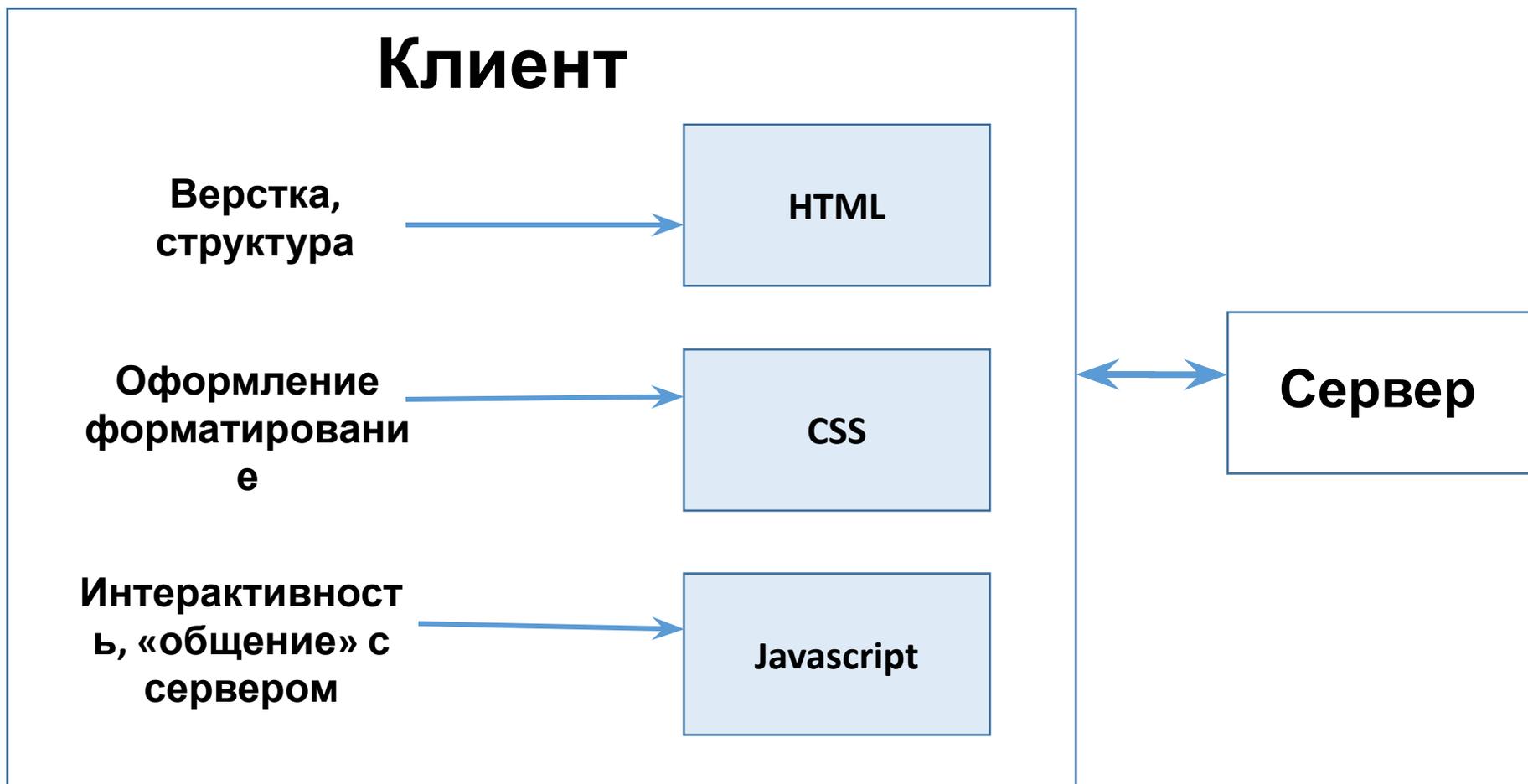
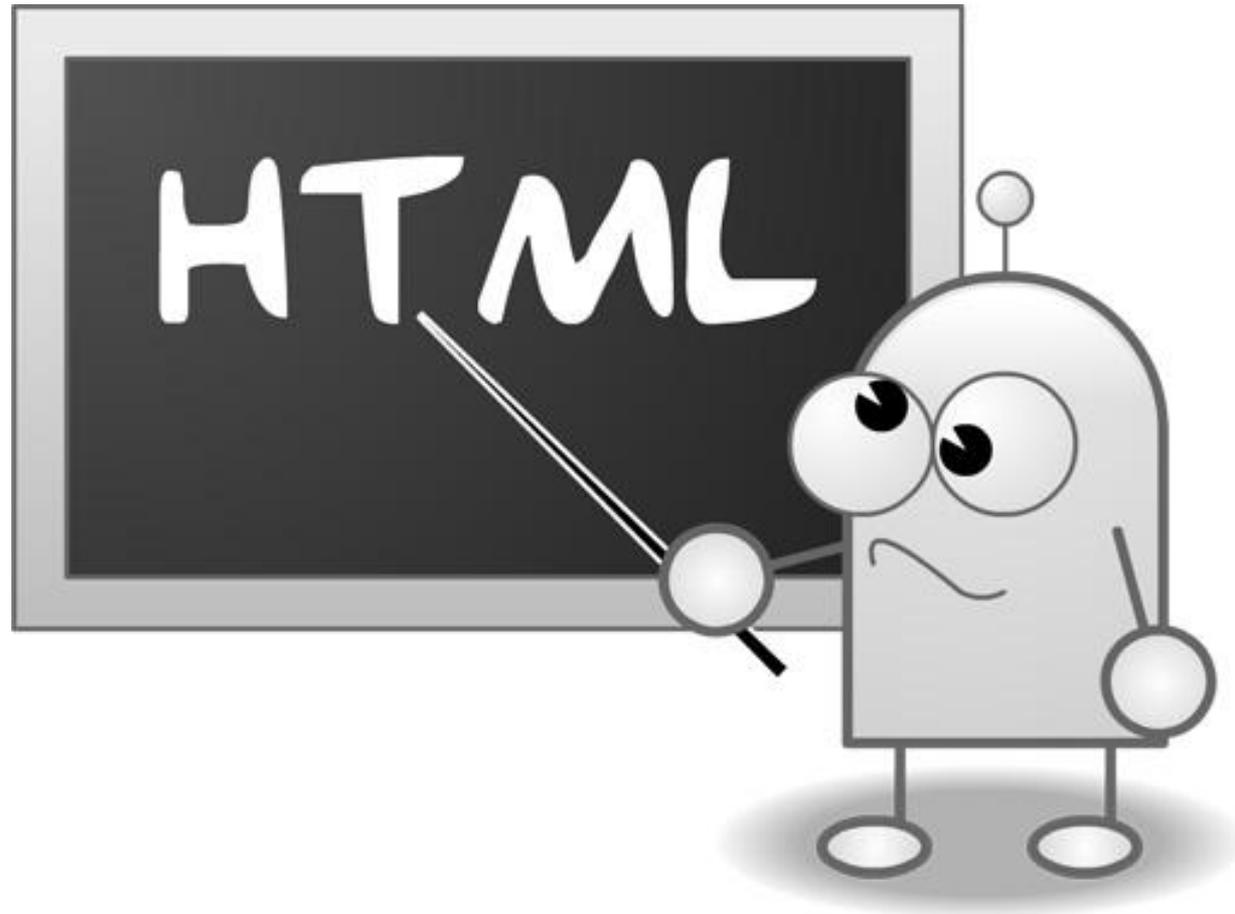


# Клиентская часть



HTML



HTML

**HTML = обычный текст + теги**

# HTML. Тэги

Текст как и в любых других файлах используется для передачи информации, а разметочные **тэги** используются для ее группировки и оформления.

Разметочные тэги HTML представляют собой специальные слова, которые окружены с обеих сторон угловыми скобками, например: **<html>**.

## HTML. Тэги

HTML тэги обычно используются в паре, например: `<b></b>`. Первый тэг называется **начальным тэгом**, а второй **конечным тэгом**.

Текст находящийся между начальным и конечным тэгом подвергается "разметке".

Например `<b>Привет</b>` будет отображено браузером как **Привет** (слово 'Привет' написанное жирным шрифтом).

# HTML. Пример

Любой HTML документ состоит из обычного текста и **<b>разметочных тэгов </b>**. Текст как и в любых других файлах используется для передачи информации, а разметочные тэги используются для ее группировки и оформления.

**<b> Разметочные тэги </b>** HTML представляют собой специальные слова, которые окружены с обеих сторон угловыми скобками, например: `<html>`. **<hr/>** HTML тэги обычно используются в паре, например: `<b> </b>`. Первый тэг называется начальным тэгом, а второй конечным тэгом.

Текст находящийся между начальным и конечным тэгом подвергается "разметке".

Например **<b>Привет</b>** будет отображено браузером как **Привет** (слово 'Привет' написанное жирным шрифтом).

# HTML

Если HTML документ будет открыт в редакторе он будет отображен как обычный текстовый файл.

Если HTML документ будет открыт в браузере он будет отображен в соответствии с разметочными тэгами и будет называться **веб-страницей**.

# HTML. Пример

```
<html>  
  <body>  
    <h1>Мой первый заголовок!</h1>  
    <p>Привет всем!</p>  
  </body>  
</html>
```

# HTML. Пример

```
1 <html>
2 <body>
3   <h1>Мой первый заголовок!</h1>
4   <p>Привет всем!</p>
5 </body>
6 </html>
```

**Мой первый  
заголовок!**

Привет всем!

# Объяснение примера

Тэг **<html>** объявляет начало и конец HTML документа;

Тэг **<body>** содержит все видимое содержимое HTML страницы;

Тэг **<h1>** позволяет "разметить" вложенный в него текст как заголовок;

Тэг **<p>** позволяет "разметить" вложенный в него текст как абзац.

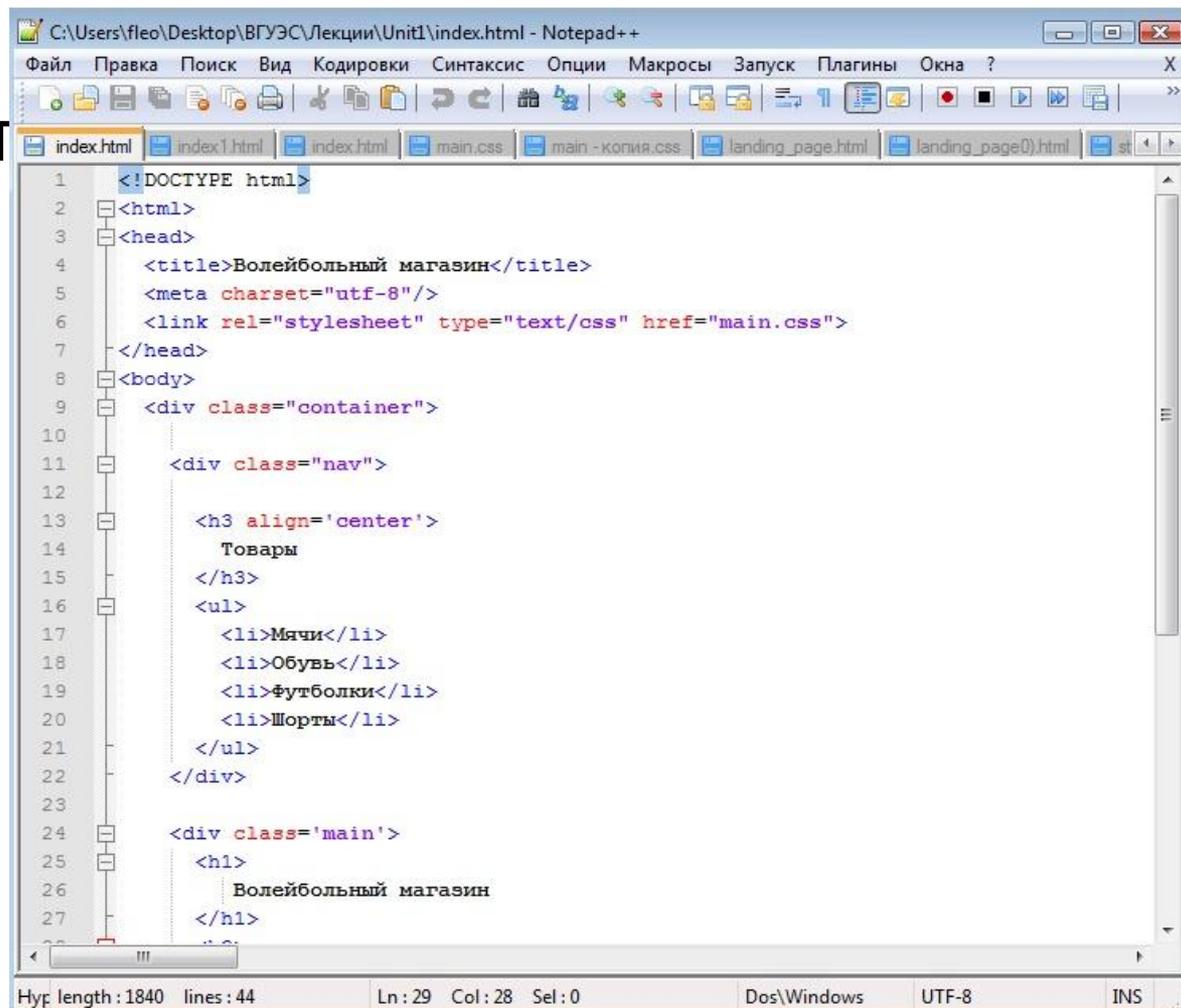
# Создание HTML-документа

Для того, чтобы создать HTML документ необходимо:

- Открыть любой текстовый редактор (например блокнот встроенный в Windows);
- Набрать произвольный текст и разметить его HTML тэгами;
- Сохранить файл с расширением **.htm** или **.html**.

# Редакторы

- Б л о к н о т
- Notepad ++
- Sublime Text
- Atom
- ...



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Волейбольный магазин</title>
5 <meta charset="utf-8"/>
6 <link rel="stylesheet" type="text/css" href="main.css">
7 </head>
8 <body>
9 <div class="container">
10
11 <div class="nav">
12
13 <h3 align='center'>
14 Товары
15 </h3>
16 <ul>
17 <li>Мячи</li>
18 <li>Обувь</li>
19 <li>Футболки</li>
20 <li>Шорты</li>
21 </ul>
22 </div>
23
24 <div class='main'>
25 <h1>
26 Волейбольный магазин
27 </h1>
```

Статус-бар: Стр: length : 1840 lines : 44 Ln : 29 Col : 28 Sel : 0 Dos\Windows UTF-8 INS

# HTML-элементы

**HTML элементом** называется комбинация начального тэга, конечного тэга и содержимого.

Примеры HTML элементов:

Начальный тэг	Содержимое элемента	Конечный тэг
<code>&lt;p&gt;</code>	Это абзац.	<code>&lt;/p&gt;</code>
<code>&lt;b&gt;</code>	Это жирный текст.	<code>&lt;/b&gt;</code>
<code>&lt;br /&gt;</code>		

# Парные и непарные теги

Парные теги: **<тег> содержимое </тег>**

Одинарные теги: **<тег />**

# Парные теги



# Заголовки <h1>...</h1>

```
1 <html>
2 <body>
3   <h1>Это заголовок</h1>
4   <h2>Это заголовок</h2>
5   <h3>Это заголовок</h3>
6   <h6>Это заголовок</h6>
7 </body>
8 </html>
9
```

**Это заголовок**

**Это заголовок**

**Это заголовок**

**Это заголовок**

# Абзац <p>...</p>

```
1 <html>
2 <body>
3   <p>Это абзац</p>
4   <p>Это другой абзац</p>
5 </body>
6 </html>
7
```

Это абзац

Это другой абзац

# ССЫЛКИ

## Внешние

`<a href="адрес">Текст ссылки </a>`

## Внутренние

`<!-- Создание гиперссылки на : -->`

`<a href="#bookmark">Текст ссылки </a>`

`<!-- Создание закладки -->`

`<a id="bookmark">Текст закладки. </a>`

# Ссылки <a href="">...</a>

```
1 <html>
2 <body>
3   <a href='http://www.wisdomweb.ru'>wisdomweb.ru</a>
4 </body>
5 </html>
```

[wisdomweb.ru](http://www.wisdomweb.ru)

# Вставка картинки

```
<img src='Адрес' width='Число' height='Число' />
```

## Пример:

```
<img src='mountimg3.jpg' width='300' height='242' />
```

# Картинки <img src="" ... />

```
1 <html>
2 <body>
3   <h2> Ергаки 2010 </h2>
4   <img src='mountimg3.jpg' width='300' height='242' />
5 </body>
6 </html>
```

## Ергаки 2010



# Атрибуты тега

**<тег атрибуты>**

содержимое

**</тег>**

# Атрибуты тега

**<тег атрибут1 атрибут2 ... атрибутN>**

содержимое

**</тег>**

# Атрибуты тега

**<тег атрибут1=значение атрибут2=значение  
...> содержимое  
</тег>**

# Примеры тегов с атрибутами

```
<a href="http://www.wisdomweb.ru/">wisdomweb.ru</a>
```

```

```

```

```

```
<a href="http://mysite.ru/" title="Комментарий  
про мой сайт">Мой сайт</a>
```

```
<div class="main" align="center">просто текст  
</div>
```

# Стандартные атрибуты

Атрибут	Описание
<b>accesskey</b>	Определяет сочетание клавиш для доступа к данному элементу.
<b>class</b>	Определяет имя класса для элемента.
<b>id</b>	Определяет уникальный идентификатор для элемента.
<b>style</b>	Определяет стиль элемента.
<b>title</b>	Содержит дополнительную информацию об элементе (значение данного атрибута отображается при наведении курсора мыши на элемент).

# Оформление текста

- **<b> Жирный текст </b>**
- *<i> Курсивный текст </i>*
- **<br />** (перевод на новую строку)

# Оформление текста

```
1 <html>
2 <body>
3 <b>Этот текст написан жирным шрифтом.
4 <i>Этот текст написан курсивом.</i>
5 <p>Это обычный текст<sub>это текст в
6 <p>Это обычный текст<sup>это текст в
7 <em>Данный тэг определяет важное соде
8 <strong>Данный тэг определяет очень
9 <big>Размер шрифта этого текста увели
10 <small>Размер шрифта этого текста ум
11 </body>
12 </html>
13
```

**Этот текст написан жирным шрифтом.**  
*Этот текст написан курсивом.*

Это обычный текст<sub>это текст в нижнем индексе.</sub>

Это обычный текст<sup>это текст в верхнем индексе.</sup>

*Данный тэг определяет важное содержимое.*  
**Данный тэг определяет очень важное  
содержимое. Размер шрифта этого  
текста увеличен. Размер шрифта этого  
текста уменьшен.**

# Перевод на новую строку: <br>

```
1 <html>
2 <body>
3 <p> Это а<br />бзац разби<br />тый на
4 </body>
5 </html>
```

Это а  
бзац разби  
тый на нескол  
ько строк

# Горизонтальная линия: `<hr />`

```
1 <html>
2 <body>
3 <p> Это первый абзац </p>
4 <hr />
5 <p> Это второй абзац </p>
6 <hr />
7 <p> Это третий абзац </p>
8 </body>
9 </html>
```

10

Это первый абзац

---

Это второй абзац

---

Это третий абзац

# Предформатированный текст

```
1 <html>
2 <body>
3 <pre>
4 Какой-то текст какой-то текст
5 Какой-то текст какой-то текст
6 Какой-то текст какой-то текст
7 Какой-то текст какой-то текст
8
9         Какой-то автор
10 </pre>
11 </body>
12 </html>
```

```
Какой-то текст какой-то текст
Какой-то текст какой-то текст
Какой-то текст какой-то текст
Какой-то текст какой-то текст
```

```
        Какой-то автор
```

# Вложенные теги

<i>

<b>

***Данный текст курсивный  
и жирный одновременно.***

</b>

</i>

# Списки

## HTML

```
<html>
  <body>
    <p>Упорядоченный список: </p>
    <ol>
      <li>Меркурий</li>
      <li>Венера</li>
      <li>Земля</li>
    </ol>
    <p>Неупорядоченный список: </p>
    <ul>
      <li>Зебра</li>
      <li>Слон</li>
      <li>Жираф</li>
    </ul>
  </body>
</html>
```

## Результат

Упорядоченный список:

1. Меркурий
2. Венера
3. Земля

Неупорядоченный список:

- Зебра
- Слон
- Жираф

# Вложенные списки

```
1 <html>
2 <body>
3 <p> Крупнейшие города различных стран
4 <ol>
5   <li> Россия
6     <ol>
7       <li> Москва </li>
8       <li> Санкт-Петербург </li>
9     </ol>
10  </li>
11  <li> США
12    <ol>
13      <li> Нью-Йорк </li>
14      <li> Чикаго </li>
15    </ol>
16  </li>
17 </ol>
18 </body>
19 </html>
20
```

Крупнейшие города различных стран  
(пример создания вложенных списков):

1. Россия
  1. Москва
  2. Санкт-Петербург
2. США
  1. Нью-Йорк
  2. Лос-Анджелес

# Теги-контейнеры

**<div>**

Содержимое

**</div>**

**<span>**

Содержимое

**</span>**

# Оформление с помощью CSS

CSS был представлен вместе с HTML 4. CSS предоставляет более удобный способ оформления HTML документов.

CSS предоставляет полную свободу при оформлении документов, с CSS Вы можете:

- Устанавливать размер, начертание и цвет шрифта;
- Изменять местоположение элементов;
- Оформлять фон элементов;
- Выравнивать текст;
- Оформлять таблицы и списки и многое другое.

## Пример

```
.....  
<p style='font-size:30px;display:inline;color:white;'> Это  
абзац оформленный с помощью CSS </p>  
<p style='display:inline;color:white;font-family:Verdana;  
border-style:solid;'> Это еще один абзац оформленный с помощью  
CSS </p>  
.....
```

# Устаревшие теги

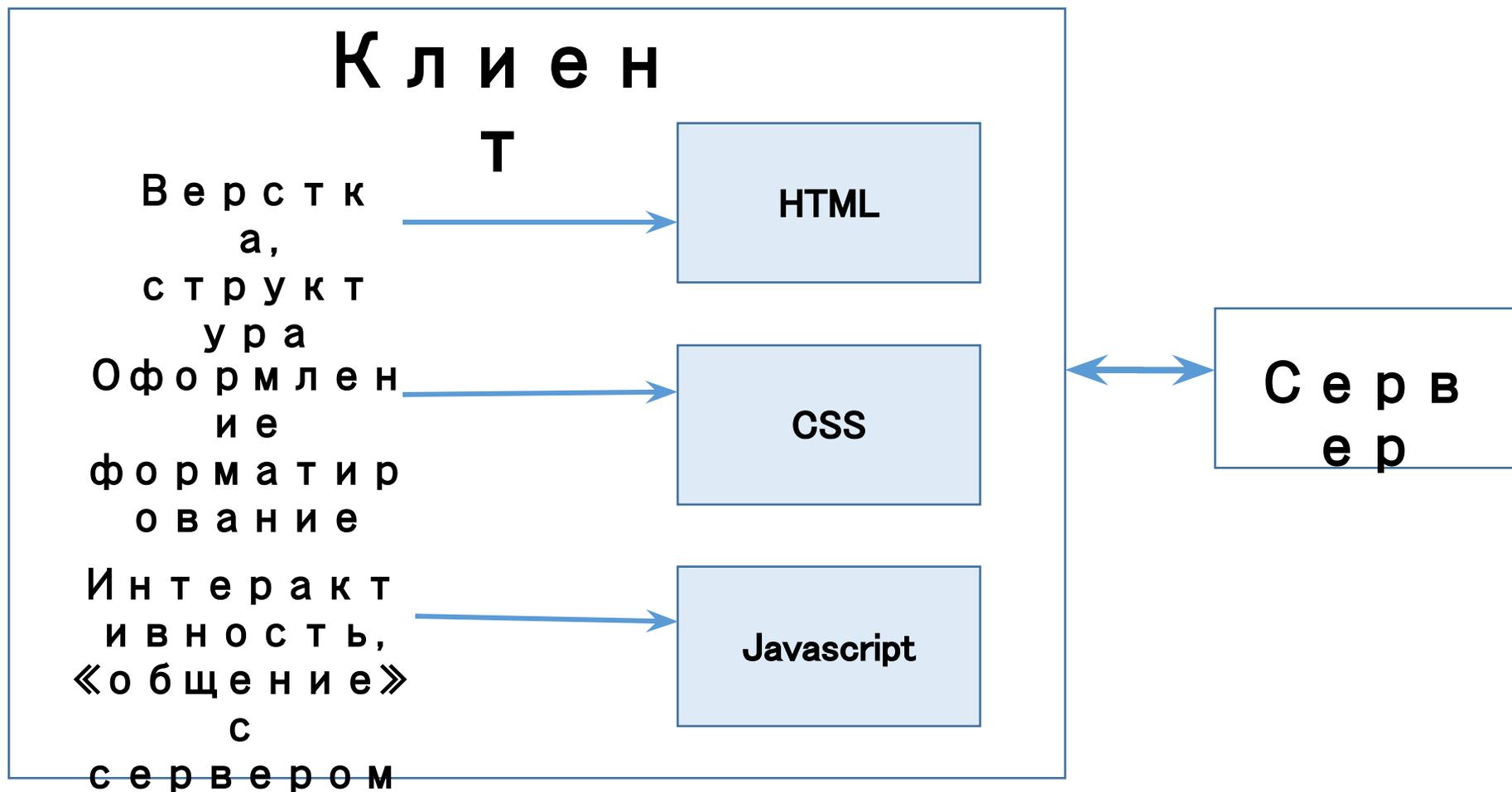
После введения CSS некоторые теги и атрибуты в HTML стали считаться **устаревшими** (с помощью CSS можно добиться большего гораздо более эффективно). Они не будут поддерживаться в будущих версиях HTML и XHTML.

Избегайте использования следующих устаревших HTML тэгов и атрибутов:

Тэги	Описание
<code>&lt;strike&gt;</code>	Определяет зачеркнутый текст.
<code>&lt;font&gt;</code> и <code>&lt;basefont&gt;</code>	Устанавливает шрифт для текста.
<code>&lt;center&gt;</code>	Выравнивает содержимое по центру.
<code>menu</code>	Определяет список меню.

Атрибуты	Описание
<code>color</code>	Определяет цвет текста.
<code>bgcolor</code>	Определяет цвет фона.
<code>align</code>	Определяет выравнивание текста.

# Клиентская часть



# CSS



```
body {  
  margin: 0;  
  padding: 0;  
  /*resetting default size to 10px or 62.5% of the 16px default*/  
  font-size: 62.5%;  
  font-family: Arial, Helvetica, sans-serif;  
  color: #666;  
}  
P {  
  font-size: 1.6em; /*equivalent to 16px*/  
  color: #333;  
  margin: 0 0 10px;  
  padding: 0;  
}
```

SiteY.ru

# CSS. Пример

```
<html>
<head>
<style type='text/css'>
#css
{
border-collapse:collapse;
font-family:'arial';
}
#css th
{
background-color:#DC63C2;
padding:7px;
font-size:1.1em;
border:1px black solid;
color:white;
}
#css td
{
padding:7px;
border:1px solid;
}
</style>
</head>
<body>
```

**Пример оформленной с помощью CSS таблицы.**

Страна	Население (млн.)
Россия	143
США	313
Китай	1347
Франция	65

**Эта же таблица без оформления.**

Страна	Население (млн.)
Россия	143
США	313
Китай	1347
Франция	65

# Синтаксис

```
Селектор — [ h1 {  
Свойство — [ color: red; ] — Значение  
}
```

# Пример.Шрифт

h1

{

font-family: Georgia, serif;

}

# Пример. Размер

```
h1  
{  
    font-size: 60px;  
}
```

# Селекторы

- название тега
- класс тега – class (.)
- идентификатор тега – id (#)
- атрибут тега – []
- \* все элементы страницы

# Селекторы тегов

```
p
{
color:green;
}
h2
{
color:red;
}
```

# Селекторы тегов

```
1 <html>
2 <head>
3
4 <style type='text/css'>
5 p
6 {
7 color:green;
8 }
9 h2
10 {
11 color:red;
12 }
13 </style>
14
15
16 </head>
17 <body>
18 <p>Я первый абзац</p>
19 <h1>Это заголовок h1</h1>
20 <p>Я второй абзац</p>
21 <h2>Это заголовок h2</h2>
22 </body>
23 </html>
24
```

Я первый абзац

# Это заголовок h1

Я второй абзац

## Это заголовок h2

# Селекторы идентификаторов

```
/* Оформим элемент с id="test1" */  
#test1  
{  
color:green;  
font-family:verdana;  
font-size:1.2em;  
}
```

# Селекторы идентификаторов

```
1 <html>
2 <head>
3
4 <style type='text/css'>
5 /* Оформим элемент с id='test1' */
6 #test1
7 {
8     color:green;
9     font-family:verdana;
10    font-size:1.2em;
11 }
12 </style>
13
14 </head>
15 <body>
16
17 <p id='test1'>Этот абзац будет оформлен
18
19 <p>А этот абзац изменения не затронут. <
20
21 </body>
22 </html>
23
```

Этот абзац будет оформлен с помощью CSS.

А этот абзац изменения не затронут.

# Селекторы классов

```
/* Свойства будут применены ко всем элементам с class="test1" */  
.test1  
{  
color:green;  
font-family:verdana;  
font-size:1.2em;  
}
```

# Селекторы классов

```
1 <html>
2 <head>
3 <style type='text/css'>
4 /* Свойства будут применены ко всем элементам с
5 .test1 {
6   color:green;
7   font-family:verdana;
8   font-size:1.2em;
9 }
10 </style>
11 </head>
12 <body>
13
14 <p class='test1'>Этот абзац будет оформлен с по
15
16 <p>А этот абзац изменения не затронут. </p>
17
18 <p class='test1'>И этот абзац тоже будет оформ
19
20 </body>
21 </html>
22
```

Этот абзац будет оформлен с помощью CSS.

А этот абзац изменения не затронут.

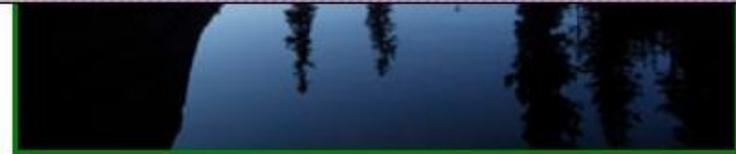
И этот абзац тоже будет оформлен с помощью CSS.

# Селекторы атрибутов

```
/* Оформит все элементы имеющие атрибут src */  
[src]  
{  
border:solid green 3px;  
}
```

# Селекторы атрибутов

```
1 <html>
2 <head>
3 <style type='text/css'>
4 /* Оформит все элементы, у которых задан атри
5 [src]
6 {
7 border-color:green;
8 border-style:solid;
9 border-width:3px;}
10 </style>
11
12 </head>
13 <body>
14
15 <img src='mountimg.jpg' alt='Ергаки 2010' />
16
17 <br /><br />
18
19 <img src='mountimg1.jpg' alt='Кузнецкий алата
20
21 <p>У данного элемента не будет зеленой рамки
22 </html>
23
```



У данного элемента не будет зеленой рамки потому, что у него нет атрибута src.

# Комбинирование селекторов

```
/* Свойства будут применены только к тем элементам с class="test1", котор  
h2.test1  
{  
color:green;  
font-family:verdana;  
font-size:1.2em;  
}
```

```
/* Свойства будут применены только к тем элементам p, которые находятся в  
div p  
{  
color:green;  
font-family:verdana;  
font-size:1.2em;  
}
```

```
/* Свойства будут применены только к тем элементам p, которые идут сразу  
div+p  
{  
color:green;  
font-family:verdana;  
font-size:1.2em;
```

# Комбинирование селекторов

```
1 <html>
2 <head>
3 <style type='text/css'>
4 /* Свойства оформления будут применены только
5 h2.test1
6 {
7   color:green;
8   font-family:verdana;
9 }
10 </style>
11
12 </head>
13 <body>
14
15 <h2 class='test1'>Данный заголовок был оформл
16
17 <p class='test1'>Данный абзац не был оформлен
18
19 <h2>Данный заголовок так же не был оформлен</
20
21 </body>
22 </html>
23
```

**Данный заголовок был оформлен с помощью CSS**

Данный абзац не был оформлен

**Данный заголовок так же не был оформлен**

# Группировка селекторов

```
h1
{
font-family:verdana;
color:green;
}
h2
{
font-family:verdana;
color:green
}
p
{
font-family:verdana;
color:green;
}
```



```
h1, h2, p
{
font-family:verdana;
color:green;
}
```

# Конфликт селекторов?

## HTML:

```
<div class="hello_class" id="hello_id">Привет, мир</div>
```

## CSS:

```
div {  
    font-size: 11pt;  
    font-weight: bold;  
}  
#hello_id {  
    font-size: 27pt;  
    text-decoration: underline;  
}  
.hello_class {  
    font-size: 14pt;  
    font-style: italic;  
}
```

# Вес селекторов

- название тега имеет вес 1 балл;
- присвоенный тегу класс — 10 баллов;
- идентификатор тега — 100 баллов.

**!important** - способ явно повысить вес определенного CSS-свойства

```
div {  
    font-size: 11pt !important;  
    font-weight: bold;  
}
```

# Каскадирование стилей.

## Вес стилей

- С т и л ь б р а у з е р а .
- С т и л ь п о л ь з о в а т е л я .
- С т и л ь а в т о р а .
- С т и л ь а в т о р а с д о б а в л е н и е м !important.
- С т и л ь п о л ь з о в а т е л я с д о б а в л е н и е м !important.

# Псевдоклассы и псевдоэлементы

**селектор :**

**псевдокласс**

- :active
- :link
- :focus
- :hover
- :visited
- :first-child
- :lang

**селектор :**

**псевдоэлемент**

- :after
- :before
- :first-letter
- :first-line

# Текст. Цвет

## **color**

- `p {color:green;}`
- `p {color:rgb(0,255,0);}`
- `p {color:#00ff00;}`

# Текст. Выравнивание

## **text-align**

- p.ta1 {text-align:center;} (П о ц е н т р у)
- p.ta2 {text-align:left;} (С л е в а)
- p.ta3 {text-align:right;} (С п р а в а)
- p.ta4 {text-align:justify;} (П о ш и р и н е)

# Текст. Выравнивание

```
1 <html>
2 <head>
3 <style type='text/css'>
4 p.ta1
5 {text-align:center;}
6 p.ta2
7 {text-align:left;}
8 p.ta3
9 {text-align:right;}
10 p.ta4
11 {text-align:justify;}
12 </style>
13 </head>
14 <p class='ta1'>Текст данного элемента
15 <p class='ta2'>Текст данного элемента
16 <p class='ta3'>Текст данного элемента
17 <div style='width:48%;float:left;'>
18 <p class='ta4'>Текст данного элемента
19 </div>
20 <div style='width:48%;float:right;'>
21 <p class='ta4'>Текст данного элемента
22 </div>
23 </html>
24
```

Текст данного элемента выравнен по центру.

Текст данного элемента выравнен по левому краю.

Текст данного элемента выравнен по правому краю.

Текст данного элемента выравнен с помощью значения justified. Как Вы можете видеть, браузер автоматически растягивает строчки до одинаковой длины путем изменения величины отступов между словами. Попробуйте убрать выравнивание или измените ширину окна браузера, чтобы лучше понять принцип действия.

Текст данного элемента выравнен с помощью значения justified. Как Вы можете видеть, браузер автоматически растягивает строчки до одинаковой длины путем изменения величины отступов между словами. Попробуйте убрать выравнивание или измените ширину окна браузера, чтобы лучше понять принцип действия.

# Текст. Оформление

```
1 <html>
2 <head>
3 |
4 <style type='text/css'>
5 p.td1 {text-decoration:underline;}
6 p.td2 {text-decoration:line-through;}
7 p.td3 {text-decoration:overline;}
8 </style>
9
10 </head>
11 <p class='td1'>Текст данного элемента
12 <p class='td2'>Текст данного элемента
13 <p class='td3'>Текст данного элемента
14 </html>
15
```

Текст данного элемента подчеркнут.

~~Текст данного элемента перечеркнут.~~

Текст данного элемента находится под  
линией.

# Текст. Отступ между словами и буквами в тексте

С помощью CSS свойства **letter-spacing** Вы можете увеличивать или уменьшать отступ между буквами в тексте HTML элементов.

```
p.ls1 {letter-spacing:10px;}
```

С помощью свойства **word-spacing** Вы можете увеличивать или уменьшать отступ между словами в тексте HTML элементов.

```
p.ws1 {word-spacing:15px;}
```

# Текст. Отступ между словами и буквами в тексте

```
1 <html>
2 <head>
3
4 <style type='text/css'>
5 p.ls1
6 {
7     letter-spacing:10px;
8 }
9 </style>
10
11 </head>
12
13 <p class='ls1'> Величина отступа
14
15 </html>
16
```

В е л и ч и н а  
о т с т у п а м е ж д у  
б у к в а м и в  
д а н н о м э л е м е н т е  
р а в н а 1 0 п и к с .

# Текст. Другие свойства

Свойство	Описание	Значения
<b>direction</b>	Устанавливает направление текста.	ltr rtl
<b>line-height</b>	Устанавливает высоту строки.	normal пиксели %
<b>text-indent</b>	Устанавливает величину отступа первого символа текста.	пиксели %
<b>text-transform</b>	Устанавливает регистр букв текста элемента.	none capitalize uppercase lowercase
<b>vertical-align</b>	Устанавливает вертикальное выравнивание элемента.	sub super top middle bottom пиксели %
<b>white-space</b>	Устанавливает как должны обрабатываться пробелы внутри элемента.	normal pre nowrap

# Текст. Шрифт

**Ш р и ф т : font-family**

p {font-family:Arial;}

**Р а з м е р ш р и ф т а : font-size**

p.fz1 {font-size:20px;}

# Текст. Безопасные шрифты

Список безопасных шрифтов:

- Arial
- Arial Black
- Courier New
- Comic Sans MS
- Georgia
- Impact
- Times New Roman
- Trebuchet MS
- Verdana

# Шрифт. Размер

```
1 <html>
2 <head>
3
4 <style type='text/css'>
5   p.fz1 {font-size:20px;}
6   p.fz2 {font-size:30px;}
7   p.fz3 {font-size:13px;}
8 </style>
9 </head>
10
11 <p class='fz1'>Величина размера шрифта д
12 <p class='fz2'>Величина размера шрифта д
13 <p class='fz3'>Величина размера шрифта д
14
15 <br />
16 <b>Величина стандартного размера шрифта
17 </html>
18
```

Величина размера шрифта данного абзаца 20 пикселей.

Величина размера  
шрифта данного абзаца  
**30 пикселей.**

Величина размера шрифта данного абзаца 13 пикселей.

**Величина стандартного размера шрифта 16 пикселей.**

# Шрифт. Стиль

```
1 <html>
2 <head>
3 <style type='text/css'>
4   p.italic {font-style:italic;}
5   p.bold {font-weight:bold;}
6 </style>
7 </head>
8
9 <p class='italic'>Данный абзац нап
10 <p class='bold'>Данный абзац напис
11
12 </html>
13
```

*Данный абзац написан курсивным шрифтом*

**Данный абзац написан жирным шрифтом**

# Оформление фона

## **background:**

background-attachment

background-color

background-image

background-position

background-repeat

```
body  
{ background-color:green; }
```

```
body  
{  
background-image:url('http://  
www.wisdomweb.ru/editor/sp  
ider2.gif'); }
```

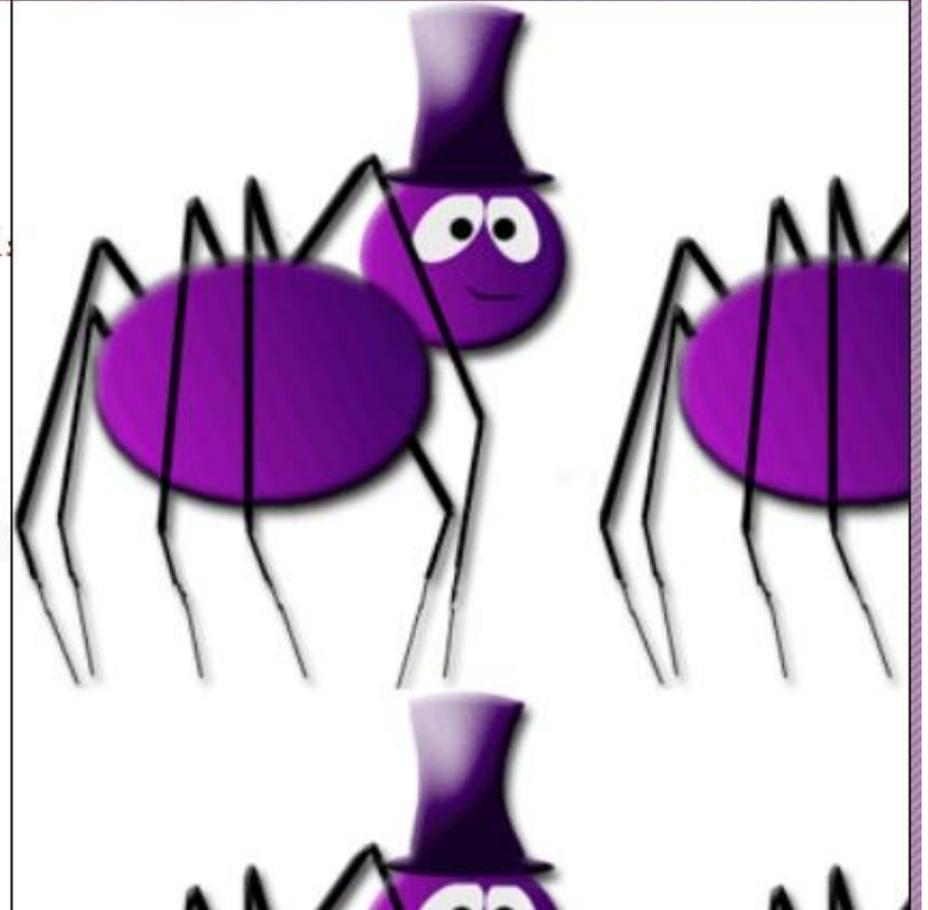
# Оформление фона: цвет

```
1 <html>
2 <head>
3 <style type='text/css'>
4 body
5 {
6   background-color:green;
7 }
8 </style>
9 </head>
10
11 <body>
12 <h2 style='color:white;'>С помощью
13 </body>
14
15 </html>
16
```

**С помощью CSS мы изменили цвет фона данной страницы на зеленый.**

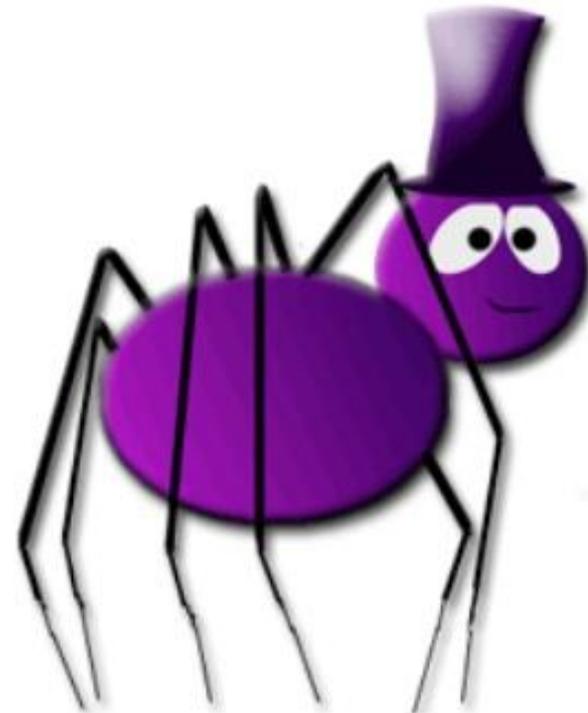
# Оформление фона: картинка

```
1 <html>
2 <head>
3 <style type='text/css'>
4 body
5 {
6   background-image:url('http://www.wi:
7 }
8 </style>
9 </head>
10
11 <body>
12 </body>
13 |
14 </html>
15
```



# Оформление фона: картинка

```
1 <html>
2 <head>
3 <style type='text/css'>
4 body
5 {
6 background-image:url('http://www.
7 background-repeat:no-repeat;
8 background-position:40px 60px;
9 }
10 </style>
11 </head>
12 <body>
13 </body>
14 </html>
15
```



Свойство	Описание	Значение
<b>background</b>	Устанавливает все возможные свойства фона за одно определение	<b>background-color</b> <b>background-image</b> <b>background-repeat</b> <b>background-attachment</b> <b>background-position</b>
<b>background-attachment</b>	Указывает будет ли фоновая картинка привязана к одному месту, или будет прокручиваться вместе с текстом.	<b>scroll</b> <b>fixed</b>
<b>background-color</b>	Устанавливает фоновый цвет элемента.	<b>rgb(x,x,x)</b> <b>#xxx</b> <b>название_цвета</b>
<b>background-image</b>	Устанавливает фоновую картинку для элемента	<b>url(URL)</b> <b>none</b>
<b>background-repeat</b>	Указывает как фоновая картинка будет повторяться	<b>repeat</b> <b>repeat-x</b> <b>repeat-y</b> <b>no-repeat</b>
<b>background-position</b>	Указывает координаты расположения фоновой картинки.	<b>left top</b> <b>left center</b> <b>left bottom</b> <b>center top</b> <b>center center</b> <b>center bottom</b> <b>right top</b> <b>right center</b> <b>right bottom</b>  <b>x% y%</b> <b>xpos ypos</b>

# Оформление ссылок

**a:link** – определяет оформление обычной не посещенной ссылки.

**a:visited** – определяет оформление посещенной пользователем ссылки.

**a:hover** – определяет оформление ссылки, на которую наведен курсор мыши.

**a:active** – определяет оформление ссылки, на которую щелкнули мышкой.

# Оформление ссылок

```
1 <html>
2 <head>
3 <style type='text/css'>
4 a:link
5 {
6 text-decoration:none;
7 color:green;
8 }
9 a:visited
10 {
11 text-decoration:none;
12 color:green;
13 }
14 a:hover
15 {
16 text-decoration:underline;
17 color:red;
18 font-size:1.1em;
19 }
20 a:active
21 {
22 text-decoration:none;
23 color:red;
24 font-size:1.1em;
25 }
```

[wisdomweb.ru](http://wisdomweb.ru)

Попробуйте навести курсор мыши на ссылку выше.

# Оформление списков

```
1 <html>
2 <head>
3 <style type='text/css'>
4 ul.lis1
5 {
6 list-style-type:square;
7 }
8 ol.lis2
9 {
10 list-style-type:upper-roman;
11 }
12 </style>
13 </head>
14 <body>
15 <p> Крупнейшие города России: </
16 <ul class='lis1'>
17 <li>Москва</li>
18 <li>Санкт-Петербург</li>
19 <li>Новосибирск </li>
20 </ul>
21 <i> Маркер списка имеет вид квад
22 <p> Города России отсортированн
23 <ol class='lis2'>
24 <li>Москва</li>
25 <li>Санкт-Петербург</li>
```

Крупнейшие города России:

- Москва
- Санкт-Петербург
- Новосибирск

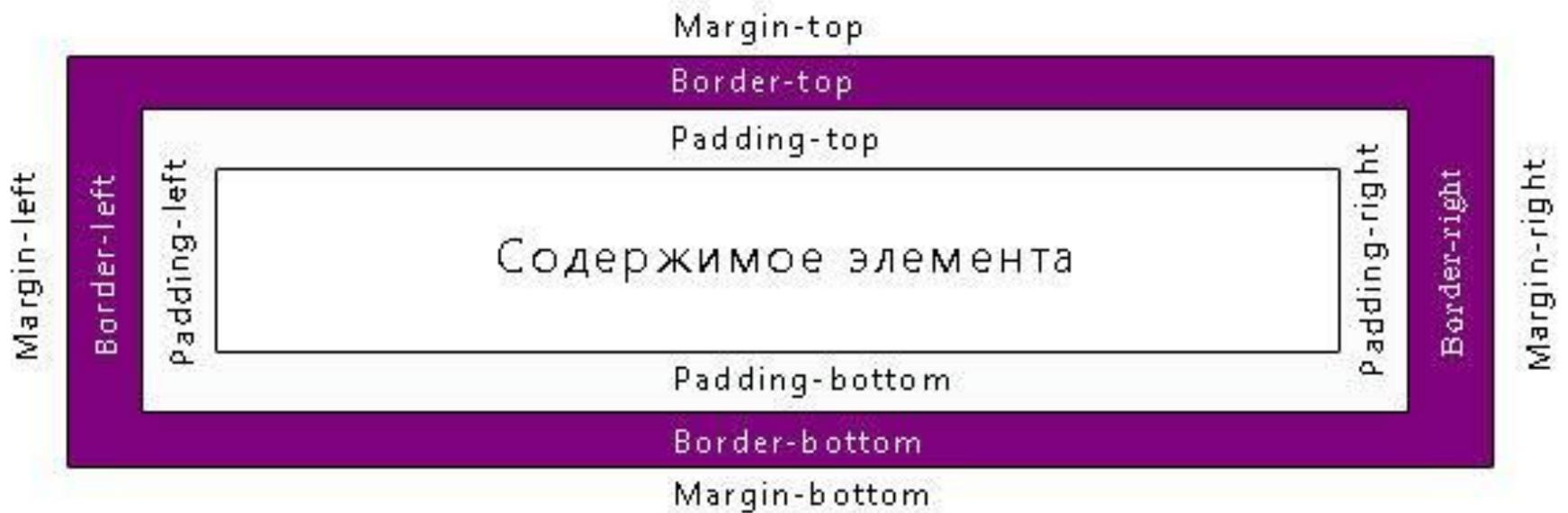
*Маркер списка имеет вид квадрата (по умолчанию имеет вид круга).*

Города России отсортированные по численности населения:

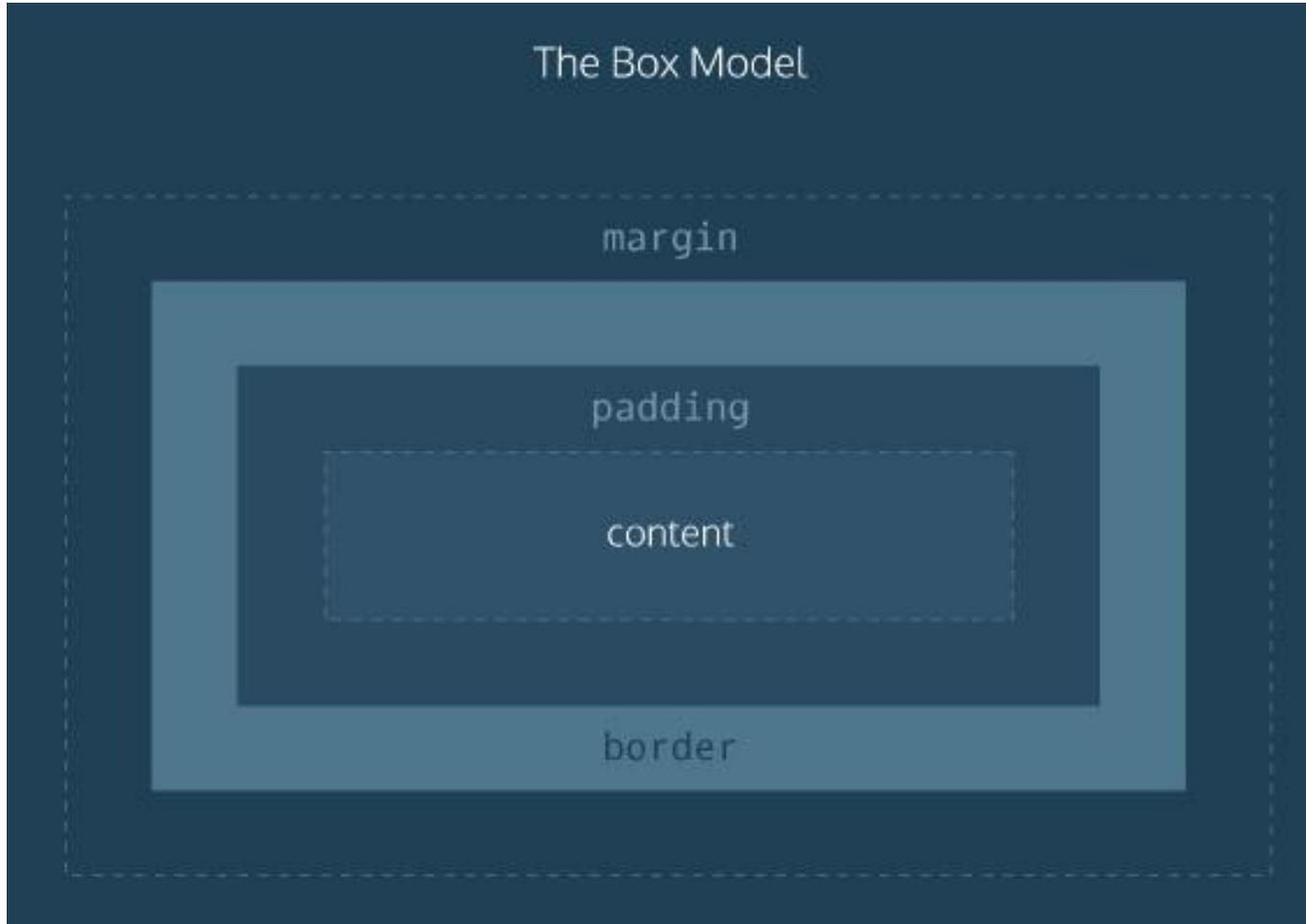
- I. Москва
- II. Санкт-Петербург
- III. Новосибирск

*Элементы списка нумеруются римскими цифрами (по умолчанию арабскими).*

# Блоковая модель

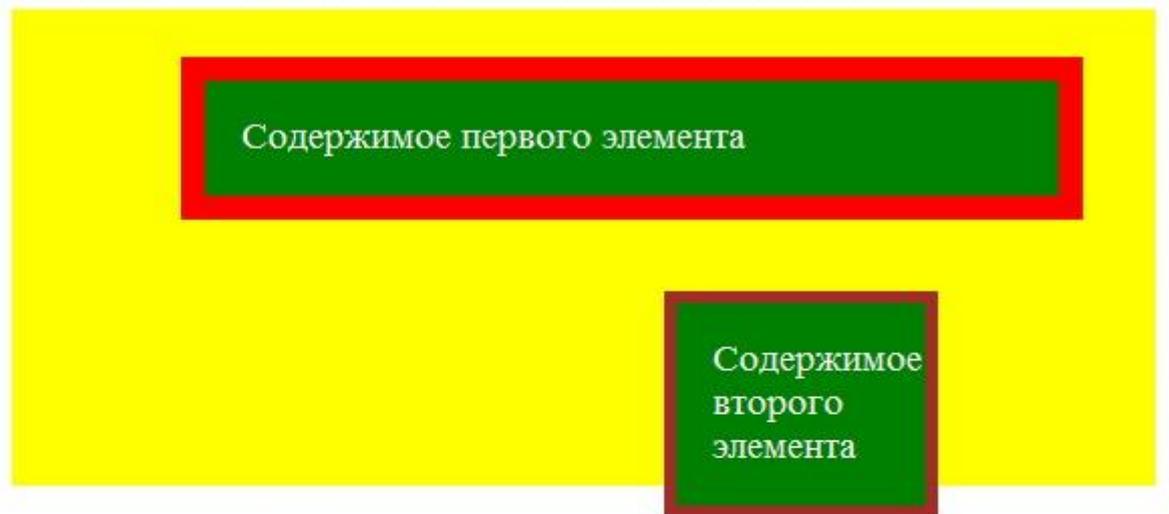


# Блоковая модель



# Блоковая модель

```
1 <html>
2 <head>
3 <style type='text/css'>
4 #wrap {
5 margin:0px;
6 padding:20px;
7 height:160px;
8 background-color:yellow;
9 }
10 .ex1 {
11 border:10px red solid;
12 margin-left:50px;
13 margin-right:10px;
14 padding:15px;
15 background-color:green;
16 color:white;
17 }
18 .ex2 {
19 border:5px brown solid;
20 margin-top:30px;
21 margin-left:250px;
22 margin-right:70px;
23 padding:15px;
24 background-color:green;
25 color:white;
26 }
```



# Стиль границ

```
1 <html>
2 <head>
3 <style type='text/css'>
4 .bor1
5 {
6   border-style:solid;
7   border-width:4px;
8   border-style:dashed;
9 }
10 .bor2
11 {
12   border-style:solid;
13   border-width:2px;
14   border-color: green;
15 }
16 .bor3
17 {
18   border-style:solid;
19   border-width:thin;
20 }
21 </style>
22 </head>
23 <body>
24 <p class='bor1'>Данный элемент имеет г
25 <p class='bor2'>Данный элемент имеет г
```

Данный элемент имеет границу толщиной 4 пикселя.

Данный элемент имеет границу толщиной 2 пикселя.

Толщина границы установлена с помощью предопределенного значения thin.

# Стиль границы в одной строке

```
1 <html>
2 <head>
3 <style type='text/css'>
4 /* Вокруг элемента с классом bor1 будет
5 толщиной 2 пикселя */
6 .bor1
7 {
8     border:2px solid green;
9 }
10 </style>
11 </head>
12 <body>
13 <p class='bor1'>Данный элемент был оформ
14 </body>
15 </html>
16
```

Данный элемент был оформлен с помощью CSS.

# Padding: внутренний отступ

```
1 <html>
2 <head>
3 <style type='text/css'>
4 .pad1
5 {
6   border-style:solid;
7   padding:20px;
8 }
9 .pad2
10 {
11   border-style:solid;
12   padding:10px;
13 }
14 .pad3
15 {
16   border-style:solid;
17   padding:5px;
18 }
19 </style>
20 </head>
21 <body>
22 <p class='pad1'>Элемент с внутренним о
23 <p class='pad2'>Элемент с внутренним о
24 <p class='pad3'>Элемент с внутренним о
25 </body>
```

Элемент с внутренним отступом равным 20 пикселям.

Элемент с внутренним отступом равным 10 пикселям.

Элемент с внутренним отступом равным 5 пикселям.

# Padding: внутренний отступ

```
1 <html>
2 <head>
3 <style type='text/css'>
4 /* Отступ от содержимого элемента до его
5 .pad1
6 {
7   border-style:solid;
8   padding-top:30px;
9   padding-left:20px;
10  padding-bottom:10px;
11  padding-right:40px;
12 }
13 </style>
14 </head>
15 <body>
16 <p class='pad1'>Элемент с заданным внутр
17 </body>
18 </html>
19
```

Элемент с заданным внутренним отступом.

# Padding: краткая форма записи

```
1 <html>
2 <head>
3 <style type='text/css'>
4 /* Отступ от содержимого до границы эле
5 .pad1
6 {
7 border:2px solid;
8 padding:60px 20px 40px 50px;
9 }
10 /* Отступ от содержимого до границы эле
11 .pad2
12 {
13 border:2px solid;
14 padding:40px 30px 10px;
15 }
16 /* Отступ от содержимого до границы эле
17 .pad3
18 {
19 border:2px solid;
20 padding:40px 30px;
21 }
22 </style>
23 </head>
24 <body>
25 <p class='pad1'>Внутренний отступ свер:
```

Внутренний отступ сверху равен 60, справа 20, снизу 40, а слева 50 пикс.

Внутренний отступ сверху равен 40, слева и справа 30, а снизу 10 пикс.

Внутренний отступ сверху и снизу равен 40, а слева и справа 30 пикс.

# Margin: ВНЕШНИЙ ОТСТУП

```
1 <html>
2 <head>
3 <style type='text/css'>
4 .dvl
5 {
6   border-style:solid;
7   border-width:1px;
8   background-color:red;
9 }
10 .mar1, .mar2, .mar3
11 {
12   border-width:1px;
13   border-style:solid;
14   background-color:white;
15 }
16 .mar1
17 {
18   margin:25px;
19 }
20 .mar2
21 {
22   margin:10px;
23 }
24 .mar3
25 {
```

Элемент с внешним отступом 25 пикселей.

Элемент с внешним отступом 10 пикселей.

Элемент с внешним отступом 5 пикселей.

**Обратите внимание:** внешний отступ в данном примере для наглядности выделен красным цветом.