

Глава 3. Форматирование текста

Язык HTML имеет широкие возможности по форматированию текста. Правильно структурированный текст дает возможность пользователям быстро и комфортно читать информацию с экрана монитора.

При наборе текста для HTML-документа следует избегать переносов слов по слогам и выравнивания текста с помощью пробелов или табуляций. Все форматирование в HTML следует осуществлять через теги.

3.1. Перевод строки

Браузер автоматически переводит строку текста по достижении правой границы окна. Однако, иногда требуется принудительно осуществить перенос части текста на новую строку. Для этого служит специальный тег.

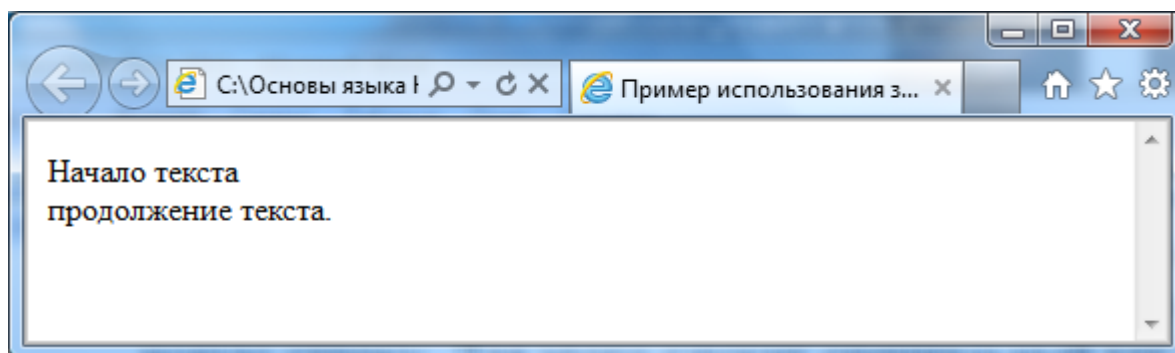
Синтаксис

<i>Начало текста</i>
 <i>продолжение текста</i>
--

Содержимое файла sample31-1.html:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title> Пример использования заголовков</title>
    <meta http-equiv="Content-Type" content="text/html;
charset=windows-1251">
  </head>
  <body>
    Начало текста<br>продолжение текста.
  </body>
</html>
```

Результат на экране:



3.2. Запрет перевода строки

Для запрета автоматического перевода строки браузером служит следующий тег.

Синтаксис

<i>Начало текста</i> <nobr> <i>продолжение текста</i> </nobr> <i>окончание текста</i>

Содержимое файла sample32-1.html:

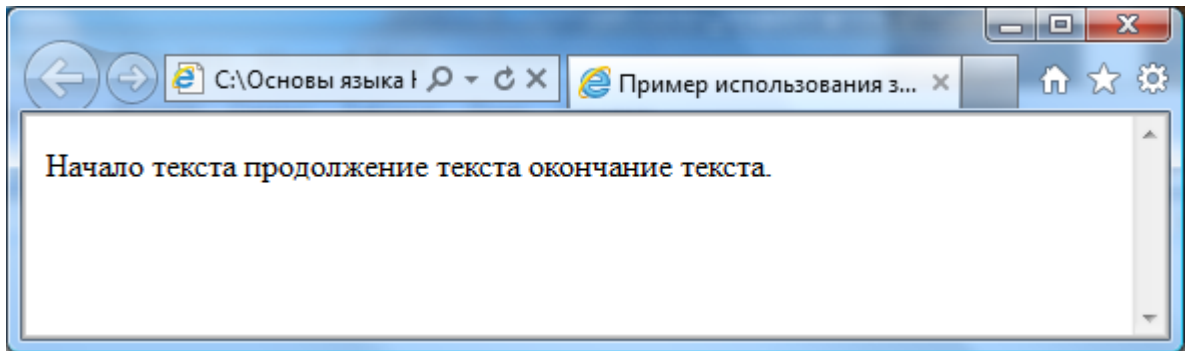
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
```

```

<head>
<title> Пример использования заголовков</title>
<meta http-equiv="Content-Type" content="text/html;
charset=windows-1251">
</head>
<body>
Начало текста <nobr>продолжение текста</nobr> окончание текста.
</body>
</html>

```

Результат на экране:



3.3. Горизонтальная линия

Для отделения одной части текста от другой горизонтальной линией необходимо использовать тег.

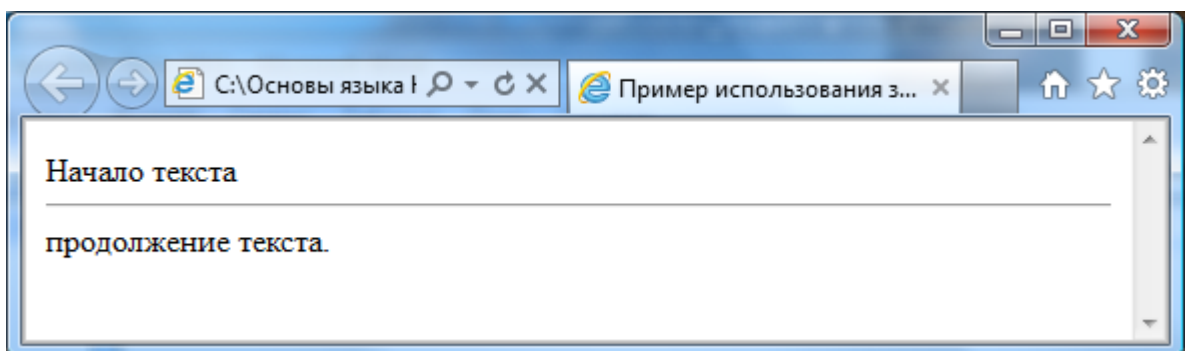
Синтаксис
<i>Начало текста</i> <hr> <i>окончание текста</i>

```

Содержимое файла sample33-1.html:
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<title> Пример использования заголовков</title>
<meta http-equiv="Content-Type" content="text/html;
charset=windows-1251">
</head>
<body>
Начало текста<hr>продолжение текста.
</body>
</html>

```

Результат на экране:



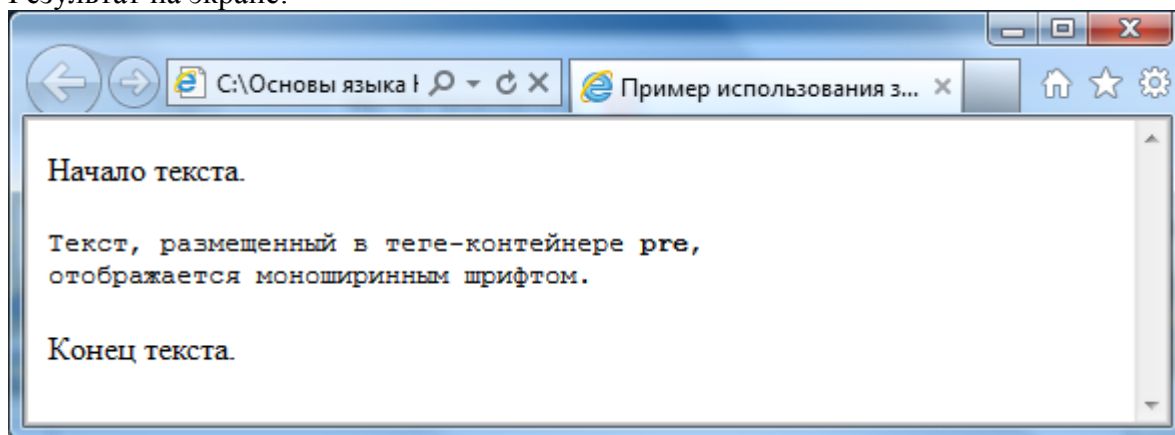
3.4. Предварительное форматирование текста

В отдельных случаях на странице необходимо сохранить форматирование текста, созданное текстовым редактором в неизменном виде. Для реализации такой возможности существует тег-контейнер `<pre>`. Браузер не будет дополнительно обрабатывать эту часть текста. Особенностью данного тега является то, что текст выводится моноширинным шрифтом.

Синтаксис
<code><pre></code> <i>текст</i> <code></pre></code>

```
Содержимое файла sample34-1.html:
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title> Пример использования заголовков</title>
    <meta http-equiv="Content-Type" content="text/html;
charset=windows-1251">
  </head>
  <body>
    Начало текста.
    <pre>
    Текст, размещенный в теге-контейнере <b>pre</b>,<br>отображается моноширинным
    шрифтом.
    </pre>
    Конец текста.
  </body>
</html>
```

Результат на экране:



3.5. Выделение блока

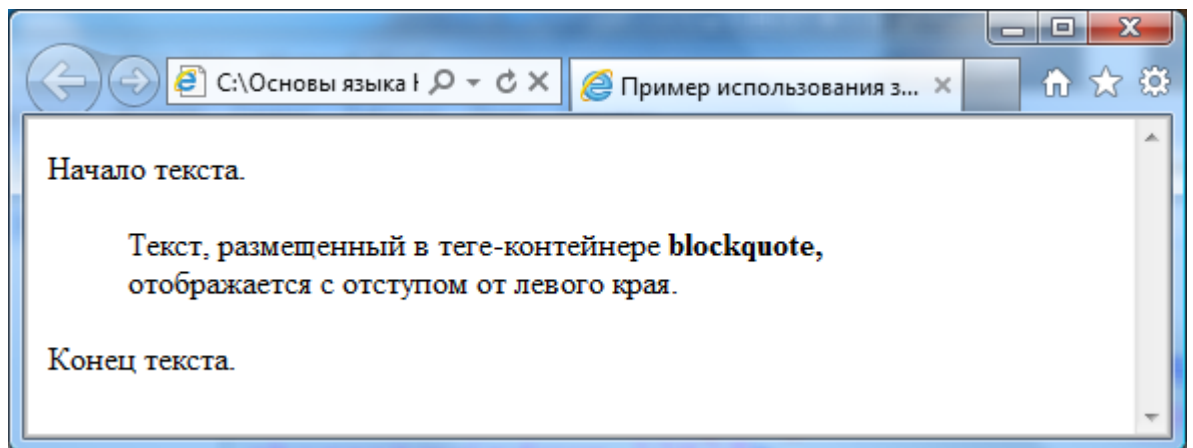
Часть текста на странице можно выделить отдельным блоком, все строки которого будут отображаться с некоторым отступом от левого края. Размер отступа зависит от браузера.

Синтаксис
<code><blockquote></code> <i>текст</i> <code></blockquote></code>

Содержимое файла sample35-1.html:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title> Пример использования заголовков</title>
    <meta http-equiv="Content-Type" content="text/html;
charset=windows-1251">
  </head>
  <body>
Начало текста.
<blockquote>
Текст, размещенный в теге-контейнере <b>blockquote</b>,<br>
отображается с отступом от левого края.
</blockquote>
Конец текста.
</body>
</html>
```

Результат на экране:



3.6. Заголовки

Заголовок кратко описывает содержимое фрагмента страницы, которому он предшествует. Заголовки отображаются жирным шрифтом.

Заголовки имеют 6 уровней с 1-го по 6-й и различаются только размером шрифта.

Синтаксис

`<hn>текст заголовка</hn>` , где n - номер уровня с 1 по 6.

Содержимое файла sample36-1.html:

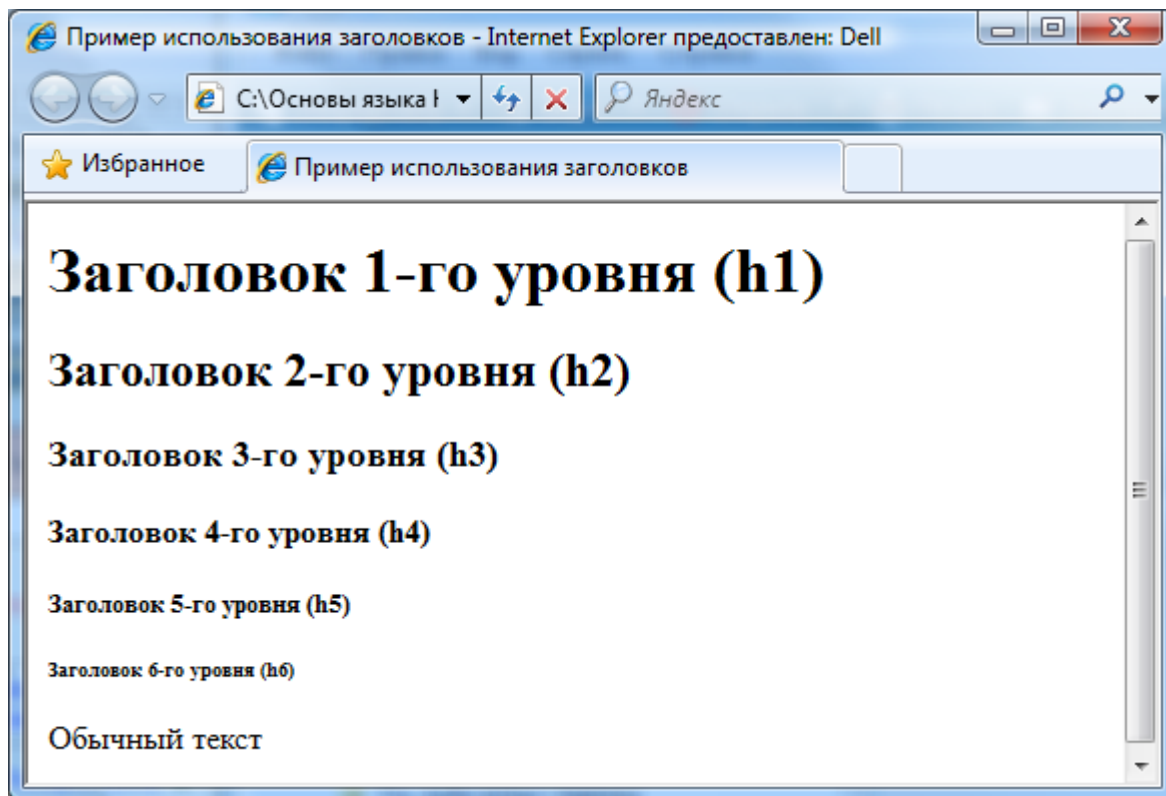
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title> Пример использования заголовков</title>
    <meta http-equiv="Content-Type" content="text/html;
charset=windows-1251">
  </head>
  <body>
<h1>Заголовок 1-го уровня (h1)</h1>
<h2>Заголовок 2-го уровня (h2)</h2>
<h3>Заголовок 3-го уровня (h3)</h3>
<h4>Заголовок 4-го уровня (h4)</h4>
```

```

<h5>Заголовок 5-го уровня (h5)</h5>
<h6>Заголовок 6-го уровня (h6)</h6>
Обычный текст
</body>
</html>

```

Результат на экране:



3.7. Параграф

Основной структурой текста в HTML является параграф (абзац). Каждый параграф начинается с новой строки. Для выделения абзаца служит тег-контейнер `<p>`. В теге можно указать способ выравнивания текста.

Синтаксис
<code><p align=выравнивание>текст параграфа</p></code>

выравнивание	Пояснение
left	выравнивание текста по левому краю
center	выравнивание текста по центру
right	выравнивание текста по правому краю
justify	выравнивание текста по ширине

Таблица 3. Выравнивание текста

Параграф не может быть вложенным, т.е. нельзя использовать тег для создания подпараграфа. Однако, решение подобной задачи можно организовать с помощью списков, которые будут рассмотрены ниже.

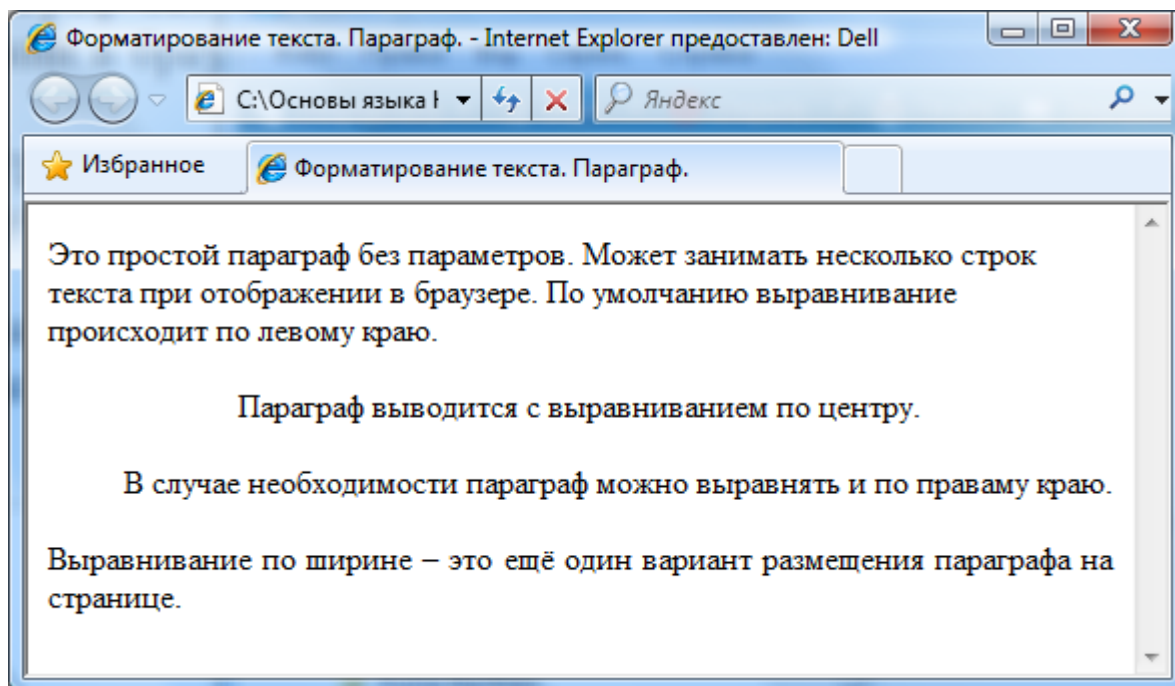
Особенности использования параметра выравнивания заметны при изменениях размера окна браузера.

Пример:

Содержимое файла sample37-1.html:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>Форматирование текста. Параграф.</title>
    <meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
  </head>
  <body>
    <p>Это простой параграф без параметров. Может занимать несколько строк текста при отображении в браузере. По умолчанию выравнивание происходит по левому краю.</p>
    <p align=center>Параграф выводится с выравниванием по центру.</p>
    <p align=right>В случае необходимости параграф можно выравнивать и по правому краю.</p>
    <p align=justify>Выравнивание по ширине – это ещё один вариант размещения параграфа на странице.</p>
  </body>
</html>
```

Результат на экране:



3.8. Увеличенный шрифт

Текст внутри тега-контейнера будет отображаться больше на 1 пункт, чем основной шрифт.

Синтаксис
<code><big>текст</big></code>

3.9. Уменьшенный шрифт

Текст внутри тега-контейнера будет отображаться меньше на 1 пункт, чем основной шрифт.

Синтаксис

<code><small>текст</small></code>

3.10. Жирный шрифт

Текст внутри тега-контейнера будет отображаться жирным шрифтом.

Синтаксис

<code>текст</code>

3.11. Наклонный шрифт (курсив)

Текст внутри тега-контейнера будет отображаться курсивом.

Синтаксис

<code><i>текст</i></code>

3.12. Подчеркнутый шрифт

Текст внутри тега-контейнера будет подчеркнут.

Синтаксис

<code><u>текст</u></code>

3.13. Перечеркнутый шрифт

Текст внутри тега-контейнера будет перечеркнут.

Синтаксис

<code><s>текст</s></code>

3.14. Верхний индекс

Текст внутри тега-контейнера будет отображаться верхним индексом.

Синтаксис

<code><sup>текст</sup></code>

3.15. Нижний индекс

Текст внутри тега-контейнера будет отображаться нижним индексом.

Синтаксис

<code><sub>текст</sub></code>

Пример:

Содержимое файла sample315-1.html:

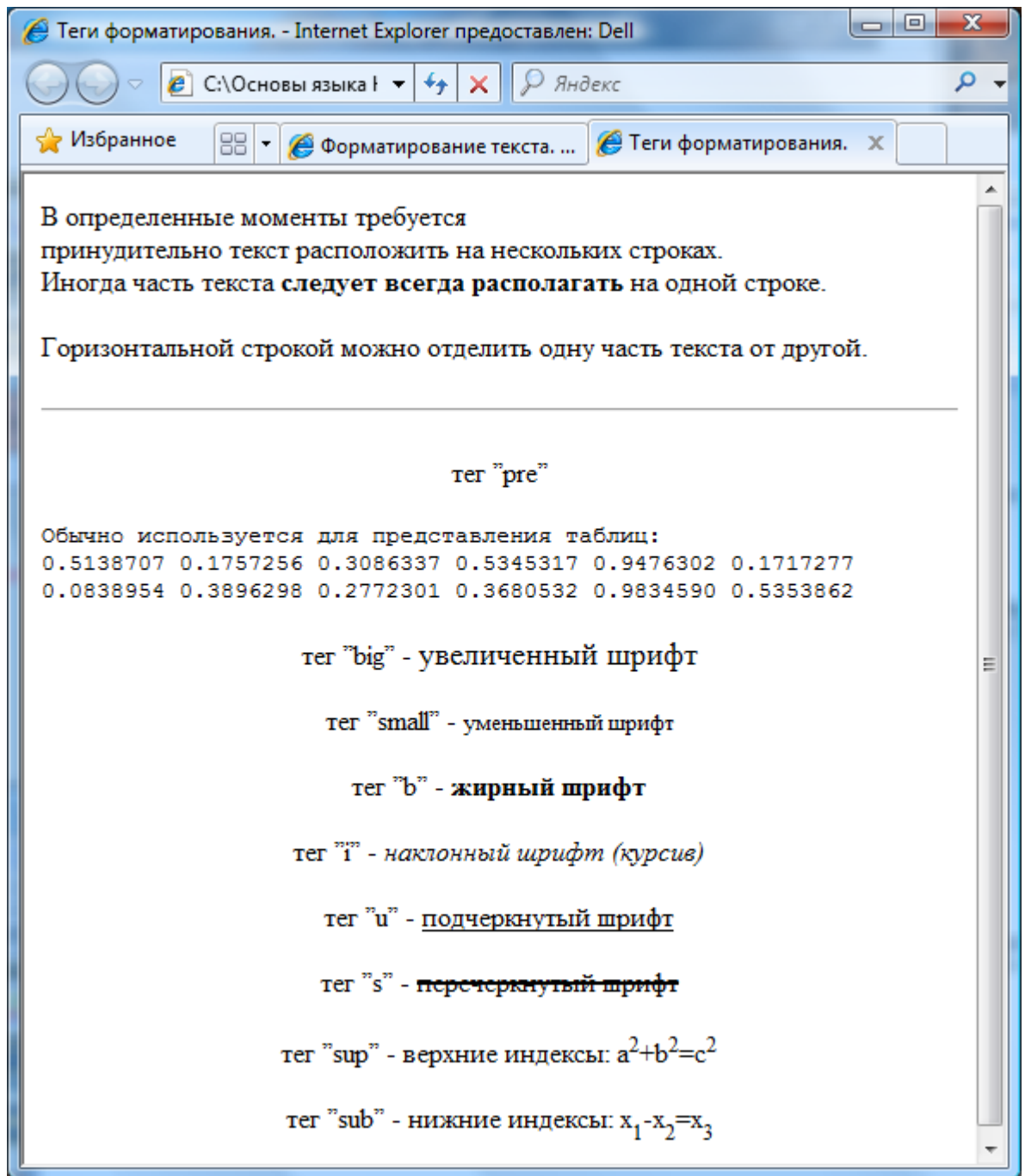
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

```

<html>
  <head>
    <title>Теги форматирования.</title>
    <meta http-equiv="Content-Type" content="text/html;
charset=windows-1251">
  </head>
  <body>
    <p>В определенные моменты требуется<br>
    принудительно текст расположить на нескольких строках.<br>
    Иногда часть текста <no>br><b>следует всегда располагать</b><no>br> на одной
    строке.</p>
    <p>Горизонтальной строкой можно отделить одну часть текста от другой.</p>
    <hr>
    <p align="center">тег "pre"</p>
    <pre>
    Обычно используется для представления таблиц:
    0.5138707 0.1757256 0.3086337 0.5345317 0.9476302 0.1717277
    0.0838954 0.3896298 0.2772301 0.3680532 0.9834590 0.5353862
    </pre>
    <p align="center">тег "big" - <big>увеличенный шрифт</big></p>
    <p align="center">тег "small" - <small>уменьшенный шрифт</small></p>
    <p align="center">тег "b" - <b>жирный шрифт</b></p>
    <p align="center">тег "i" - <i>наклонный шрифт (курсив)</i></p>
    <p align="center">тег "u" - <u>подчеркнутый шрифт</u></p>
    <p align="center">тег "s" - <s>перечеркнутый шрифт</s></p>
    <p align="center">тег "sup" - верхние индексы: a<sup>2</sup>+b<sup>2</sup>=c
    <sup>2</sup></p>
    <p align="center">тег "sub" - нижние индексы: x<sub>1</sub>-x<sub>2</sub>=x
    <sub>3</sub></p>
  </body>
</html>

```

Результат на экране:



3.16. Тип, размер и цвет шрифта

Размер шрифта на странице устанавливается браузером по умолчанию. Тег-контейнер `` позволяет изменить размер, цвет и тип шрифта.

Синтаксис
<code>текст</code>

Параметр **face** может не поддерживаться некоторыми браузерами. Следующие виды шрифтов, типичные на ПК на основе Windows:

шрифт	Пояснение
Times New Roman	шрифт Times New Roman

Arial Narrow	шрифт Arial Narrow
Arial	шрифт Arial
Courier New	шрифт Courier New
Tahoma	шрифт Verdana
Comic Sans MS	шрифт Comic Sans MS
Verdana	шрифт Verdana

Таблица 4. Стандартные шрифты

Можно указать несколько шрифтов через запятую - в этом случае для форматирования текста будет использован первый подходящий шрифт из списка, например: "Comic Sans MS, Tahoma". Таким образом, если у пользователя на компьютере нет шрифта Comic Sans MS, браузер подставит следующий в этом списке - Tahoma.

Использовать необходимо шрифты, которые установлены на компьютере пользователя, в противном случае браузер будет использовать шрифт, определенный по умолчанию (обычно Times New Roman). К стандартным шрифтам можно отнести шрифты, поставляемые с MS Windows, MS Plus, MS Office.

Параметр **size** определяет размер шрифта, начиная с цифры 1 по цифру 7 либо целое со знаком, например, "+1" или "-2". Число со знаком добавляется к текущему базовому размеру шрифта.

Параметр **color** определяет цвет шрифта. По умолчанию черный - "#000000" (black). Цвет может быть представлен как символьной нотацией, например, red, blue и т.д, так и представлен в шестнадцатичном виде - "#FF0000" (красный), "#0000FF" (синий), при этом обозначение должно быть взято в кавычки.

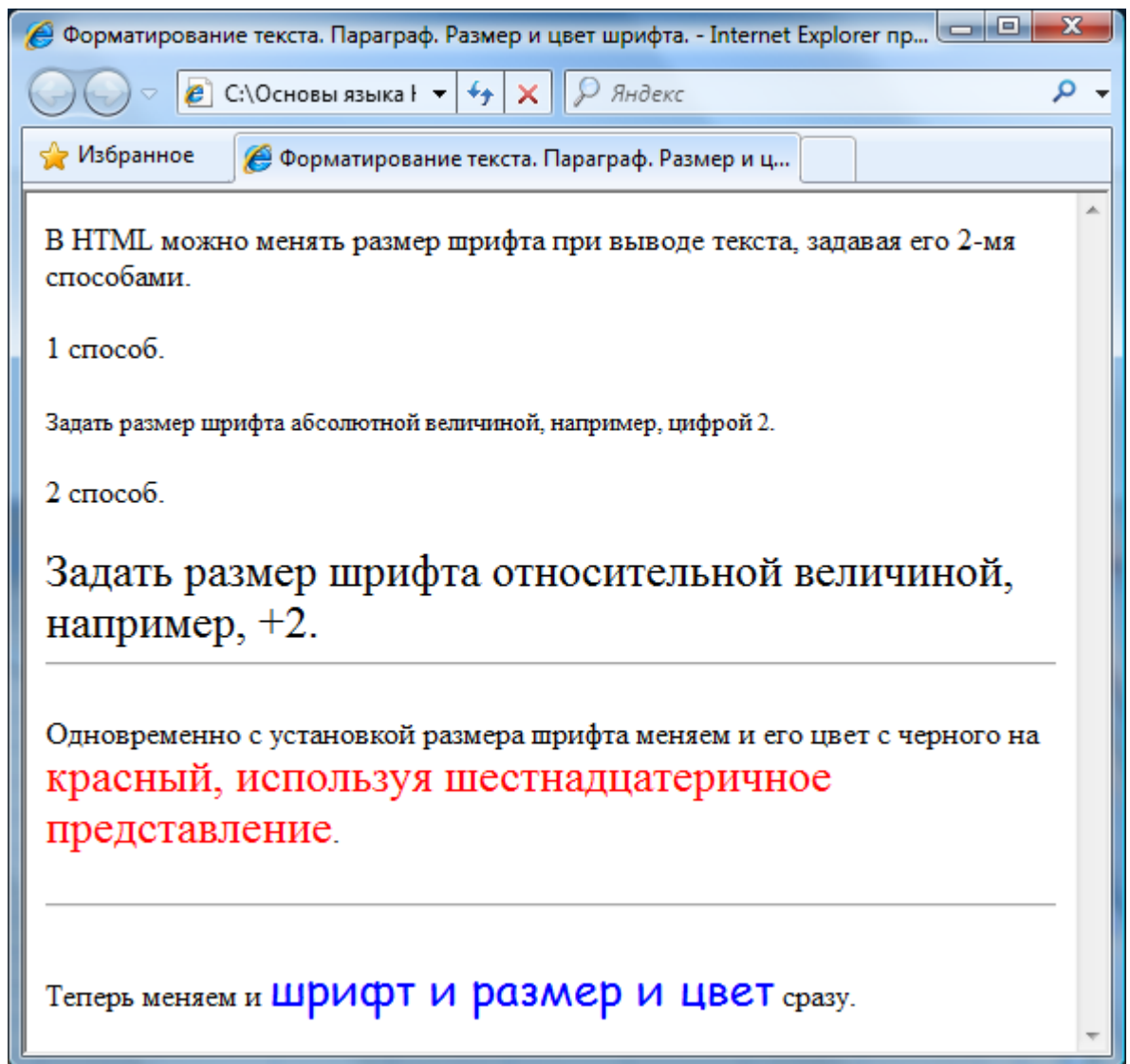
Пример:

```

Содержимое файла sample316-1.html:
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>Теги форматирования.</title>
    <meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
  </head>
  <body>
    <p>В HTML можно менять размер шрифта при выводе текста, задавая его 2-мя способами.</p>
    <p>1 способ.</p>
    <font size=2>Задать размер шрифта абсолютной величиной, например, цифрой 2.</font>
    <p>2 способ.</p>
    <font size=+2>Задать размер шрифта относительной величиной, например, +2.</font>
    <hr>
    <p>Одновременно с установкой размера шрифта меняем и его цвет с черного на <font size=+2 color="#FF0000">красный, используя шестнадцатичное представление.</font>.</p>
    <hr>
    <p>Теперь меняем и <font face="Comic Sans MS" size=+2 color="#0000FF">шрифт и размер и цвет.</font> сразу.</p>
  </body>
</html>

```

Результат на экране:



3.17. Списки

В языке HTML поддерживаются упорядоченные списки, неупорядоченные списки и списки определений. Отличаются эти разновидности списков лишь способом оформления. Перед пунктами неупорядоченных списков обычно ставятся символы-буллеты (bullets), например, точки, ромбики и т.п., в то время как пунктам упорядоченных списков предшествуют их номера или символы латинского алфавита.

3.17.1. Маркированные (Неупорядоченные) списки

Маркированный список начинается с тегом-контейнером ``. Каждый элемент списка начинается с тега ``.

Синтаксис
<pre><ul compact type=тип_маркеров> элемент 1 элемент 2 ... элемент N </pre>

Параметр **compact** делает отображение списка более компактным.
 Параметр **type** определяет тип маркеров для списка:

тип_маркеров	Пояснение
circle	○ - окружность
disc	● - диск (круг)
square	▪ - квадрат

Таблица 5. Типы маркеров списка

В качестве элемента списка могут служить параграфы, переносы строк, изображения, ссылки, другие списки, и т.д.

Не рекомендуется использовать параметр **type**, если планируется использовать вложенные списки. Следует воспользоваться возможностями браузеров заданных по умолчанию.

3.17.2. Нумерованные (упорядоченные) списки

Элементы списка маркируются с помощью чисел или букв. Начинается с тега-контейнера ****. Каждый элемент списка начинается с тега ****.

Синтаксис
<pre> <ol compact type="стиль_нумерации" start="число"> элемент 1 элемент 2 ... элемент N </pre>

Параметр **compact** делает отображение списка более компактным.

Параметр **start** определяет первое число или символ, с которого начинается нумерация пунктов.

Параметр **type** определяет стиль нумерации:

стиль_нумерации	Пояснение
"A"	заглавные буквы A, B, C ...
"a"	строчные буквы a, b, c ...
"I"	большие римские числа I, II, III ...
"i"	маленькие римские числа i, ii, iii ...
"1"	арабские числа 1, 2, 3 ...

Таблица 6. Стили нумерации

В качестве элемента списка могут служить параграфы, переносы строк, изображения, ссылки, другие списки, и т.д.

Не рекомендуется использовать параметр **type** и параметр **start**, если планируется использовать вложенные списки. Следует воспользоваться возможностями браузеров заданных по умолчанию.

3.17.3. Списки определений

Список определений начинается с тег-контейнера `<dl>`. Перед каждым термином присутствует тег `<dt>`. После термина указывается тег `<dd>`, после которого следует определение термина.

Синтаксис
<pre><dl compact> <dt>термин 1 <dd>определение термина 1 <dd>другое определение термина 1 <dt>термин 2 <dd>определение термина 2 <dd>другое определение термина 2 ... <dt>термин N <dd>определение термина N <dd>другое определение термина N </dl></pre>

Параметр `compact` делает отображение списка более компактным.

3.17.4. Вложенные списки

Элемент любого списка может содержать в себе целый список любого вида. Число уровней вложенности в принципе не ограничено, однако злоупотреблять вложенными списками все же не следует.

Вложенные списки очень удобны при подготовке разного рода планов и оглавлений.

3.17.5. Графические элементы при оформлении списков

В качестве маркеров списка можно использовать графические изображения для создания привлекательных, красиво оформленных HTML-документов. Такая возможность не предоставляется непосредственно языком HTML, а реализуется искусственно.

Как реализовать такую возможность?

Тег списка выполняет единственную задачу - указывает браузеру, что вся информация, располагаемая после данного тега должна отображаться с отступом. Теги ``, указывающие на отдельные элементы списка, обеспечивают вывод стандартных маркеров элементов списка.

Если же нам требуется построить список с графическими маркерами, то можно вообще обойтись без тегов ``. Достаточно будет перед каждым элементом списка вставить желаемое графическое изображение (работа с графическими изображениями подробно будет рассмотрена в одной из глав). Далее необходимо отделение элементов списка друг от друга. Для этого можно использовать теги абзаца `<p>` или перевода строки `
`.

Следует заметить, что подобное не рекомендуется делать согласно стандартам HTML. Однако, позволяет продемонстрировать широкие возможности языка при создании HTML-документов.

Пример:

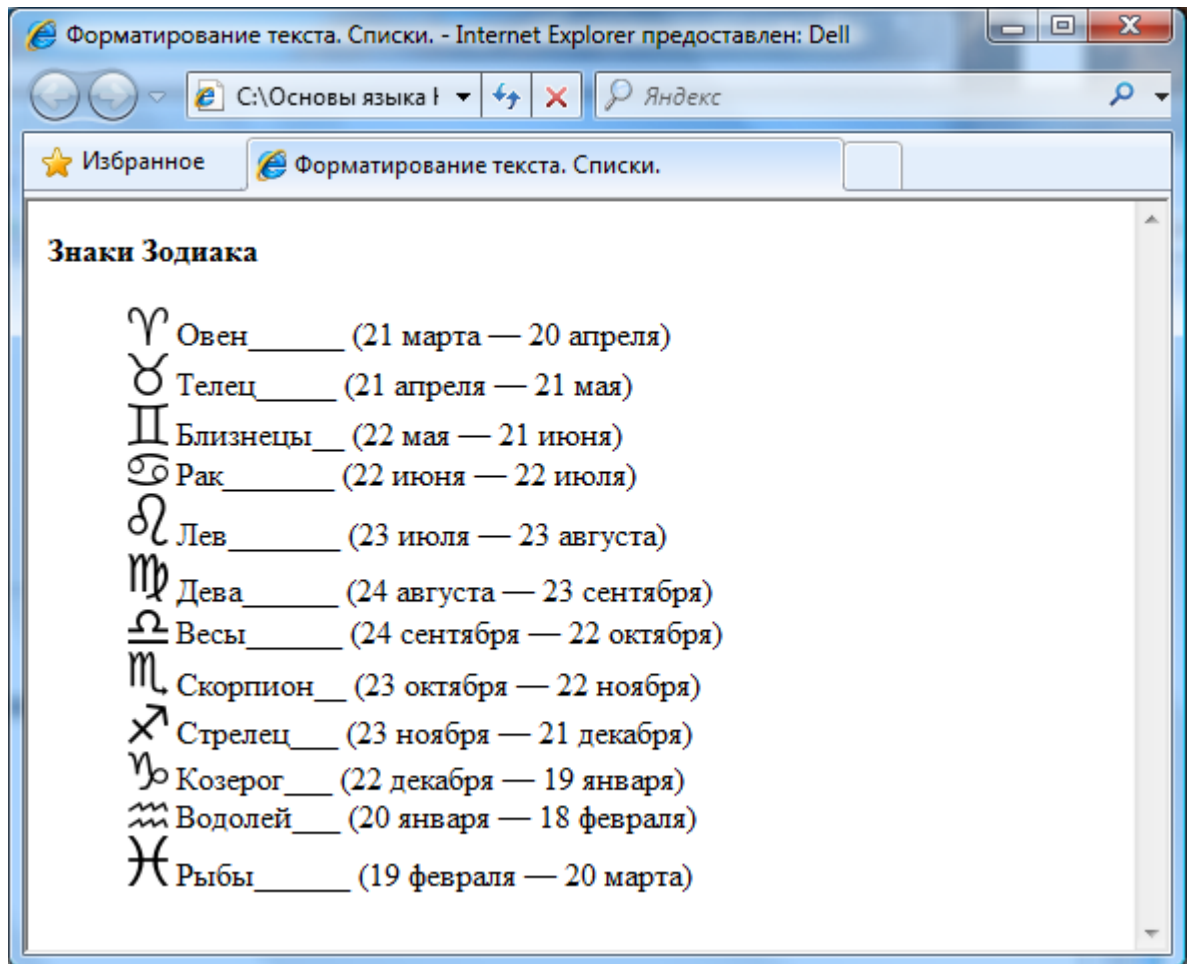
⋮ Содержимое файла sample3175-1.html: ⋮

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>Теги форматирования.</title>
    <meta http-equiv="Content-Type" content="text/html;
charset=windows-1251">
  </head>
  <body>
    <p><b>Знаки Зодиака</b></p>
    <ul >
       Овен_____ (21 марта — 20 апреля)<br>
       Телец_____ (21 апреля — 21 мая)<br>
       Близнецы_ (22 мая — 21 июня)<br>
       Рак_____ (22 июня — 22 июля)<br>
       Лев_____ (23 июля — 23 августа)<br>
       Дева_____ (24 августа — 23 сентября)<br>
       Весы_____ (24 сентября — 22 октября)<br>
       Скорпион_ (23 октября — 22 ноября)<br>
       Стрелец__ (23 ноября — 21
декабря)<br>
       Козерог__ (22 декабря — 19 января)<br>
       Водолей___ (20 января — 18 февраля)<br>
       Рыбы_____ (19 февраля — 20 марта)
    </ul>
  </body>
</html>

```

Результат на экране:



3.17.6. Пример использования списков

Пример:

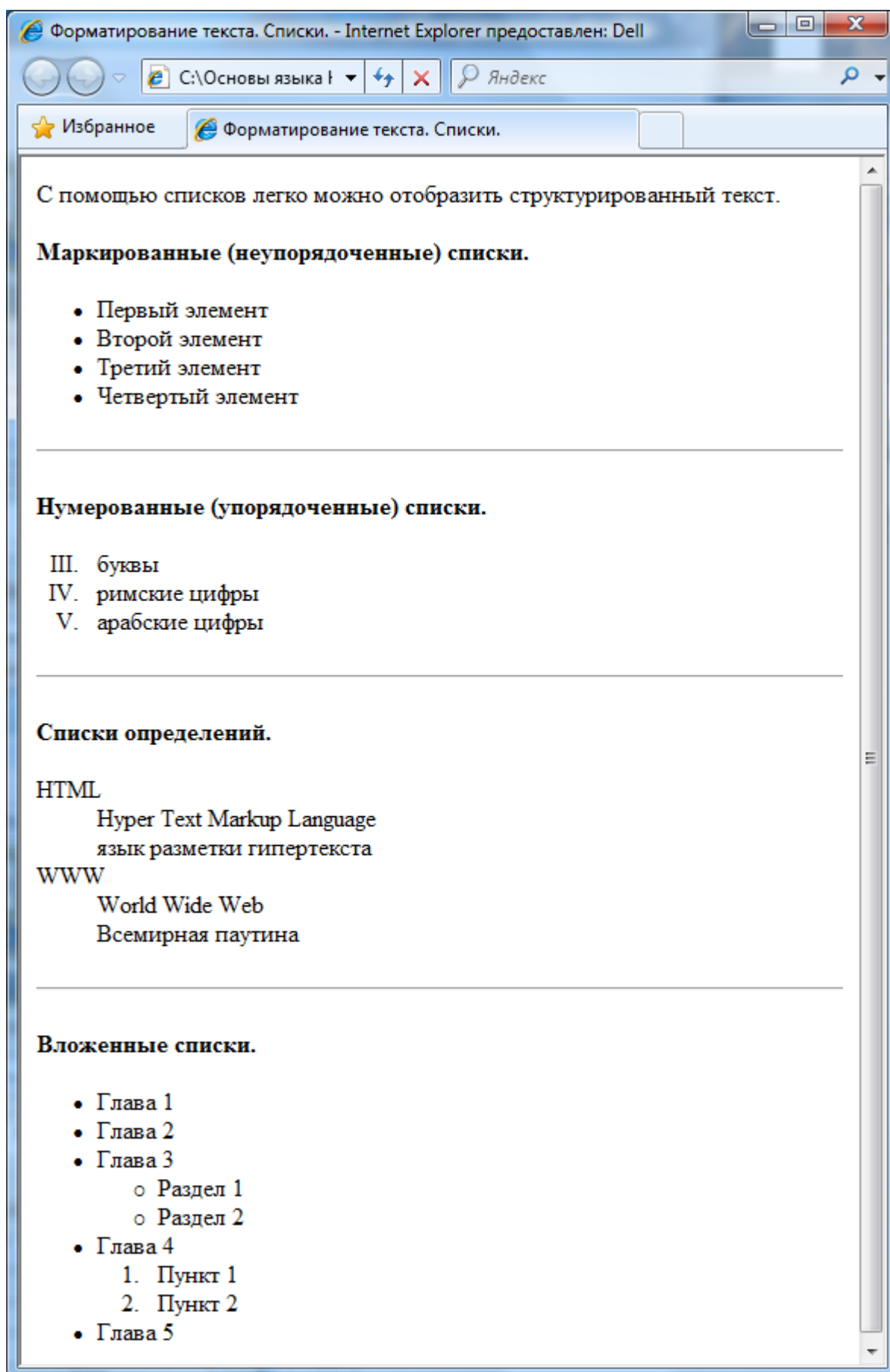
```
Содержимое файла sample3176-1.html:
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>Теги форматирования.</title>
    <meta http-equiv="Content-Type" content="text/html;
charset=windows-1251">
  </head>
  <body>
    <p>С помощью списков легко можно отобразить структурированный текст.</p>
    <p><b>Маркированные (неупорядоченные) списки.</b></p>
    <ul type="disc">
      <li>Первый элемент
      <li>Второй элемент
      <li>Третий элемент
      <li>Четвертый элемент
    </ul>
    <hr>
    <p><b>Нумерованные (упорядоченные) списки.</b></p>
    <ol type="I" start="3">
      <li>буквы
      <li>римские цифры
      <li>арабские цифры
```

```

</ol>
<hr>
<p><b>Списки определений.</b></p>
<dl compact>
  <dt>HTML
    <dd>Hyper Text Markup Language
    <dd>язык разметки гипертекста
  <dt>WWW
    <dd>World Wide Web
    <dd>Всемирная паутина
</dl>
<hr>
<p><b>Вложенные списки.</b></p>
<ul>
  <li>Глава 1
  <li>Глава 2
  <li>Глава 3
  <ul>
    <li>Раздел 1
    <li>Раздел 2
  </ul>
  <li>Глава 4
  <ol>
    <li>Пункт 1
    <li>Пункт 2
  </ol>
  <li>Глава 5
</ul>
</body>
</html>

```

Результат на экране:



3.17.7. Символьные объекты

Escape-последовательности или, точнее, символьные объекты, представляют собой метод представления специальных символов. Символьная последовательность начинается с символа "&".

Некоторые символы, такие как "<", используются в HTML в специальном значении, поэтому должен существовать способ их выражения, как символьных данных внутри самого документа. Принято использовать следующие нотации (соответствия):

Символ	Пояснение
<	<
>	>
&	&
"	"
©	©
®	®
_	 (непрерывный пробел)

Таблица 7. Примеры символьных объектов

Точка с запятой является частью escape-последовательности, но в принципе она может быть опущена. Однако одни браузеры обработают такой код HTML, а другие могут и не понять.

В escape-последовательности важен вариант написания символов, то есть различаются заглавные и строчные буквы. Например, амперсанд (&) не может быть представлен как & (escape-последовательность – не определена).

Необходимость в вышеупомянутых escape-последовательностях проистекает из синтаксиса HTML. На самом деле escape-последовательности существуют для всех символов в наборе ISO 8859-1 Latin 1.

Существуют последовательности вида &#n;, где n - позиционный код символа в десятичном исчислении (в диапазоне от 0 до 255)

Некоторые браузеры поддерживают и другие escape-последовательности, например, ™, ­, . Использование таких нотаций не рекомендуется. (Нотация ™ представляет символ, который вообще не принадлежит к ISO 8859-1 Latin 1. Вместо этого можно использовать проверенную нотацию `^{<small>TM</small>}`.

Если нужно вставить в текст один или несколько неизменяемых пробелов, для этого используется сочетание символов .

Пример:

```
Содержимое файла sample3177-1.html:
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>Теги форматирования.</title>
    <meta http-equiv="Content-Type" content="text/html;
charset=windows-1251">
  </head>
  <body>
    <p><b>Использование символьных объектов</b></p>
    <p><i>Фрагмент HTML-кода на странице</i></p>
```

