

HTML CSS

**Урок 3**



# ОСНОВЫ CSS

# План урока

1. Что такое CSS
2. Синтаксис CSS
3. Способы объявления CSS
4. Селекторы (id, class, tag)
5. Селекторы атрибутов
6. Основные свойства стилей
7. Вложенность, наследование и группирование свойств



# Что такое CSS

CSS (англ. Cascading Style Sheets — каскадные таблицы стилей) — формальный язык описания внешнего вида документа, написанного с использованием языка разметки. Преимущественно используется как средство описания, оформления внешнего вида веб-страниц.

Простыми словами CSS дополнение к HTML, которое значительно расширяет его возможности.



# Синтаксис CSS

Селектор {

    свойство1: значение1;

    свойство2: значение2;

}

Пример:

p {

    color: blue;

}



# Оформление

1. Селектор {свойство1: значение1;свойство2: значение2;}
2. Селектор  
{  
    свойство1: значение1;  
    свойство2: значение2;  
}
3. Селектор {  
    свойство1: значение1;  
    свойство2: значение2;  
}



# Комментарии

```
/*Внешний вид*/
```

```
p {  
    color: blue;  
}
```

```
/*  
Стили  
для  
параграфа  
*/
```



# Способы объявления CSS

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



# 1. inline-стили

Плюсы:

- + Можно быстро прописывать стиль для определенного элемента.

Минусы:

- Необходимо прописывать стили для каждого тега
- Сложно редактировать

Пример:

```
<body style="background:#0f0;">  
  <h1 style="color: blue;">  
    Заголовок  
  </h1>  
</body>
```





## 2. Стили в разделе head

Плюсы:

- + Можно прописывать стили для нескольких элементов.

Минусы:

- Необходимо добавлять стили для каждой страницы

Пример:

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



### 3. Подключение внешнего файла

Плюсы:

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

Пример:

```
body {  
    color: blue;  
    background: #0f0;  
}  
h1 {  
    text-align: center;  
}
```



# Подключение внешнего файла

index.html

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

style.css

```
body {  
    color: blue;  
    background: #0f0;  
}  
h1 {  
    text-align: center;  
}
```



# Селекторы



# Tags

HTML:

`<h1>`Для всех заголовков первого  
уровня цвет текста будет  
синим`</h1>`

CSS:

```
h1 {  
    color: blue;  
}
```



# id

HTML:

```
<p id="first">Цвет фона данного  
параграфа будет серым</p>
```

CSS:

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

Идентификатор в коде документа должен быть в единственном экземпляре,  
иными словами, встречаться только один раз.



# class

HTML:

```
<h1 class="border">Заголовок с  
рамкой</h1>
```

```
<p class="border">Параграф с  
рамкой</p>
```

CSS:

```
.border {  
    border: 1px solid black;  
}
```

В значении допускается указывать сразу несколько классов, разделяя их между собой пробелом.



# Селекторы атрибутов

HTML:

```

```

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

CSS:

```
img[alt] {  
    width: 100px;  
}
```

```
input[type="text"] {  
    font-size: 10px;  
}
```





# Основные свойства стилей



# Единицы измерения в CSS

## Относительные:

- px - пиксел
- % - процент
- em – высота текущего шрифта
- ...

## Абсолютные:

- cm -сантиметр
- mm – миллиметр
- in - дюйм
- pt - пункт
- ...



# Цвета в CSS

## Именованные

red, green, blue, black

...

## Функциональные RGB

RGB(255, 130, 0)

RGB(100%, 70%, 0%)

0 - 255, 0 - 100%

## Шестнадцатичные RGB

#FA96CF;

#FFAA00 => #FA0;

(0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F)



# width, height – ширина и высота

```
h1 {  
    width: 300px;  
    height: 200px;  
}
```



# 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;`

Объединённое значение:

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

(любое из свойств может отсутствовать)



# border – рамка

- **border-color**: red; (#f00 | RGB(255, 0, 0));
- **border-style**: solid; (dotted | dashed | groove | ridge | 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



# Добавление рамки

Для создания рамки вокруг текста, изображения или блока используется объединённое свойство:

```
h1 {  
    border: 1px solid black;  
}
```

(однопиксельная сплошная рамка, вокруг заголовка)



# color – цвет текста

- `color: red;`
- `color: #78fa2e;`
- `color: RGB(34, 21, 56);`





# 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 | bold);
- **font-variant**: small-caps;
- **font-weight**: bold; (bolder|lighter| 100 | 200);
- **font-size**: 20px; (small | medium | large);

Объединённое значение:

**font**: bold 24px Arial, Verdana;



# list-style – вид маркера

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

Универсальное свойство, позволяющее одновременно задать стиль маркера, его положение, а также изображение, которое будет использоваться в качестве маркера.



# Редактирование текста

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



# Вложенность



# Контекстные селекторы

HTML:

```
<p class="main">В этом параграфе  
  <strong><a href="#">эта ссылка</a></strong>  
  будет размером 18 px и красного цвета,  
  <a href="#">а эта будет обычной</a>.  
</p>
```

CSS:

```
.main strong a {  
  font-size: 18px;  
  color: red;  
}
```



# Дочерние селекторы

HTML:

```
<p class="main">
  <a href="#">ссылка 1</a>
</p>
<p class="main">
  <i>
    <a href="#">ссылка 2</a>
  </i>
</p>
```

CSS:

```
.main > a {
  font-size: 18px;
  color: red;
}
```



# Соседние селекторы

HTML:

```
<p class="main">  
  <a href="#">ссылка 1</a>  
  
</p>  
  
<p class="main">  
  <i>Соседний элемент</i>  
  
  <a href="#">ссылка 2</a>  
  
</p>
```

CSS:

```
.main i + a {  
  font-size: 18px;  
  color: red;  
}
```





# Наследование

HTML:

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

CSS:

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



# Не все свойства CSS наследуются

HTML:

```
<p>В этом параграфе весь текст  
<b>будет</b>красного цвета и шрифтом  
<b><i>размером 18 px</i><b>, только  
<a href="#"><u>эта ссылка</u></a> не будет красной  
</p>
```

CSS:

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



# Наследование свойств для ссылок

HTML:

```
<p>В этом параграфе весь текст  
<b>будет</b>красного цвета и шрифтом  
<b><i>размером 18 px</i><b>, включая  
<a href="#">эту ссылку</a>  
</p>
```

CSS:

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



# Группирование свойств



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

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



# Приоритеты применения стилей



# Почему каскадные?

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

```
h1 {  
    color: black;  
}
```

```
h1 {  
    color: red;  
}
```

(Цвет заголовка будет **красного цвета**)



# 1. Приоритеты стилей автора

CSS-свойство получит максимальный приоритет в каскаде стилей, при добавлении `!important`.

```
h1 {  
    color: black !important;  
}
```

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

(Цвет заголовка будет черного цвета)





## 2. Стиль в теге style

HTML:

```
<h1 style="color: red;">
```

Заголовок первого уровня

```
</h1>
```

CSS:

```
h1 {
```

```
color: green;
```

```
}
```

Цвет текста будет красным.



3. Уровень приоритета селекторов
4. Стили, заданные в разделе <head>
5. Стили, во внешних файлах
6. Наследуемые стили от предков



# Домашнее задание

- Создайте файл `style.css`
- Создание стилей для интернет магазина
- Сделанные ДЗ - это ваше будущее портфолио. Это Важно!
- Портфолио - основное, что интересует работодателя.



Вопросы участников ...

