**План курса:**

1. Компоненты веб сайтов
2. Как клиенты взаимодействуют с серверами
3. Структура HTML страницы и теги HTML
4. Создание HTML страницы с разными тегами
5. Подключение библиотеки таблиц стилей CSS Bootstrap
6. Несколько HTML страиниц и навигация между ними
7. **Компоненты Веб сайтов**

HTML – HyperText Markup Language (язык гипертекстовой разметки)

CSS – Cascading Style Sheets (стили)

JavaScript – язык програмирования

1. **Как клиенты взаимодействуют с серверами**

Для взаимодействия клиента с сервером используется протокол HTTP (HyperText Transfer Protocol)

Для того, чтоб зашифровать данные, которые передаются по протоколу HTTP используется TLS (Transport Layer Security)

Сейчас используется HTTPS (HyperText Transfer Protocol Security)

Комп (запрос)🡪 сервер (ответ)🡪Комп

1. **Структура HTML страницы и теги HTML**

Структрура html файла начинается с «!» + tab

Теги это инструкция для веб сайта.

Открывающий и закрывающий тег должны быть на одном уровне

Основные теги:

**<html>** - корневой тег, внутри которого все остальные теги и он означает, что файл является именно html-формата.

**<head>** - мета информация, которая описывает html страницы, внутри <head> подключаются таблицы стилей CSS либо на JavaScript файл. В нем так же можно добавить текстовое описание для страницы.

**<body>** - основной тег (контейнер) для добавления всех остальных тегов, которые будут размещаться в html файле

**<div> (division – разделение)** - самый популярный тег, который ничего не означает, а является контейнером для отдельных элементов страницы, к которым нужно применить конкретные стили – стилизовать тот или иной блок на веб странице.

**<a>** - добавление гиперссылок.

**<img />** - добавление на веб странице изображение. Должен содержать атрибут, который указывает местоположение изображения на веб странице. **Самозакрывающийся тег.**

**<h1>, <h2>, <h3>, <h4>, <h5>, <h6>** - заголовки разный уровней, где 1 самый большой, 6 самый маленький. Как правило, <h1> используется на странице только 1 раз.

**<p> (параграф)** – добавление больших блоков текста, состоящих из нескольких абзацев.

**<meta>** - используется «внутри» тега <head> и добавляет мета информацию, которая описывает страницу. К примеру, с помощью тега <description> можно добавить информацию, которая описывает страницу и именно это текстовое описание будет использовано роботами Google для корректного индексирования страницы в поиске

**<link>** - указать ссылку на другой файл, который нужно подключить к html файлу, к примеру на таблицу стилей CSS или на файл JavaScript

**<style>** - можно подключать стили CSS непосредственно в файле HTML страницы (но так делать не рекомендуется, а лучше подключать внешние файлы с помощью тега <link>)

**<script>** - можно добавлять JavaScript код непосредственно в файле HTML страницы (но так делать не рекомендуется, а лучше подключать внешние файлы с помощью тега <link>)

**<title>** - название вкладки сайта (находится внутри тега <head>)

В теге может размещаться атрибут (либо несколько атрибутов) тега, она находится в открывающемся теге.

Записывается, например

<button **class=”btn**”> </button>

<img ***src*=”img.ipg” *class*=”rounded-img” *alt*=”My photo”** /> (“src” – fnnhb

**Атрибуты**

Указываются внутри открывающегося тега и **состоят из названия атрибута, знака равенства и значения атрибута в двойных кавычках “ ”**

**src** – путь к файлу изображения

**class** – устанавливает класс элемента

**alt** – название изображение (если неверно прописать путь к изображению выведется только название, если верно, то при наведении курсора на изображение появится прямоугольник с названием)

Необходимо соблюдать разметку (иерархическую структуру) HTML страницы. Тег, вложенный в другой тег, должен быть на 2 отступа правее тега, в который он вложен.

1. **Создание HTML страницы с разными тегами**

**Visual Studio Code – редактор кода от Microsoft**

**Расширения VSCode, которые желательно установить:**

* **Live Server** – в реальном времени обновляет страницу в браузере
* **Prettier** – помогает соблюдать иерархию кода, сам выстраивает структуру
* **Auto Close Tag** – автозакрытие тега
* **Auto Complete Tag** – автозаполнение тега
* **Rainbow Brackets** – выделение цветом скобочек
* **Code Spell Checker** – проверка орфографии кода

1. **Подключение библиотеки таблиц стилей CSS Bootstrap**

<https://getbootstrap.com/docs/5.2/getting-started/introduction/> - подключаем библиотеку bootstrap с помощью ссылки в теге <head>

**<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-beta1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-0evHe/X+R7YkIZDRvuzKMRqM+OrBnVFBL6DOitfPri4tjfHxaWutUpFmBp4vmVor" crossorigin="anonymous">**

Теперь, по ссылке <https://bootstrapshuffle.com/classes/> можно добавлять классы стилизации в объекты кода.

2:17

1. **Несколько HTML страниц и навигация между ними**

Class “row” – колонки

Class “col” – столбцы

[**https://favicon.io/favicon-generator/**](https://favicon.io/favicon-generator/) **- генератор иконки в названии вкладки**