

Урок 3

ОСНОВЫ CSS

Содержание урока

- Что такое CSS
- Синтаксис CSS
- Способы объявления CSS
- Селекторы (id, class, tag)
- Селекторы атрибутов
- Основные свойства стилей
- Вложенность, наследование и группирование свойств
- Приоритеты применения стилей
- Псевдоклассы и псевдоэлементы

Что такое CSS

CSS - Cascading Style Sheets – Каскадные
листы стилей

HTML – структура документа

CSS – его оформление

Так делать не надо!!!

```
<body bgcolor="green">  
  <h1 align="center">  
    <font color="red">Заголовок</font>  
  </h1>  
</body>
```

Синтаксис CSS

```
Селектор {  
    свойство1: значение1;  
    свойство2: значение2;  
}
```

Оформление

Селектор { свойство1: значение1; свойство2: значение2; }

Селектор

```
{  
    свойство1: значение1;  
    свойство2: значение2;  
}
```

```
Селектор {  
    свойство1: значение1;  
    свойство2: значение2;  
}
```

Комментарии

/* здесь пишем комментарий */

/*

и здесь можно

тоже

его указать

*/

Способы объявления CSS

1. inline-стили
2. CCS в разделе head
3. Подключение внешнего файла

Inline-стили

```
<div style="свойство1: значение1;  
            свойство2: значение2; ">
```

```
<body style="background: #0f0;" >  
  <h1 style="text-align: center; color: #f00;">
```

Заголовок

```
</h1>
```

```
</body>
```

Стили в разделе head

```
<head>
  <style type="text/css">
    body {
      СВОЙСТВО1: значение1;
      СВОЙСТВО2: значение2;
    }
  </style>
</head>
<body>
  ...
</body>
```

```
<head>
  <style type="text/css">
    body {
      background: #0f0;
    }
    h1 {
      text-align: center;
      color: #f00;
    }
  </style>
</head>
<body>
  <h1>Заголовок</h1>
</body>
```

Подключение внешнего файла css/style.css

```
body {  
    СВОЙСТВО1: значение1;  
    СВОЙСТВО2: значение2;  
}
```

index.html

```
<head>  
  <link  
    rel="stylesheet"  
    href="css/style.css"  
    type="text/css" />  
</head>
```

Какой способ выбрать?

Inline-стили

- + Можно быстро прописывать стиль для определенного элемента
- Необходимо прописывать стили для каждого тега
- Сложно редактировать

Стили в разделе head

- + Можно прописывать стили для нескольких элементов
- Необходимо добавлять стили для каждой страницы

Подключение файла

- + Можно применить стили ко всему сайту
- + Удобно редактировать
- + Файл кэшируется

Селекторы

Tags

```
h1 {  
    background: #ccc;  
}
```

<h1>

Для всех заголовков первого уровня цвет
фона будет серым

</h1>

id

```
#first {  
  color: red;  
}
```

```
<p id="first">
```

цвет этого абзаца будет красным

```
</p>
```

!!! id должен быть уникальным !!!

class

```
.is_border {  
    border: 1px solid #000;  
}
```

```
<h1 class="is_border">
```

Заголовок с рамкой

```
</h1>
```

```
<p class="is_border">
```

параграф с рамкой

```
</p>
```

Селекторы атрибутов

```
img[title]{  
    width: 200px;  
}
```

```
input[type="text"]{  
    font-size: 18px;  
}
```

```

```

```
<input type="text" />
```

Основные свойства стилей

Единицы измерения в CSS

Относительные

- `px` - пиксел
- `%` - процент
- `em` — высота текущего шрифта
- `ex` — высота буквы x

Абсолютные

- `cm` - сантиметр
- `mm` — миллиметр
- `in` - дюйм
- `pt` - пункт
- `pc` - пика

Цвета в CSS

- Именованные

red, green, blue, black, yellow, white ...

- Функциональные RGB

RGB(255, 130, 0) 0 - 255

RGB(100%, 70%, 0%) 0 – 100%

- Шестнадцатичные RGB

#FA96CF; 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

#FFAA00 => #FA0

background – фон элемента

- background-color: #ff0;
- background-image: url(img/foto.jpg);
- background-position: top; (bottom | left | right)
- background-repeat: repeat-x; (repeat-y | no-repeat)
- background-attachment: fixed;

```
background: #ff0 url(img/foto.jpg) top repeat-x;
```

border - рамка

- **border-color:** red; (#f00 | RGB(255, 0, 0))
- **border-style:** solid; (dotted | solid | double | inset | outset)
- **border-width:** 2px;
(1px 2px) - 1px: верхняя и нижняя, 2px: левая и правая
(1px 2px 3px) - 1px: верхняя, 2px: левая и правая, 3px: нижняя
(1px 2px 3px 4px) - 1px: верхняя, 2px: правая, 3px: нижняя, 4px: левая

border: 1px solid black;

`border-top: 2px dotted green;`
`border-bottom: 3px double blue;`
`border-left: 1px solid red;`
`border-right: 4px inset #000;`

color – цвет текста

color: red;

color: #78fa2e;

color: RGB(34, 21, 56);

width, height – ширина и высота

width: 200px;

height: 300px;

font – шрифт текста

- font-family: "Times New Roman", serif, Verdana;
- serif — шрифты с засечками
- sans-serif — рубленые шрифты, без засечек
- cursive — курсивные шрифты
- fantasy — декоративные шрифты
- monospace — моноширинные шрифты

font – шрифт текста

- font-style: italic; (oblique | normal)
- font-variant: small-caps;
- font-weight: bold; (bolder | lighter | 100 | 200)
- font-size: 20px; (small | medium | large)

font: font-style
font-variant
font-weight
font-size
font-family;

font: bold 24px Arial, Verdana;

list-style – вид маркера

- `list-style-type: circle;` (disc | square | armenian | decimal)
- `list-style-position: inside;`
- `list-style-image: url(img/list.png);`

list-style: list-style-type
list-style-position
list-style-image;

list-style: circle inside;

- **text-align:** center; (justify | left | right)
- **text-decoration:** none; (blink | line-through | overline | underline | none)
- **text-transform:** capitalize; (lowercase | uppercase)

Вложенность

Контекстные селекторы

```
p.main strong a {  
  font-size: 28px;  
  color: red;  
}
```

<p class="main">В этом параграфе
 эта ссылка будет
 размером 28 px и красного цвета,
 а эта будет обычной.

</p>

<p>А в этом параграфе
 ссылка тоже будет обычной.
</p>



В этом параграфе **ссылка** будет размером 28 px и красного цвета, а эта будет обычной.

А в этом параграфе ссылка тоже будет обычной.

```
p.main a {  
  font-size: 28px;  
  color: red;  
}
```

```
<p class="main">В этом параграфе  
  <strong><a href="#">эта ссылка</a></strong> будет  
  размером 28 px и красного цвета,  
  <a href="#">и эта тоже</a>.  
</p>
```

```
<p>А в этом параграфе  
  <a href="#">ссылка</a> тоже будет обычной.  
</p>
```



В этом параграфе **ссылка** будет размером 28 px и красного цвета, **и эта тоже**.

А в этом параграфе ссылка тоже будет обычной.

Дочерние селекторы

```
p.main > a {  
    font-size: 20px;  
    color: green;  
    background: yellow;  
}
```

```
<p class="main">  
    <a href="#">Ссылка 1</a>
```

```
</p>
```

```
<p class="main">  
    <em><a href="#">Ссылка 2</a></em>
```

```
</p>
```



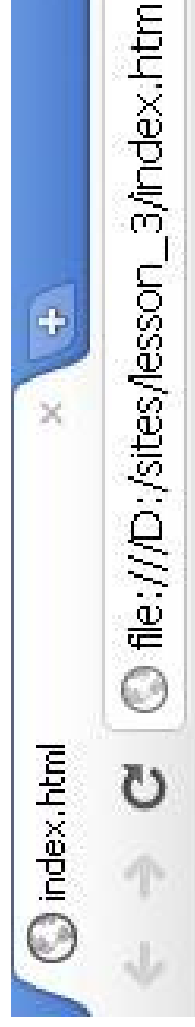
Ссылка 1

Ссылка 2

Соседние селекторы

```
p. main em + a {  
  font-size: 20px;  
  color: green;  
  background: yellow;  
}
```

```
<p class="main">  
  <a href="#">Ссылка 1</a>  
</p>  
<p class="main">  
  <em>Соседним элементом будет</em>&nbsp;<a href="#">Ссылка 2</a>  
</p>
```



[Ссылка 1](#)

Соседним элементом будет

[Ссылка 2](#)

Наследование


```
p {  
  font-size: 18px;  
  color: red;  
}
```

<p>В этом случае весь текст в этом параграфе
 будет красного цвета и шрифтом
 <i>размером 18 px</i>
</p>

Но!!!

не все свойства CSS наследуются

```
p {  
    font-size: 18px;  
    color: red;  
}
```

<p>В этом случае весь текст в этом параграфе
 будет красного цвета и шрифтом
 <i>размером 18 px</i>, только
 эта ссылка не будет красной
</p>

```
p {  
  font-size: 30px;  
  color: red;  
}  
p a {  
  color: inherit;  
}
```

**В этом случае весь текст в этом параграфе
будет** красного цвета и шрифтом
размером 18 px, включая
[эту ссылку](#)

Группирование свойств

```
h1 {  
    color: blue;  
    text-align: center;  
    font-family: Verdana;  
}
```

```
h3 {  
    color: blue;  
    text-align: center;  
    font-family: Arial;  
}
```

```
p {  
    color: blue;  
    text-align: center;  
    font-size: 12px;  
}
```

```
h1, h3, p {  
    color: blue;  
    text-align: center;  
}  
h1 {  
    font-family: Verdana;  
}  
h3 {  
    font-family: Arial;  
}  
p {  
    font-size: 12px;  
}
```

Приориты применения стилей

Почему каскадные?

Каскадирование применяется тогда, когда одному и тому же элементу пытаются присвоить разные стили.

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

Приоритеты источников стилей

1. Стиль автора документа
2. Стиль, заданный пользователем в настройках браузера
3. Стиль самого браузера

Приоритеты стилей автора

1. !important

```
p {  
    color: red !important;  
}
```

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

!!! Стиль пользователя важнее стиля автора !!!

2. Стил в теге style

```
h1 {  
    color: green;  
}
```

```
<h1 style="color: red;">Цвет будет красным</h1>
```

3. Уровень приоритета селекторов (специфичность)

- *Селекторы тегов* и *псевдоэлементы* — по 1 баллу (0, 0, 0, 1)
- *Селекторы атрибутов, классы* и *псевдоклассы* — по 10 баллов (0, 0, 1, 0)
- *Идентификаторы* — по 100 баллов (0, 1, 0, 0)
- Атрибут style – 1000 баллов (1, 0, 0, 0)

- ✓ `p { }` - 1 балл (селектор тегов)
- ✓ `p::first-letter { }` - 2 балла (1 - селектор тегов и 1 - псевдоэлемент)
- ✓ `input[type="submit"] { }` - 11 баллов (по 1 селектору тегов и атрибутов)
- ✓ `div.head .new { }` - 21 балл (2 класса и 1 селектор тегов)
- ✓ `#header a:hover { }` - 111 баллов (идентификатор, селектор тегов и псевдокласс)

4. Стили, заданные в разделе `<head>`
5. Стили, во внешних файлах
6. Наследуемые стили от предков

Псевдоклассы и псевдоэлементы

Псевдоклассы

```
Селектор:псевдокласс {  
    свойство:значение;  
}
```

```
a:hover{  
    color: #ccc;  
}
```

Определяющие состояние

- **:hover** {} – курсор мыши в пределах элемента
- **:active** {} – при активации элемента
- **:focus** {} – при получении фокуса элемента
- **:link** {} – непосещенные ссылки
- **:visited** {} – посещенные ссылки

```
a:link {  
    color: #0D0;  
    font-family: monospace;  
}  
a:visited {  
    color: #F00;  
    font-family: Arial;  
}  
a:hover {  
    color: #FFF;  
    font-family: Verdana;  
    background: #911;  
    font-weight: bold;  
}  
a:active {  
    color: #001;  
    font-family: "Times New Roman";  
}
```

Выбор первого дочернего элемента

```
li:first-child{  
    font-size: 24px;  
    color: #F23401;  
}
```

```
<ul>  
    <li>Первый элемент</li>  
    <li>Второй элемент</li>  
    <li>Третий элемент</li>  
</ul>
```



- **Первый элемент**
- Второй элемент
- Третий элемент

Управления языками

```
strong:lang(en){  
    color: #f00;  
}
```

```
<p><strong>Меня зовут</strong> -  
    <strong lang="en">My name is</strong>  
</p>
```



Меня зовут - **My name is**

Комбинирование псевдоклассов

```
a:link:hover{  
    color: #FD56SE;  
}
```

```
a:visited:hover{  
    color: #FD5;  
}
```


Псевдоэлементы

- :first-letter {}
- :first-line {}
- :after {}
- :before {}