

CSS

Основные понятия

Структура HTML-документа

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title>Пример</title>  
    <meta charset="utf-8" />  
  </head>  
  <body>  
    <p>Текст</p>  
  </body>  
</html>
```

Определение

CSS (англ. *Cascading Style Sheets* — каскадные таблицы стилей) — формальный язык описания внешнего вида документа, написанного с использованием языка разметки.

Подключение CSS (1 вариант)

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title>Пример</title>  
    <meta charset="utf-8"/>  
    <link rel="stylesheet" href="style.css"/>  
  </head>  
  <body>  
    <p>Текст</p>  
  </body>  
</html>
```

Подключение CSS (2 вариант)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Пример</title>
    <meta charset="utf-8"/>
    <style media="all">
      @import url(style.css);
    </style>
  </head>
  <body>
    <p>Текст</p>
  </body>
</html>
```

Подключение CSS (3 вариант)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Пример</title>
    <meta charset="utf-8"/>
    <style>
      body {
        color: red;
      }
    </style>
  </head>
  <body>
    <p>Текст</p>
  </body>
</html>
```

Синтаксис CSS

```
селектор, селектор {  
    свойство: значение;  
    свойство: значение;  
    свойство: значение;  
}
```

Синтаксис CSS

```
p, .warning {  
  font-size: 10px;  
  font-weight: bold;  
  color: red;  
}
```

Селектор тэгов

HTML:

```
<p>Жирный текст</p>
```

CSS:

```
p{  
  font-weight: bold;  
}
```

Селектор классов

HTML:

```
<p class="warning">Внимание</p>
```

CSS:

```
.warning{  
    font-weight: bold;  
}
```

Тэг с несколькими классами

HTML:

```
<p class="warning text">Внимание</p>
```

CSS:

```
.warning{  
    font-weight: bold;  
}  
.text{  
    font-size: 14px;  
}
```

Селектор идентификаторов

HTML:

```
<p id="warning">Внимание</p>
```

CSS:

```
#warning{  
    font-weight: bold;  
}
```

Селектор атрибутов

HTML:

```
<a href="google.by">Ссылка</a>
```

CSS:

```
a[href="google.by"]{  
  font-weight: bold;  
}
```

Контекстный селектор

HTML:

```
<p>Внимание<b> на экран</b></p>
```

CSS:

```
p b{  
    font-weight: bold;  
}
```

Контекстный селектор

HTML:

```
<p>Внимание  
  <b class="warning"> на экран</b>  
</p>
```

CSS:

```
p .warning{  
  font-weight: bold;  
}
```

Селектор дочерних элементов

HTML:

```
<p>  
  <i>Это дочерний элемент</i>  
  <b>  
    <i>Это не дочерний элемент</i>  
  </b>  
</p>
```

CSS:

```
p > i {  
  font-color: yellow;  
}
```

Селектор сестринских элементов

HTML:

```
<p>  
  <b>Сестра</b>  
  <i>Сестра</i>  
  <h2><b>Не сестра</b></h2>  
</p>
```

CSS:

```
b + i {  
  font-color: yellow;  
}
```

Селектор псевдоклассов

HTML:

```
<p>Текст</p>
```

CSS:

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

Селектор псевдоэлементов

HTML:

`<p>Текст</p>` 

CSS:

```
p::first-letter{  
  color: red;  
  font-size: 60px;  
}
```

Универсальный селектор

HTML:

```
<p>Красный текст</p>
```

```
<b>Красный текст</b>
```

CSS:

```
*{
```

```
color: red;
```

```
}
```

Наследование

HTML:

```
<p>текст<b>полужирный</b></p>
```



текст **полужирный**

CSS:

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

Каскадирование

HTML:

```
<p id="p">текст<b>полужирный</b></p>
```

текст полужирный

CSS:

```
p{  
    color: yellow;  
}
```

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

```
b{  
    color: red;  
}
```

Приоритеты

1. Стили браузера
2. Пользовательские стили в браузере
3. Стили разработчиков
 - a) количество идентификаторов (#id) в селекторе – (1,0,0)
 - b) количество классов (.class), атрибутов ([attr], [attr="value"]) и псевдоклассов (:pseudo-class) - (0,1,0)
 - c) количество элементов (h1, input) и псевдоэлементов (::pseudo-element) - (0,0,1)
4. Атрибут **style**
5. Свойства с **!important**

Стили разработчиков

HTML:

```
<p class="text s-text">ТЕКСТ  
  <b id="warning" class="important">жирный  
    <i class="italic"> и курсивный</i>  
  </b>  
</p>
```



CSS:

```
p.text.s-text .important .italic{  
  color: yellow;  
}  
  
.text #warning i{  
  color: red;  
}  
#warning .italic{  
  color: green;  
}
```

Задание цвета

1. По имени цвета
2. Функции `rgb()` и `rgba()`
3. Функция `hsl()` и `hsla()`
4. Шестнадцатеричная запись (краткая и полная)

Имена цветов

```
p{  
  color: yellow;  
}
```

```
.colored{  
  color: red;  
}
```

```
.not-red{  
  color: green;  
}
```

Функции rgb() и rgba()

- Красный от 0 до 255
- Зеленый от 0 до 255
- Синий от 0 до 255
- Прозрачность от 0 до 1

Функции rgb() и rgba()

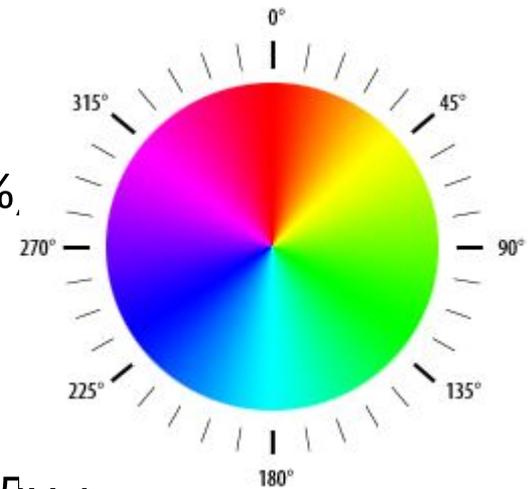
```
p{  
  color: rgb(255,255,0);  
}
```

```
.colored{  
  color: rgb(255,255,0,0.5);  
}
```

```
.not-red{  
  color: rgb(255,255,255);  
  background: rgb(0,0,0);  
}
```

Функция hsl() и hsla()

- **оттенок** — значение в диапазоне от 0 до 360; определяет, какой цвет вы хотите;
- **насыщенность** — в диапазоне от 0% до 100%; определяет, сколько этого цвета вы хотите;
- **яркость** — в диапазоне от 0% до 100%; определяет, насколько ярким вы желаете цвет.
- **прозрачность** — значение в диапазоне от 0 до 1.



Функция hsl() и hsla()

```
p{
```

```
  color: hsl(45,100%,100%);
```

```
}
```

```
.colored{
```

```
  color: hsla(45,100%,100%,0.5);
```

```
}
```

Шестнадцатеричная запись

- Красный от 00 до FF
- Зеленый от 0 до FF
- Синий от 0 до FF

Шестнадцатеричная запись

```
p{  
  color: #FFFF00;  
}
```

```
.colored{  
  color: #FF0; // равно #FFFF00  
}
```

```
.not-red{  
  color: #FFF;  
  background: #000;  
}
```

Единица измерения

1. Пиксели - 10px
2. Проценты – 10%, 200%
3. Относительные единицы
 1. em (от родительского элемента)
 2. rem (от корневого элемента (html)) для размера шрифта
4. Абсолютные единицы
 1. in (дюйм = 2.54 см)
 2. cm
 3. mm
 4. pt (пункт = 1/72 in)
 5. pc (пика = 12 pt)
5. От ширины окна
 1. vw (1% от ширины окна)
 2. vh (1% от высоты окна)
 3. vmin (меньшее из vw и vh)
 4. vmax (большее из vw и vh)