# Курс лекций. Часть 2. Основы CSS.

Составитель: Нуромская НН.

# Таблицы каскадных стилей

**CSS Cascading Style Sheets (Таблицы каскадных стилей)** – это набор правил оформления и форматирования, который может быть применен к различным элементам страницы.

**При использовании CSS мы можем получить следующее:**

- добиться однообразия всех страниц сайта;
- существенно сократить размер сайта (так как не нужно будет писать кучу параметров к каждому тэгу);
- улучшить внешний вид сайта.

Любое правило каскадных таблиц стилей состоит из 2-х частей:

**селектора и определения.**

***Селектором*** может быть любой тег HTML, для которого определение задает, каким образом необходимо его форматировать.

***Определение*** также состоит из 2-х частей: свойства и его значения, разделенных знаком двоеточия.

Например, если необходимо, чтобы в документе все заголовки первого уровня отображались синим цветом и размером в 16 пунктов, то в таблице следует задать правило:

```
H1 {color: blue;
      font-size: 16pt}
```

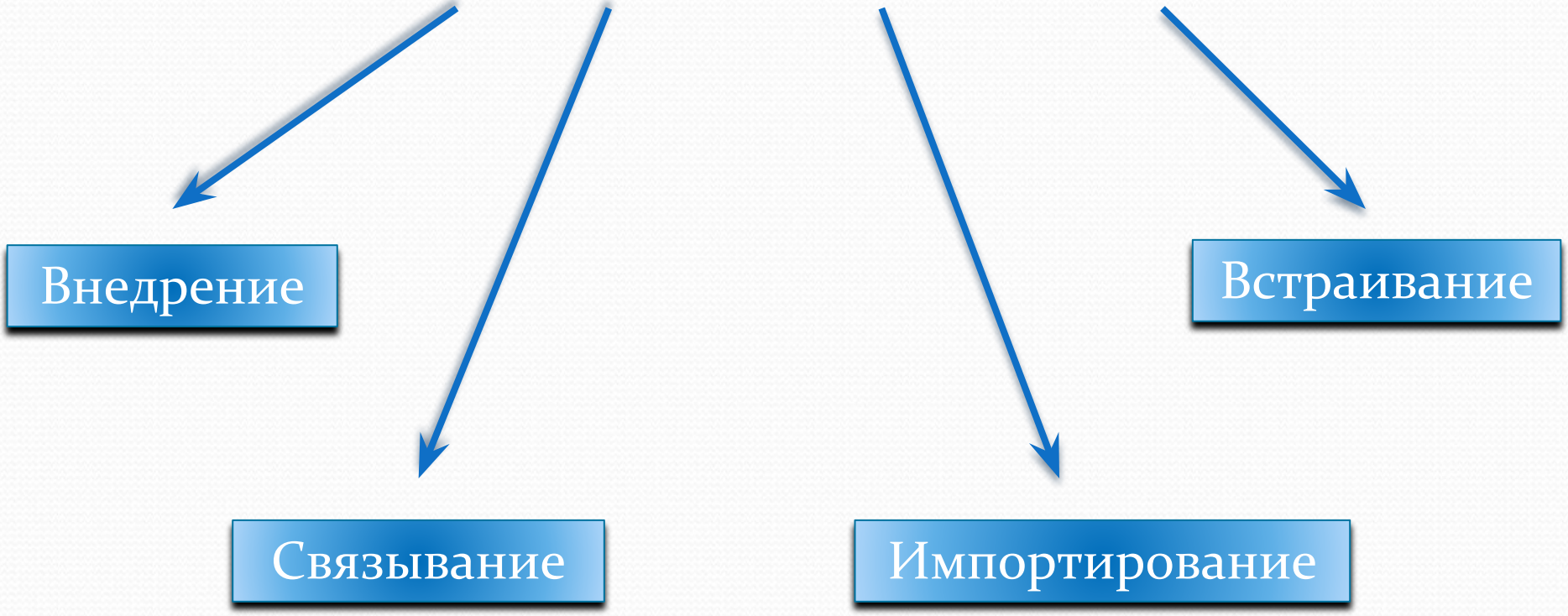
**H1** – селектор;

**{color: blue;**
**font-size: 16pt}** – определение;

**color, font-size –** свойства;

**blue, 16pt –** значение.

# Существует четыре способа встраивания CSS в документ:



# Внедрение

При **внедрении** описание стилей находится между тегами <STYLE></STYLE> и размешаются между тегами <HEAD></HEAD>

**Пример:**

```
<html>
<head>
<STYLE>
    <!--
    P {
    text-align : justify;
    font-size : 10pt;
    font-family : sans-serif;
    }
    -->
    </STYLE>
</head>
<body>
<p>Пример внедрения</p>
</body>
<html>
```

# Связывание

При **связывании** Вы располагаете стили в отдельном файле. Тогда между тегами

<HEAD></HEAD> каждого HTML документа нужно добавить ссылку на CSS файл:

**Пример:**

```
<html>
<head>
<LINK REL=STYLESHEET TYPE="text/css" HREF="Example.css">
</head>
<body>
<p>Пример связывания</p>
</body>
</html>
```

Example.css - это Ваш CSS файл, содержащий описание применяемых стилей. Если он находиться в другом каталоге, нужно указать к нему путь. Создается CSS файл в любом текстовом редакторе, например, в Блокноте, нужно будет только изменить расширение текстового файла на CSS. В CSS файле не должны указываться теги <STYLE></STYLE>.

# Импортирование

В текущую стилевую таблицу можно импортировать содержимое CSS-файла с помощью команды @import. Общий синтаксис следующий.

```
@import url("имя файла") типы носителей;
@import "имя файла" типы носителей;
```

```
<html>
 <head>
<title>Импорт стиля</title>
<style type="text/css">
 @import "/style/main.css" screen; /* Стиль для вывода результата на монитор */
@import "/style/palm.css" handheld, print; /* Стиль для печати */
</style>
 </head>
 <body>
<p>...</p>
</body>
</html>
```

В данном примере импортируется два файла — main.css предназначен для изменения вида документа при его просмотре на экране монитора, и palm.css — при печати страницы и отображении на КПК.

## *Типы носителей*

| Тип | Описание |
|---|---|
| all | Все типы. Это значение используется по умолчанию. |
| aural | Речевые синтезаторы, а также программы для воспроизведения текста вслух. Сюда, например, можно отнести речевые браузеры. |
| braille | Устройства, основанные на системе Брайля, которые предназначены для слепых людей. |
| handheld | Наладонные компьютеры и аналогичные им аппараты. |
| print | Печатающие устройства вроде принтера. |
| projection | Проектор. |
| screen | Экран монитора. |
| tv | Телевизор. |

# Встраивание в тэги

***Встраивание в тэги*** - тэги HTML имеют параметр style, в котором можно задать стили.

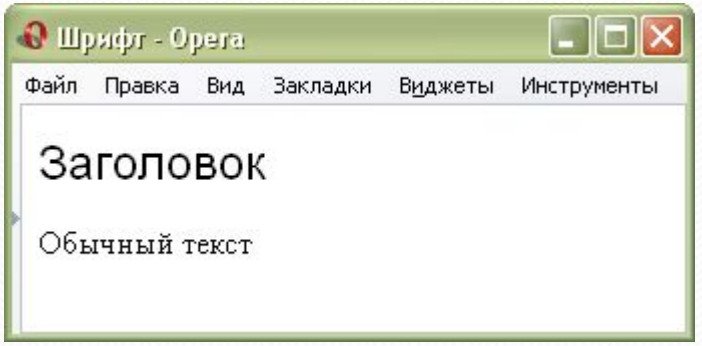
Например, в следующем примере задаётся форматирование заголовка втрого уровня, определяющее его отображение шрифтом красного цвета:

**Пример:**

```
<html>
<head>
<title>встраивание </title>
</head>
<body>
<H2 style="color: red">Пример встраивания</h2>
</body>
<html>
```

# Пример использования CSS при работе со шрифтами

```
<html>
    <head>
    <title>Шрифт</title>
    <style type="text/css">
    H1 {
     font-family: Arial, Helvetica, Verdana,
    sans-serif; /* Гарнитура шрифта */
     font-size: 150%; /* Размер текста */
     font-weight: lighter; /* Светлое начертание */
    }
    </style>
    </head>
    <body>
    <H1>Заголовок</H1>
    <p>Обычный текст</p>
    </body>
</html>
```



# Виды селекторов

# Классы

***Классы*** применяют, когда необходимо определить стиль для индивидуального элемента веб-страницы или задать разные стили для одного тега. При использовании совместно с тегами синтаксис для классов будет следующий.

**Тег.Имя класса { свойство1: значение; свойство2: значение; ... }**

Внутри стилевой таблицы вначале пишется желаемый тег, а затем, через точку пользовательское имя класса. Чтобы указать в коде HTML, что тег используется с определенным стилем, к тегу добавляется параметр class="Имя класса“.

Имена классов выбираются по желанию, главное, чтобы они были понятны и соответствовали их использованию, при этом имя должно всегда начинаться с символа.

## Пример использования классов

```
<html>
<head>
<title>Классы</title>
 <style type="text/css">
  P
   { /* Обычный параграф */
    text-align: justify; /* Выравнивание текста по ширине */
   }
  P.cite
   { /* Параграф с классом cite */
    text-align: justify; /* Выравнивание текста по ширине */
    color: navy; /* Синий цвет текста */
    font-style: italic; /* Курсивное начертание */
   }
 </style>
</head>
<body>
 <p>Параграф без использования классов.</p>
 <p class="cite">Параграф с использованием классов.</p>
</body>
</html>
```

Можно, также, использовать классы и без указания тега. Синтаксис в этом случае будет следующий.

**.Имя класса { свойство1: значение; свойство2: значение; ... }**

При такой записи, класс можно применять к любому тегу.

```
<html>
<head>
<title>Классы</title>
 <style type="text/css">
  .cite
    {
      color: navy; /* Синий цвет текста */
      font-style: italic; /* Курсивное начертание */
    }
 </style>
</head>
<body>
<p>Следует тщательно позаботиться о своем рабочем месте. Освещение в помещении отрегулировать таким образом, чтобы источник света находился сбоку. Во избежании медицинских осложнений <b class="cite">стул рекомендуется выбирать с мягким сидением</b>
.</p>
</body>
</html>
```

# *Идентификаторы*

**Идентификатор** (называемый также «ID селектор») определяет уникальное имя элемента, которое используется для изменения его стиля.

Синтаксис использования идентификатора следующий.

**#Имя идентификатора { свойство1: значение; свойство2: значение; ... }**

В отличие от классов идентификаторы должны быть уникальны, т.е. встречаться в коде документа только один раз.

Обращение к идентификатору происходит аналогично классам, но в качестве ключевого слова у тега используется параметр id, значением которого выступает имя идентификатора. Символ решетки при этом уже не указывается.

## Пример использования идентификаторов.

```
<html>
<head>
<title>Идентификаторы</title>
<style type="text/css">
#help
{
position: absolute; /* Абсолютное позиционирование */
left: 160px; /* Положение элемента от левого края */
top: 50px; /* Положение от верхнего края */
width: 225px; /* Ширина блока */
height: 180px; /* Высота блока */
background: #f0f0f0; /* Цвет фона */
}
</style>
</head>
<body>
<div id="help"> Этот элемент помогает в случае, когда вы находитесь в осознании того факта, что совершенно не понимаете, кто и как вам может помочь. Именно в этот момент вы и догадываетесь, что помочь вам никто не сможет. </div>
</body>
</html>
```

Как и при использовании классов, идентификаторы можно применять к конкретному тегу. Синтаксис при этом будет следующий.

Тег#Имя идентификатора { свойство1: значение; свойство2: значение; ... }

Вначале указывается имя тега, затем без пробелов символ решетки и название идентификатора.

```
<html>
<head>
<title>Идентификаторы</title>
<style type="text/css">
P
{
color: green; /* Зеленый цвет текста */
font-style: italic; /* Курсивное начертание текста */
}
P#opa
{
color: red; /* Красный цвет текста */
border: 1px solid #666; /* Параметры рамки */
background: #eee; /* Цвет фона */
padding: 5px; /* Поля вокруг текста */
}
</style>
</head>
<body>
<p>Обычный параграф</p>
<p id="opa">Параграф необычный</p>
</body>
</html>
```

# Контекстные селекторы

При создании веб-страницы часто приходится вкладывать одни теги внутрь других. Чтобы стили для этих тегов использовались корректно, помогут селекторы, которые работают только в определенном контексте.

Например, задать стиль для тега **<B>** только когда он располагается внутри контейнера **<P>**. Т. о. можно одновременно установить стиль для отдельного тега, а также для тега, который находится внутри другого.

Контекстный селектор состоит из простых селекторов разделенных пробелом. Так, для селектора тега синтаксис будет следующий.

```
Тег1 Тег2 { ... }
```

В этом случае стиль будет применяться к Тегу2 когда он размещается внутри Тега1.

```
<Тег1>
<Тег2> ... </Тег2>
</Тег1>
```

## *Пример использования контекстного селектора.*

```
<html>
<head>
<title>Контекстные селекторы</title>
  <style type="text/css">
   P B
    {
     font-family: Times, serif; /* Семейство шрифта */
     font-weight: bold; /* Жирное начертание */
     color: navy; /* Синий цвет текста */
    }
  </style>
</head>
<body>
<div>
<b>Жирное начертание текста</b>
</div>
<p>
<b>Одновременно жирное начертание текста и выделенное цветом</b>
</p>
</body>
</html>
```

# Пример использования контекстного селектора.

```
<html>
<head>
<title>Контекстные селекторы</title>
<style type="text/css">
A
{ color: green; /* Зеленый цвет текста для всех ссылок */ }
.menu
{ padding: 7px; /* Поля вокруг текста */
border: 1px solid #333; /* Параметры рамки */
background: #fc0; /* Цвет фона */}
.menu A
{ color: navy; /* Темно-синий цвет ссылок */ }
.menu A:hover
{ color: red; /* Красный цвет ссылок при наведении на нее */ }
</style>
</head>
<body>
<div class="menu">
<a href="link1.html">Русская кухня</a> |
<a href="link2.html">Украинская кухня</a> |
<a href="link3.html">Кавказская кухня</a>
</div>
<p><a href="link4.html">Другие материалы по теме</a></p>
</body>
</html>
```

# Соседние селекторы

**Соседними** называются элементы веб-страницы, когда они следуют друг за другом в коде документа.

```
<p>Lorem ipsum <b>dolor</b> <var>sit</var> amet.</p>
```

Теги **<VAR>** и **<B>** никак не перекрываются и представляют собой соседние элементы. То, что они расположены внутри контейнера **<P>**, никак не влияет на их отношение.

```
<p>Lorem <b>ipsum </b> dolor sit amet, <i>consectetuer</i> adipiscing <tt>elit</tt>.</p>
```

Соседними здесь являются теги **<B>** и **<I>**, а также **<I>** и **<TT>**. При этом **<B>** и **<TT>** к соседним элементам не относятся из-за того, что между ними расположен контейнер **<I>**.

Для управления стилем соседних элементов используется символ плюса (+), который устанавливается между двумя селекторами. Общий синтаксис следующий.

```
Селектор 1 + Селектор 2 { Описание правил стиля }
```

Пробелы вокруг плюса не обязательны, стиль при такой записи применяется к Селектору 2, но только в том случае, если он является соседним для Селектора 1 и следует сразу после него.

## *Пример использования соседних селекторов.*

```
<html>
<head>
<title>Соседние селекторы</title>
  <style type="text/css">
   B + I
    {
      color: red; /* Красный цвет текста */
    }
  </style>
</head>
<body>
<p>Lorem <b>ipsum </b> dolor sit amet, <i>consectetuer</i>
adipiscing elit.</p>
<p>Lorem ipsum dolor sit amet, <i>consectetuer</i> adipiscing elit.</p>
</body>
</html>
```

**Замечание**

Соседние селекторы не поддерживаются браузером Internet Explorer.

# Дочерние селекторы

**Дочерним** называется элемент, который непосредственно располагается внутри родительского элемента.

Синтаксис применения таких селекторов следующий:

**Селектор 1 > Селектор 2 { Описание правил стиля }**

Стиль применяется к Селектору 2, но только в том случае, если он является дочерним для Селектора 1.

По своей логике дочерние селекторы похожи на селекторы контекстные. Разница между ними следующая. Стиль к дочернему селектору применяется только в том случае, когда он является прямым потомком, иными словами, непосредственно располагается внутри родительского элемента. Для контекстного селектора допустим любой уровень вложенности.

**Замечание**

Дочерние селекторы не поддерживаются браузером Internet Explorer.

# Пример использования дочерних селекторов.

```
<html>
<head>
<title>Дочерние селекторы</title>
<style type="text/css">
DIV I
{ /* Контекстный селектор */
color:yellow; /* Зеленый цвет текста */ }
DIV > I
{ /* Дочерний селектор */
color: red; /* Красный цвет текста */ }
</style>
</head>
<body>
<div>
<p><i>Lorem ipsum dolor sit amet</i>, consectetuer adipiscing elit, sed diem nonummy nibh
euismod tincidunt ut lacreet dolore magna aliguam erat volutpat.</p>
</div>
<div>
<p><i>Lorem ipsum dolor sit amet</i>, consectetuer adipiscing elit, sed diem nonummy nibh
euismod tincidunt ut lacreet dolore magna aliguam erat volutpat.</p>
</div>
</body>
</html>
```