

Основы верстки CSS. Каскадные таблицы стилей CSS

Каскадные таблицы стилей CSS (Cascading Style Sheets) – стандарт стилей, объявленный консорциумом W3C. Термин *каскадные* указывает на возможность слияния различных видов стилей и на наследование стилей внутренними тегами от внешних.

CSS – это язык, содержащий набор свойств для определения внешнего вида документа. Спецификация CSS определяет свойства и описательный язык для установления связи с HTML-элементами.

CSS – абстракция, в которой внешний вид Web-документа определяется отдельно от его содержания.

По методам добавления стилей в документ различают три вида стилей.

Внутренние стили

Внутренние стили определяются атрибутом **style** конкретных тегов. Внутренний стиль действует только на определенные такими тегами элементы. Этот метод мало отличается от традиционного HTML.

Пример

```
<p style="color:blue">Абзац с текстом синего цвета</p>
<p style="color:red">Абзац с текстом красного цвета</p>
```

РЕЗУЛЬТАТ:

Абзац с текстом синего цвета

Абзац с текстом красного цвета

Не стоит использовать внутренние стили слишком часто, так как тогда Web-документ оказывается перегружен кодом и его внешний вид трудно изменить.

Глобальные стили

Глобальные стили CSS располагаются в контейнере **<style>...</style>**, расположенном в свою очередь в контейнере **<head>...</head>**.

Атрибут **type="text/css"**, ранее обязательный для тега **<style>**, в стандарте HTML5 можно опускать.

Глобальные стили являются универсальным средством, позволяющим не только оперативно изменять внешний вид Web-страницы, но и бороться с перегруженностью документа оформительскими тегами. Проблема в том, что такие стили надо прописывать на каждой странице сайта.

Пример

```
<html>
<head>
.....
<style type="text/css">
  p {color:#808080;}
</style>
.....
</head>
<body>
  <p>Серый цвет текста во всех абзацах Web-страницы</p>
  <p>Серый цвет текста во всех абзацах Web-страницы</p>
```

```
</body>
</html>
```

РЕЗУЛЬТАТ:

Серый цвет текста во всех абзацах Web-страницы

Серый цвет текста во всех абзацах Web-страницы

Внешние (связанные) стили

Внешние (связанные) стили определяются в отдельном файле с расширением **css**. Внешние стили позволяют всем страницам сайта выглядеть единообразно.

Для связи с файлом, в котором описаны стили, используется тег **<link>**, расположенный в контейнере **<head>...</head>**.

В этом теге должны быть заданы два атрибута: **rel="stylesheet"** и **href**, определяющий адрес файла стилей.

Пример

```
<html>
<head>
.....
<link rel="stylesheet" href="style.css">
.....
</head>
<body>
.....
</body>
</html>
```

Подключение стилей

Правило подключения глобальных и внешних стилей состоит из *селектора* и *объявлений* *стиля*.

Селектор, расположенный в левой части правила, определяет элемент (элементы), для которых установлено правило. Далее, в фигурных скобках перечисляются объявления стиля, разделенные точкой с запятой. Например:

```
p {
  text-indent: 30px;
  font-size: 14px;
  color: #666;
}
```

Объявление стиля – это пара *свойство CSS: значение CSS*.

Например: **color: red**

color свойство CSS, определяющее цвет текста;

red значение CSS, определяющее красный цвет.

При внутреннем подключении стиля правило CSS, которое является значением атрибута **style**, состоит из объявлений стиля, разделенных точкой с запятой. Например:

```
<p style="text-indent: 30px; font-size: 14px; color: #666;">...</p>
```

Селекторы CSS

Селектор	Описание	Подробнее
----------	----------	-----------

*	Любой элемент	Универсальный селектор
E	Элемент, определенный тегом E	Селекторы тегов
E#myid	Элемент E с идентификатором "myid"	Селекторы идентификаторов
E.myclass	Элемент E класса "myclass"	Селекторы классов
E[atr]	Селектор существования атрибута	Селекторы атрибутов
E[atr="val"]	Селектор равенства атрибута	Селекторы атрибутов
E[atr~="val"]	Селектор атрибута со списком значений	Селекторы атрибутов
E[atr^="val"]	Селектор префикса атрибута	Селекторы атрибутов
E[atr\$="val"]	Селектор суффикса атрибута	Селекторы атрибутов
E[atr*="val"]	Селектор подстроки атрибута	Селекторы атрибутов
E:link	Элемент E – еще не посещенная пользователем ссылка	Динамические псевдоклассы
E:visited	Элемент E – уже посещенная пользователем ссылка	Динамические псевдоклассы
E:hover	Элемент E при наведении на него указателя мышки	Динамические псевдоклассы
E:active	Элемент E, активированный пользователем	Динамические псевдоклассы
E:focus	Элемент E в фокусе	Динамические псевдоклассы
E:target	Элемент E, который является целью ссылки	Целевой псевдокласс
E:lang	Элемент E, написанный на указанном языке	Псевдокласс языка
E:enabled	Элемент E – доступный элемент формы	Псевдоклассы состояний
E:disabled	Элемент E – недоступный элемент формы	Псевдоклассы состояний
E:checked	Элемент E – включенный флажок или переключатель	Псевдоклассы состояний
E:indeterminate	Элемент E – неопределенный флажок или переключатель	Псевдоклассы состояний
E:root	Элемент E, корень документа	Структурные псевдоклассы
E:nth-child(n)	Элемент E, n-й ребенок родительского элемента	Структурные псевдоклассы
E:nth-last-child(n)	Элемент E, n-й ребенок родительского элемента, считая с конца	Структурные псевдоклассы
E:nth-of-type(n)	n-й элемент типа E	Структурные псевдоклассы
E:nth-last-of-type(n)	n-й элемент типа E, считая с конца	Структурные псевдоклассы
E:first-child	Элемент E, первый дочерний элемент родителя	Структурные псевдоклассы
E:last-child	Элемент E, последний дочерний элемент родителя	Структурные псевдоклассы
E:first-of-type	Первый элемент типа E	Структурные псевдоклассы
E:last-of-type	Последний элемент типа E	Структурные псевдоклассы
E:only-child	Единственный у родителя дочерний элемент	Структурные псевдоклассы
E:only-of-type	Единственный у родителя элемент типа E	Структурные псевдоклассы

E:empty	Элемент E, не содержащий дочерних элементов	Структурные псевдоклассы
E:not(X)	Элемент E, который не соответствует простому селектору X	Псевдокласс отрицания
E::first-line	Первая строка элемента E	Псевдоэлементы
E::first-letter	Первая буква элемента E	Псевдоэлементы
E::before	Содержимое до элемента E	Псевдоэлементы
E::after	Содержимое после элемента E	Псевдоэлементы
E::selection	Выделение в элементе E	Псевдоэлементы
E F	Элемент F, который находится внутри элемента E	Контекстные селекторы
E > F	Элемент F, который находится непосредственно внутри элемента E	Дочерние селекторы
E + F	Элемент F, который следует сразу после элемента E	Соседние селекторы
E ~ F	Элемент F, который следует после элемента E	Смежные селекторы

Универсальный селектор

Универсальный селектор соответствует любому элементу html-документа.

Для обозначения универсального селектора применяется символ "звёздочка" (*).

Его используют, если надо установить одинаковый стиль для всех элементов Web-страницы. Например:

```
* {
  margin: 0;
  padding: 0;
}
```

В некоторых случаях символ "звёздочка" (*) может быть опущен:

***.myclass** и **.myclass** эквиваленты,

***#myid** и **#myid** эквивалентны

Селекторы тегов

В качестве селектора может выступать любой html-тег, для которого определяются правила стилового оформления. Например:

```
h1 {color: red; text-align: center;}
```

Если несколько элементов будут иметь общий стиль, то соответствующие им селекторы можно перечислить в таблице стилей через запятую. Например:

```
h1, h2, h3, h4 {color: red; text-align: center;}
```

Селекторы идентификаторов

HTML предоставляет возможность присвоить уникальный идентификатор любому тегу. Идентификатор задается атрибутом **id**. Например:

```
<div id="a1">...</div>
```

Значение идентификатора должно начинаться с латинской буквы и может содержать буквы ([A-Z],[a-z]), цифры ([0-9]), "дефисы" (-) и "подчеркивания" (_).

Значения всех атрибутов **id** в html-документе обязаны быть уникальными. Если встречаются **id** с одинаковыми значениями, то эти идентификаторы игнорируются, а код Web-страницы становится невалидным.

В CSS-коде селектор идентификатора обозначается знаком "решетка" (#). Так как идентификатор **id** применяется только к уникальным элементам, название тега перед знаком "решетка" (#) обычно опускают:

```
div#a1 {color: green;}
```

аналогично

```
#a1 {color: green;}
```

Желательно использовать **id** не для стилового оформления элемента, а для обращения к нему через скрипты или перехода по ссылке.

Селекторы классов

Для стилового оформления чаще всего используются селекторы классов. Класс для тега задается атрибутом **class**. Например:

```
<div class="c1">...</div>
```

Имя класса должно начинаться с латинской буквы и может содержать буквы ([A-Z],[a-z]), цифры ([0-9]), "дефисы" (-) и "подчеркивания" (_).

Если атрибут **id** применяется для уникальной идентификации, то при помощи атрибута **class** тег относят к той или иной группе.

В CSS-коде селектор идентификатора обозначается знаком "точка" (.). Разные теги можно отнести к одному классу. В таком случае имя тега перед знаком "точка" (.) опускают:

```
i.green {color: #008000;}  
b.red {color: #f00;}  
.blue {color: #00f;}
```

Для тега можно одновременно указать несколько классов, перечисляя их в атрибуте **class** через пробел. В этом случае к элементу применяются стили каждого из указанных классов.

```
<div class="left w100">...</div>
```

Если некоторые из этих классов содержат одинаковые свойства стиля, но с разными значениями, то будут применены значения стиля класса, который в CSS-коде расположен ниже.

Селекторы атрибутов

Существует множество селекторов атрибутов, с помощью которых можно задать стиловое оформление для тега в зависимости от его атрибутов.

[**atr**] элемент с атрибутом **atr**, независимо от его значения. Например:

```
h1[title] {color: #800000;} /* элемент h1, у которого есть атрибут title */
```

[atr="val"] элемент с атрибутом **atr**, значение которого равно **val**. Например:

```
input[type="text"] {  
  border: 1px solid #ccc;  
  padding: 4px 6px;  
  width: 300px;  
}
```

[atr~="val"] элемент с атрибутом **atr**, значение которого представляет собой список разделенных пробелами слов, одно из которых равно **val**. Например:

```
a[rel~="sidebar"] {  
  text-decoration: none;  
  border-bottom: 1px solid #06c;  
  color: #06c;  
}
```

[atr^="val"] элемент с атрибутом **atr**, значение которого начинается строкой **val**.
Например:

```
span[class^="icon"] {  
  display: inline-block;  
  background-image: url("/img/icon_sprite.png");  
}
```

[atr\$="val"] элемент с атрибутом **atr**, значение которого заканчивается строкой **val**.
Например:

```
a[href$="doc"], a[href$="docx"] {  
  background: url("pic.gif") bottom left no-repeat;  
  display: inline-block;  
  width: 32px;  
}
```

[atr*="val"] элемент с атрибутом **atr**, значение которого содержит строку **val**.
Например:

```
[class*="cat-"] {  
  display: block;  
  float: left;  
  width: 280px;  
}
```

Между именем тега и квадратной скобкой ([) не должно быть пробела!

Универсальный селектор, селекторы тегов, идентификаторов, классов и атрибутов, а также псевдоклассы относятся к простым селекторам.

Простые селекторы можно связать в определенные последовательности на основе отношений элементов в дереве Web-документа. Такие конструкции называют комбинаторами.

Контекстные селекторы

Один из самых часто используемых комбинаторов – контекстный селектор.

Контекстные селекторы или селекторы потомков определяют несколько элементов, один из которых находится внутри другого. В контекстном селекторе простые селекторы разделены пробелом.

Пример

```
<style>
  ol li {border: 1px solid #06c;}
</style>
<ol>
  <li>Ч.Т.Айтматов
    <ul>
      <li>«Белый пороход»</li>
      <li>«Первый учитель»</li>
      <li>«Лицом к лицу»</li>
    </ul>
  </li>
  <li>
    <ul>
      <li>«Ревизор»</li>
      <li>«Тарас Бульба»</li>
      <li>«Мертвые души»</li>
    </ul>
  </li>
  <li>Толстой Л.Н.
    <ul>
      <li>«Война и мир»</li>
      <li>«Анна Каренина»</li>
      <li>«Воскресение»</li>
    </ul>
  </li>
</ol>
```

РЕЗУЛЬТАТ:

1. Ч.Т.Айтматов	<ul style="list-style-type: none">○ «Белый пороход»○ «Первый учитель»○ «Лицом к лицу»
2. Гоголь Н.В.	<ul style="list-style-type: none">○ «Ревизор»○ «Тарас Бульба»○ «Мертвые души»
3. Толстой Л.Н.	<ul style="list-style-type: none">○ «Война и мир»○ «Анна Каренина»

- «Воскресение»

Дочерние селекторы

Дочерний селектор определяет элемент, который находится внутри другого непосредственно. В дочернем селекторе простые селекторы разделены знаком "больше" (>).

Пример

```
<style>
  ol > li {border: 1px solid #06c;}
</style>
<ol>
  <li>Пушкин А.С.
    <ul>
      <li>«Выстрел»</li>
      <li>«Метель»</li>
      <li>«Дубровский»</li>
    </ul>
  </li>
  <li>Гоголь Н.В.
    <ul>
      <li>«Ревизор»</li>
      <li>«Тарас Бульба»</li>
      <li>«Мертвые души»</li>
    </ul>
  </li>
  <li>Толстой Л.Н.
    <ul>
      <li>«Война и мир»</li>
      <li>«Анна Каренина»</li>
      <li>«Воскресение»</li>
    </ul>
  </li>
</ol>
```

РЕЗУЛЬТАТ:

1. Пушкин А.С.	<ul style="list-style-type: none">○ «Выстрел»○ «Метель»○ «Дубровский»
2. Гоголь Н.В.	<ul style="list-style-type: none">○ «Ревизор»○ «Тарас Бульба»○ «Мертвые души»
3. Толстой Л.Н.	<ul style="list-style-type: none">○ «Война и мир»○ «Анна Каренина»○ «Воскресение»

Соседние селекторы

Соседний селектор определяет знак "плюс" (+), разделяющий две последовательности простых селекторов. Элементы, представленные этими последовательностями, находятся

внутри одного контейнера и следуют второй за первым непосредственно, не разделенные никакими другими тегами.

Пример

```
<style>
  h3 + p { padding-left: 260px; font-style: italic;}
</style>
<h3>РЕФЛЕКСОТЕРАПИЯ</h3>
<p>
  "Все акты сознательной и бессознательной жизни
  по способу происхождения суть рефлексы". И.М. Сеченов
</p>
<p>
  Рефлексотерапия – лечение заболеваний через управление рефлексами.
  Успешно применяется в программах комплексного лечения или как индивидуальная
  методика.
</p>
```

РЕЗУЛЬТАТ:

РЕФЛЕКСОТЕРАПИЯ

"Все акты сознательной и бессознательной жизни по способу происхождения суть рефлексы". И.М. Сеченов

Рефлексотерапия – лечение заболеваний через управление рефлексами. Успешно применяется в программах комплексного лечения или как индивидуальная методика.

Смежные селекторы

Смежный селектор определяет знак "тильда" (~), разделяющий две последовательности простых селекторов. Элементы, представленные этими последовательностями, находятся внутри одного контейнера и следуют второй за первым (необязательно непосредственно).

Пример

```
<style>
  h3 + p { padding-left: 260px; font-style: italic; text-indent: 0;}
  hr { background-color: #06c; border: none; height: 1px;}
  h3 ~ hr { background-color: #666; margin: 5px 0 5px 260px;}
</style>
<hr>
<h3>РЕФЛЕКСОТЕРАПИЯ</h3>
<p>
  "Все акты сознательной и бессознательной жизни
  по способу происхождения суть рефлексы". И.М. Сеченов
</p>
<hr>
<p>
  Рефлексотерапия – лечение заболеваний через управление рефлексами.
  Успешно применяется в программах комплексного лечения или как индивидуальная
  методика.
</p>
```