

- FTP 17, 21, 86
- ◊ пассивный 89

G

- GIF 108
- ◊ анимированный 109
- Go To URL 284
- Grid Settings 200
- Guides 203

H

- HTML 25, 43, 68, 266
- HTML-редактор 34
- HTTP 21

I

- Image Tag Accessibility Attributes 111
- Insert Div Tag 240
- Insert Navigation Bar 303
- Insert Other Character 81
- Insert Rollover Image 119
- Insert Rows or Columns 135
- IP 20
- IP-адрес 23

J

- JavaScript 277

L

Layers 271
Link Checker 97
List Properties 60

M

Manage Sites 84

N

Named Anchor 76
New CSS Rule 152, 160, 174
New Document 49, 226
New Editable Region 224
New Optional Region 230
New Repeating Region 233

O

Open Browser Window 285

P

Page Properties 177
Play Sound 287
PNG 109
POP3 21
Popup Message 287
Preferences 43, 46, 90, 212
Properties 55

R

Reference 73, 160
Reports 95, 111
RGB 64

S

Save As Template 221
Save Spacer Image File As 212
Select File 111
Select Image Source 109
Server Compatibility 89

Set Text of Frame 288
Set Text of Layer 289
Set Text of Status Bar 289
SFTP 90
Sharpen 126
Shockwave/Flash 109
Show-Hide Layers 290
Site Definition 84, 86, 87, 91, 98
Site Reports 96
SMTP 21
Swap Image 291
Synchronize 101, 102
Synchronize Files 100

T

Table 130
TCP 20
Template Properties 232

U

Update Files 93, 229, 252
Update Template Files 228

W

W3C 306
Web 17
Web-обозреватель 25—30
◇ Firefox 29
◇ Internet Explorer 29
◇ Mozilla 29
◇ Navigator 29
◇ Opera 29
◇ Safari 30
◇ история 53
◇ несовместимость 282
Web-редактор 33, 34
◇ визуальный 34
◇ гибридный 34
◇ невизуальный 34
Web-сайт 25
◇ выбор 91
◇ регистрация 84

- Web-сервер 25—31
 - ◊ Apache 30
 - ◊ Internet Information Server 30
 - ◊ Netscape Web Server 31
 - ◊ Personal Web Server 30
 - ◊ Sambar 31
 - ◊ локальный 86, 99
 - ◊ удаленный 86, 98, 102
- Web-страница 25
 - ◊ главная 83, 94
 - ◊ закрытие 54
 - ◊ название 53, 72, 177, 195, 225
 - ◊ обновление 228
 - ◊ общие параметры 176

- ◊ открепление 228
- ◊ открытие 54, 93
- ◊ по умолчанию 27, 91, 94
- ◊ просмотр 54, 94
- ◊ создание 49, 93, 226, 227
- ◊ сохранение 53
- ◊ текущая 27
- Web-сценарий 118, 277, 284
- Workspace Setup 34
- WWW 17

Z

- Z-индекс 264—273

A

- Автономный режим 28
- Администратор 26
- Архитектура 19, 20
 - ◊ двухзвенная 19
 - ◊ клиент-сервер 19
 - ◊ однозвенная 20
- Атрибут:
 - ◊ необязательный 69
 - ◊ обязательный 69
 - ◊ стиля 158, 242, 266
 - ◊ тега 69, 73

B

- Безопасный набор цветов 66
- Блок 175
- Брандмауэр 90

B

- Вложенность тегов 70, 73
- Врезка 301
- Вставка 52, 235
- Выделение:

- ◊ границы контейнера 263
- ◊ группы контейнеров 269, 271
- ◊ изменяемой области 225
- ◊ изображения 112
- ◊ наборов фреймов 192
- ◊ необязательной области 232
- ◊ повторяющейся области 234
- ◊ свободного контейнера 263, 271
- ◊ содержимого тега 72
- ◊ столбца 138, 139
- ◊ строки 138
- ◊ таблицы 138, 139, 207
- ◊ текста 52
- ◊ файла или папки 91
- ◊ фреймов 192
- ◊ ячейки 138, 205
- Выравнивание 56, 115, 131, 140—144, 165—167, 209, 269
- Вырезание 52, 235

G

- Гиперссылка 74, 78, 93, 174, 295
 - ◊ активная 174, 296
 - ◊ внешняя 98
 - ◊ графическая 120

Гиперссылка (*прод.*):

- ◊ мертвая 97
 - ◊ оборванная 97
 - ◊ обычная 296
 - ◊ посещенная 174, 296
 - ◊ почтовая 76
 - ◊ правка 75
 - ◊ создание 74—78, 94, 117
 - ◊ текстовая 117
 - ◊ удаление 75
 - ◊ цель 75, 195
- Горячая область 120—123
 Граница обрезки 125
 Группа контейнеров 269, 270
 Группа панелей 37, 38

Д

Дизайн Web-страниц:

- ◊ табличный 197, 214
 - ◊ фреймовый 182
- Док 38
 Домен 23
 Доменная зона 23
 Доменное имя 23

ЕЕдиницы измерения в стандарте CSS
162, 243**З**

- Заголовок 55
 Запрос 18
 Захват 263
 Зачеркивание 62
 Звук 287
 Значение атрибута 69, 158, 266
 Значок свободно позиционируемого
 контейнера 263

И

- Иерархия тегов 70
 Изображение:

- ◊ вставка 109
- ◊ гиперссылка 117
- ◊ изменение размеров 125
- ◊ маркер 172
- ◊ настройка яркости и контрастности 125
- ◊ обрезка 124
- ◊ параметры 113
- ◊ перекрывающее 118, 302
- ◊ плавающее 115
- ◊ полутоновое 109
- ◊ размеры 112, 114
- ◊ распорка 211—214
- ◊ смена на событие 291, 292
- ◊ увеличение резкости 126
- ◊ фоновое 141—143, 163, 268—270
- ◊ черновое 114
- ◊ штриховое 108

Имя:

- ◊ активного изображения 119
- ◊ изменяемой области 224, 225
- ◊ карты-изображения 122
- ◊ необязательной области 231, 232
- ◊ повторяющейся области 233
- ◊ пользователя 24, 87
- ◊ свободного контейнера 267, 271
- ◊ стилевое класса 152, 155
- ◊ стиля 158
- ◊ фрейма 191—195
- ◊ шаблона 222
- ◊ элемента полосы навигации 303
- ◊ якоря 76

Индекс 166

Инструментарий:

- ◊ документа 40, 41, 53
 - ◊ объектов 40, 41
 - ◊ панели Files 91
 - ◊ стандартный 40, 41
- Интернет-сервис 17
 Интернет-адрес 22, 74, 78, 86, 120
- ◊ абсолютный 27
 - ◊ относительный 28
- Интернет-провайдер 16

К

- Карта-изображение 120
- Каскадная таблица стилей 150
- Каскадность 156
- Кегль 63
- Клиент 18
 - ◇ почтовый 76
- Кнопка:
 - ◇ выключатель 62
 - ◇ переключатель 57
- Кодировка 43—45, 72, 79, 177
- Контейнер 175, 239, 241—245
 - ◇ блочный 175, 239, 301
 - ◇ видимая область 268
 - ◇ видимость 267, 271, 272, 290
 - ◇ вложенный 263, 272
 - ◇ встроенный 176, 239
 - ◇ координаты 267
 - ◇ параметры 267
 - ◇ переименование 272
 - ◇ перекрытие 264, 265, 271, 272
 - ◇ перемещение 263
 - ◇ переполнение 268
 - ◇ плавающий 244, 245, 255
 - ◇ размеры 263, 267
 - ◇ саморастягивающийся 251, 255
 - ◇ свободно позиционируемый 261, 271, 289, 290, 298
 - ◇ свободный 261
 - ◇ создание 240, 262
 - ◇ удаление 263
 - ◇ фиксированный 244
- Конфликт стилей 157
- Координатная сетка 199
 - ◇ настройка 200
- Копирование 52, 235
- Копия Web-сайта:
 - ◇ локальная 84, 99, 220
 - ◇ удаленная 84, 86, 92, 99
- Красная строка 167
- Курсор текстовый 51

Л

- Линейка измерительная 199
- Линия:
 - ◇ горизонтальная 78
 - ◇ размерная 134, 202
- Локальный хост 86

М

- Маркер:
 - ◇ изменения размера 112, 123—125, 139, 205, 263, 269
 - ◇ плавающего изображения 116
- Меню:
 - ◇ вкладок 40
 - ◇ главное 36
 - ◇ Избранное 301
 - ◇ контекстное 52
 - ◇ панели дополнительное 38
 - ◇ поведений 279
 - ◇ событий 280
 - ◇ строки размеров 139, 212, 214

Н

- Набор фреймов 182, 190—192
- Направляющая 201—204
- Начертание 61, 162

О

- Область шаблона:
 - ◇ заголовок 225, 232—234
 - ◇ изменяемая 219, 225
 - ◇ копия 233, 234
 - ◇ необязательная 230, 232
 - ◇ переименование 232, 234
 - ◇ перемещение 225, 234
 - ◇ повторяющаяся 233, 234
 - ◇ присутствие на странице 232
 - ◇ создание 223, 230
 - ◇ удаление 225, 234
- Объединение ячеек 145

Окно:

- ◊ всплывающее 285
 - ◊ выбора цвета 65
 - ◊ главное 35
 - ◊ документа 37, 41, 72, 94
 - ◊ предупреждение 287
- Откат 53, 127
 Отступ 58, 168, 170
- ◊ между символами 165
 - ◊ между словами 165
 - ◊ отрицательный 253
 - ◊ фреймов 194
 - ◊ ячейки 130, 144, 213
- Ошибка 404 97

П

- Пакет 20
 Палитра 65
 Панель 38
 Папка:
- ◊ виртуальная 26
 - ◊ корневая 26, 86, 99
 - ◊ переименование 92
 - ◊ перемещение 94
 - ◊ создание 93
 - ◊ удаление 92
- Пароль 24, 87
 Перенаправление 282—284
 Перенос строк 141, 142, 210
 Поведение 278—280
 Подтверждение 19
 Подчеркивание 62
 Полоса навигации 302
 Порт 22, 24, 90
 Потомок 71
 Приклеивание 200—205
 Пробел неразрывный 81, 132
 Проверка:
- ◊ HTML-кода 95
 - ◊ гиперссылок 97
- Прокрутка 193
 Прокси-сервер 90
 Протокол 20—24
 Псевдостиль гиперссылок 173, 295

Пункт меню:

- ◊ выключатель 41
- ◊ переключатель 57

Р

- Рабочая среда 34
 Разделитель 245
 Разрыв строк 51, 80
 Разъединение ячеек 147
 Рамка 130, 169
- ◊ невидимая 130
 - ◊ стиль линии 170
 - ◊ толщина 115, 130, 144, 170, 192, 297
 - ◊ фреймов 192, 194
 - ◊ цвет 141—143, 171, 192—194
- Редактор свойств 55
 Режим отображения:
- ◊ кода HTML 68
 - ◊ обычных таблиц 139, 199, 214
 - ◊ разметка страниц 198, 199, 214
 - ◊ страницы 66, 72
 - ◊ страницы и кода HTML 66
 - ◊ увеличенных таблиц 138, 199
- Родитель 71
 Русификация 43

С

- Секция Web-страницы:
- ◊ заголовка 71, 158
 - ◊ тела 71
- Секция строки статуса 72
- ◊ тегов 73, 155
- Селектор цвета 64
 Сервер 18
- ◊ DNS 24
 - ◊ FTP 86
- Символ:
- ◊ код 43
 - ◊ специальный 80, 81
- Синхронизация 100
 Событие 277—280
 Соединение:
- ◊ разрыв 19, 103

Сообщение об ошибке 19
Список 59
◇ маркер 171
◇ маркированный 59
◇ меню 60
◇ нумерованный 59
◇ определений 60
◇ папок 60
◇ параметры 60
◇ пункт 59—61
◇ стиль 61
◇ тип 60
Справка:
◇ по CSS 160
◇ по Dreamweaver 42
◇ по HTML 73
Стартовая страница 36
Стилевой класс 152—157
Стиль 150—158, 242
◇ внедренный 159
◇ встроенный 159
◇ гибридный 159
◇ переопределения тега 152, 157
◇ правка 172
◇ привязка 155, 156
◇ селектор 266, 267
◇ создание 152, 155
◇ создание копии 173
◇ удаление 173
Столбец 133—135
Строка 136
◇ вертикальный размер 163
◇ высота 214
◇ добавление 134
◇ параметры 142
◇ размеры 133, 134, 207, 212
◇ с относительной высотой 145
◇ с фиксированной высотой 133, 145
◇ текущая 135
◇ удаление 135
Строка статуса 72, 91, 289
Сценарий 277

Т

Таблица 136, 296
◇ высота 143, 144
◇ заголовок 207
◇ кодовая 43
◇ название 131, 136, 138
◇ параметры 143, 213
◇ перемещение 208
◇ примечание 131
◇ размеры 134, 139, 208
◇ разметки 197
◇ резиновая 133
◇ с относительной шириной 133
◇ с фиксированной высотой 134
◇ с фиксированной шириной 133, 134, 213, 216
◇ саморастягивающаяся 213
◇ создание 129, 207
◇ удаление 208
◇ шапка 131, 141
◇ ширина 130, 143, 144
Таблица стилей 150
◇ внешняя 150—158
◇ внутренняя 150—158, 266
◇ привязка 154, 158
◇ удаление 173
Тег 25, 69, 73, 113, 152
◇ видимый 72
◇ дочерний 71
◇ закрывающий 69
◇ имя 152
◇ название 69
◇ невидимый 72
◇ одинарный 72
◇ открывающий 69
◇ парный 69
◇ пустой 96
◇ родительский 71
◇ содержимое 69, 73
◇ удаление 73
Текст 115, 166
Текст замены 111, 115

Точка:

- ◇ начала отсчета 199
- ◇ угловая 123

Ф

Файл:

- ◇ копирование 94
- ◇ переименование 92
- ◇ перемещение 94
- ◇ сирота 98
- ◇ удаление 92, 101
- Фрейм 182—192, 284, 288
 - ◇ высота 190, 192
 - ◇ граница 183
 - ◇ заголовок 185—191
 - ◇ параметры 193
 - ◇ размеры 186
 - ◇ резиновый 190
 - ◇ создание 183
 - ◇ текущий 184
 - ◇ фиксированный 193
 - ◇ ширина 190, 192

Х

Хостинг-провайдер 32

Ц

Цвет 163, 295

- ◇ код 64
- ◇ линий сетки 200
- ◇ направляющих 203
- ◇ размерных линий 203
- ◇ рамки 298
- ◇ фона 141—143, 163, 209, 213, 268—270
- ◇ шрифта 64—66

Ш

- Шаблон 49, 219—229
- Шрифт 62, 161
- ◇ размер 161
 - ◇ стандартный 63

Э

Элемент Web-страницы:


- ◇ внедренный 108
- ◇ высота 168, 242
- ◇ неизменяемый 219
- ◇ текстовый 107
- ◇ ширина 168, 242

Я

Якорь 76—78

Ячейка 136—144

- ◇ высота 141, 144, 209
- ◇ заголовка 131, 136, 141
- ◇ параметры 140, 209
- ◇ перемещение 205
- ◇ размеры 205
- ◇ разметки 197
- ◇ резиновая 209
- ◇ с относительной шириной 145
- ◇ с фиксированной шириной 145, 209—216
- ◇ саморастягивающаяся 209—212, 216
- ◇ создание 204
- ◇ удаление 206
- ◇ ширина 141—144, 208—214



Как заставить
гигагерцы
работать?

Самый полный каталог «софта»

Чтобы техника правильно работала, ей необходимы правильные программы. Линейка каталогов SoftLine-direct предлагает вам программное обеспечение от ведущих мировых производителей. Оформите **БЕСПЛАТНУЮ** подписку, и вы всегда будете в курсе новинок и тенденций IT-рынка. Самый простой и надежный способ купить софт — заказать его по каталогу SoftLine-direct.

softline®
ЛИЦЕНЗИРОВАНИЕ. ОБУЧЕНИЕ. КОНСАЛТИНГ

119991 Москва, ул. Губкина, 8. Тел./факс: (095) 232 00 23
E-mail: info@softline.ru <http://www.softline.ru>

• Москва • Санкт-Петербург • Екатеринбург • Нижний Новгород • Новосибирск • Ростов-на-Дону • Хабаровск • Минск • Киев • Ташкент • Алматы



www.bhv.ru

Книги издательства "БХВ-Петербург" в продаже:

Магазин "Новая техническая книга": СПб., Измайловский пр., д. 29, тел.: (812) 251-41-10
Отдел оптовых поставок: e-mail: opt@bhv.spb.su

Серия "Учебное пособие"/ "Учебник"

Барсегян А. Методы и модели анализа данных: OLAP и Data Mining (+CD-ROM)	336 с.
Белов Д. и др. Разработка экспертных систем. Среда CLIPS	608 с.
Бенькович Е. Практическое моделирование динамических систем (+CD-ROM)	464 с.
Бенькович Е. и др. Практическое моделирование динамических систем (+CD-ROM)	464 с.
Боев В. Моделирование систем. Инструментальные средства GPSS World	368 с.
Бойко В. Схемотехника электронных систем. Микропроцессоры и микроконтроллеры	464 с.
Бойко В. Схемотехника электронных систем. Цифровые устройства	512 с.
Бойко В. Схемотехника электронных систем. Аналоговые и импульсные устройства	450 с.
Большаков В. Инженерная и компьютерная графика. Практикум	592 с.
Вуль В. Электронные издания (+дискета)	560 с.
Гомоюнов К. Транзисторные цепи	240 с.
Грушвицкий Р. и др. Проектирование систем на микросхемах программируемой логики	608 с.
Доманова Ю. и др. Высшая математика на базе Mathcad. Общий курс	608 с.
Ирвин Дж., Харль Д. Передача данных в сетях: инженерный подход	448 с.
Иртегов Д. Введение в сетевые технологии	560 с.
Кубенский А. Структуры и алгоритмы обработки данных: объектно-ориентированный подход и реализация на C++ (+CD-ROM)	464 с.
Кузнецов А. и др. Математика для экономистов на базе Mathcad	496 с.
Культин Н. Delphi в задачах и примерах (+CD-ROM)	288 с.
Культин Н. C/C++ в задачах и примерах	288 с.
Культин Н. Turbo Pascal в задачах и примерах	256 с.
Лурье Б., Энрайт П. Классические методы автоматического управления	640 с.
Макаров Е. Сопrotивление материалов на базе Mathcad (+CD-ROM)	512 с.
Никулин Е. Основы теории автоматического управления. Частотные методы анализа и синтеза систем	640 с.