

CLIENT-SIDE WEB

CSS BASICS

19

EXTERNAL TRAINING .NET/WEB

EPAM SARATOV · SUMMER 2020



ПЛАН ЗАНЯТИЯ

- Введение в CSS
- Подключение CSS стилей к документу
- Синтаксис CSS
- CSS селекторы
- Каскадирование и наследование
- Шрифты
- Изображения и background
- List properties
- Работа с CSS используя Developer Tools



ПРЕИМУЩЕСТВА ИСПОЛЬЗОВАНИЯ CSS

- Разделение структуры документа от стиля отображения. CSS можно написать независимо от HTML.
- Поддержка большим количеством устройств (в том числе телефоны и принтеры).
- Оптимизация производительности. CSS файлы кэшируются браузерами.

```
h1 { color: white;
      background: orange;
      border: 1px solid black;
      padding: 0 0 0 0;
      font-weight: bold;
    }
/* begin: seaside-theme */

body {
  background-color: white;
  color: black;
  font-family: Arial, sans-serif;
  margin: 0 4px 0 0;
  border: 12px solid;
}
```

CSS

ПОДКЛЮЧЕНИЕ CSS СТИЛЕЙ К ДОКУМЕНТУ

- Inline (атрибут `style` в HTML-элементе);
- Тег `<style>` в HTML-документе
- Подключение внешнего CSS файла при помощи тега `<link>`
- Подключение внутри CSS файла (использование `@import`)



INLINE-СТИЛЬ

- Записывается в качестве значения атрибута **style** целевого элемента HTML;
- Достоинства:
 - Позволяет (пере)определить стиль конкретного элемента на конкретной странице;
- Недостатки:
 - Невозможно повторное использование даже в рамках страницы;
 - Затрудняется поддержка;
 - Смешение разметки и стилизации.

```
<h4 style="font-family: sans-serif; color: maroon;">Heading</h4>
```

ТЕГ <STYLE>

- Записывается в качестве содержимого элемента <style>, размещаемого в секции <head> документа HTML;
- Правила стилизации закрепляются за **селекторами**, описывающими целевые элементы;
- Достоинства:
 - Позволяет описать набор стилей для всех элементов конкретной страницы;
- Недостатки:
 - Невозможно повторное использование между страницами;
 - Смешение разметки и стилизации.

```
<head>
  <!-- meta, title etc. -->
  <style type="text/css">
    h3, h4 {
      font-family: sans-serif;
      color: maroon;
    }
  </style>
</head>
```


ВНЕШНИЙ ФАЙЛ CSS

- Стилиевые правила и селекторы записываются в отдельный файл с расширением .css;
- Файл стилей подключается к странице при помощи элемента `<link>` в секции `<head>`;
- Достоинства:
 - Возможность использовать один файл на множестве страниц;
 - Кэширование браузером;
 - Логическое отделение разметки от стилизации;
- Недостатки:
 - Подключение целого файла, если нужен всего лишь один стиль.

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

ИМПОРТ СТИЛЯ ИЗ ДРУГОГО СТИЛЯ

- Позволяет импортировать файл стилей в другой файл стилей при помощи директивы **@import**;
- Достоинства:
 - Можно добавлять стили для конкретных устройств и разрешений экрана;
 - Возможность связать несколько файлов в один;
- Недостатки:
 - Неочевидность подключаемых файлов к странице;
 - Приостанавливает загрузку файла до тех пор, пока не загрузится файл, указанный в **@import**.

```
@import url('styles_common.css');  
@import url('styles_tv.css') tv;
```


БАЗОВЫЙ СИНТАКСИС CSS

- Любой стиль (кроме inline) состоит из селектора и набора правил, заданных в формате свойство: значение;
- Селектор описывает параметры элементов, к которым будут применяться описанные в его теле правила;
- В качестве селекторов могут использоваться теги, классы и другие характеристики.

```
селектор
{
    свойство1: значение1;
    свойство2: значение2;
    ...
    свойствоN: значениеN;
}
```

```
body {
    font-family: 'Times New Roman';
    color: yellow;
    background: green;
}

.active-link {
    color: red;
}
```

КОММЕНТАРИИ В CSS

- Добавление пояснений по поводу использования того или иного стилевого свойства;
- Разделение на блоки;
- Собственные заметки;
- Позволяют легко вспомнить логику и структуру селекторов;
- Повышают разборчивость кода.

```
/*  
    комментарий  
*/  
  
body {  
    font-family: 'Times New Roman'; /* комментарий */  
    color: yellow;  
    background: green;  
}  
  
.active-link {  
    color: /* комментарий */ red;  
}
```

СЕЛЕКТОРЫ CSS

Описание	Пример
Универсальный селектор	<code>* { color: red; }</code>
Элемент	<code>p { font-family: Garamond, serif; }</code>
Класс	<code>.note { color: red; background: yellow; }</code>
Идентификатор	<code>#paragraph1 { margin: 0; }</code>
Атрибут	<code>a[href="http://epam.com"] { font-weight: bold; }</code>
Потомок	<code>div p { color: red; }</code>
Дочерний элемент	<code>p.note > b { color: green; }</code>
Родственный элемент	<code>h1 ~ p { font-size: 24pt; }</code>
Соседний элемент	<code>h1 + p { font-size: 24pt; }</code>
Псевдо-класс	<code>a:active { color: yellow; }</code>
Псевдо-элемент	<code>.new::before { content: '!'; color: red; }</code>

<http://jsfiddle.net/Nd6S5/2/embedded/result/>

УНИВЕРСАЛЬНЫЙ И СЕЛЕКТОРЫ ЭЛЕМЕНТОВ

- В качестве селектора может выступать любой тег HTML;
- Применить стиль можно к любому тегу, но результат будет заметен только для тегов, непосредственно отображающихся на странице.

```

[ * {
  border: 1px solid black;
}

[ span {
  color: lightblue;
  padding: 10px;
}

```

Selector:

```

<section id="solar">
  <div class="planet" id="mercury"></div>
  <div class="planet" id="venus"></div>
  <div class="planet" id="earth">
    <p id="moon" class="sattelite">
      <span class="crater"></span>
      <span class="crater"></span>
      <span class="crater"></span>
    </p>
  </div>
  <div id="mars">
    <p class="sattelite" id="fobos">
      <span class="crater"></span>
      <span class="crater"></span>
    </p>
    <p class="sattelite" id="deimos"></p>
  </div>
</section>

```

СЕЛЕКТОР КЛАССА

- Определение стиля для индивидуального элемента страницы;
- Задание разных стилей для одного тега;
- Применение одного стиля к разным элементам страницы.

```
.planet{  
    color: lightblue;  
    padding: 10px;  
}
```

Selector: `.planet`

```
<section id="solar">
```

```
<div class="planet" id="mercury"></div>
```

```
<div class="planet" id="venus"></div>
```

```
<div class="planet" id="earth">
```

```
  <p id="moon" class="sattelite">
```

```
    <span class="crater"></span>
```

```
    <span class="crater"></span>
```

```
    <span class="crater"></span>
```

```
  </p>
```

```
</div>
```

```
<div class="planet" id="mars">
```

```
  <p class="sattelite" id="fobos"></p>
```

```
  <p class="sattelite" id="deimos"></p>
```

```
</div>
```

```
<div id="pluto">
```

```
  <p class="sattelite" id="haron"></p>
```

```
</div>
```

```
</section>
```

ЗАДАНИЕ НЕСКОЛЬКИХ КЛАССОВ

- К любому элементу можно добавить одновременно несколько классов, перечислив их в атрибуте `class` через пробел;
- Элемент будет найден при использовании селектора по любому из его классов;
- Если правила определяют несколько значений одного и того же свойства, в силу вступает т.н. **сила селекторов (specificity)**.

```
[-] .crater{  
    margin:5px;  
}  
  
[-] .object{  
    margin:10px;  
}
```

Selector: `.crater.object`

```
<section id="solar">  
  <div class="planet" id="mercury"></div>  
  <div class="planet" id="venus"></div>  
  <div class="planet" id="earth">  
    <p id="moon" class="satellite">  
      <span class="crater"></span>  
      <span class="crater object"></span>  
      <span class="crater"></span>  
    </p>  
  </div>  
  <div id="mars">  
    <p class="satellite" id="fobos">  
      <span class="crater"></span>  
      <span class="crater"></span>  
    </p>  
    <p class="satellite" id="deimos"></p>  
  </div>  
</section>
```

СЕЛЕКТОР ИДЕНТИФИКАТОРА

- Идентификатор определяет уникальное имя элемента в рамках всей страницы;
- Задаётся при помощи атрибута `id`;
- Используется для точечного изменения стиля конкретного элемента и обращения к нему через скрипты.

```
#venus{  
    color: aquamarine;  
    border-width: 2px;  
}
```

Selector: `#venus`

```
<section id="solar">  
  <div class="planet" id="mercury"></div>  
  <div class="planet" id="venus"></div>  
  <div class="planet" id="earth">  
    <p id="moon" class="satellite">  
      <span class="crater"></span>  
      <span class="crater object"></span>  
      <span class="crater"></span>  
    </p>  
  </div>  
  <div id="mars">  
    <p class="satellite" id="fobos">  
      <span class="crater"></span>  
      <span class="crater"></span>  
    </p>  
    <p class="satellite" id="deimos"></p>  
  </div>  
</section>
```


СЕЛЕКТОР ПОТОМКА

- Позволяет применять правила только для элементов (потомков), вложенных в другой элемент (родительский);
- Записывается как селектор родителя и селектор потомка, разделённые пробелом.

```
#solar div{  
    color: lightblue;  
    padding: 10px;  
}
```

Selector:

```
<section id="solar">  
  <div class="planet" id="mercury"></div>  
  <div class="planet" id="venus"></div>  
  <div class="planet" id="earth">  
    <div id="moon" class="satellite">  
      <span class="crater"></span>  
      <span class="crater object"></span>  
      <span class="crater"></span>  
    </div>  
  </div>  
  <div id="mars">  
    <p class="satellite" id="fobos">  
      <span class="crater"></span>  
      <span class="crater"></span>  
    </p>  
    <p class="satellite" id="deimos"></p>  
  </div>  
</section>
```

СЕЛЕКТОР ДОЧЕРНЕГО ЭЛЕМЕНТА

- Дочерним называется элемент, расположенный непосредственно внутри родительского элемента;

```
#solar > div{  
    color: lightblue;  
    padding: 10px;  
}
```

Selector: `#solar > div`

```
<section id="solar">  
  <div class="planet" id="mercury"></div>  
  <div class="planet" id="venus"></div>  
  <div class="planet" id="earth">  
    <div id="moon" class="satellite">  
      <span class="crater"></span>  
      <span class="crater object"></span>  
      <span class="crater"></span>  
    </div>  
  </div>  
  <div id="mars">  
    <p class="satellite" id="fobos">  
      <span class="crater"></span>  
      <span class="crater"></span>  
    </p>  
    <p class="satellite" id="deimos"></p>  
  </div>  
</section>
```

СЕЛЕКТОР РОДСТВЕННОГО И СОСЕДНЕГО ЭЛЕМЕНТА

- Родственными называются элементы, следующие друг за другом в рамках одного родительского элемента (на одном уровне вложенности);
- Соседние элементы — родственные, непосредственно следующие друг за другом.

```
#mercury ~ div {  
    padding: 10px;  
}
```

```
#mercury + div {  
    padding: 10px;  
}
```

Selector: <code>#mercury + div</code>	Selector: <code>#mercury ~ div</code>
<pre><section id="solar"> <div class="planet" id="mercury"></div> <div class="planet" id="venus"></div> <div class="planet" id="earth"> <div id="moon" class="satellite"> </div> </div> <div id="mars"> <p class="satellite" id="fobos"> </p> <p class="satellite" id="deimos"></p> </div> </section></pre>	<pre><section id="solar"> <div class="planet" id="mercury"></div> <div class="planet" id="venus"></div> <div class="planet" id="earth"> <div id="moon" class="satellite"> </div> </div> <div id="mars"> <p class="satellite" id="fobos"> </p> <p class="satellite" id="deimos"></p> </div> </section></pre>

СЕЛЕКТОР АТТРИБУТА

- Многие элементы могут менять своё действие в зависимости от используемых в них атрибутов;
- Можно выбрать элементы, просто имеющие указанный атрибут или с конкретным значением.

```
img[alt] {  
    border: 1px solid blue;  
}
```

```
input[type="text"] {  
    background-color: aqua;  
}
```

```
input[type="text"][name="firstName"] {  
    background: ■ blue;  
    color: □ white;  
    border: 3px solid ■ royalblue;  
}  
  
<input type="text" name="firstName" value="Richard" />
```

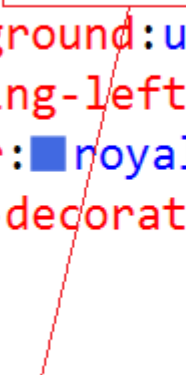
СЕЛЕКТОРЫ АТТРИБУТОВ

Наличие атрибута	div[attr]
Значение атрибута равно тексту	div[attr=text]
... начинается с текста	div[attr^=text]
... заканчивается текстом	div[attr\$=text]
... содержит текст	div[attr*=text]
... содержит блок, отделённый пробелом	div[attr~=text]
... содержит блок, отделённый дефисом	div[attr =text]

ЗНАЧЕНИЕ АТТРИБУТА НАЧИНАЕТСЯ С ТЕКСТА

- Устанавливает стиль для элемента в том случае, если значение атрибута тега начинается с указанного текста.

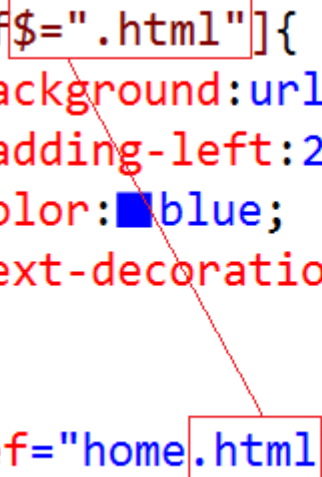
```
a[href^="mailto:"]{  
    background:url('envelope.png') no-repeat left center;  
    padding-left:25px;  
    color:royalblue;  
    text-decoration:none;  
}  
  
<a href="mailto:webmaster@example.com">Email the webmaster</a>
```



ЗНАЧЕНИЕ АТТРИБУТА ЗАКАНЧИВАЕТСЯ С ТЕКСТА

- Устанавливает стиль для элемента в том случае, если значение атрибута оканчивается указанным текстом.

```
a[href$=".html"]{  
    background:url('firefox.png') no-repeat left center;  
    padding-left:25px;  
    color:blue;  
    text-decoration:none;  
}  
  
<a href="home.html">Go back to the home page.</a>
```



ЗНАЧЕНИЕ АТТРИБУТА СОДЕРЖИТ ТЕКСТ

Возможны варианты, когда стиль следует применить к тегу с определённым атрибутом, при этом частью его значения является некоторый текст.

```
a[href*=".php"]{
    background:url('thunderbird.png') no-repeat left center;
    padding-left:25px;
    color:blue;
    text-decoration:none;
}
<a href="/address_book.php?sid=a0da625a83123f133e72436f20254baf">
    Launch the Address Book
</a>
```

СЕЛЕКТОРЫ ПСЕВДО-КЛАССОВ

- Псевдо-классы определяют динамическое состояние элементов, изменяющееся при действиях пользователя, а также положение в дереве документа.

```
a:link{  
    color:■mediumblue;  
}  
a:visited{  
    color:■magenta;  
}  
a:hover{  
    text-decoration:underline;  
}  
a:active{  
    color:■crimson;  
}
```

ПСЕВДО-КЛАСС ПЕРВОГО ДОЧЕРНЕГО ЭЛЕМЕНТА

- Применяется к первому дочернему элементу селектора, который расположен в дереве элементов документа.

```
span:first-child{  
    padding:10px  
}
```

Selector:

```
<section id="solar">  
  <div class="planet" id="mercury"></div>  
  <div class="planet" id="venus"></div>  
  <div class="planet" id="earth">  
    <div id="moon" class="sattelite">  
      <span class="crater"></span>  
      <span class="crater object"></span>  
      <span class="crater"></span>  
    </div>  
  </div>  
  <div id="mars">  
    <p class="sattelite" id="fobos">  
      <span class="crater"></span>  
      <span class="crater"></span>  
    </p>  
    <p class="sattelite" id="deimos"></p>  
  </div>  
</section>
```

ДРУГИЕ ПСЕВДО-КЛАССЫ

Selector: `div:nth-child(3)`

```
<section id="solar">
  <div class="planet" id="mercury"></div>
  <div class="planet" id="venus"></div>
  <div class="planet" id="earth">
    <p id="moon" class="sattelite">
      <span class="crater"></span>
      <span class="crater"></span>
      <span class="crater"></span>
    </p>
  </div>
  <div id="mars">
    <p class="sattelite" id="fobos"></p>
    <p class="sattelite" id="deimos"></p>
  </div>
</section>
```

Selector: `div:last-child`

```
<section id="solar">
  <div class="planet" id="mercury"></div>
  <div class="planet" id="venus"></div>
  <div class="planet" id="earth">
    <p id="moon" class="sattelite">
      <span class="crater"></span>
      <span class="crater"></span>
      <span class="crater"></span>
    </p>
  </div>
  <div id="mars">
    <p class="sattelite" id="fobos"></p>
    <p class="sattelite" id="deimos"></p>
  </div>
</section>
```

ДРУГИЕ ПСЕВДО-КЛАССЫ

Selector: `div:nth-child(odd)`

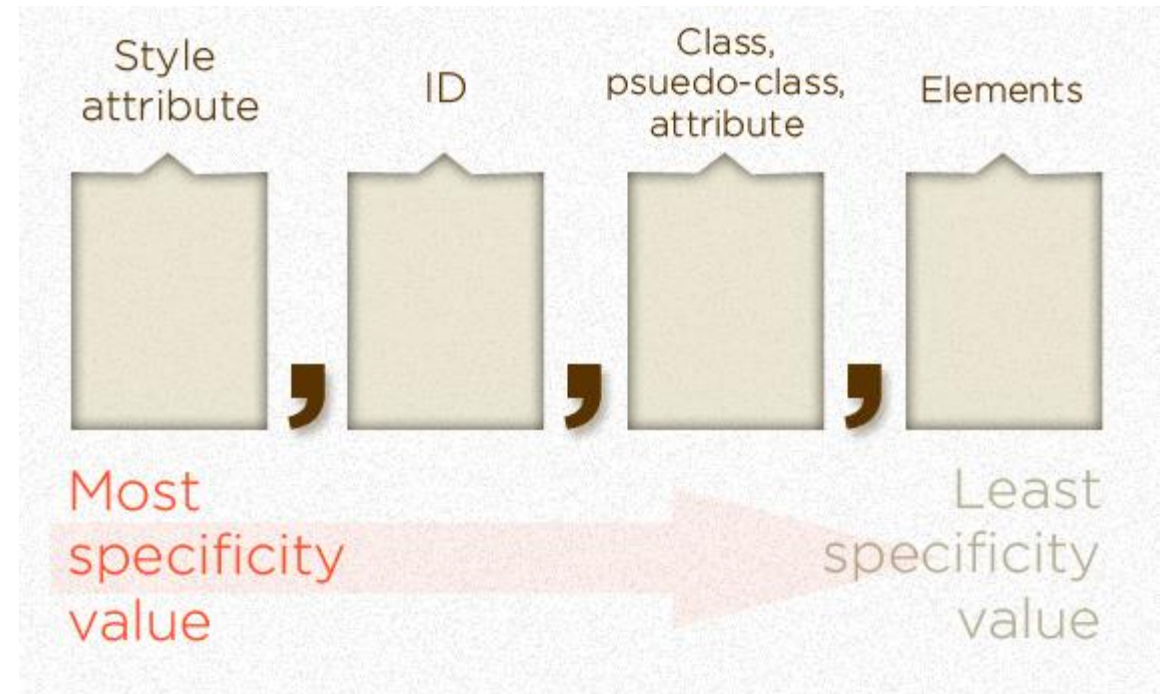
```
<section id="solar">
  <div class="planet" id="mercury"></div>
  <div class="planet" id="venus"></div>
  <div class="planet" id="earth">
    <p id="moon" class="sattelite">
      <span class="crater"></span>
      <span class="crater"></span>
      <span class="crater"></span>
    </p>
  </div>
  <div id="mars">
    <p class="sattelite" id="fobos"></p>
    <p class="sattelite" id="deimos"></p>
  </div>
</section>
```

Selector: `div:nth-child(even)`

```
<section id="solar">
  <div class="planet" id="mercury"></div>
  <div class="planet" id="venus"></div>
  <div class="planet" id="earth">
    <p id="moon" class="sattelite">
      <span class="crater"></span>
      <span class="crater"></span>
      <span class="crater"></span>
    </p>
  </div>
  <div id="mars">
    <p class="sattelite" id="fobos"></p>
    <p class="sattelite" id="deimos"></p>
  </div>
</section>
```

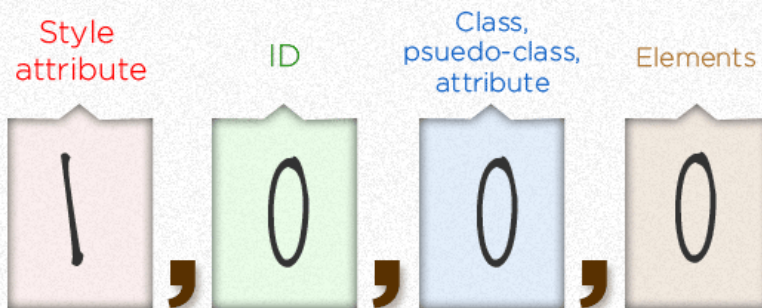
КАСКАДИРОВАНИЕ

- Аббревиатура CSS расшифровывается как Cascading Style Sheets (каскадные таблицы стилей);
- Под каскадом понимается одновременное применение разных стилевых правил к элементам документа.

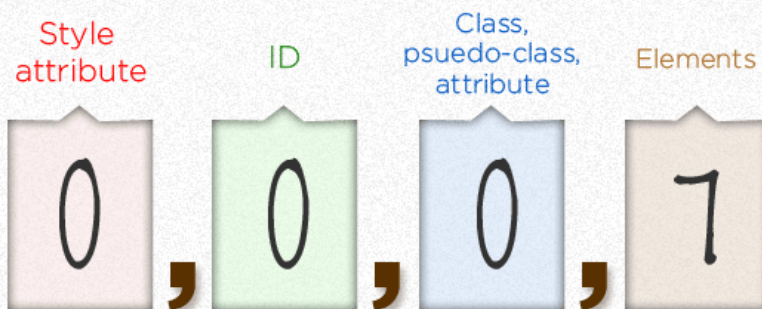


ВЫЧИСЛЕНИЕ СИЛЫ СЕЛЕКТОРА

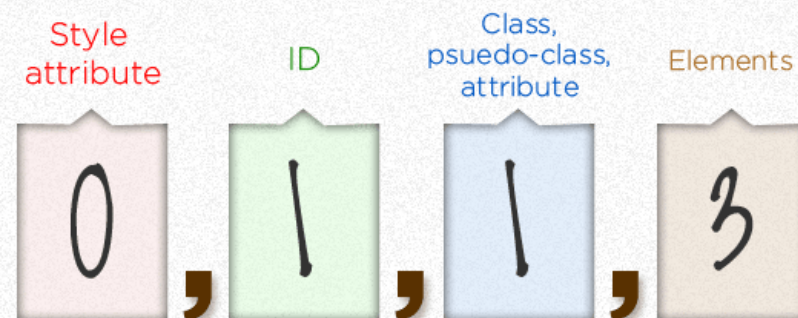
`<li style="color: red;">`



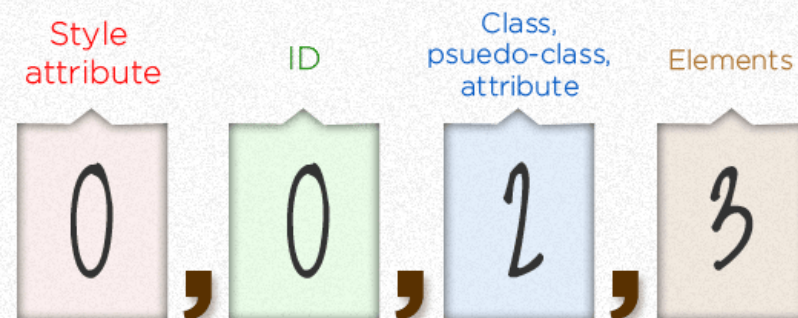
`ul > li ul li ol li:first-letter`



`ul#nav li.active a`



`body.ie7 .col_3 h2 ~ h2`



ВЫЧИСЛЕНИЕ СИЛЫ СЕЛЕКТОРА ПРИМЕР

Какого цвета будет параграф P?

```
1 <section class="galaxy">
2   <div class="solar">
3     <div>
4       <p class="planet object">Earth</p>
5     </div>
6   </div>
7 </section>
```

HTML

```
1 section .solar > div p{
2   color:red;
3   text-decoration:underline;
4 }
5 .galaxy div .planet{
6   color:green;
7 }
```

Earth

ИСПОЛЬЗОВАНИЕ !IMPORTANT

- Играет роль в том случае, когда пользователи подключают свою собственную таблицу стилей;
- Если возникает противоречие, когда стиль автора страницы и пользователя для одного и того же элемента не совпадает, то !important позволяет повысить приоритет стиля.

```
<p id="earth"> Lorem ipsum <p>  
<p style="color: green;" > Dolor sit amet <p>
```

```
#earth {  
    color: blue;  
}
```

Lorem ipsum

Dolor sit amet

```
#earth {  
    color: blue;  
}  
  
p {  
    color: red !important;  
}
```

Lorem ipsum

Dolor sit amet

НАСЛЕДОВАНИЕ СВОЙСТВ

- Наследование позволяет задавать значения некоторых свойств единожды, определяя их для родителей верхнего уровня. Допустим, требуется установить цвет и шрифт для основного текста. Достаточно воспользоваться селектором **BODY**, добавить для него желаемые свойства, и цвет текста внутри абзацев и других текстовых элементов поменяется автоматически.
- Свойства **background** и **border** не наследуются.

```
body{  
  font-size:24px;  
}
```

```
div{  
  color:■crimson;  
  text-align:right;  
  border:1px solid ■crimson;  
  padding:10px;  
}
```

```
<div>  
  <h1>Inheritance</h1>  
  <p>  
    Some properties in CSS are inherited to children elements  
  </p>  
</div>
```

Inheritance

Some properties in CSS are inherited to children elements as you can see here. The <h1> heading and the <p> element inherit color and alignment from the <div>, but not the border and the padding.

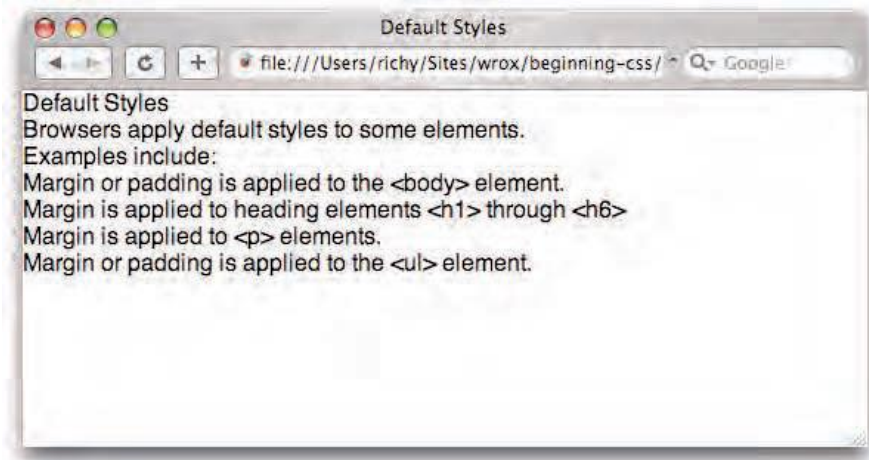
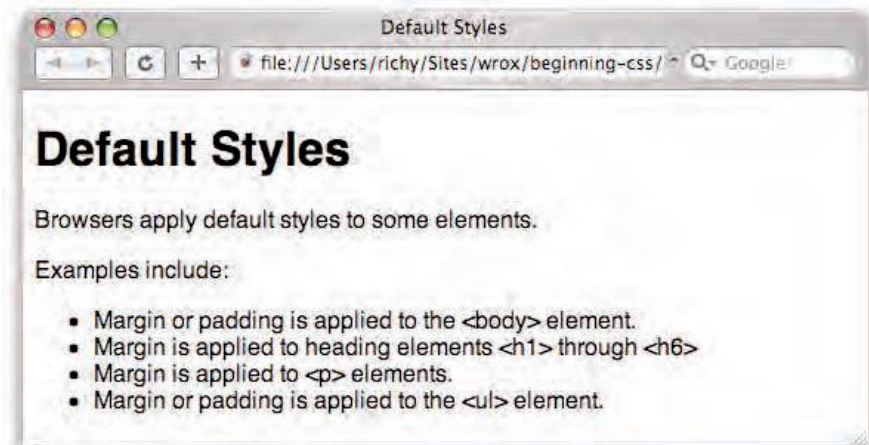
Inheritance

Some properties in CSS are inherited to children elements as you can see here. The <h1> heading and the <p> element inherit color and alignment from the <div>, but not the border and the padding.

RESET CSS И СТИЛИ БРАУЗЕРА ПО УМОЛЧАНИЮ

Каждый браузер устанавливает свои значения стилей по умолчанию для различных HTML-элементов. С помощью CSS **Reset** мы можем нивелировать эту разницу для обеспечения кроссбраузерности стилей.

```
body{
    margin:0;
    padding:0;
}
h1{
    margin:0;
    font-weight:normal;
    font-size:16px;
}
p{
    margin:0;
}
ul{
    margin:0;
    padding:0;
    list-style:none
}
```



<http://meyerweb.com/eric/tools/css/reset/>

ПОЧТИ ВСЁ: ЕДИНИЦЫ ИЗМЕРЕНИЯ

- Сантиметры, дюймы, пункты, пики 😊
- Em, rem, px – основные.



ЕДИНИЦЫ ИЗМЕРЕНИЯ: АБСОЛЮТНЫЕ

Единица	Описание
cm	сантиметры
mm	миллиметры
in	дюймы (1in = 96px = 2.54cm)
pt	пункты (1pt = 1/72 дюйма)
pc	пики (1pc = 12 pt)

ЕДИНИЦЫ ИЗМЕРЕНИЯ: ОТНОСИТЕЛЬНЫЕ

Единицы	Описание
em	высота шрифта текущего элемента (2em означает два размера текущего шрифта)
ex	высота символа x текущего шрифта (используется редко)
ch	ширина символа "0" (ноль)
rem	размер относительно размера шрифта элемента <html>
vw	1% ширины окна
vh	1% высоты окна
vmin	наименьшее из (vw, vh)
vmax	наибольшее из (vw, vh)
%	процент
px	пиксел

ШРИФТЫ

- **font** - универсальное свойство, которое позволяет одновременно задать несколько характеристик шрифта и текста.
- **font-family** - устанавливает семейство шрифта, которое будет использоваться для оформления текста содержимого.
- **font-size** - определяет размер шрифта элемента.
- **font-stretch** - устанавливает узкое, нормальное или широкое начертание шрифта, что позволяет уплотнять или расширять текст.
- **font-style** - определяет начертание шрифта — обычное, курсивное или наклонное.
- **font-variant** - определяет, как нужно представлять строчные буквы — оставить их без модификаций или делать их все прописными уменьшенного размера.
- **font-weight** - устанавливает насыщенность шрифта.

PROPERTY	VALUE
font-family	<code>[[<family-name> <generic-family>] [, <family-name> <generic-family>]*]</code> Initial value: Varies depending on the browser or user agent.

PROPERTY	VALUE
font-style	<code>normal italic oblique</code> Initial value: normal

PROPERTY	VALUE
font-weight	<code>normal bold bolder lighter 100 200 300 400 500 600 700 800 900</code> Initial value: normal

PROPERTY	VALUE
font-size	<code><absolute-size> <relative-size> <length> <percentage></code> Initial value: medium

PROPERTY	VALUE
font	<code>[<'font-style'> <'font-variant'> <'font-weight'>]? <'font-size'> [/ <'line-height'>]? <'font-family'>] caption icon menu message-box small-caption status-bar</code>

ШРИФТЫ ПРИМЕР

```
1 <p> Default font </p>
2 <p class="p1"> 24px Arial bold </p>
3 <p class="p2"> 16px Helvetica green </p>
4 <p class="p3"> 20px Courier italic blue </p>
5 <p class="p4"> 20px Arial Narrow italic bold
  </p>
```

HTML

```
1 p {
2   padding: 0 0 10px 0;
3   margin: 0;
4 }
5 .p1 {
6   font-family: Arial;
7   font-size: 24px;
8   font-weight: bold;
9 }
10 .p2 {
11   font-family: Helvetica;
12   font-size: 16px;
13   color: #0F0;
14 }
```

Default font

24px Arial bold

16px Helvetica green

20px Courier italic blue

20px Arial Narrow italic bold

<http://jsfiddle.net/Kep22/2/>

BACKGROUND

Универсальное свойство `background` позволяет установить одновременно до пяти характеристик фона. Значения могут идти в любом порядке, браузер сам определит, какое из них соответствует нужному свойству. Для подробного ознакомления смотрите информацию о каждом свойстве отдельно. В CSS3 допустимо указывать параметры сразу нескольких фонов, перечисляя их через запятую.

PROPERTY	VALUE
<code>background-color</code>	<code><color></code> <code>transparent</code> Initial value: <code>transparent</code>

PROPERTY	VALUE
<code>background-image</code>	<code><uri></code> <code>none</code> Initial value: <code>none</code>

PROPERTY	VALUE
<code>background-repeat</code>	<code>repeat</code> <code>repeat-x</code> <code>repeat-y</code> <code>no-repeat</code> Initial value: <code>repeat</code>

PROPERTY	VALUE
<code>background-position</code>	<code>[[<percentage> <length> left center right] [<percentage> <length> top center bottom]?]</code> <code>[[left center right] [top center bottom]]</code> <code>inherit</code> Initial value: <code>0% 0%</code>

PROPERTY	VALUE
<code>background</code>	<code><'background-color'></code> <code><'background-image'></code> <code><'background-repeat'></code> <code><'background-attachment'></code> <code><'background-position'></code> Initial value: <code>n/a</code>

<http://jsfiddle.net/HXRJC/4/>

BACKGROUND-IMAGE VS IMAGE

Использование «background-image»

- Используйте это свойство, если картинка не является частью контента.
- Используйте «background-image», если Вы делаете текст картинкой (например, в параграфах или заголовках).
- Это свойство можно использовать, если Вы предполагаете, что Вашу страницу захотят распечатать, и Вы не хотите, чтобы рисунок попал на печать по умолчанию.
- Применение «background-image» также может быть уместным, если Вы хотите улучшить время загрузки страниц.
- «background-image» используют, если необходимо сделать видимой только часть картинки.

Использование IMG

- Использование IMG (вместо с атрибутом «alt») уместно, если Ваше изображение является частью контента, как например, рисунок, схема или диаграмма.
- Используйте тег IMG, когда Вы предполагаете, что люди могут захотеть распечатать Вашу страницу, и картинка должна по умолчанию попасть на печать.
- Используйте этот тег, если изображение имеет важное семантическое значение, например, предупреждающая иконка. Это будет гарантировать, что картинка отобразится во всех программах-агентах, например, скринридерах, а не только в браузерах.
- Используйте тег IMG, когда Вы предполагаете использование масштабирования изображения браузером.
- Используйте IMG для многослойного наложения изображений в IE6.
- Использование IMG вместе со свойством «z-index» уместно, если необходимо растянуть фоновую картинку по всей ширине окна.
- А ещё применение IMG может дать разительный прирост производительности по сравнению со свойством «background-image».

СТИЛЕВЫЕ СВОЙСТВА LIST

- Списком называется взаимосвязанный набор отдельных фраз или предложений, которые начинаются с маркера или цифры. Списки предоставляют возможность упорядочить и систематизировать разные данные и представить их в наглядном и удобном для пользователя виде.
- Любой список представляет собой контейнер:
 - `` - маркированный список.
 - `` - нумерованный список.
- Каждый элемент списка должен начинаться с тега ``.

Property	Value
list-style-type	disc circle square decimal decimal-leading-zero lower-roman upper-roman lower-greek lower-latin upper-latin armenian georgian none Initial value: disc

Property	Value
list-style-image	<uri> none Initial value: none

Property	Value
list-style-position	inside outside Initial value: outside

СТИЛЕВЫЕ СВОЙСТВА LIST ПРИМЕР

```
1 <ul class="u1">
2   <li>Mercury</li>
3   <li>Venus</li>
4   <li>Earth</li>
5 </ul>
6 <ul class="u2">
7   <li>Mercury</li>
8   <li>Venus</li>
9   <li>Earth</li>
10 </ul>
11 <ul class="u3">
12   <li>Mercury</li>
13   <li>Venus</li>
14   <li>Earth</li>
15 </ul>
16 <ul class="u4">
17   <li>Mercury</li>
18   <li>Venus</li>
19   <li>Mars</li>
20 </ul>
```




```
1 ul {
2   font-size: 18px;
3 }
4 .u1 {
5   list-style: none;
6 }
7 .u2 {
8   list-style-type: square;
9 }
10 .u3 {
11   list-style-image: url('http://www
12 }
13 .u4 {
14   list-style-position: inside;
15 }
16
```

1

JavaScript

- Mercury
- Venus
- Earth

- Mercury
- Venus
- Earth

-  Mercury
-  Venus
-  Earth

- Mercury
- Venus
- Mars

<http://jsfiddle.net/EWmJ2/2/>

СТИЛИЗАЦИЯ ТАБЛИЦ

Определяет, как браузер должен вычислять ширину ячеек таблицы, основываясь на ее содержимом.

auto - Браузер загружает всю таблицу, анализирует ее для определения размеров ячеек и только после этого отображает.

fixed - Ширина колонок в этом случае определяется либо с помощью тега `<col>`, либо вычисляется на основе первой строки. Если данные о форматировании первой строки таблицы по каким-либо причинам получить невозможно, в этом случае таблица делится на колонки равной ширины. При использовании этого значения, содержимое, которое не помещается в ячейку указанной ширины, будет «обрезано» либо наложено поверх ячейки.

Property	Value
table-layout	auto fixed Initial value: auto

Пример: <http://jsfiddle.net/y2a7g>

СТИЛИЗАЦИЯ ТАБЛИЦ

border-collapse - устанавливает, как отображать границы вокруг ячеек таблицы.

border-spacing - задает расстояние между границами ячеек в таблице. border-spacing не работает в случае, когда для таблицы установлено свойство border-collapse со значением collapse.

Property	Value
border-collapse	collapse separate Initial value: separate

Property	Value
border-spacing	<length> <length>? Initial value: 0

Пример: <http://jsfiddle.net/F3BN4/>

ВЕНДОРНЫЕ ПРЕФИКСЫ

Эти свойства работают только в определённых браузерах.

-moz- для Firefox

-ms- для Internet Explorer

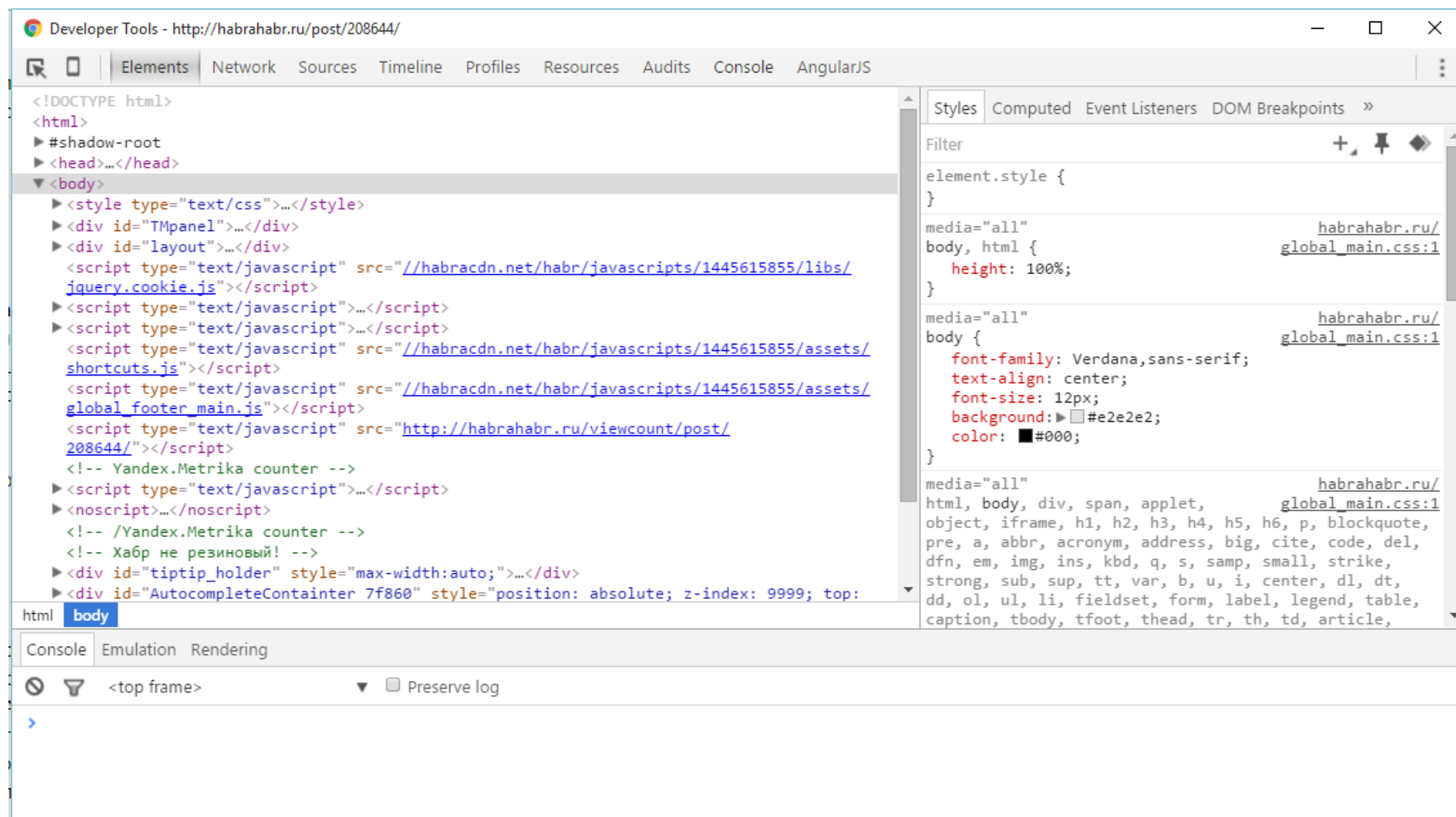
-webkit- для Chrome и Safari

-o- для Opera.

```
div {  
  -webkit-transition: all 4s ease; /* chrome, safari, opera */  
  -moz-transition: all 4s ease; /* firefox */  
  -ms-transition: all 4s ease; /* internet explorer */  
  -o-transition: all 4s ease; /* opera (old) */  
  transition: all 4s ease; /* standart CSS property */  
}
```

DEVELOPER TOOLS

- Developer Tools — удобный инструмент веб-разработки. Он содержит в себе необходимый функционал для создания и отладки полноценных веб-приложений.



ПОЛЕЗНЫЕ ССЫЛКИ

1. CSS справочник: <http://htmlbook.ru/css>
2. Поддержка CSS стилей различными браузерами: <http://caniuse.com/>
3. Спецификации W3C : <http://www.w3.org/Style/CSS/>



**THANKS FOR
ATTENTION!**

VIKENTII EKGART, SARATOV, RUSSIA