

# Урок 6. Введение в CSS

CSS – Cascading Style Sheet – каскадная таблица стилей. По сути в большинстве своём представляет собой файлы с расширением css и содержит набор правил, по которым стилизуются теги в HTML. Однако стилизация тегов HTML может быть и в самих тегах – инлайн-стилизация, без отдельного файла.

## Подключение CSS

Существует три основных способа стилизации тегов с помощью CSS:

1. Инлайн-стилизация
2. Стилизация с помощью тега `<style>`
3. Стилизация с помощью подключённого отдельно файла с расширением `.css`

Конечно, в большинстве случаев подключается внешний файл (используется 3й способ), для удобства чтения (чтобы не нагромождать лишнее в файле html). Но бывают моменты, когда необходимо использовать и другие способы, поэтому рассмотрим все.

## Инлайн-стилизация

Для того, чтобы стилизовать какой-то конкретный тег инлайн (это значит, что CSS-правила для него прописываются прямо внутри тега), нужно добавить тегу специальный атрибут `style` и в нём прописать необходимые правила.

```
<div style="color: red">Красный текст</div>
```

В данном теге мы задали красный цвет текста, используя правило `color`.

Правил может быть несколько, они разделяются точкой с запятой:

```
<div style="color: red; background-color: green">Красный текст</div>
```

В данном примере мы добавили новое правило `background-color`, и теперь красный текст находится на зелёном фоне.

Из примеров ясно, что структура правила в CSS следующая:

`название-правила: значение;`

## Стилизация с помощью тега `<style>`

В отличие от инлайн-стилизации здесь CSS-правила пишутся в одном месте: внутри тега `<style>`, который необходимо разместить внутри тега `<head>`:

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Стилизация с помощью тега style</title>
  <style>
    <!-- здесь будут правила CSS -->
    <style>
  </head>
<body>
</body>
</html>
```

В отличие от инлайн-стилизации, здесь все правила пишутся в одном месте, и необходимо сделать так, чтобы было понятно, для какого тега нужно применять то или иное правило. Для этого есть селекторы – специальные классы в CSS, объединяющие правила для определённых тегов.

```
<style>
  div {
    background-color: blue;
    color: red;
  }
</style>
```

В примере выше мы написали класс, используя селектор `div`, который выберет все элементы `<div>` на странице и применит к ним указанные правила.

Подробнее про селекторы можно почитать здесь:

[https://developer.mozilla.org/ru/docs/Web/CSS/CSS\\_Selectors](https://developer.mozilla.org/ru/docs/Web/CSS/CSS_Selectors)

## Стилизация в отдельном файле

В принципе, этот способ стилизации отличается от предыдущего только тем, что классы CSS с правилами вынесены в отдельный файл, а не располагаются прямо в теге `<style>` в html.

Для подключения отдельного файла необходимо сначала его создать рядом со страницей html. Пусть это будет `style.css`.

Для подключения используется тег `<link>`, который необходимо разместить внутри `<head>`:

```
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Подключение CSS-файла</title>
  <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
</body>
</html>
```

Тег `<link>` не имеет закрывающего тега. Для подключения стилей значения его атрибутов `rel` и `type` всегда одинаково. Меняется только атрибут `href`, значение которого является относительным путём до файла css (аналогично подключению картинки)

## Материалы

Подробнее введение в CSS описано в следующих материалах:

1. [https://developer.mozilla.org/ru/docs/Learn/CSS/First\\_steps](https://developer.mozilla.org/ru/docs/Learn/CSS/First_steps)
2. <https://html5book.ru/osnovy-css/>