

Модуль - Верстка

4/10 занятие

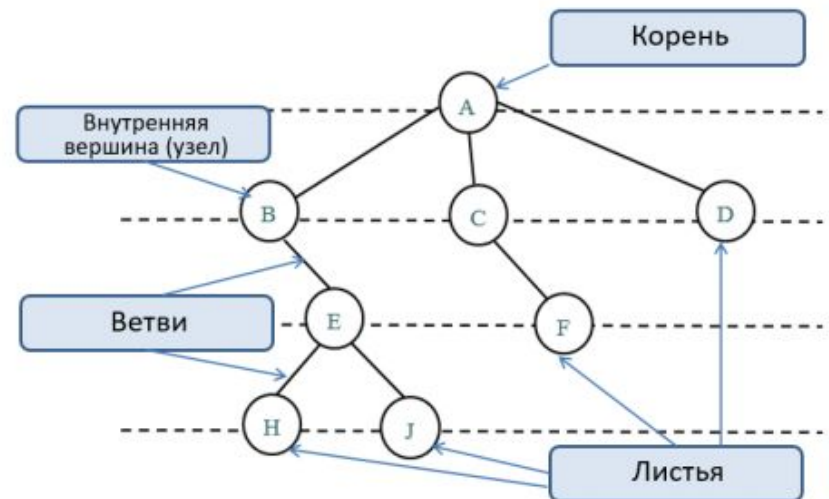
DOM

Объекты и деревья

Объект как тип данных используется для хранения коллекций различных значений(свойств) и более сложных сущностей

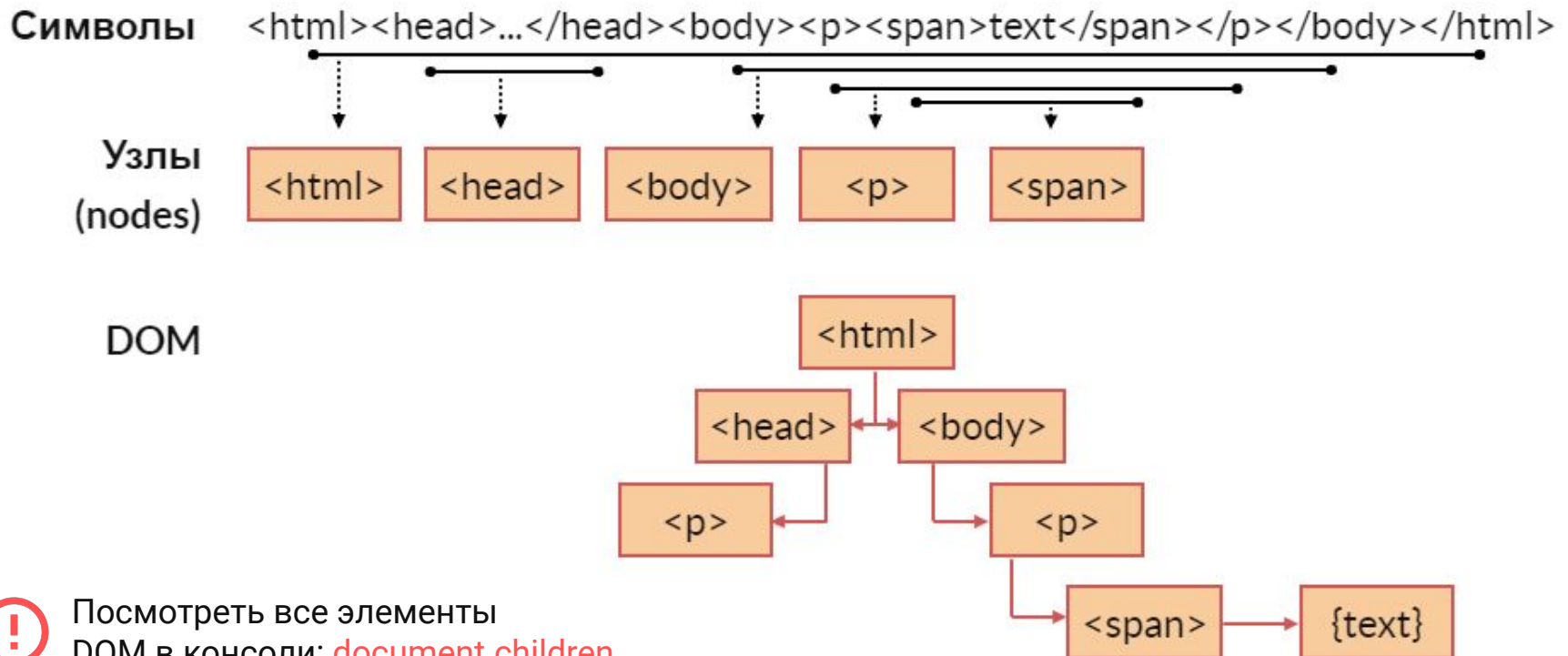
```
ОбъектСтол = {  
    высота: 60  
    ширина: 120  
    материал: {ОбъектБереза}  
}
```

Значением свойства объекта может быть другой **объект**, так образуются **иерархическое дерево**



DOM

Document Object Model (объектная модель документа) - это программный интерфейс, позволяющий получить доступ к содержимому HTML документов, а также изменять содержимое, структуру и их оформление.



Посмотреть все элементы
DOM в консоли: `document.children`

ПОСТРОЕНИЕ DOM

HTML

```
<html>
<head>
  <title>Beautiful page</title>
</head>
<body>

  <p>
    Once upon a time there was
    a looong paragraph...
  </p>

  <div style="display: none">
    Secret message
  </div>

  <div></div>
  ...

</body>
</html>
```

(теги)

DOM tree

```
documentElement (html)
  head
    title
  body
    p
      [text node]
    div
      [text node]
    div
      img
    ...
```

Render tree

```
root (RenderView)
  body
    p
      line 1
      line 2
      ...
    div
      img
    ...
```

(элементы)

ЭЛЕМЕНТ КАК ОБЪЕКТ


Каждый **html-элемент** обладает набором **свойств** и их значений, которые дают браузеру представление о том, как его отображать

Можно представить стили в виде "объекта":

```
стили_элемента_div = {  
  border: 1px solid #000  
  font-size: 16px,  
  border-radius: 0  
}
```

А так выглядит объект свойств ноды DOM:

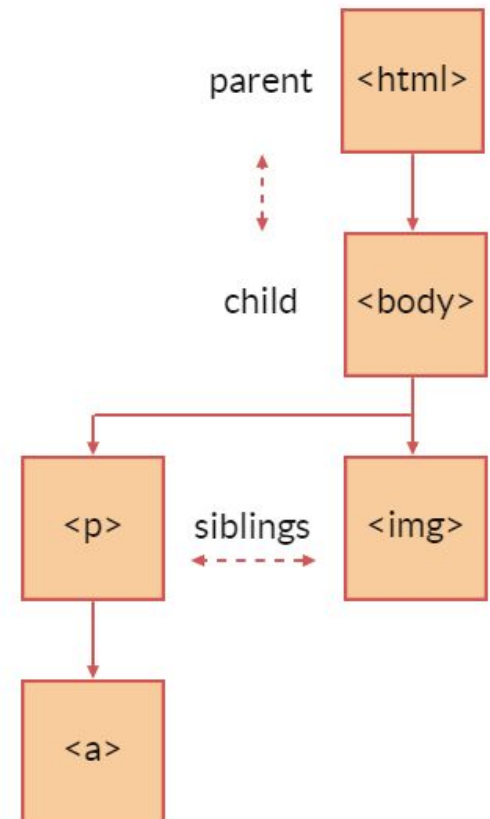
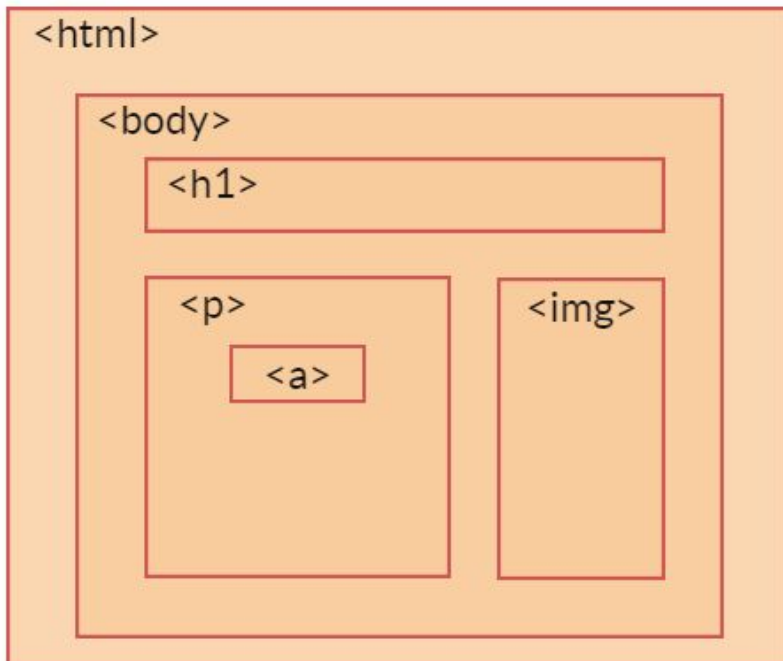
```
CSSStyleDeclaration : {  
  borderStyle: "solid"  
  borderTop: "0px solid rgb(0, 0, 0)"  
  borderTopColor: "rgb(0, 0, 0)"  
  borderTopLeftRadius: "0px"  
}
```

 Посмотреть все свойства html элемента можно в DevTool браузера:
Computed -> Filter -> "show all"

НАСЛЕДОВАНИЕ СВОЙСТВ

Наследование - это передача свойств от родительского элемента к его потомкам

Например, все вложенные в **<html>** теги по-умолчанию наследуют от него *размер шрифта*



Задание 1

- Откройте любой файл index.html и в devTool проверьте наследование font-size от body к дочерним элементам
- Создайте inheritance.html с базовой разметкой, структурой и ответьте на вопросы:

```
<body style="color:red;">
  <div style="font-size:20px;">
    <div>
      <p>почему текст красного цвета? <br>
        <b> - почему эта часть выделена жирным?</b>
      </p>
    </div>
  </div>
  <div style="color:green;">
    <p>почему текст меньше и зеленого цвета?</p>
  </div>
</body>
```


CSS

CSS

Cascading Style Sheets (каскадные таблицы стилей) - средство, позволяющее изменять визуальные свойства HTML-элементов

```
<!DOCTYPE html>
<html lang="ru">
  <head>
    <title>Заголовок страницы</title>
    <link rel="stylesheet" href="/css/style.css">
  </head>
  <body>
    <style>
      h1 {
        color: orange;
      }
    </style>
    <h1>Оранжевый заголовок</h1>
    <p style="color:red; font-weight: bold;">Этот текст
      будет красного цвета и с жирным начертанием</p>
  </body>
</html>
```

Связывание

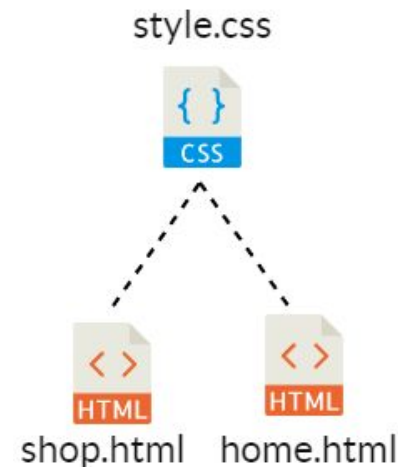
Вложение

Встраивание (inline)

ФАЙЛЫ СТИЛЕЙ

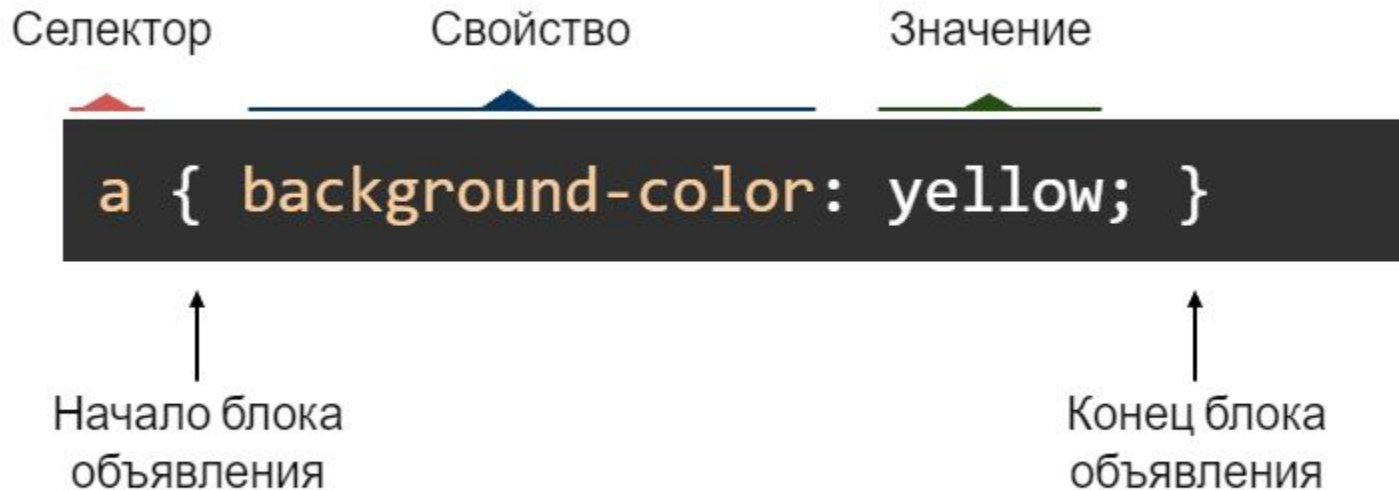
Наиболее важные причины вынесения CSS в отдельный файл:

- ✓✓ Для каждой логической части сайта свой файл стилей
Например, для меню - menu.css, для слайдера - slider.css,
для остальных элементов страницы - style.css
- ✓✓ Синтаксис html отделен от css
- ✓✓ Создание шаблона и темы для сайта
- ✓✓ Можно создавать библиотеки, дополнять темы



СИНТАКСИС

Селектор - указывает браузеру какой элемент DOM необходимо выбрать, для изменения его свойств



Селектор и блок объявлений образуют **правило**

КЛАССЫ И ИДЕНТИФИКАТОРЫ

id - уникальный идентификатор, назначается типовому тегу для выделения его среди подобных

class - для группирования тегов с одинаковыми признаками и свойствами

```
<h2>Заголовок</h2>  
<a class="text-red">Ссылка красного цвета</a>  
<h2 id="secondHeading">Заголовок</h2>  
<p class="text-red">Текст красного цвета</p>  
<p class="text-red text-big">Текст красного цвета и увеличенным размером шрифта</p>
```

 Классов у одного тега может быть несколько, а идентификатор один

NAMING CONVENTION

- ★ Это негласная **договоренность** использовать единый стиль именования классов, идентификаторов и переменных в JavaScript
- ✓✓ Классы несут смысловую нагрузку о предназначении элемента или группы элементов, понятную другим разработчикам
- ✓✓ Допускается использование символов (- и _) для разделения слов
- ✓✓ Часто применяется методология БЭМ
- ✓✓ Идентификаторы обычно записывают в стиле "CamelCase"

```
<div class="about-australia" id="textSection045">  
  <p class="about-australia--info_01">Кенгуру является одним из символов...</p>  
  <p class="about-australia--info_02">Австралийский доллар — валюта Австралийского Союза...</p>  
</div>
```

Задание 2

1. внешние стили

- Подключите новый файл css к inheritance.html
- Напишите правило для селектора “p” на определение свойства font-size, равного 26px;
- Убедитесь, что размер текста стал больше

2. внешний шрифт

- Подключить через css <https://fonts.google.com/specimen/Inter>
- Заменить его в теге body

перерыв

СЕЛЕКТОРЫ

РАЗНОВИДНОСТИ СЕЛЕКТОРОВ

Основные:

- По названию тега (tagName)
- По классу (.className)
- По идентификатору (#identifierName)
- "Все потомки" и "прямые потомки" (ul li; ul > li)
- Универсальный селектор (*) - выбор всех элементов

По ситуации:

- По атрибуту ([htmlAttrName])
- По соседним (p + p; p ~ h1)
- По псевдоклассам и псевдоэлементам (:last-child)

СЕЛЕКТОРЫ ПО ТЕГУ

Селектор тега - выберет все теги на странице с указанным названием

HTML

```
<div class="my_newclass-1" >
  <h1 class="title" >Заголовок</h1>
  <p>Первый абзац текста</p>
  <p>Второй абзац текста</p>
</div>
```

CSS

```
p {
  text-align: left;
  padding-left: 15px;
  font-size: 14px;
}
h1 {
  text-align: center;
  padding-bottom: 25px;
  color: #5A8385;
}
```

СЕЛЕКТОР ПО КЛАССУ

Селектор класса в CSS начинается с символа точка "." и выбирает все элементы имеющие класс с указанным названием

HTML

```
<div class="animal cat">кот</div>
<div class="animal dog">пёс</div>
<div class="brick">кирпич</div>
```

CSS

```
.animal {
  margin: 0 auto;
  background: url(../images/bg.png);
  background-color: grey;
  max-width: 800px;
}

.cat {
  font-size: 20px;
}
```

СЕЛЕКТОР ПО ИДЕНТИФИКАТОРУ

Селектор идентификатора начинается с символа "#" и выбирает один элемент с указанным id, так как идентификатор всегда **уникальный(!)**


HTML


```
<div class="cat">кот</div>  
<div class="dog" id="sharik">пёс</div>  
<div class="brick">кирпич</div>
```

CSS

```
#sharik {  
  font-weight: 400;  
  line-height: 1em;  
}
```

КОМБИНИРОВАНИЕ СЕЛЕКТОРОВ

 **A B** - будут выбраны элементы по селектору **B**, обязательно являющиеся потомками для элемента **A**

 **AB** - будут выбраны только те элементы, которые можно выбрать как по селектору **A** так и по **B**

HTML

```
<ul class="list list-1" >
  <li>Пункт меню</li>
  <li><span>Пункт меню</span></li>
  <li>Пункт меню</li>
</ul>
<ol class="list list-2">
  <li>Пункт меню</li>
  <li><span>Пункт меню</span></li>
  <li>Пункт меню</li>
</ol>
```

CSS

```
.list-2 li span {
  color: green;
}
.list span {
  font-size: 14px;
}

ol.list {
  text-align: center;
}
```

ГРУППИРОВАНИЕ СЕЛЕКТОРОВ

```
h1, h2, h3, h4, h5 {  
  text-align: left;  
  font-family: "Open Sans", Arial, sans-serif;  
}  
h1 {  
  font-size: 24px;  
}  
h2 {  
  font-size: 20px;  
}  
h3 {  
  font-size: 18px;  
}  
h4 {  
  font-size: 16px;  
}
```





ПРИОРИТЕТЫ СЕЛЕКТОРОВ

ПРИОРИТЕТЫ СТИЛЕЙ

К html элементу может применяться **несколько "стилей"** (изменений значений свойств) одновременно, но при отображении страницы будет использован **только один из них**

В таком случае, выбор стиля определяется приоритетом в следующем порядке(от меньшего к большему):

- 1. Стили браузера
 - 2. Стили автора:
 - 1. Специфичность css селекторов(будет рассмотрена далее)
 - 2. Inline вставка в html(атрибут style)
 - 3. Указание значения с **!important** (color: #fdfdfd**!important**)
- 

 При равном приоритете выбирается значение из правила "ниже в коде"

ОПРЕДЕЛЕНИЕ СПЕЦИФИЧНОСТИ

Специфичность - условная величина, определяющая какие значения свойства будут применены браузером, если селекторы нескольких правил выбрали один и тот же элемент

HTML

```
<div class="pets" >  
  <h1>Домашние питомцы</h1>  
  <p class="cat">Кот Барсик</p>  
  <p class="cat">Кот Бегемот</p>  
</div>
```

CSS

```
.pets p {  
  font-size: 14px;  
}  
  
.cat {  
  font-size: 16px;  
}  
  
p {  
  font-size: 18px;  
}
```

ПРИМЕР РАСЧЕТА СПЕЦИФИЧНОСТИ

Встраивание свойств (inline) = 1000

#id = 100

.class = 10

Tag = 1

<p style="color:red;">

#id_name {display:block; }

.classname {width:100%; }

a {cursor:pointer; }

*	/* ~> специфичность = 0	*/
li	/* ~> специфичность = 1	*/
li:first-line	/* ~> специфичность = 2	*/
ul li	/* ~> специфичность = 2	*/
ul ol+li	/* ~> специфичность = 3	*/
ul li.red	/* ~> специфичность = 12	*/
li.red.level	/* ~> специфичность = 21	*/
#t34	/* ~> специфичность = 100	*/
#content#wrap	/* ~> специфичность = 200	*/

Задание 4

Стилизируйте текст, как это описано в предложениях, обязательно добавив следующий css код: `p { font-size:20px; }` `li { text-decoration: underline }:`

(<p>)Этот текст крупный

(<p>)Этот текст красного цвета

(<p>)Этот текст крупный и подчеркнутый

(<h1>)Заголовок зеленого цвета

(()Элемент списка крупный и подчеркнутый

 ()Элемент списка крупный и подчеркнутый

 ()Элемент списка крупный, подчеркнутый и красного цвета

 ()Элемент списка подчеркнутый

Конец занятия

Задание на дом

1. Вынести стили buttons_collection.html и cta_section.html в отдельный css файл
2. Выполнить 1- 14 задания в тренажере <http://flukeout.github.io/>
3. *Выполнить задания 14 - 32