

CSS

Каскадные таблицы стилей

Что такое CSS?

CSS (анг. Cascading Style Sheets — каскадные таблицы стилей) — стандартизованная технология описания представления документа, оформленного языком разметки (преимущественно HTML).

Зачем оно надо?

Цель разработки CSS - *разделение содержания* (написанного на HTML или другом языке разметки) *и оформления документа* (написанного на CSS).

Слова

Стиль — совокупность правил, применяемых к элементу разметки и определяющих способ его отображения.

Стиль включает все типы элементов дизайна: шрифт, фон, текст, цвета ссылок, поля и расположение объектов на странице.

Таблица стилей — совокупность стилей, применимых к гипертекстовому документу.

Слова

Каскадирование — порядок применения различных стилей к веб-странице.

Браузер будет последовательно применять стили в соответствии с приоритетом: сначала *связанные*, затем *внедренные* и, наконец, *встроенные* стили.

Наследование (стилей) — означает, что если не указано иное, то конкретный стиль будет применен ко всем дочерним элементами гипертекстового документа.

div->p.*->span.*

История CSS

Level 1 (CSS1, 1996 г.) - шрифты, цвета, параметры текста

Level 2 (CSS2, 1998 г.) - позиционирование, @media, @page, звуковые стили, генерируемое содержимое

Level 2, rev. 1 (CSS2.1, 2011 г.) - некоторые изменения и исправления

Level 3 (CSS3, актуально) - существенные изменения и дополнения, учитывающие современные требования к юзабилити

Level 4 (CSS4, ?) ...

Все подробности - на сайте W3.ORG (<http://www.w3.org/TR/CSS/>)

Синтаксис

selector [, *selector*] {*property: value*; [*property:value*;]}

property + *value* => *description*

selector + *description* => *rule*

Селекторы элементов

```
div, th, td {  
    padding-left: 10px;  
    padding-right: 10px;  
    background-color: yellow;  
}
```



Классы

Q: Как применить разные стили к одному селектору?

A: Нужно использовать именованные стили - **классы**
`[element].classname {description}`

...

```
<style type="text/css">
  p.red {color: #FF00;}
  p.bold {font-weight: bold;}
</style>
```

...

```
<p class=" red bold">Красный полужирный</p>
```


Псевдоклассы и псевдоэлементы

“Как бы элементы” и “как бы классы” определяют *фрагмент* содержимого или его текущее *состояние*. Они *предопределены* и применимы не ко всем элементам

```
<style type="text/css">  
  p:first-letter {font-size:200%;font-weight:bold;} - буквица в  
  начале абзаца  
  a {text-decoration: none;} - запрет подчеркивания ссылки  
  a:hover {text-decoration: underline;} - подчеркивание ссылки под  
  курсором  
</style>
```


Идентификаторы

Идентификатор - уникальное в пределах документа имя элемента

задается атрибутом `id`, используется в скриптах, поддерживается CSS

`[element]#identifier {description}`

...

```
<style type="text/css">  
    #wrapper {text-align: left; width: 980px; margin: 0 auto;}  
</style>
```

...

```
<div id="wrapper">Hello?</div>
```

Еще про селекторы

Универсальный селектор

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

Селектор атрибутов

```
a[href="http://example.com"]  
{font-weight: bold;}
```

Контекстный селектор (потомков)

```
div#container div.listener {margin:  
5px;}
```

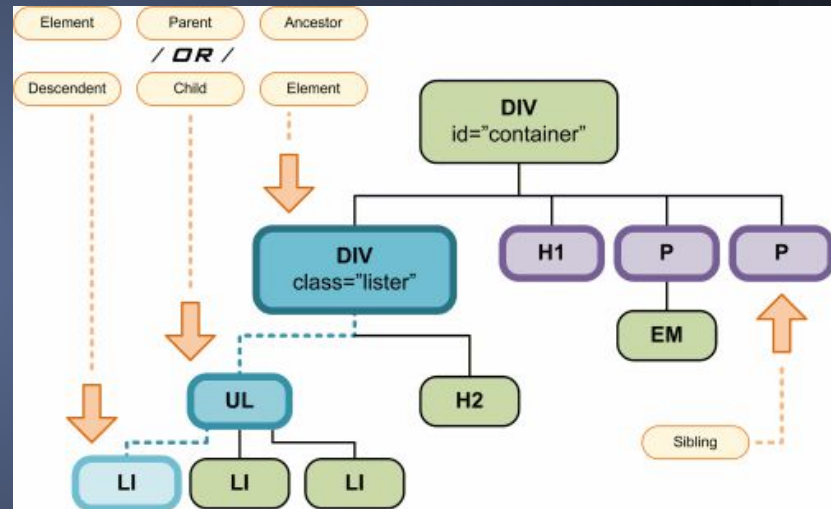
Селектор дочерних элементов;

```
div.listener > h2 {color: green;}
```

Селектор соседних элементов

```
h1 + p {font-size: 24pt;}
```

Есть еще, но это уже путь в параллельные миры...



Подключение стилей

Встраивание (Inline). Встроенные стили применяются к конкретному элементу.

Внедрение (Embedded). Внедренные стили позволяют управлять отображением той страницы, где они определены.

Связывание (Linked или External). Описание стилей вынесено во внешний файл, ссылаясь на него можно контролировать отображение всех страниц сайта.

Inline

Встроенные стили предоставляют максимальный контроль над любым элементом разметки.

Встроенный стиль применяется к любому тегу HTML с помощью атрибута *style*:

Пример:

```
<div style="font-family: Garamond; font-size: 18 pt;">Весь текст в этом разделе имеет  
размер 18 точек и шрифт Garamond. <span style="color:#ff3300;">А этот фрагмент еще и  
выделен оттенком красного цвета.</span>  
</div>
```

Встроенные стили полезны, когда необходима тонкая настройка отображения некоторого элемента страницы или небольшой веб-страницы.

Embedded

Внедренные стили используют тег `<style>`, который размещается в заголовке HTML-документа:

```
<html>
<head>
  ...
  <style type="text/css">
    правила CSS
  </style>
  ...
</head>
<body>
  ...
```

Linked (external)

Связанные (linked), или *внешние (external)* стили — наиболее удобное решение, когда речь идет об оформлении целого сайта. Описание правил помещается в отдельный файл, который тегом `<link>` подключается ко всем страницам:

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

К странице можно подключить несколько стилевых таблиц
Файлы со стилями могут находиться на другом веб-сервере
Таблица стилей может быть “собрана” из нескольких файлов с помощью директивы `@import`

Обработка правил

Каскадирование применяется при объявлении более одного правила CSS для одного элемента. Чтобы избежать конфликта устанавливается **приоритет обработки** стилевых правил (по возрастанию):

1. Наиболее низким приоритетом обладают стили браузера (default styles);
2. Пользовательские стили (заданные в его настройках браузера);
3. Авторские стили:
 - Inherit - наследуемые стили, имеют самый низкий приоритет;
 - Внешние стили (linked);
 - Внедренные стили (embedded);
 - Встроенные стили (inline);
4. Высшим приоритетом обладают авторские и пользовательские стили, объявленные с директивой **!important**. Если таких свойств несколько, то предпочтение отдаётся пользовательским стилям.

Гол “в раздевалку”

Начинайте использовать стилевую разметку сразу, ибо через неделю разобраться в ваших `<font ...` и т.д. вы уже не сможете.

Запомните мантру “СПЕЦИФИКАЦИЯ” (<http://www.w3.org/TR/CSS2/>) – туда смотрят производители браузеров

Пользуйтесь СПЕЦИФИКАЦИЕЙ. Никто не знает все свойства CSS наизусть.

СПЕЦИФИКАЦИЯ обширна, но не все свойства востребованы и поддерживаются: помните о кросс-браузерности.

Теперь все