

Как читать эту книгу

Если эта книга послужит вам приятным послеобеденным чтением, — что ж, буду рада за вас. Но если вы всерьез собрались освоить HTML, то ее нужно читать иначе.

Каждая глава несет в себе информацию трех видов: теоретическую, практическую и справочную. Как усваивать теорию — дело ваше. Одним людям достаточно просто вдумчиво прочесть главу. Другие ведут конспект. Третьи делают нечто наподобие школьной шпаргалки, которой потом пользуются в работе. Подберите тот вариант, который вам удобнее и больше нравится. Здесь все зависит от привычек и особенностей вашей памяти.

Но в любом случае будет не лишним проверить себя: правильно ли вы поняли прочитанное? Для этого в конце каждой главы приводится раздел *Тесты*. Там вы найдете вопросы с несколькими вариантами ответа на каждый. Отметьте те варианты, которые вы считаете правильными, и проверьте свои ответы по *Приложению А*. Обратите внимание, что среди множества предлагаемых вариантов может не оказаться ни одного правильного или несколько правильных ответов. Кроме того, встречаются ответы частично правильные, например, вариант кода, который в принципе работает, но не является оптимальным. В общем, все как в жизни... Так что будьте внимательны.

Однако основные навыки приобретаются, конечно же, на практике. Поэтому настоятельно советую, после того как прочтаете главу, выполнить все описанные в ней примеры самостоятельно. Для этого вам понадобится компьютер, на котором установлены HTML-редактор и браузер. Желательно также иметь доступ к Internet. Но если у вас его нет, не беда. Он нам будет нужен не так часто. При необходимости можете сходить в ближайшее Internet-кафе.

Наконец, в каждой главе содержится информация справочного характера, такая как описание дескрипторов и их параметров. На мой взгляд, зазубривать эти сведения ни к чему. В конце учебника вы найдете сводную таблицу дескрипторов HTML (см. *Приложение Б*). Можете пользоваться ею, а можете по ходу чтения составить собственную.

Некоторые главы снабжены разделами, озаглавленными *Технические подробности*. В них содержатся более углубленные технические сведения, касающиеся темы, рассматриваемой в данной главе.

Некоторые заметки и рекомендации в книге снабжены пиктограммами. Ответы на наиболее типичные вопросы начинающих помечены пиктограммой *Совет*. Пиктограммой *Внимание* выделены важные моменты, о которых следует помнить при создании Web-страниц. Под пиктограммой *На заметку* приводятся удобные практические приемы, наблюдения — результат многолетнего опыта многих Web-дизайнеров. Это не правила и не стандарты. Вы можете согласиться с ними и взять на вооружение, а можете поспорить, перепроверить и только потом решить, насколько они вам подходят.

В любом случае, я желаю вам успехов в изучении HTML и удовольствия от создания собственных HTML-документов.

*Елена ПОЛОНСКАЯ,
автор*

Введение. Первые “почему”

Уважаемый читатель! Позвольте задать вам вопрос: зачем вам понадобилось изучить HTML? Для того чтобы стать Web-дизайнером? Чтобы создать виртуальную книгу, учебник, презентацию? Сдать зачет? Для общего развития?

Как бы то ни было, какая-то причина все же побудила вас взять в руки эту книгу, пролистать первые страницы. Пойдете ли вы до конца? Решитесь ли подарить себе удовольствие нового знания, умения? Не берусь судить. Но, говорят, тот, кто однажды отказал себе в таком удовольствии, потом долго об этом жалеет...

Домашняя Web-страничка — это, конечно, далеко не всегда источник дохода (хотя нередко бывает и так). Гораздо чаще это виртуальная визитная карточка, средство общения, “живой” учебник или презентация, которые можно без особых хлопот изготовить самому, не прибегая к помощи программиста. Нередко это первая проба Web-дизайнерского пера, за которой потом появляются другие, лучшие, профессиональные... И всегда — средство самовыражения.

Начинать всегда немного боязно и страшно. Боязно — как нырять в холодную воду после того, как угреешься на теплом песке. Страшно — как все новое. Тебя удивляют правила новой среды. А старожилы удивляются твоим вопросам. Но постепенно образуется некий общий знаменатель. А потом начинает *получаться*. И тогда приходит удовольствие. Как от всего, что делаешь своими руками и делаешь хорошо, — будь то гениальный роман или ровно вбитый гвоздь.

Но поначалу все кажется немного странным. Особенно до тех пор, пока не найдешь ответы на основные “почему”.

Почему HTML

Языков программирования и так в последнее время расплодилось довольно много. Зачем было изобретать еще одну тарабарскую систему кодов, на сей раз для Web? Неужели нельзя было взять обычный формат текстовых документов, например Word, со всеми его шрифтами, таблицами, картинками и, само собой, гиперссылками, и написать для него программу-просмотрщик? Зачем было изобретать какой-то язык разметки гипертекста и набирать в куцем Notepad нечто жуткое, не имея возможности даже увидеть, как это выглядит на самом деле? Ведь вот он — Word, под рукой. И все видно, и все понятно, и его документы везде открываются. Похоже, эти интернетчики — просто ненормальные. Создают сложности на ровном месте себе и другим...

У вас никогда не возникало подобных мыслей?

У меня — возникали.

Оставим в покое историю развития Internet. Вряд ли вас удовлетворит ответ в духе “все всегда так делали и мы будем делать так же”. Займемся днем сегодняшним. И самыми практическими его вопросами.



Почему объем HTML-страницы так важен? Говоря о месте на жестком диске, мы давно оперируем гигабайтами или, на худой конец, сотнями мегабайт. А тут трясемся над каждым десятком килобайт!

В чем причина? В каналах связи.

подавляющее большинство отечественных пользователей Internet подключаются к сети по коммутируемой телефонной линии через модем. Повысить качество такой связи они обычно могут только в очень ограниченных пределах, так как определяется оно не столько модемом, сколько бабушкой АТС, из которой весь песок уже давно высыпался.

Попробуем приблизительно оценить, какой объем должна иметь HTML-страница, чтобы посетитель не заскучал и не ушел дальше бродить по Internet, не дожидаясь конца ее загрузки.

Скорость загрузки страницы на компьютер ее посетителя определяется многими факторами. Учесть их все при таком поверхностном расчете не представляется возможным. Поэтому будем опираться только на один из них, который нам известен, — скорость модема. Предположим, она составляет 33,6 Кбит/с. Это значит, что (подчеркнем, не учитывая массу других факторов!) за секунду модем передает $33,6 \text{ Кбит} / 8 = 4,2 \text{ Кбайт}$ данных. Но количество полезных данных меньше, так как на линии часто возникают ошибки, отчего данные приходится передавать повторно. Насколько оно меньше, точно сказать нельзя. Предположим, что в полтора раза. Тогда получится, что пользователь принимает $4,2 / 1,5 = 2,8 \text{ Кбайт}$ полезной информации в секунду. Сколько времени он будет терпеливо ждать загрузки страницы? Конечно, это зависит и от темперамента пользователя, и от того, насколько ему нужна именно эта страница, и еще от массы причин. Но предположим, что это время составляет 1 минуту. Тогда, как нетрудно подсчитать, максимальный размер Web-страницы составляет $2,8 * 60 = 168 \text{ Кбайт}$. Нетрудно подсчитать, что для скорости 14,4 кбит/с этот размер составит 72 Кбайт, а для скорости 54,6 Кбит/с — 273 Кбайт.

Конечно, мы многого не учли в нашем примитивном расчете. Однако он позволяет сказать, что размер Web-страницы должен составлять 100–150 Кбайт и уж, во всяком случае, не должен превышать 250 Кбайт.

Но ведь файлы с HTML-кодом гораздо меньше, скажете вы. Действительно, размер самого кода редко превышает несколько десятков килобайт. Но следует помнить, что вместе с кодом на компьютер посетителя страницы передаются файлы с картинками и, возможно, другими нетекстовыми объектами. А их размер может значительно превышать определенный нами лимит. О том, как поступать в таких случаях, вы узнаете из главы 8.

Следует отметить, что это довольно жесткое ограничение. Оно не позволяет использовать на Web-страницах интенсивную графику, аудио- и видеоэффекты. Для того чтобы такие вещи, как просмотр через Internet видеофильмов и телеканалов, стали реальностью, нужны гораздо более мощные каналы передачи данных, — такие, которые позволили бы оперировать не сотнями килобайт, а гигабайтами или хотя бы сотнями мегабайт.

Скачайте из Internet и сохраните на диске любую Web-страницу. Можно даже без картинок. Откройте ее, скажем, в Word 2000 или XP и сохраните как документ Word. Готово? Теперь откройте Explorer (Проводник) или какой-нибудь Commander и сравните размеры обоих файлов. Ну как? Могу поспорить, что файл с кодом HTML не просто меньше, а раз в десять меньше. Или даже в пятьдесят... А теперь припомните свои впечатления, когда вы последний раз загружали объемистую Web-страницу и минут пять глазели на одинокий баннер. Не мелькала ли у вас тогда мысль заставить автора этой страницы частично оплатить вам доступ к Internet?

А у того — свои заботы. Каждый хозяин Web-собственности желает, чтобы его страницу в Internet посещали часто и с удовольствием. Иначе зачем было городить весь этот виртуальный огород? А раз так — страницы должны быть, во-первых, интересными, а во-вторых, *быстро открываться*. За первое HTML ответственности не несет. Зато за второе — отвечает головой. И, поверьте, неплохо справляется. Вы могли убедиться в этом сами, сравнив объем HTML-страницы с объемом документа Word.

Идем дальше. У вас какой компьютер? Pentium II? Прекрасно. P-IV? Ого! “Двойка”? Смотрите-ка, жива ведь старушка... Мака ни у кого нет? Ладно, шучу. А впрочем, вон кто-то тянет руку в заднем ряду. Карманный ПК... Мобильный телефон... Телевизор с Web-приставкой... Что? Холодильник?! Н-да, дожили...

А какая у вас операционная система? Windows 98... NT... 2000... XP... “Могу-чая кучка” линуксоидов с фанатично горящими глазами... UNIX? Вы, наверное, системный администратор или скоро им будете... Владельцам Mac и карманных ПК выбирать особо не из чего... Плюс то, что заменяет ОС в мобильном телефоне, Web-приставке и... гм, в холодильнике.

То, что нас так много, — это еще ничего. Это даже очень хорошо. Хуже, что у нас такие разные *платформы* — компьютеры и операционные системы. Поэтому нам, *сообществу Internet*, для обмена информацией нужен такой формат, который бы “понимали” *все* браузеры *всех* производителей и *всех* операционных систем, настоящие и по возможности будущие. В PC, Macintosh, карманном ПК и холодильнике. Можем ли мы положиться на двоичный формат отдельно взятого приложения, разработанный отдельно взятой — пусть даже очень большой — компанией? Голосуем: кто за то, чтобы вверить формат наших Web-страниц заботам г-на Гейтса? Заснули, что ли... Ладно, кто против? Нет, похоже, не заснули.

Раз так — возникает следующий вопрос: какой формат самый универсальный? Очевидно, такой, который прочтет и поймет не только программа, но и, в случае чего, человек. Какой формат для этого лучше всего подходит? Правильно — *текстовый*.

Если взять обычный текстовый файл и разметить в нем фрагменты, которые нужно выделить, скажем, курсивом или цветом, обозначить нумерованные и маркированные списки, места вставки рисунков и т.п. (рисунки и другие нетекстовые объекты придется поместить в отдельных файлах), то любой мало-мальски толковый программист практически за неделю “нарисует” вам программку, которая бы отображала такие файлы на экране в более или менее приятно читаемом виде. Вот вам и браузер. Осталось разработать единую систему разметки текста. Или, точнее, *гипертекста*, по-скольку, кроме собственно текста — и не простого, а форматированного — в нем ведь есть еще картинки, ссылки на другие документы и файлы, а иногда аудио- и даже ви-

деовставки. Что у нас получится? Язык разметки гипертекста, или, по-английски, HyperText Markup Language — HTML. Вот мы и изобрели велосипед...

И последний, но немаловажный аргумент в пользу HTML. К сожалению, в силу — как бы это сказать помягче? — некоторых специфических традиций отечественного рынка программного обеспечения этот аргумент не вполне очевиден. Вы когда-нибудь интересовались, *сколько стоит* текстовый процессор Microsoft Word? Нет? Поинтересуйтесь. За эту сумму можно устроить себе неплохой летний отпуск. А текстовых редакторов для ASCII — пруд пруди. Помимо Notepad и редакторов, встроенных во всяческие “Коммандеры”, на Web-узлах бесплатного ПО их сотни мегабайт. И это не считая специализированных HTML-редакторов, среди которых тоже немало бесплатных и условно-бесплатных. Если бы вся Всемирная Паутина перешла на стандарт Word, она бы, пожалуй, быстро лишилась многих небогатых, но законопослушных авторов. А поскольку слишком много народу уже жизни не мыслит без Internet, то происходит как раз обратное: Word и весь Microsoft Office, да и другие офисные пакеты, дружно принимают HTML в виде еще одного текстового формата.

Почему Notepad

Notepad, он же Блокнот Windows — любимый редактор Web-дизайнеров. Хотя есть много других хороших и не менее уважаемых ими специализированных редакторов для HTML, таких как MacroHTML или HomeSite. Уважают их, главным образом, за простоту, доступность, компактность. А еще за невмешательство в творческие дела дизайнера. Все очень просто: вы вводите в редакторе HTML-код и любуетесь результатом в окне параллельно работающего броузера. Редактор в лучшем случае ускорит ввод ключевых слов и поможет с синтаксической проверкой. Остальное — за вами.

Другой тип HTML-редакторов — *визуальные*, такие как известный Microsoft FrontPage Express или гораздо более любимый Web-дизайнерами Macromedia DreamWeaver. Их интерфейс построен по тому же принципу, что и интерфейс текстового процессора, скажем Word. Кстати, Word 2000 или XP вполне сойдет в качестве визуального HTML-редактора.

Преимущество визуальных HTML-редакторов в том, что для работы с ними можно вообще не знать HTML. К сожалению, визуальные редакторы страдают тем же недостатком, что и трансляторы языков программирования, преобразующие текст на C или Pascal в ассемблерный код. А именно — неоптимальностью этих самых кодов. Спросите любого программиста, и он вам скажет, что ассемблерный код, написанный вручную, в несколько раз меньше кода такой же программы, написанной, скажем, на Pascal и преобразованной в ассемблерный код с помощью транслятора. Удивляться нечему: решения человека-специалиста всегда красивее решений машины. Зато и труда, и времени они требуют несравнимо больше. Впрочем, вероятно, недалек тот день, когда при загрузке страницы разницы в несколько килобайтов или даже мегабайтов не будет так заметна, как сейчас. Тогда визуальные HTML-редакторы займут на компьютере Web-дизайнера такое же достойное место, какое сейчас занимают на компьютере программиста визуальные среды разработки, такие как, например, Delphi.

Увы, этим грехи визуальных редакторов не исчерпываются. Часто страница выглядит в браузере совсем не так, как до этого она выглядела в окне редактора. Приходится лезть в длинный, запутанный код, искать там ошибки и вручную их исправлять.

Когда-то давно мне встретились в одной книжке десять шуточных заповедей программиста. Одна из них гласила: “Не возжелай программы ближнего твоего!”. В применении к нашей задаче она звучит так: “Не возжелай HTML-кода ближнего твоего, в особенности если этот ближний — FrontPage, а ты собрался изучать HTML”.



Где взять хороший HTML-редактор?

HTML-редакторов множество, и у каждого из них есть свои преимущества и недостатки. Поэтому однозначно сказать, какой именно вам подойдет лучше всего, нельзя. Одним нравится HomeSite, другим — 1st Page, третьим — Arachnophilia или Easy HTML. Или другой из десятков им подобных. Здесь все в конечном счете сводится к личным вкусам.

Для того чтобы подобрать редактор по душе, советую посетить несколько Web-хранилищ бесплатного и условно-бесплатного программного обеспечения и там подобрать себе редактор по душе и по карману. Поэтому вместо того, чтобы сравнивать редакторы (тем более что регулярно появляются новые версии с новыми функциями), я лучше приведу несколько адресов, по которым стоит обратиться.

- 1st Page: www.evrsoft.com
- Arachnophilia: www.arachnoid.com
- HomeSite: www.macromedia.com
- Easy HTML: www.ukrwest.net
- Также виртуальные архивы ПО: FreeSoft (www.freesoft.ru), Freeware.ru (www.freeware.ru), ListSoft (www.listsoft.ru), Softarea (www.softarea.ru) и др.

Почему HTML не является языком программирования

Многие ошибочно называют HTML языком программирования. К сожалению, это не так. И дело здесь даже не в том, что у HTML нет компилятора, а только встроенный в браузер интерпретатор: BASIC и JavaScript тоже обходятся одними интерпретаторами, но это не мешает им называться языками программирования. Дело в том, что в HTML отсутствует главный атрибут, присущий любому языку программирования, — команды. На HTML нельзя задать *последовательность действий*, а можно только описать, как браузер должен вводить на экран тот или иной документ. Если же на Web-странице действительно должно что-то выполняться, например вестись форум, то используются настоящие языки программирования, такие как Java и JavaScript (см. главу 16). Поэтому говорить “программа на HTML” не вполне корректно. Мы будем пользоваться термином “HTML-код”.

Резюме

Язык HTML, или универсальный язык разметки гипертекста, используется для создания самых разных интерактивных документов с гиперссылками и элементами мультимедиа — Web-страниц, интерфейсов, презентаций, электронных книг и учебных пособий. Файлы с HTML-кодом — это обычные текстовые файлы, доступные для чтения как программе, так и человеку. Благодаря этому HTML-страницы можно редактировать и просматривать на любом компьютере и в любой операционной системе.

Для создания HTML-страниц можно пользоваться любым текстовым редактором, но существуют и специализированные программы. Эти HTML-редакторы делятся на два типа: визуальные и невизуальные. Визуальные HTML-редакторы обладают интуитивно понятным интерфейсом и не требуют много времени на освоение, но генерируют очень длинный, неоптимальный и малопонятный HTML-код, который потом трудно редактировать. К редакторам этого типа относится, например Microsoft FrontPage Express.

Невизуальные редакторы требуют знания HTML, но лишены недостатков визуальных редакторов. Кроме того, это, как правило, небольшие, компактные программы, бесплатные или условно-бесплатные. К HTML-редакторам этого типа относятся, например MacroHTML и HomeSite.

HTML не является языком программирования. Поэтому для того чтобы на HTML-странице что-то выполнялось, например, выводился какой-то текст или менялась картинка в ответ на определенные действия пользователя, необходимо пользоваться специальными средствами, расширяющими возможности HTML, такими как язык программирования JavaScript.

Тесты

1. Что из нижеперечисленного является браузером?
 - а) Microsoft Word
 - б) Microsoft Internet Explorer
 - в) Microsoft FrontPage Express
 - г) HomeSite
 - д) Netscape Navigator
 - е) Opera
 - ж) Mozilla
 - з) Notepad
 - и) PhotoShop
 - к) Google.com
2. Что из нижеперечисленного можно использовать для просмотра HTML-страницы?

- а) Microsoft Word
 - б) Microsoft Internet Explorer
 - в) Microsoft FrontPage Express
 - г) HomeSite
 - д) Netscape Navigator
 - е) Opera
 - ж) Mozilla
 - з) Notepad
 - и) Windows
 - к) Linux
 - л) PhotoShop
 - м) Google.com
3. Что из нижеперечисленного можно использовать для просмотра кода HTML-страницы?
- а) Microsoft Word
 - б) Microsoft Internet Explorer
 - в) Microsoft FrontPage Express
 - г) HomeSite
 - д) Netscape Navigator
 - е) Opera
 - ж) Mozilla
 - з) Notepad
 - и) Windows
 - к) Linux
 - л) PhotoShop
 - м) Google.com