

HTML&CSS

НАШИ ПРАВИЛА



Включенная камера



Вопросы по поднятой руке



Не перебиваем друг друга



Все вопросы, не связанные с тематикой курса (орг-вопросы и т. д.), должны быть направлены куратору



Подготовьте свое рабочее окружение для возможной демонстрации экрана (закройте лишние соцсети и прочие приложения)

ЦЕЛЬ

Изучить способы подключения CSS. Изучить синтаксис CSS: селекторы, блоки деклараций.

ПЛАН ЗАНЯТИЯ

- Разбор домашнего задания
- Контейнеры: div, span
- Семантические теги: header, footer, main, section и пр.
- CSS. Основы
- Стили: inline styles, internal CSS, external CSS
- CSS селекторы
- Class и id атрибуты

Абсолютные ссылки

Абсолютные ссылки создаются путем добавления в атрибут **href** тега `<a>` абсолютного URL адреса.

Абсолютный URL предполагает указание полного адреса страницы, на которую вы ссылаетесь, включая протокол и доменное имя.

Примеры значения атрибута **href**:

`“https://www.example.com/”`

`“https://www.example.com/courses/”`

Относительные ссылки

Относительные ссылки создаются путем добавления в атрибут **href** тега `<a>` относительного URL адреса.

Относительный путь означает, что указание пути на нужный файл или страницу вашего сайта начинается относительно каталога, в котором расположена страница со ссылкой, либо относительно корневого каталога сайта

Примеры значения атрибута **href**:

`"main.html"`

`"./flowers.pdf"`

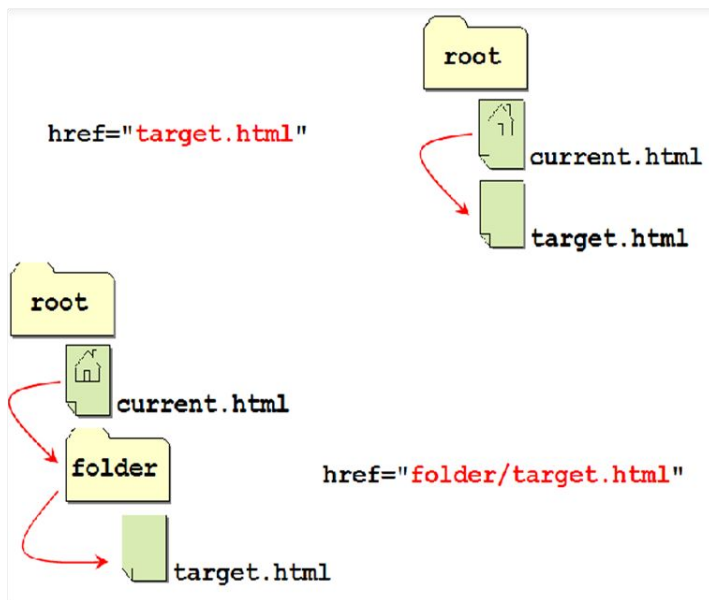
`"file1/main.html"`

`"folder1/folder2/main.html"`

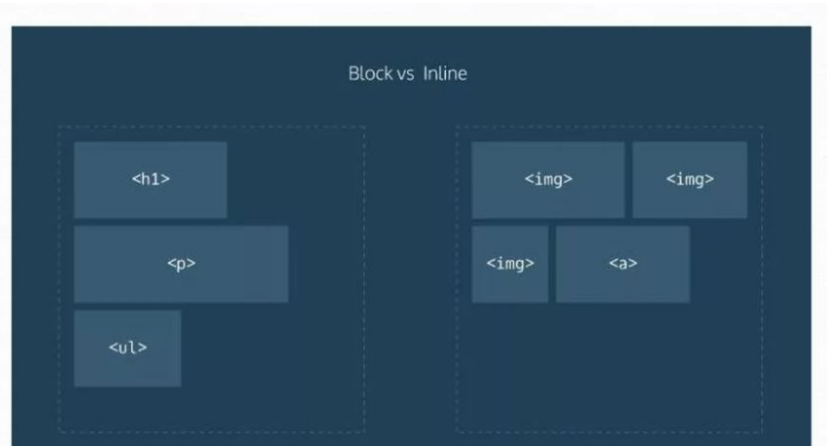
`"../main.html"`

`"../../main.html"`

Схема адреса относительной ссылки



Блочный и строчный контейнеры



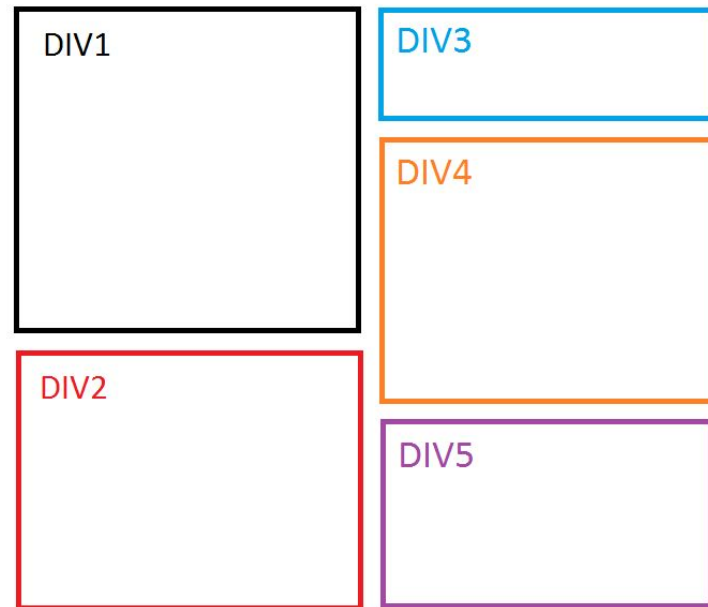
Блочные элементы - это элементы, которые занимают всю строку горизонтального пространства на веб-странице. Это означает, что они автоматически начинаются с новой строки, а последующие элементы автоматически переносятся на новую строку.

Строчные элементы – это элементы, которые являются частью строки и занимают такое количество пространства, которое необходимо для отображения их содержимого.

div - блочный контейнер

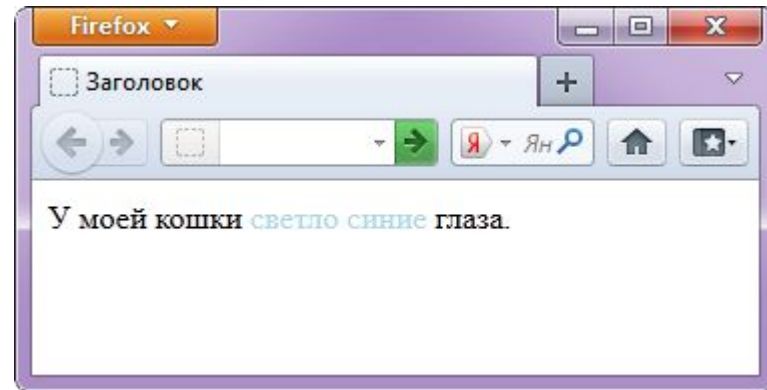
Является универсальным контейнером для контента веб-страницы.

```
<div>Контент</div>
```



span - строчный контейнер

Тег `` предназначен для определения строчных элементов документа. С помощью тега `` можно выделить часть информации внутри других тегов и установить для нее свой стиль.

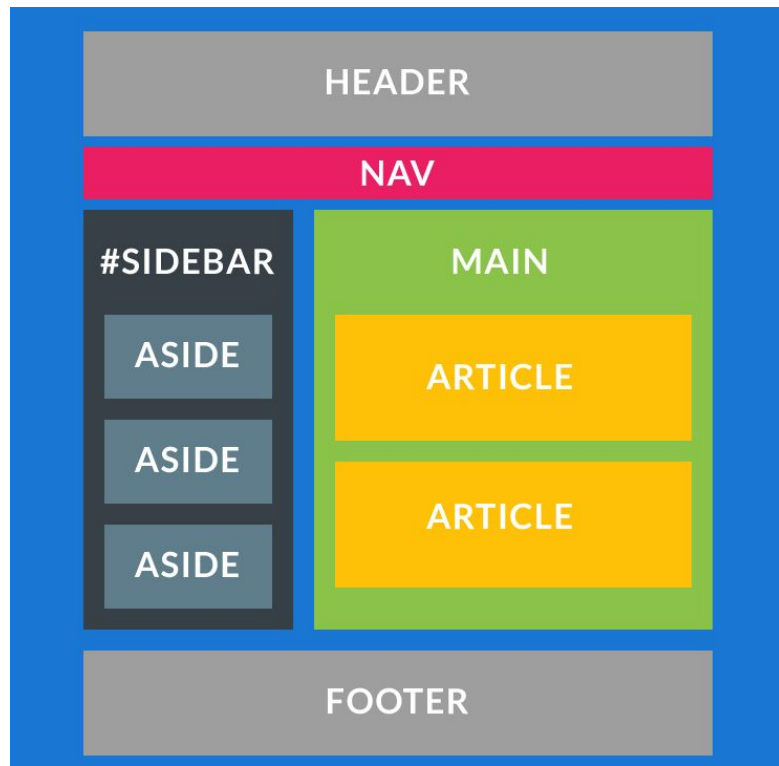


``Контент``

Семантические теги



Семантическая вёрстка —
подход к разметке, который
опирается не на содержание
сайта, а на смысловое
предназначение каждого
блока и логическую
структуру документа.



Для чего нужна семантика:

1. Для accessibility*. Основной инструмент незрячих или частично незрячих людей для просмотра сайтов не браузер, а скринридер, который читает текст со страницы вслух. Этот инструмент «зачитывает» содержимое страницы, и семантическая структура помогает ему лучше определять, какой сейчас блок, а пользователю понимать, о чём идёт речь. Таким образом семантическая разметка помогает большему количеству пользователей работать с вашим сайтом.
2. Чтобы сайт был выше в поисковиках. Поисковики не разглашают правила ранжирования, но известно, что наличие семантической разметки страниц помогает поисковым ботам лучше понимать, что находится на странице, и в зависимости от этого ранжировать сайты в поисковой выдаче.
3. Семантика прописана в стандартах. При работе, например, над поддержкой существующего приложения вам будет проще работать с кодом, написанным по общим стандартам

**Accessibility – это область знания, которая занимается изучением вопросов доступности сайтов, мобильных приложений и программного обеспечения для людей с ограниченными возможностями.*



В html, чтобы выполнить требования семантики используют специальные теги

<header>: содержит логотип и основную навигацию.

<main>: включает основной контент, такой как статьи и по

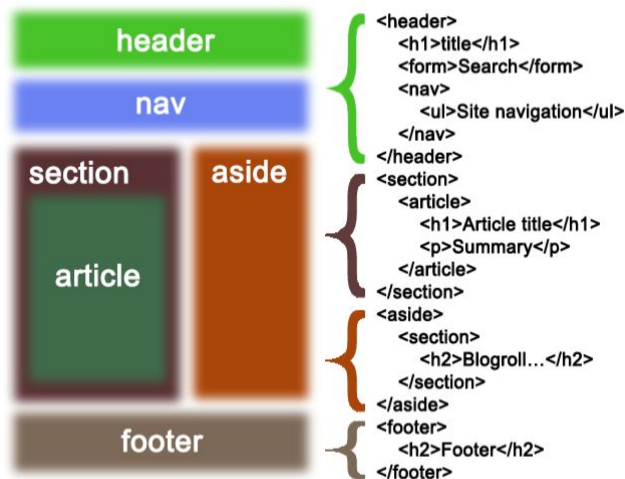
<footer>: содержит информацию об авторских правах и дополнительные ссылки.

<nav>: представляет собой меню навигации по сайту.

<article>: используется для отдельных статей или постс

<section>: разделяет контент на логические блоки.

<aside>: содержит дополнительную информацию, такую боковые панели или виджеты.



<https://developer.mozilla.org/ru/docs/Glossary/Semantics>



CSS. Введение

HTML



HTML the Skeleton



CSS



CSS the Skin



JS



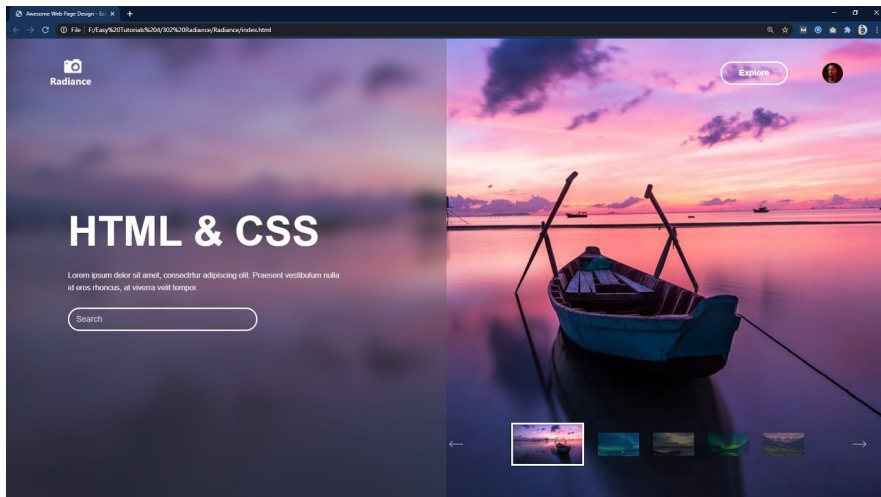
Javascript the Brain



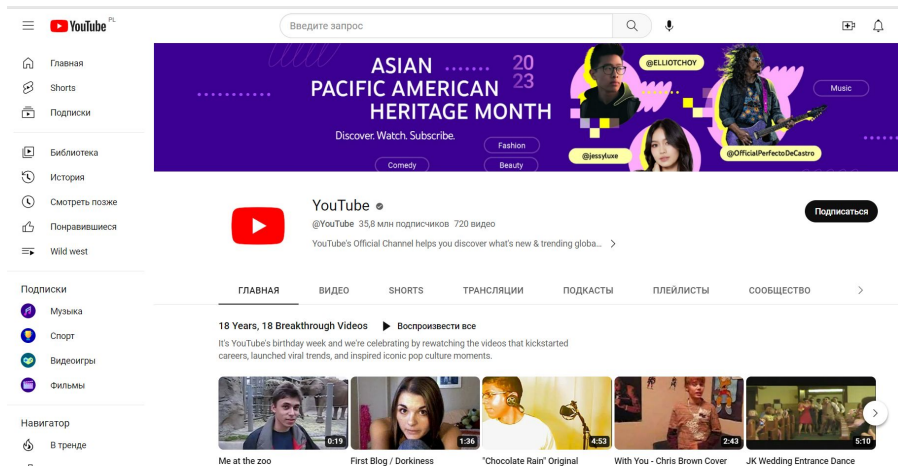
CSS (Cascading Style Sheets) - язык, который отвечает за описание внешнего вида HTML-документа.

Что делает CSS?

Если HTML структурирует контент на странице, то CSS позволяет отформатировать его, сделать более привлекательным для читателя.



HTML + CSS



HTML



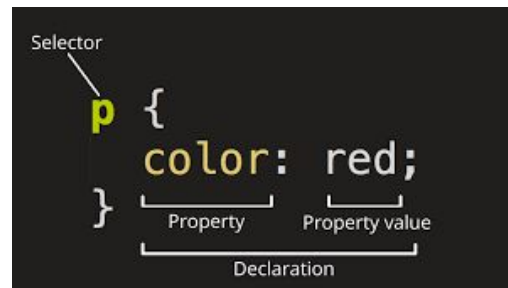
Синтаксис CSS

У языка CSS относительно простой синтаксис. Сначала прописывается селектор — он выбирает конкретный элемент на странице. Потом, после фигурных скобок, указываются свойства со значениями — между ними ставится двоеточие. Сами свойства отделяются друг от друга точкой с запятой.

Схема



Пример

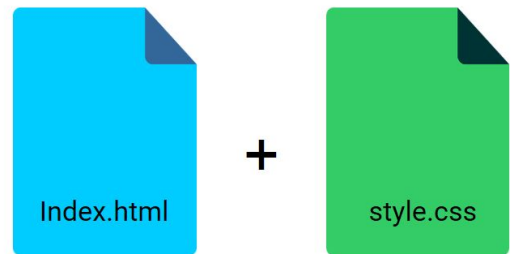


Варианты подключения стилей



Существует 3 основных способа подключения или добавления CSS стилей к HTML документу, каждый из которых подходит для определенного круга задач.

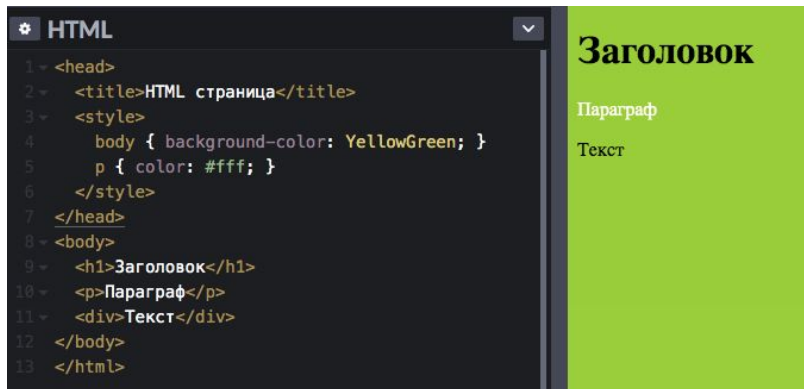
- CSS стили для конкретного тега атрибутом `style`
- Подключение с помощью тега `link` через внешний файл стилей CSS
- Добавление CSS с помощью тега `style` в HTML файл



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

С помощью атрибута style для тега

`<p style="color: red">Текст сообщения</p>`



Подключение с помощью тега link через внешний файл стилей CSS

Подключение стилей CSS является использование элемента `link`, который позволяет подключать к HTML странице внешние файлы. Ссылка на внешний CSS файл помещается в контейнер `<head>` страницы:

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

В атрибуте `href` необходимо указать URL адрес файла, содержащего набор стилей CSS. Атрибуты `rel="stylesheet"` и `type="text/css"` указывают, что указанный файл является таблицей стиля в формате CSS.

Подключение с помощью тега link через внешний файл стилей CSS

Преимущества внешних CSS:

- Меньший размер страницы HTML и более чистая структура файла.
- Быстрая скорость загрузки.
- Для разных страниц может быть использован один и тот же .css файл.

Недостатки внешних CSS:

- Страница может некорректно отображаться до полной загрузки внешнего CSS.

Добавление CSS с помощью тега style в HTML файл

Этот метод подойдет когда нужно вставить (определить) стили для группы уникальных элементов страницы, то есть, набор стилей, которые используются только в пределах одной страницы и не нужны для корректной работы остальных страниц сайта.

В любом месте областей `<head>` и `<body>` HTML документа используйте тег `<style>`, внутри которого поместите необходимые CSS правила.

```
<style>

    h2 {

        color: red;

    }

</style>
```

CSS Селектор



Что такое селекторы?

Селекторы - это выражения, которые говорят браузеру, к какому элементу HTML нужно применить те или иные свойства CSS, определённые внутри блока объявления стиля.

```
h1 {  
  color: blue;  
  background-color: yellow;  
}  
  
p {  
  color: red;  
}
```

Типы базовых CSS селекторов

- Универсальный селектор
- Селекторы по типу элемента
- Селекторы по идентификатору
- Селекторы по классу
- Селекторы по атрибуту



Селектор по элементу (тегу)

Этот селектор CSS применяет стили ко всем элементам с одинаковым тегом. Например, для всех <div>, <h2>, <p> и так далее.

```
p {  
    color: red;  
}
```

Селекторы по классу

Для выделения какой-то группы объектов (элементов), которые необходимо наделить одними и теми же свойствами CSS, необходимо создать класс.

```
.class_name {  
  
    свойство1: значение;  
  
    свойство2: значение;  
  
}
```

Примечание: Имена классов должны начинаться с латинского символа и могут содержать в себе символ дефиса (-) и подчеркивания (_). Использование русских букв в именах классов недопустимо.

Селекторы по идентификатору

Селектор по идентификатору обозначается символом **#** (решётка) и применяет стили к элементу, для которого задан атрибут **id** с соответствующим значением. При этом у элемента может быть только один **id**, и этот **id** должен быть уникальным в пределах веб-страницы.

html code

```
<p id="article_content">В этой  
статье:</p>
```

css code

```
#article_content {  
  
    font-family: sans-serif;  
  
    font-weight: bold;  
  
}
```


Добавление класса

Чтобы добавить класс, необходимо для элементов (тегов), к которым будут применены свойства класса, прописать атрибут **class** с придуманным названием класса в качестве значения атрибута:

```
<h1 class="class_name">Текст сообщения</h1>
```



Ваша новая IT-профессия – Ваш новый уровень жизни

Программирование с нуля в
немецкой школе AIT TR GmbH