

Министерство науки и высшего образования Российской Федерации

Рязанский институт (филиал)
федерального государственного автономного образовательного учреждения
высшего образования
«Московский политехнический университет»

Кафедра «Информатика и информационные технологии»

Ю.И. Арабчикова, Т.А. Асаева

ИНТЕРНЕТ-ТЕХНОЛОГИИ
Часть 1

Учебно-методическое пособие

Рязань
2023

УДК 004.4
ББК 16.263
А-79

Арабчикова, Ю.И.

А-79 Интернет-технологии. Часть 1: учебно-методическое пособие / Ю.И. Арабчикова, Т.А. Асаева. – Рязань: Рязанский институт (филиал) Московского политехнического университета, 2023. – 24 с.

Данное пособие подходит как для самостоятельного изучения основ HTML-технологий, так и для проведения занятий со студентами направлений подготовки 09.03.01 Информатика и вычислительная техника, 09.03.02 Информационные системы и технологии и 27.03.04 Управление в технических системах в рамках курса «Интернет-технологии». Рассматриваются вопросы использования языка разметки гипертекста HTML для верстки web-страниц. Основное внимание уделяется разбору теоретических правил верстки.

Печатается по решению методического совета Рязанского института (филиала) Московского политехнического университета.

УДК 004.4
ББК 16.263

© Арабчикова Ю.И., Асаева Т.А., 2023
© Рязанский институт (филиал)
Московского политехнического
университета, 2023

Содержание

Введение	4
1 Структура HTML-документа	5
2 Текстовое оформление страниц.....	6
2.1 Абзацы в HTML.....	6
2.2 Заголовки в HTML. Размер текста.....	7
2.3 Цвет текста и фон страницы	10
2.4 Оформление цитат	11
3 Создание списков	12
3.1 Нумерованный список	12
3.2 Маркированный список	13
4 Создание таблиц	14
Библиографический список.....	23

Введение

Современное информационное пространство – это всё в большей степени интернет-пространство, обладающее целым рядом принципиально новых свойств. В частности, благодаря поисковым системам, кардинально меняется ситуация с доступностью информации и её сортировки по различным критериям.

Интернет-технологии стремительно развиваются, проникая в самые разнообразные сферы профессиональной деятельности. Для компаний присутствие в Интернете – это возможность рассказать о своих товарах и услугах, найти потенциальных партнеров и клиентов, а также снизить издержки за счет Интернет-торговли, использования «облачных» сервисов. Даже такие традиционно замкнутые системы, как промышленные автоматизированные системы управления производством, в том числе и в критических отраслях, также в большинстве случаев прямо или косвенно подключены к Интернету.

Освоение необходимых теоретических основ для эффективного использования современных интернет-технологий в профессиональной деятельности и приобретение соответствующих практических навыков по созданию профессионально-ориентированных сайтов и их продвижение в интернет-пространстве приобретают статус мощного и незаменимого инструментария в арсенале специалистов.

Вместе с тем, новые возможности порождают и новые угрозы безопасности, которые не всегда в достаточной мере осознаются как рядовыми пользователями, так и владельцами ресурсов. Для компаний наиболее тревожной является тенденция роста атак на корпоративные сайты и web-приложения. Это является следствием того, что большинство компаний недооценивают опасность подобных атак, а разработчики web-приложений недостаточно тщательно следят за безопасностью собственных продуктов. Однако для рассмотрения вопросов, касающихся угроз безопасности, уязвимостей и методов защиты web-приложений, необходимо обладать базовыми навыками использования основных web-технологий.

Данное учебное пособие в сжатой форме знакомит с основными web-технологиями, такими как HTML и CSS, современными принципами и инструментальными средствами разработки web-сайтов и серверным программным обеспечением.

1 Структура HTML - документа

Веб-страница – это текстовый документ, который обычно является представлением некоторого ресурса всемирной паутины (WWW – World Wide Web). Веб-страница может содержать списки, таблицы, графические изображения, а также ссылки на другие ресурсы. Благодаря наличию ссылок содержимое веб-страницы часто называют гипертекстом.

Подобно разработке программы создание веб-страницы состоит в написании исходного кода с той лишь разницей, что в первом случае исходный код записывается на некотором языке программирования, а во втором – на специальном языке разметки гипертекста (HTML – Hypertext Markup Language). При написании исходного кода веб-страницы, так же, как и исходного кода программы, можно использовать как обычные текстовые редакторы типа «Блокнот», так и специализированные редакторы, и среды, поддерживающие подсветку синтаксиса, подсказки и автопродолжение.

Основой языка HTML являются теги. Каждый элемент страницы обозначен тем или иным тегом.

Тег – это «метка», состоящая из буквы или нескольких букв, заключенных между символами «меньше» – "<" и «больше» - ">". Теги бывают парные и непарные (одиночные).

У парных сначала идет открывающий тег – между символами "<" и ">", сообщающий браузеру о начале своего действия в данном месте; затем закрывающий тег – между символами "</" и ">" (то есть у закрывающего тега к символу «меньше чем» – "<" добавляется слэш "/"), сообщающий браузеру о завершении своего действия.

Закрывающий тег не всегда является обязательным. Одиночные теги, например, не имеют закрывающего элемента.

Любой *html*-документ должен начинаться тегом `<!DOCTYPE>`, который указывает тип текущего документа – DTD (Document Type Definition – описание типа документа). Указывается используемая версия HTML или XHTML. Благодаря тегу `<!DOCTYPE>` браузер понимает, как интерпретировать текущую страницу для ее корректного отображения.

Далее, после тега `<!DOCTYPE>` – указания типа текущего документа, следует тег `<html>`, который включает все содержимое страницы: теги `<head>` и `<body>`.

Тег `<head>` (голова документа) содержит служебную информацию о текущей странице.

Обязательным элементом головы документа является тег `<title>`, который отображает имя документа в заголовке окна браузера – в левом верхнем углу окна.

Тег `<body>` (тело документа) включает в себя все содержимое страницы – все что Вы видите в окне браузера.

У любого *html*-документа одна голова и одно тело. Важно, что сначала идет голова, затем тело.

Закрывающим тегом `</html>` заканчивается любой *html*-документ.

Фрагмент кода:

```
<!DOCTYPE>
<html> <!--Начало документа-->
  <head> <!--Голова документа-->
    <title> Заголовок </title> <!--Имя документа-->
  </head>
  <body>
<!--Тело документа - содержание страницы-->
</body>
</html> <!--Конец документа-->
```

Замечание. Для вставки комментария используется тег <!-- -->. Комментарий в браузере не отображается.

2 Текстовое оформление страниц

Форматированию текста в HTML в первую очередь следует уделить внимание, так как практически на любой странице большая часть информации представлена в текстовом виде.

Группу тегов HTML, предназначенных для работы с текстом можно разделить на две основные подгруппы: теги физического форматирования (Basefont, Big, Blockquote, Br, Font, H1...H6, Nobr, P, Pre, Small, Strike, Sub, Sup, U, Wbr) и теги логического форматирования текста (Abbr, Em, Q, Samp, Strong и др.).

Теги физического форматирования физически воздействуют на текст: отвечают за выделение, размещение, оформление текста, которое необходимо для его корректного и одинакового отображения в браузерах.

Теги логического форматирования в большей степени несут смысловую нагрузку, и некоторые из них в основном предназначены для удобства работы поисковых систем.

2.1 Абзацы в HTML

Важнейшим тегом физического форматирования текста является тег <p>, который предназначен для разбивки текста на абзацы.

Каждый тег <p> создает новый абзац. При этом текст начинается с новой строки, и между абзацами образуется промежуток.

Атрибут *align* тега <p> позволяет позиционировать абзац по горизонтали, размещая его слева (*align="left"*), справа (*align="right"*), по центру

(*align*="center") или выравнивая по ширине (*align*="justify") окна браузера или родительского элемента.

Текст после тега `
` также начинается с новой строки. Но тег `
` осуществляет лишь перенос строк - он не образует промежутка между ними и его можно использовать внутри тега `<p>`.

Тег `<nobr>` наоборот запрещает перенос строк. Если длина строки превысит ширину окна браузера, то появится горизонтальная полоса прокрутки. В свою очередь тег `<wbr>` разрешает перенос строки внутри тега `<nobr>`.

Фрагмент кода:

```
<p align="center"> Текст разбит на параграфы. </p>
```

```
<p> При просмотре в браузере новый абзац отделяется от предыдущего пустой строкой. Тег <br> используется для переноса строки. </p>
```

Результат:

Текст разбит на абзацы.

При просмотре в браузере новый абзац отделяется от предыдущего пустой строкой.
Тег " br " используется для переноса строки.

2.2 Заголовки в HTML. Размер текста

Заголовки в HTML представлены шестью тегами `<h1>...<h6>`. Каждый из них обозначает заголовок определенного уровня.

Тегом `<h1>` выделяют наиболее значимые части документа (тема страницы или статьи), а текст заключенный в него – самый крупный.

Тегами `<h2>` и `<h3>` можно выделить подтемы, крупные разделы. Текст в таких заголовках уже не такой крупный, как в заголовках высшего уровня.

И так далее, по значимости и иерархии, при помощи тегов `<h1>...<h6>`, выделяют заголовки различных уровней.

Фрагмент кода:

```
<h1> Заголовок 1-го уровня </h1>
```

```
<h2> Заголовок 2-го уровня </h2>
```

```
<h3> Заголовок 3-го уровня </h3>
```

```
<h4> Заголовок 4-го уровня </h4>
```

```
<h5> Заголовок 5-го уровня </h5>
```

```
<h6> Заголовок 6-го уровня - мелкий шрифт! </h6>
```

Результат:

Заголовок 1-го уровня

Заголовок 2-го уровня

Заголовок 3-го уровня

Заголовок 4-го уровня

Заголовок 5-го уровня

Заголовок 6-го уровня - мелкий шрифт!

Замечание. Теги `<h1>...<h6>`, как и тег `<p>` имеют атрибут *align*, позволяющий выравнивать заголовок по горизонтали.

Теги `<big>` и `<small>` тоже позволяют изменить размер текста: они соответственно увеличивают и уменьшают текущий размера шрифта на одну единицу.

В HTML размер шрифта задается в условных единицах от 1 до 7. По умолчанию текст имеет средний размер шрифта, относящийся к 3 условной единице.

Теги `<sup>` и `<sub>` предназначены для отображения текста в виде верхнего и нижнего индекса соответственно. При этом размер шрифта уменьшается на единицу.

И наконец рассмотрим тег ``. Он предназначен не только для изменения размера текста, но и для задания таких его характеристик как цвет и указания семейства шрифтов.

Атрибут *size* тега `` задает размер шрифта в условных единицах от 1 до 7, атрибут *color* предназначен для задания цвета, и атрибут *face* определяет семейство шрифтов.

Фрагмент кода:

```
<p> <big> Увеличение </big> и <small> уменьшение текущего размера шрифта на одну условную единицу. </small> </p>
```

```
<p> Индекс <sup> верхний </sup> и индекс <sub> нижний. </sub> </p>
```

```
<p> <font color="red" size="+2" face="Arial, sans-serif"> Изменение шрифта при помощи тега font. </font> </p>
```

Результат:

Увеличение и уменьшение текущего размера шрифта на одну условную единицу.

Индекс ^{верхний} и индекс _{нижний}.

Изменение шрифта при помощи тега font.

По начертанию шрифт может отличаться постановкой – прямой или курсив, и насыщенностью (плотностью) – светлый и жирный.

По умолчанию в браузере виден светлый текст прямой постановки. Тег `` позволяет отобразить текст курсивом, а тег `` сделать его жирным.

Фрагмент кода:

```
<p> Изменим начертание шрифта: <em> его постановку</em> и <strong> насыщенность! </strong> </p>
```

Результат:

Изменим начертание шрифта: *его постановку* и **насыщенность!**

Многие среды разработки и редакторы исходного текста программ по умолчанию настроены на моноширинные шрифты. Это сделано для удобства работы программистов: улучшается удобочитаемость кода.

Шрифты этого семейства отличаются тем, что все буквы имеют одинаковую ширину.

В HTML несколько тегов отображают текст моноширинным шрифтом.

Один из них – тег физического форматирования текста `<pre>` – кроме этого, отображает текст в браузере со всеми пробелами между словами (по умолчанию браузеры воспринимают несколько пробелов, следующих друг за другом, как один).

Тег `<u>` предназначен для подчеркивания текста, а тег `<strike>` для его перечеркивания. Это теги физического форматирования текста, и, кроме визуального оформления, никакой смысловой нагрузки они не несут.

Фрагмент кода:

```
<p> <u> ПОДЧЕРКНУТЫЙ </u> и <strike> ПЕРЕЧЕРКНУТЫЙ </strike> текст. </p>
```

Результат:

ПОДЧЕРКНУТЫЙ и ~~ПЕРЕЧЕРКНУТЫЙ~~ текст

2.3 Цвет текста и фон страницы

Фоновый цвет страницы и цвет текста на всей странице можно изменить. Задать их можно при помощи атрибутов *bgcolor* и *text* тега `<body>`.

Значения этих свойств могут задаваться следующими способами:

1) с помощью ключевых слов *aqua* (голубой), *black* (черный), *blue* (синий), *fuchsia* (фуксия), *gray* (серый), *green* (зеленый), *lime* (лайм), *maroon* (бордовый), *navy* (темно-синий), *olive* (оливковый), *orange* (оранжевый), *purple* (пурпурный), *red* (красный), *silver* (серебряный), *teal* (сине-зеленый), *white* (белый), *yellow* (желтый) и др.;

2) на основе цветовой модели **RGB** путем задания трех цветовых компонентов. Числовые значения компонентов могут быть заданы в десятичной нотации в целыми числами от 0 до 255 либо в процентах от 0% до 100% либо в шестнадцатеричной нотации целыми числами от 0 до F либо 00 до FF. При использовании десятичной нотации компоненты представляются как аргументы функции **RGB** (т. е. пишутся после ключевого слова `rgb` будучи взятыми в круглые скобки и разделенными запятыми). При использовании шестнадцатеричной нотации компоненты пишутся без деления в виде трех либо шести шестнадцатеричных цифр, которым предшествует решетка (#); если используются три цифры, то считается, что каждая из них задает как старший, так и младший полубайт.

Фрагмент кода:

```
<h3 style="background-color: rgb (127, 255, 127)"> rgb (127, 255, 127)</h3>
```

```
<h3 style="background-color: rgb (50%, 100%, 50%)"> rgb (50%, 100%, 50%)</h3>
```

```
<h3 style="background-color: #00FF00"> зеленый: #00FF00</h3>
```

```
<h3 style="background-color: green"> зеленый: green </h3>
```

Результат:

rgb(127, 255, 127)

rgb(50%, 100%, 50%)

зеленый: #00FF00

зеленый: green

Замечание. Цвета, заданные в теге <body>, действуют до парного тега </body>, т. е. до конца документа.

2.4 Оформление цитат

Возможно, в процессе создания сайта, может понадобиться вставить цитату на страницу.

Цитата – это дословная выдержка текста из какого-либо произведения. Желательно, чтобы цитируемый текст идентифицировался читающим лицом как вставленный. Для этого цитату необходимо оформить соответствующим образом.

В HTML для этой цели существует специальный тег – <blockquote>. Текст, заключенный в этот тег, оформляется следующим образом: сверху и снизу от основного текста образуются промежутки, а слева и справа отступы (примерно по 40 пикселей).

Тег <q> применяется для заключения текста в кавычки. Этот тег можно использовать при оформлении цитат или при употреблении слов и выражений, которые имеют переносный смысл.

Фрагмент кода:

```
<p> Как сказал Эдвард Теллер: <blockquote> <q> Сегодняшняя наука – это технология завтрашнего дня. </q> </blockquote> </p>
```

Результат:

Как сказал Эдвард Теллер:

“ Сегодняшняя наука - это технология завтрашнего дня. ”

Аббревиатура – принятое на письме сокращение слов или слово, составленное из нескольких частей других слов (ВУЗ, НИИ, исполком и др.). При употреблении в тексте какого-либо акронима или аббревиатуры следует воспользоваться тегом <abbr>. С помощью атрибута *title* можно задать всплывающую подсказку с расшифровкой аббревиатуры. При этом поисковые роботы индексируют именно полный вариант расшифровки, определенный в атрибуте *title*. Чтобы отличать аббревиатуры от обычного текста, они подчеркиваются пунктирной линией.

Фрагмент кода:

```
<p> Стремительное развитие <abbr title="World Wide Web"> WWW </abbr> началось в середине 90-х годов XX века. </p>
```

Результат:

Стремительное развитие WWW началось в середине 90-х годов XX века.

World Wide Web

3 Создание списков

В HTML различают два основных типа списков: нумерованные и маркированные.

В нумерованном списке пункты обозначены цифрами или буквами, а в маркированном – маркерами: закрашенный/незакрашенный круг, квадрат.

3.1 Нумерованный список

Нумерованный список создает тег ``. Тег `` создает пункт списка. Если для тега `` задать стиль, то он будет применен ко всем пунктам списка.

С помощью атрибута *type* тега `` устанавливают тип нумерации: заглавные или строчные латинские буквы, большие или маленькие римские цифры, или арабские цифры.

С помощью атрибута *start* устанавливают первое число/букву, с которого начинается нумерация.

Фрагмент кода:

```
<ol>
```

```
<li> Пункты обозначены
```

```
<li> Арабскими цифрами
```

```
<li> Эта нумерация используется по умолчанию </ol>
```

```
<ol type="a">
```

```
<li> Пункты обозначены
```

```
<li> Строчными латинскими буквами </ol>
```

```
<ol type="I" start="3">
```

```
<li> Пункты обозначены
```

```
<li> Большими римскими цифрами
```

```
<li> Нумерация начинается с цифры III </ol>
```

Результат:

1. Пункты обозначены
2. Арабскими цифрами
3. Эта нумерация используется по умолчанию

- a. Пункты обозначены
- b. Строчными латинскими буквами

- III. Пункты обозначены
- IV. Большими римскими цифрами
- V. Нумерация начинается с цифры III

3.2 Маркированный список

Маркированный список создает тег ``. Тег `` создает пункт списка. Если для тега `` задать стиль, то он будет применен ко всем пунктам списка.

С помощью атрибута *type* тега `` устанавливают тип маркера: незакрашенный круг, закрашенный круг или квадрат.

Фрагмент кода:

```
<ul>
```

```
<li> Маркером пунктов списка
```

```
<li> Является закрашенный круг
```

```
<li> Этот тип маркера используется по умолчанию </ul>
```

```
<ul type="circle">
```

```
<li> Маркером пунктов списка
```

```
<li> Является
```

```
<li> Незакрашенный круг </ul>
```

```
<ul type="square">
```

```
<li> Маркером пунктов списка
```

```
<li> Является
```

```
<li> Квадрат </ul>
```

Результат:

- Маркером пунктов списка
- Является закрашенный круг
- Этот тип маркера используется по умолчанию

- Маркером пунктов списка
- Является
- Незакрашенный круг

- Маркером пунктов списка
- Является
- Квадрат

Кроме нумерованного и маркированного списков существуют так называемые списки определений.

Список определений представляет собой список из терминов и список из определений этих терминов.

Такой список образуют теги <dl>, <dt> и <dd>.

Тег <dl> создает список; тег <dt> задает термин, а тег <dd> задает определение заданного термина.

Фрагмент кода:

```
<dl>
<dt> PHP </dt>
  <dd> Язык программирования для создания веб приложений. </dd>
<dt> JavaScript </dt>
  <dd> Язык написания программ, исполняемых Вашим браузером. </dd>
</dl>
```

Результат:

```
PHP
  Язык программирования для создания веб приложений.
JavaScript
  Язык написания программ, исполняемых Вашим браузером.
```

4 Создание таблиц

Таблицы довольно широко применяются при создании веб-страниц: они служат не только способом представления и упорядочения данных, но и являются инструментом форматирования страниц - их используют для создания каркаса страницы – применяют табличную верстку.

При табличной верстке создается таблица, занимающая в основном всю ширину окна браузера, в каждой ячейке которой располагается тот или иной элемент (или элементы) страницы.

Таблица создается с помощью тега `<table>`, который открывает таблицу. Обязательный тег `</table>` информирует браузер о завершении таблицы.

Любая таблица состоит из столбцов и строк. Тег `<tr>` создает строку, а тег `<td>` ячейку в строке. Соответственно сколько ячеек включено в строку, столько будет и столбцов в таблице.

Тег `<th>` также создает ячейку. Его отличие от тега `<td>` в том, что ячейка, созданная тегом `<th>`, является ячейкой – заголовком: ее содержимое располагается по центру, а если это текст, то браузер отображает его жирным шрифтом.

Содержимое ячейки, созданной тегом `<td>` по умолчанию располагается в ее левой части.

Тег `<caption>` создает заголовок таблицы, он располагается внутри тега `<table>` – сразу после открывающего тега. Заголовок по умолчанию располагается над таблицей и выравнивается по ее центру.

Фрагмент кода:

```
<table border="1" width="700px">
  <caption> Заголовок таблицы </caption>
<tr>
  <td> 1-я ячейка 1-ой строки </td>
  <td> 2-я ячейка 1-ой строки </td> </tr>
<tr>
  <th> 1-я ячейка 2-ой строки </th>
  <th> 2-я ячейка 2-ой строки </th> </tr>
</table>
```

Результат:

Заголовок таблицы

1-я ячейка 1-ой строки	2-я ячейка 1-ой строки
1-я ячейка 2-ой строки	2-я ячейка 2-ой строки

Для выравнивания таблицы используется атрибут *align* тега <table>.

При помощи атрибут *align* можно разместить таблицу в левой или правой (*align="left"* и *align="right"*) части окна браузера (родительского элемента) или по его центру (*align="center"*).

Выравнивание содержимого строк (тег <tr>) и ячеек (тег <td>) по горизонтали также осуществляется при помощи атрибута *align*, а по вертикали при помощи атрибута *valign*:

– атрибут *align* принимает значения *left*, *right*, *center* и *justify*, которые задают выравнивание содержимого строк и ячеек по их левому, правому краю, по центру и по ширине соответственно;

– атрибут *valign*, принимающий значения *top*, *bottom* и *middle*, задает выравнивание содержимого строк и ячеек по их верхнему, нижнему краю и по середине соответственно.

Атрибут *align* служит также для выравнивания заголовка (тег <caption>) по горизонтали и определения его расположения – над таблицей или под ней.

По умолчанию содержимое ячеек расположено по левому краю по горизонтали, и выравнено по середине в вертикальной плоскости.

По умолчанию размеры (высота и ширина) и таблицы, и ячеек меняются в зависимости от размеров их содержимого.

Но высоту и ширину как таблицы, так и отдельных ее ячеек, можно задать явно – при помощи атрибутов *hieght* и *width* соответственно.

Тег <tr>, создающий строку таблицы, не имеет атрибутов *hieght* и *width*. Высота строки соответствует высоте ячеек, расположенных в ней. А ширина строки равна ширине таблицы.

Значения высоты и ширины задаются в пикселях или в процентах относительно свободного пространства. Для задания значений указывается целое положительное число (в этом случае размер будет задан в пикселях) или целое положительное число с символом %.

Если содержимое таблицы или ячейки превышает заданные размеры – они будут проигнорированы браузером, а новые размеры автоматически подобраны в соответствии с размерами содержимого.

Фрагмент кода:

```
<table border="1" width="300px" align="center">
<tr>
  <td height="70px"> Ячейка 1 </td>
  <td valign="top"> Ячейка 2 </td> </tr>
<tr>
  <td width="150px" valign="bottom"> Ячейка 3 </td>
  <td height="50px" align="right"> Ячейка 4 </td> </tr>
```

```
<tr align="center">
  <td> Ячейка 5</td>
  <td height="30px"> Ячейка 6 </td> </tr>
</table>
```

Результат:

Ячейка 1	Ячейка 2
Ячейка 3	Ячейка 4
Ячейка 5	Ячейка 6

Замечание. Указывая ширину одной из ячеек столбца, задается ширина всего столбца; и, указывая высоту одной из ячеек строки, – высоту всей строки.

Таблица и каждая ее ячейка имеют свои собственные границы, которые по умолчанию не видны.

Атрибут *border* тега `<table>` позволяет сделать границы видимыми и задать их толщину. При этом будут отображены границы и вокруг таблицы, и вокруг каждой ячейки.

Толщина границы (или рамки) задается в пикселях. В качестве значения атрибута *border* указывается целое положительное число. Если атрибут *border* указан без значения, толщина рамки составит 1 пиксель.

Толщина границы задается только для таблицы. Толщина рамки вокруг ячеек всегда составляет 1 пиксель (либо отсутствует).

По умолчанию граница отображается с эффектом трехмерности и имеет черный цвет.

Атрибут *bordercolor* задает цвет границы и устраняет эффект трехмерности. Атрибут может использоваться для задания цвета границы таблицы (тег `<table>`), строки (тег `<tr>`) или ячейки (тег `<td>`).

Замечание. Атрибут *bordercolor* поддерживается не всеми браузерами и поэтому не рекомендуется его использовать.

Фрагмент кода:

```
</table>
<table width="500px" border="3px">
<tr>
  <td> Толщина границы таблицы составляет 3 пикселя. </td>
```

```

        <td> Ячейки имеют границы толщиной 1 пиксель! </td> </tr>
<tr>
        <td>Ячейка </td>
        <td>Ячейка </td> </tr>
</table>

```

Результат:

Толщина границы таблицы составляет 3 пикселя.	Ячейки имеют границы толщиной 1 пиксель!
Ячейка	Ячейка

Граница таблицы и рамка вокруг ячеек могут быть отображены частично. Атрибут *frame* тега `<table>` указывает, где рисовать границу таблицы. Атрибут *rules* указывает, как отобразить границы ячеек.

Фрагмент кода:

```

<table height="50px" width="500px" frame="hsides" rules="rows">
<tr>
        <td> Установлены </td>
        <td> горизонтальные </td>
        <td> границы таблицы </td> </tr>
<tr>
        <td> и </td>
        <td> отображены границы </td>
        <td> между строками </td> </tr>
</table>

```

Результат:

Установлены	горизонтальные	границы таблицы
и	отображены границы	между строками

При форматировании таблиц в HTML, для наглядного представления некоторой информации и удобного ее восприятия, может оказаться полезным использование отступов внутри и снаружи ячеек.

Внутренние отступы – от границ ячеек до их содержимого, задаются атрибутом *cellpadding* тега <table>.

Внешние отступы – расстояние между границами соседних ячеек и расстояние от границ ячеек до границы таблицы, задаются атрибутом *cellspacing* тега <table>.

В качестве значений атрибутов указываются целые положительные числа, которые задают расстояние в пикселях.

Фрагмент кода:

```
<table border="1px" width="500" cellpadding="10" cellspacing="25">
<tr>
  <td> Расстояние от содержимого ячеек </td>
  <td> до их границ составляет 10 пикселей </td> </tr>
<tr>
  <td> Расстояние между ячейками и от ячеек </td>
  <td> до границы таблицы составляет 25 пикселей </td> </tr>
</table>
```

Результат:

Расстояние от содержимого ячеек	до их границ составляет 10 пикселей
Расстояние между ячейками и от ячеек	до границы таблицы составляет 25 пикселей

При оформлении и форматировании таблиц в HTML нередко требуется объединить рядом расположенные ячейки. И если такая необходимость возникла, то следует воспользоваться атрибутами *colspan* и *rowspan* тега <td>.

Атрибут *colspan* указывает количество ячеек, которые будут объединены по горизонтали, а атрибут *rowspan* устанавливает количество ячеек, объединяемых по вертикали.

Оба атрибута имеют смысл, если таблица состоит из нескольких строк.

Фрагмент кода:

```
<table border="1px" width="500">
<tr>
  <td colspan="2"> 1 </td>
  <td rowspan="2"> 2 </td> </tr>
<tr>
  <td> 3 </td>
  <td> 4 </td> </tr>
</table>
```

Результат:

1		2
3	4	

В HTML есть возможность задать общий фон как для всей таблицы, так и фон для отдельно выделенной ячейки.

Атрибут *background* тега `<table>` задает изображение, которое будет фоновым рисунком таблицы. В качестве значения атрибута указывается адрес файла с изображением – абсолютный или относительный путь к файлу.

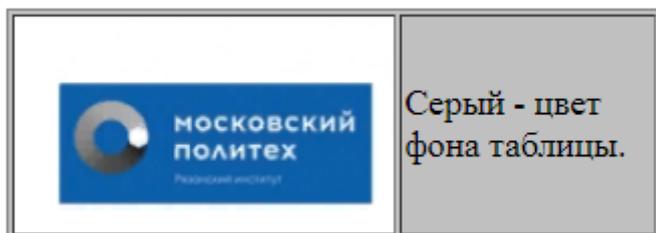
Атрибут *bgcolor* тега `<table>` задает цвет фона таблицы.

При помощи этих же атрибутов задают фоновый рисунок и цвет фона для любой ячейки таблицы (тег `<td>`).

Фрагмент кода:

```
<table width="25%" border="1" bgcolor="#c0c0c0">
<tr>
  <td width="60%" height="100" background = "Логотип.jpg"> </td>
  <td width="40%"> Серый - цвет фона таблицы. </td> </tr>
</table>
```

Результат:



Рассмотрим теги, которые применяют при редактировании сразу нескольких элементов таблицы. Эти теги можно разделить на две группы.

К первой группе относятся теги `<col>` и `<colgroup>`. Они практически идентичны и служат для задания некоторых свойств и изменения характеристик одной или нескольких колонок таблицы.

Один из этих тегов располагают сразу после тега `<table>`. Допустим это тег `<col>`.

При помощи атрибута *span* тега `<col>` указывают количество колонок, к которым будут применены атрибуты *align*, *valign* или *width* (выравнивание содержимого ячеек колонок по горизонтали, вертикали или задание ширины колонок).

Если атрибут *span* в теге `<col>` отсутствует, то будут изменены характеристики только первой колонки таблицы. При втором использовании тега `<col>` задаются свойства для следующих колонок таблицы и т.д.

Фрагмент кода:

```
<table width="700px" border="1">
  <col width="250px">
  <col span="2" width="70px">
<tr>
  <td> 1 </td>
  <td> 2 </td>
  <td> 3 </td>
  <td> 4 </td>
  <td> 5 </td> </tr>
</table>
```

Результат:

1	2	3	4	5
---	---	---	---	---

Ко второй группе тегов относятся также практически идентичные между собой теги <thead>, <tbody> и <tfoot>.

Любой из них можно применить для изменения некоторых свойств одной или нескольких строк таблицы: это опять же выравнивание содержимого ячеек в строках по горизонтали и по вертикали – при помощи атрибутов *align* и *valign* соответственно; и задание фонового цвета ячеек при помощи атрибута *bgcolor*.

При использовании этих тегов следует знать несколько нюансов, которые как раз и определяют различия между ними.

Теги <thead> и <tfoot> должны быть расположены перед тегом <tbody>, сразу после открывающего таблицу тега <table>. Строки, помещенные в тег <thead>, представлены вверху таблицы, а строки, заключенные в тег <tfoot>, будут расположены внизу таблицы. Оба тега могут быть применены только один раз в пределах одной таблицы.

Тег <tbody> допускается использовать несколько раз внутри тега <table>.

Фрагмент кода:

```
<table width="600" border="1">
<thead bgcolor="#99FFFF" align="center">
  <tr> <td width="350"> 1 </td> <td> 2 </td> </tr>
</thead>
<tfoot bgcolor="red">
  <tr> <td> 3 </td> <td> 4 </td> </tr>
  <tr> <td> 5 </td> <td> 6 </td> </tr>
</tfoot>
<tbody bgcolor="#00FFFF">
  <tr> <td> 7 </td> <td> 8 </td> </tr>
</tbody>
</table>
```

Результат:

1	2
7	8
3	4
5	6

Библиографический список

1. Дронов, В. HTML 5, CSS 3 и Web 2.0. Разработка современных Webсайтов. – СПб.: БХВ-Петербург, 2011. – 416 с.
2. Кисленко, Н.П. HTML. Самое необходимое. – СПб.: БХВ-Петербург, 2012. – 352 с.
3. Хеник, Б. HTML и CSS: путь к совершенству / Б. Хеник. – СПб.: Питер, 2011. – 336 с.

Учебное издание

Арабчикова Юлия Ивановна
Асаева Татьяна Александровна

ИНТЕРНЕТ-ТЕХНОЛОГИИ
Часть 1

Учебно-методическое пособие

Подписано в печать _____. Тираж 5 экз.
Рязанский институт (филиал) Московского политехнического университета
390000, г. Рязань, ул. Право-Лыбедская, 26/53