

Лабораторная работа 2. Работа с текстом.

Цель работы: Освоение приемов форматирования текста при помощи гипертекстовой разметки.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ.

Как правило, блоки текста разделяют между собой абзацами (параграфами). По умолчанию между параграфами существует небольшой вертикальный отступ, называемый отбивкой. Синтаксис создания абзацев следующий.

```
<p>Абзац 1</p>
```

```
<p>Абзац 2</p>
```

Каждый абзац начинается с тега `<p>` и заканчивается необязательным закрывающим тегом `</p>`.

В любой книге для выделения следующего абзаца используется отступ первой строки, еще называемый «красная строка». Это позволяет читателю легко отыскивать взглядом новую строку и повышает, таким образом, читабельность текста. На веб-странице этот прием обычно не используется, а для разделения абзацев применяется отбивка.

В примере 1 показано применение абзацев для создания отступов между строками.

Все примеры выполнять!

Пример 1. Использование абзацев.

Создайте страницу по образцу, набрав стихотворение, применяя абзац.

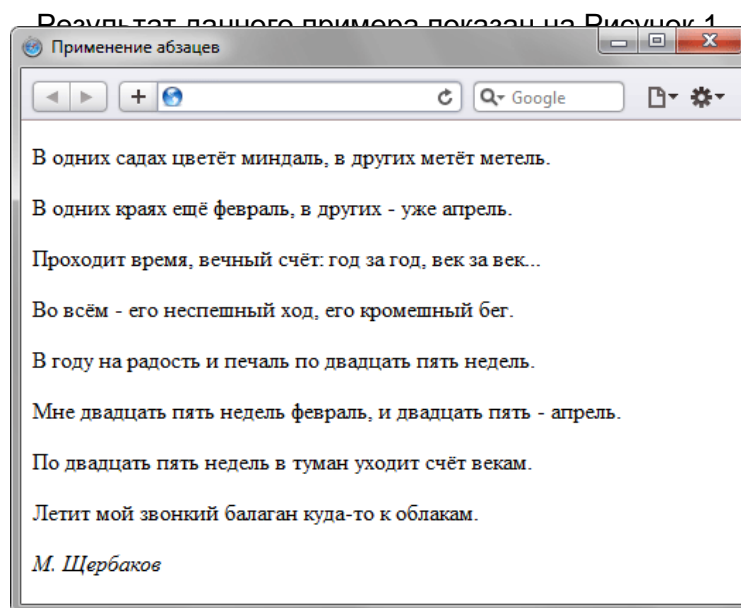


Рисунок 1. Отступы на веб-странице при использовании абзацев

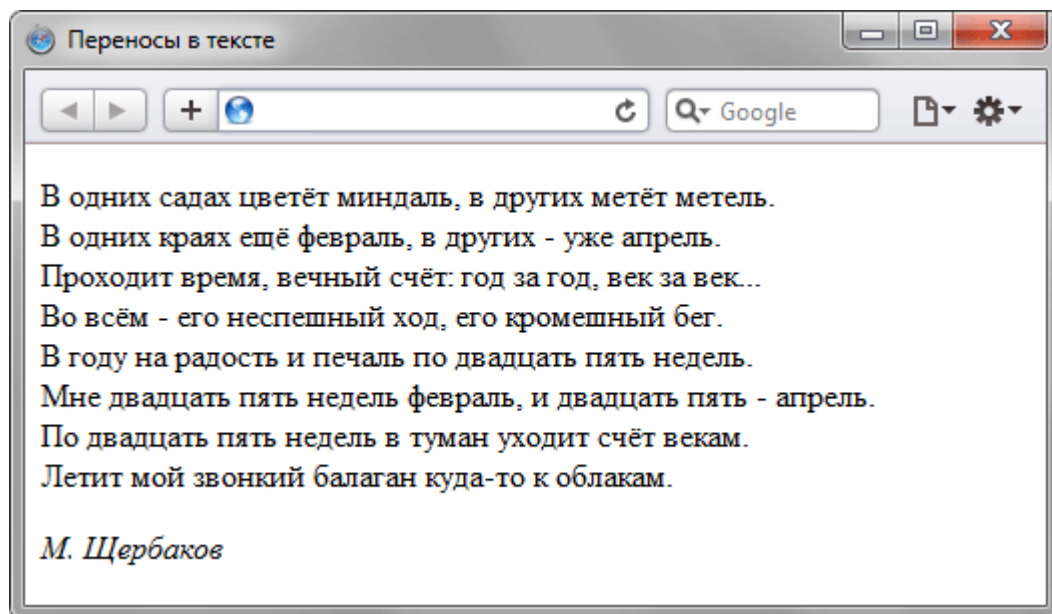
Как видно из рисунка, при использовании тега `<p>` между абзацами возникают слишком большие отступы. От них можно избавиться, если в местах переноса строк добавлять тег `
`. В отличие от абзаца, тег переноса строки `
` не создает дополнительных вертикальных отступов между строками и может применяться практически в любом тексте.

Так, текст примера 1. с учетом переноса строк будет преобразован следующим образом (пример 2).

Пример 2. Тег `
`

Создайте страницу по образцу, набрав стихотворение в одном абзаце, применяя обрыв строки.

Результат примера продемонстрирован на Рисунок 2. Видно, что расстояние между строками текста уменьшилось и он приобрел более компактный вид.



Заголовки выполняют важную функцию на веб-странице. Во-первых, они показывают важность раздела, к которому относятся. Чем больше заголовок и его «вес», тем более он значимый. Вспомните, что в газетах и журналах передовицы набраны крупным шрифтом, тем самым, привлекая к ним внимание и говоря: «вот это надо читать обязательно». Во-вторых, с помощью различных заголовков легко регулировать размер текста. Чем выше уровень заголовка, тем больше размер шрифта. Самым верхним уровнем является уровень 1 (`<h1>`), а самым нижним — уровень 6 (`<h6>`). И, в-третьих, поисковики добавляют рейтинг тексту, если он находится внутри тега заголовка. Это важно для раскрутки сайта и для его занятия первых строк выдачи результата в поисковой системе по ключевым словам.

Синтаксис создания заголовков показан в примере 3.

Пример 3. Добавление заголовков

Создайте страницу по образцу.

Результат данного примера показан на Рисунок 3. Содержимое тега `<h1>` отображается самым крупным шрифтом жирного начертания, а `<h6>` — самым мелким.

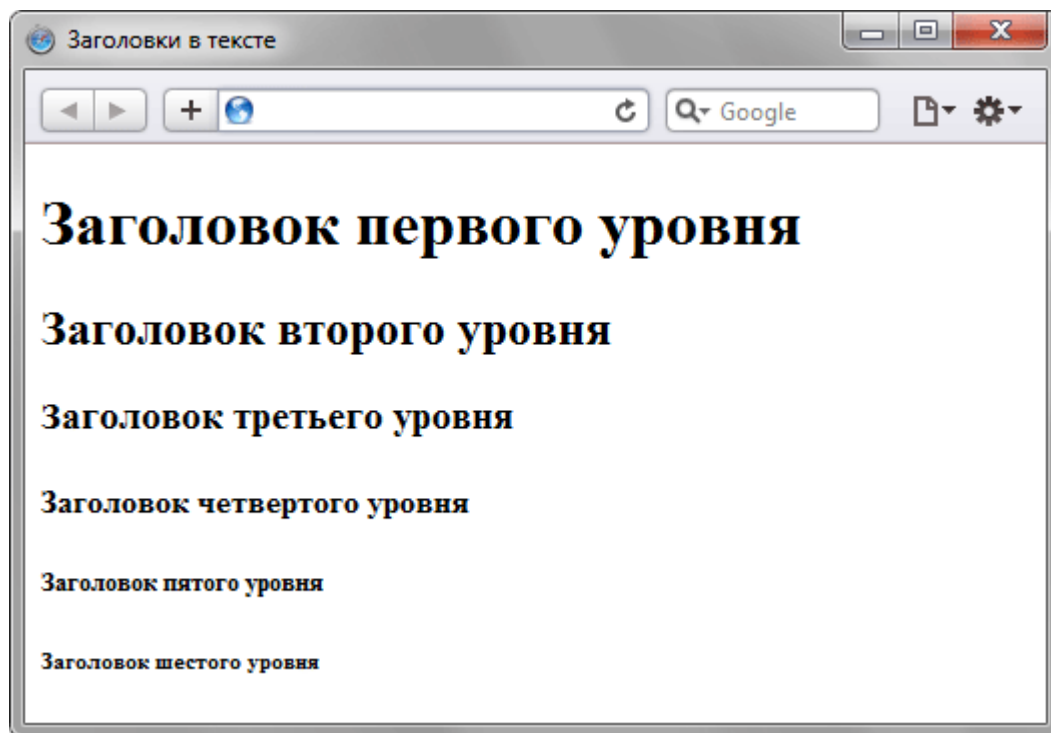


Рисунок 3. Вид заголовков на веб-странице

Как правило, на веб-странице применяют заголовки с первого по третий уровень, их вполне достаточно. Редко когда приходится использовать заголовки более низкого уровня.

Выравнивание текста определяет его внешний вид и ориентацию краев абзаца и может выполняться по левому краю, правому краю, по центру или по ширине. Наиболее распространенный вариант — выравнивание по левому краю, когда слева текст сдвигается до края, а правый остается неровным. Выравнивание по правому краю и по центру в основном используется в заголовках и подзаголовках. Следует иметь в виду, что при использовании выравнивания по ширине, в тексте между словами могут появиться большие интервалы, что не очень красиво.

Для установки выравнивания текста обычно используется тег параграфа `<p>` с атрибутом `align`, который определяет способ выравнивания. Также блок текста допустимо выравнивать с помощью тега `<div>` с аналогичным атрибутом `align`. Он может принимать следующие значения:

- `left` — выравнивание по левому краю, задается по умолчанию;
- `right` — выравнивание по правому краю;
- `center` — выравнивание по центру;
- `justify` — выравнивание по ширине (одновременно по правому и левому краю). Это значение работает только для текста, длина которого более, чем одна строка.

Атрибут `align` можно применять как для текста, так и для заголовков (пример 4).

Пример 4. Способы выравнивания текста

```
><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>Выравнивание текста</title>
  </head>
  <body>
    <h1 align="center">Как поймать льва?</h1>
    <h2 align="right">Метод перебора</h2>
    <p align="justify">Делим пустыню на ряд элементарных участков, размер
      которых совпадает с габаритными размерами льва, но при этом меньше
      размера
      клетки. Далее простым перебором определяем участок, в котором находится
      лев,
      что автоматически приводит к его поимке.</p>
    <h2 align="right">Метод дихотомии</h2>
    <p align="justify">Делим пустыню на две половины. В одной части – лев, в
      другой его нет. Берем ту половину, в которой находится лев, и снова
      делим
      ее пополам. Так повторяем до тех пор, пока лев не окажется пойман.</p>
  </body>
</html>
```

Результат данного примера показан на Рисунок 4.

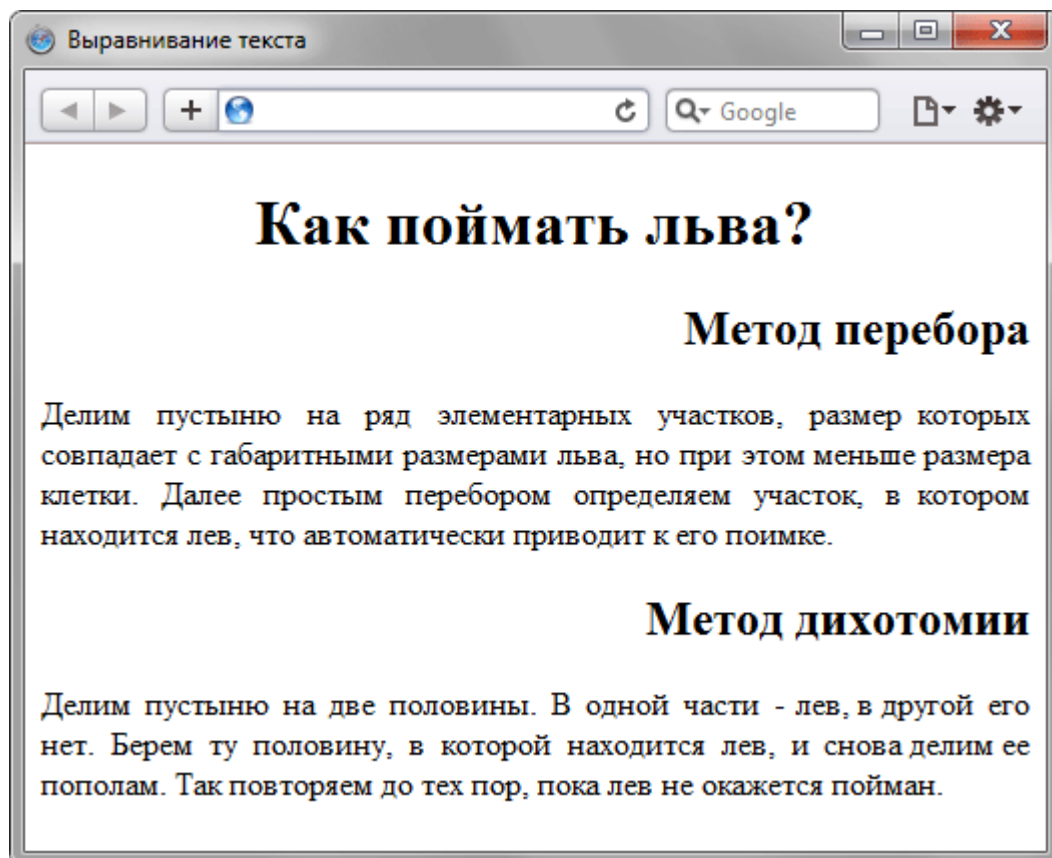


Рисунок 4. Вид текста при его выравнивании

Жирное начертание

Насыщенностью называют увеличение толщины линий шрифта и соответственно контраста. Обычно различают четыре вида насыщенности: светлое начертание, нормальное, полужирное и жирное. Однако с помощью HTML можно установить только нормальное и жирное начертание. Для установки текста жирного начертания применяется два тега: `` и ``.

``Жирное начертание шрифта``

``Сильное выделение текста``

Курсивное начертание

Курсивный шрифт представляет собой не просто наклон отдельных символов, для некоторых шрифтов это полная переделка под новый стиль, имитирующий рукописный. Курсив для текста определяют два тега: `<i>` и ``.

`<i>`Курсивное начертание шрифта`</i>`

``Выделение текста``

Следует отметить, что теги `` и ``, также как `<i>` и `` хотя и похожи по своему действию, являются не совсем эквивалентными и заменяемыми. Первый тег `` — является тегом физической разметки и устанавливает жирное начертание текста, а тег `` — тегом логической разметки и выделяет помеченный текст. Такое разделение тегов на логическое и физическое форматирование изначально предназначалось, чтобы сделать HTML универсальным, в том числе не зависящим от устройства вывода информации. Теоретически, если воспользоваться, например, речевым браузером, то текст, оформленный с помощью тегов `` и ``, будет отмечен по-разному. Однако получилось так, что в популярных браузерах результат использования этих тегов равнозначен.

В примере 5 показано использование тегов `` и `` для оформления текстов.

Пример 5. Теги `` и ``

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>Оформление текста</title>
  </head>
  <body>
    <p><strong>А где же печенье и самогоноваренье?!</strong> —
    <em>воскликнул Мальчиш-плохиш</em>. </p>
  </body>
</html>
```

Результат данного примера показан на Рисунок 5.

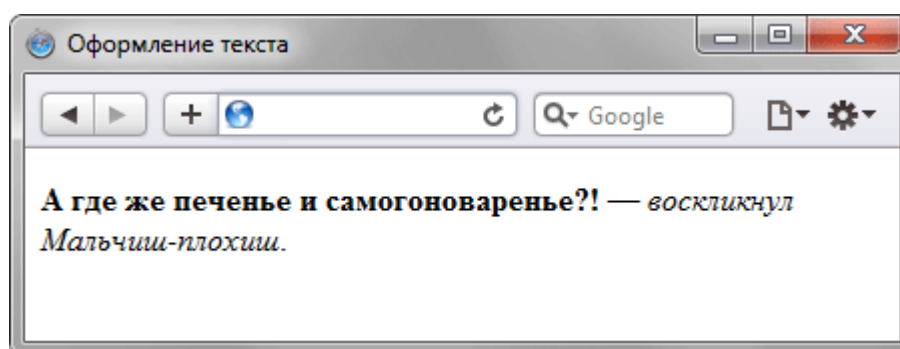


Рисунок 5. Жирное и курсивное начертание текста

Индексом по отношению к тексту называется смещение символов относительно базовой линии вверх или вниз.

В зависимости от положительного или отрицательного значения, индекс называется, соответственно, верхним или нижним. Они активно применяются в математике, физике, химии и для обозначения единиц измерения. HTML предлагает два тега для создания индекса: `<sup>` — верхний индекс и `<sub>` — индекс нижний. Текст, помещенный в один из этих контейнеров, обозначается меньшим размером, чем базовый текст и смещается относительно горизонтали.

В примере 6 показано, где применяется подобный текст

Пример 6. Использование нижнего индекса

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>Нижний индекс</title>
  </head>
  <body>
    <p><b>Формула серной кислоты:</b> <i>H<sub>2</sub>SO<sub>4</sub></i></p>
  </body>
</html>
```

Результат данного примера показан на рис.6.

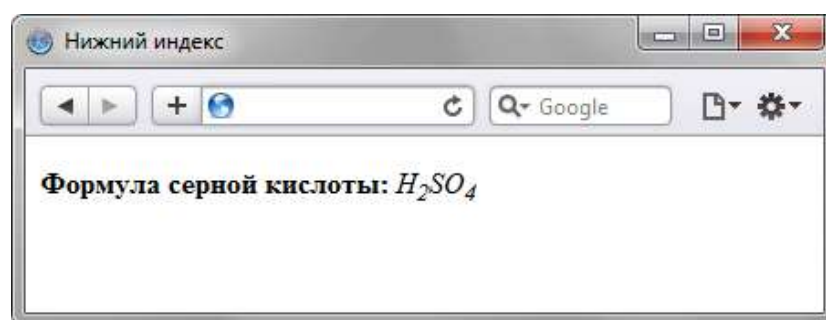


Рисунок 6. Отображение текста в виде нижнего регистра

Тег ``

Тег `` представляет собой контейнер для изменения характеристик шрифта, таких как размер, цвет и гарнитура. Хотя этот тег до сих пор поддерживается всеми браузерами, он считается устаревшим и от его использования рекомендуется отказаться в пользу стилей.

Синтаксис

font>Текст

Атрибуты:

color-Устанавливает цвет текста.

face-Определяет гарнитуру шрифта.

size-Задаёт размер шрифта в условных единицах.

Пример

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html;
charset=utf-8">
  <title>Тег FONT</title>
</head>
<body>
  <p><font size="5" color="red" face="Arial">П</font>ервая буква
этого предложения написана шрифтом Arial, выделена красным
цветом и увеличена в размерах.</p>
</body>
</html>
```

Задание 1. Создайте веб-страницу с зеленым фоном и белым текстом, как показано на Рисунок 7

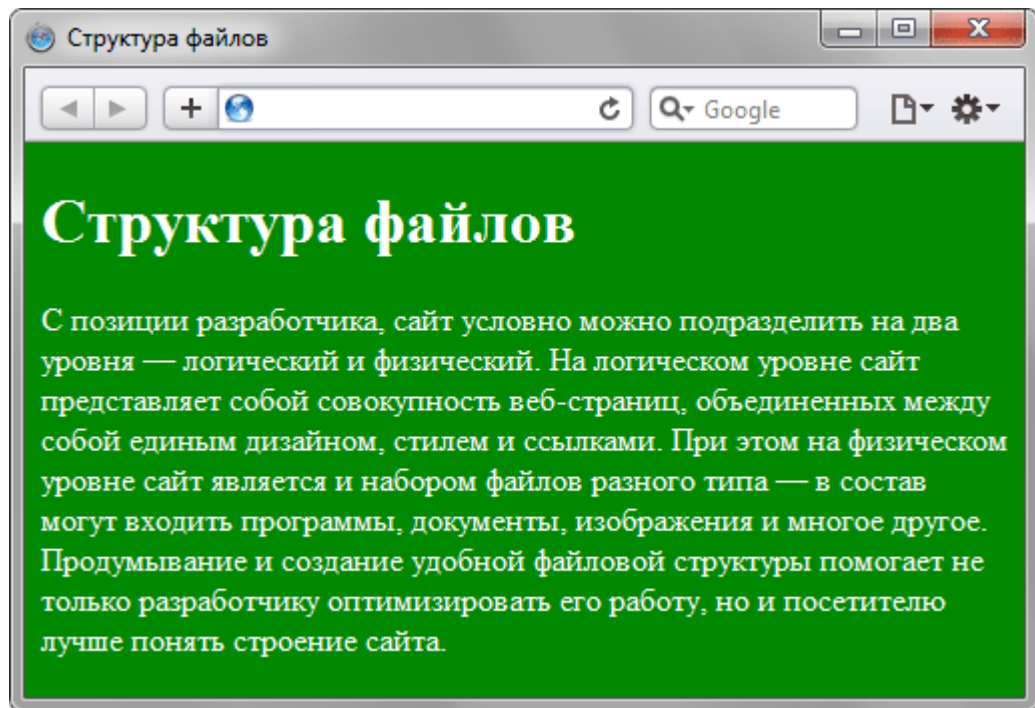


Рисунок 7

Задание 2. Напишите код HTML, чтобы получить результат, приведенный на Рисунок 8.

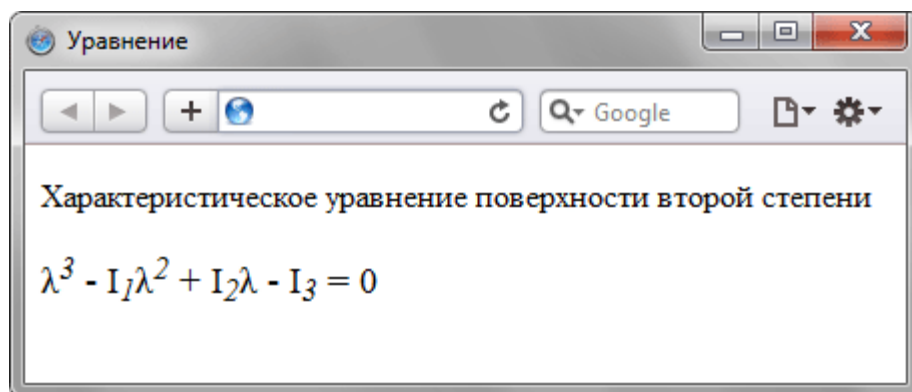


Рисунок 8