Муниципальное бюджетное общеобразовательное учреждение  
 «Средняя школа №26 с углубленным изучением предметов естественнонаучного цикла»

**Индивидуальный проект  
 по теме:  
 «Современные языки web-разработки»**

Обучающийся: Кулигин Олег Алексеевич

Руководитель проекта: Семёнова Марина Юрьевна,  
 учитель математики

г. Иваново, 2022

**Оглавление**

1. Введение ……………………………………………………3
2. История появления…………………………………………4
3. Серверные и клиентские языки программирования……..5
   1. Серверные языки программирования……………….5
   2. Клиентские языки программирования………………6
4. Языки web-программирования
   1. Язык HTML……………………………………………7
   2. Язык CSS………………………………………………8
   3. Язык JavaScript……………………………………….10
5. Практическая часть………………………………………...11
6. Анкетирование……………………………………………..12
7. Заключение…………………………………………………14
8. Список используемых источников………………………..14

**Введение**

**WEB-**программирование – это раздел программирования для создания различных web-приложений, в основном динамичных сайтов. Web-языки программирования не стоят на месте и постоянно развиваются, именно поэтому невозможно выучить хотя бы один *язык программирования* (далее ЯП) наизусть. В основном сайты пишутся на таких языках как: **HTML**, **CSS** и **JavaScript** (далее JS), первые технологии на их основе учатся за несколько недель, но уже позволяют верстать небольшие сайты, пусть и малофункциональные. Языки разметки **HTML** и **CSS** легко читаются и понимаются даже детьми, которые совсем немного знают английский и еще не умеют программировать. Взрослому же человеку хватит и одной недели плотного изучения, чтобы изучить все основы и уже начать брать небольшие заказы. Но веб-разработка не ограничивается языками разметки (это еще не языки программирования). Пройдя очень низкий порог входа и начав изучать **JavaScript** мы увидим один из самых универсальных и быстроразвивающихся языков на сегодняшнем рынке. Если же продолжим дальше углубляться во **фронтенд**, то встретим достаточно мощные и быстрые инструменты, чтобы делать быстрые и современные веб-приложения. Примеры таких приложений: **Discord**, **Figma**, **Google Drive**, приложения карт и другие.

**Цель работы:** исследовать строение сайта, изучить языки web-программирования, создание личного сайта.

**Актуальность:** данная проблема актуальна для общества, так как IT-сфера, это наше будущее

**Объект исследования:**WEB-программирование.

**Предмет****исследования:** языки WEB-программирования

**История появления**

**HTML** (*от англ*. *HyperText Markup Language* – “Язык гипертекстовой разметки”) Был разработан Британским учёным Тимоном Берсеном-Ли приблизительно в 1986-1991 годах. HTML создавался как язык для обмена научной и технической документацией, пригодный для использования людьми, не являющимися специалистами в области вёрстки. С помощью HTML можно легко создать относительно простой, но красиво оформленный документ.

**CSS** (*от англ. Cascading Style Sheets* – “Каскадные таблицы стилей”) Был введён в HTML версии 3.0 в 1995 году W3C (World Wibe Web Concsortium – Консортиум Всемирной паутины). CSS это формальный язык стилей web-документа, написанного с использованием языка разметки (HTML). О том, как подключить CSS к HTML описано в приложении

**JavaScript** (*аббр*. JS) – это *динамически типизированный язык программирования высокого уровня*, разработанный компанией **Nombas** в 1992 году. JS используется для написания сайтов, конкретнее, он делает сайты “живыми”, добавляя анимации, события и создавая интерактивность web-страниц.

**Серверные и клиентские языки программирования**

Существующие языки программирования принято разделять на 2 вида, а именно серверные и клиентские. **Сервером** является удалённый ПК (хостинг), на котором находятся файлы сайта. **Клиент** – это сам пользователь, точнее его браузер, от которого поступают запросы на сайт.

Начнём с **клиентских** языков программирования. Их главная особенность уже заключена в названии. Программы написанные и запущенные на ПК пользователя, обрабатываются на нём же. Если говорить проще, то обработка клиентских языков производится на компьютере пользователя, составившего программу. Самый распространённый клиентский язык программирования – **JavaScript.** Я считаю, что главный недостаток клиентских языков, это то, что любой пользователь может посмотреть исходный код сайта, и использовать его в своих целях. Преимущество же заключается в удобстве использования, а именно то, что обработка скриптов выполняется на компьютере пользователь, без отправки на сервер. Так же, не все браузеры поддерживают клиентские языки. Приведу несколько примеров браузеров, который поддерживают данные языки: *Google Chrome, Mozilla Firefox, Internet Explorer* (лично я пользуюсь вторым вариантом).

Итак, с клиентскими языками познакомились, теперь перейдём к **серверным** языкам программирования. Их особенность заключается в том, что запросы, поступающие на сайт от пользователя, обрабатываются на удалённом сервере, и возвращаются обратно на сайт. В отличии от клиентских языков, код программы посмотреть может только разработчик сайта. Самый распространённый СЕРВЕРНЫЙ язык программирования – это PHP (HyperText Preprocessor), препроцессор гипертекста изначально, а позже названный Personal Home Page - интенсивно применяемый для разработки веб-приложений.

Представим следующую ситуацию:

Пользователь захотел что-либо сделать на вашем сайте (кликнуть на кнопку, перейти по ссылке и т.д). Что делает наш сайт в этот момент? Он отправляет запрос на сервер(хостинг), этот запрос там обрабатывается и возвращается гостю в виде обычной HTML-страницы. Таким образом, работа ваших скриптов полностью зависит от сервера, на котором находится ваш сайт

Наверно, у вас возник вопрос, что такое сервер и как на него разместить сайт? Далее я постараюсь ответить на этот вопрос.

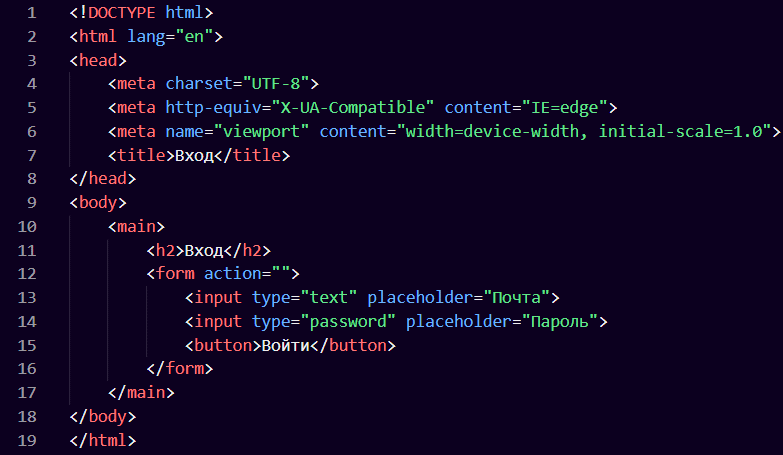
Сервер (далее хост), это некий компьютер, к которому можно подключиться из любой точки мира. Он выглядит не как обычный ПК, а немного иначе. Это вытянутая коробка, содержащая в себе накопители с высоким объемом памяти, большое кол-во оперативной памяти и мощный процессор. изображение такого сервера вы сможете найти в приложении. Как же запустить наш сайт на таком сервере? При покупке сервера (он покупается на месяц\год) вы можете подключиться к нему по FTP протоколу (File Transfer Protocol) – это протокол для передачи файлов по сети. И переместить свой сайт на жёсткий диск сервера. Но на этом установка сайта на сервер не закончилась, да, мы добавили его на сервер, но мы не знаем как на него зайти. Для этого нам нужен домен. Домен, это имя нашего сервера (например school26.ivedu.ru), домен покупается на один год. После покупки домена и хостинга, можно заходить на наш сайт.

**Языки WEB-Программирования**

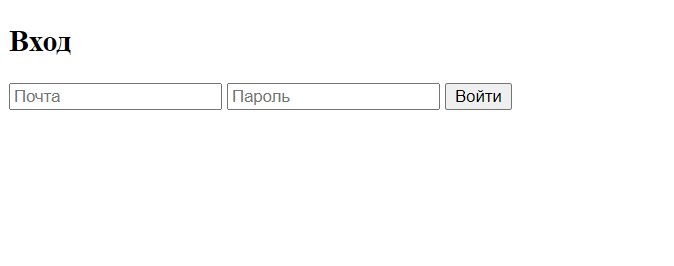
**Язык HTML**

Это не язык программирования, а всего лишь технология, позволяющая создавать веб-страницы. Правда без стилей такие страницы будут больше похожими скорее на Word-документы, чем на сайты, которые мы привыкли видеть. Изначально, на заре развития интернета, все сайты выглядели невзрачно. К примеру, посмотрите на пример ниже. Так может выглядеть сайт из 90х.

**Код**



**Результат**



Весь **HTML** строится на тегах и их наполнении. **Тег** – это элемент страницы (как в примере <h2>Вход</h2>)

Структура **HTML**-документа:

Стоит заметить, что вся страница делится на два больших раздела.

1. <head> - это информация о странице
2. <body> - это контент на странице

При создании сайта, мы в основном работаем в разделе <body>.

**Основные текстовые теги:**

1. **<h1> - <h6\>** - заголовки 1-6 уровней
2. **<p>** - абзац текста
3. <**ol>** - нумерованный список
4. <**ul>** - маркированный список
5. **<li>**- элемент списка
6. **<a href="путь" target="blank">** - ссылка (target="blank" чтобы открыть в нужной вкладке)

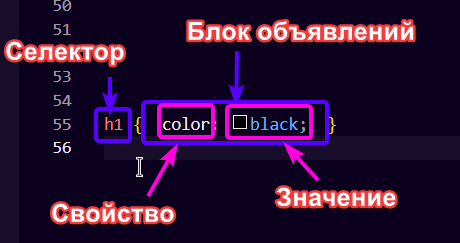
**Язык CSS**

**CSS (Cascading Style Sheet) –** используется для оформления страницы, написанной на HTML. Каскадные таблицы стилей обычно описывают правила форматирования элементов, находящихся на странице с помощью свойств и допустимых значений этих свойств. CSS - язык стилей. Сегодня неотъемлемая часть любого сайта. Именно с помощью него элементы на странице приобретают свои цвета, шрифты, размеры, расположение и даже могут стать подвижными. Хоть CSS и сложнее HTML, он все равно изучается намного проще и быстрее, чем языки программирования. После того, как вы его изучите и начнете применять, вы сможете сверстать любой, даже достаточно сложно сделанный визуально сайт. Но пока что он будет статичным - это не рабочее приложение, а просто его графическая часть, которую еще предстоит программировать, если нужно добиться какого-то интерактива. Но есть сайты, которые используют только HTML и CSS - например лендинги (одностраничные рекламные сайты). Они несут статическую информацию о каком-то товаре или услуге, не меняется и уже могут быть вам по силам, если вы знаете только эти две технологии. Но лучше двигаться дальше.

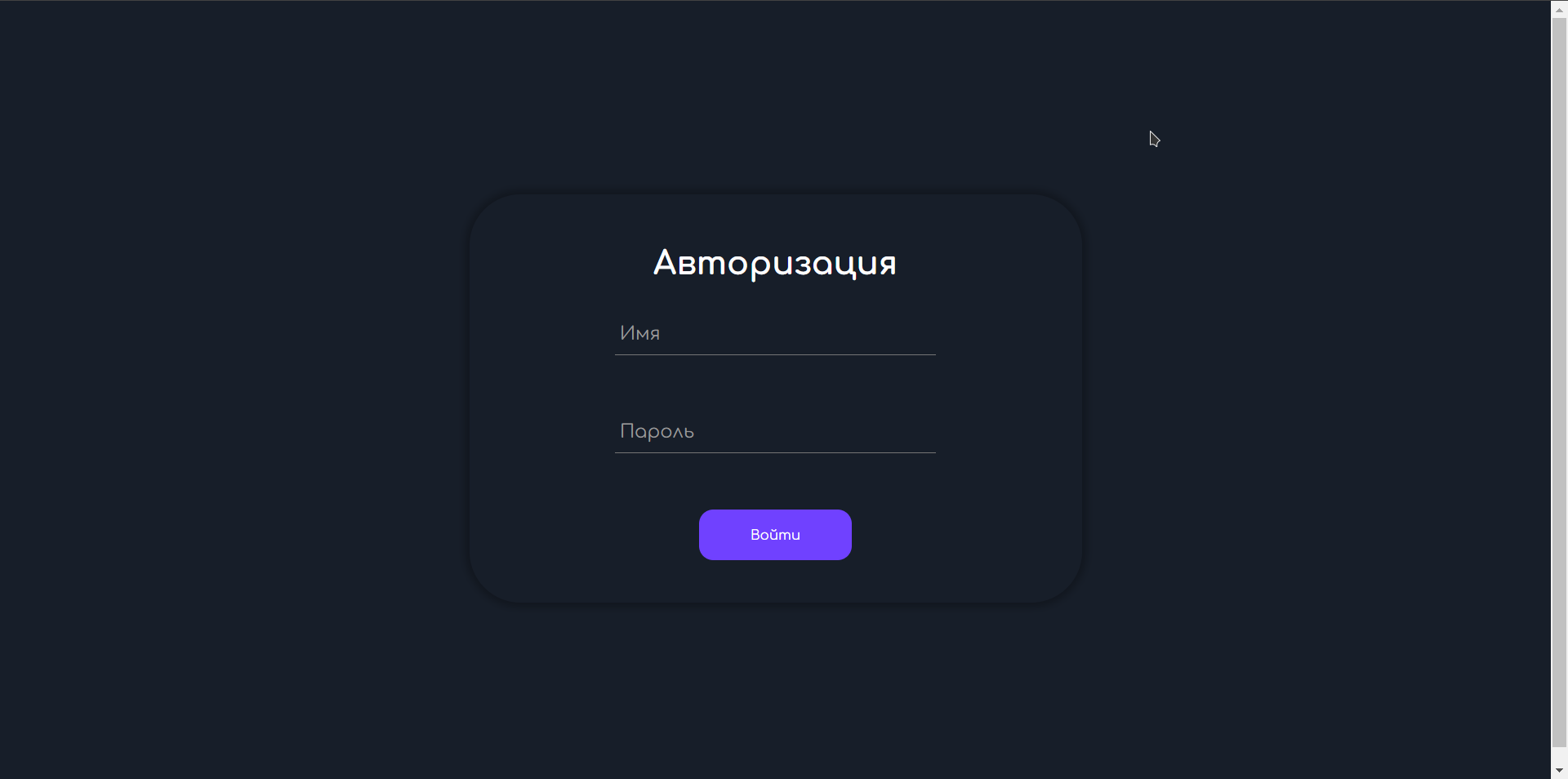
**Объявление** свойств состоит из двух частей:

1. **Селектор –** элемент веб-страницы
2. **Блок объявления –** команда форматирования.

Селектор сообщает браузеру о том, какой элемент нужно форматировать, а в блоке объявления указываются свойства, которые нужно применить



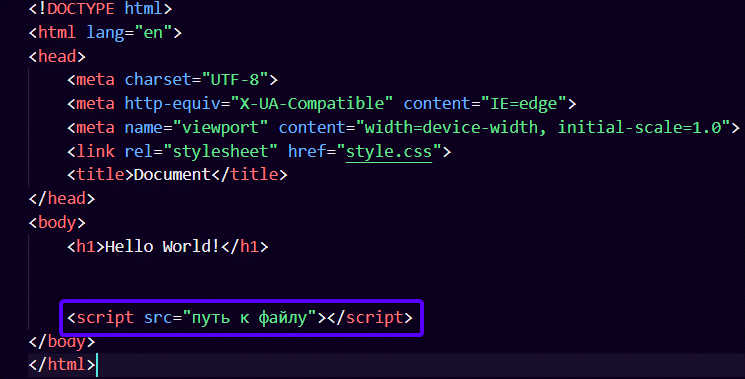
**Пример сайта с использованием CSS**



**Язык JavaScript**

**Javascript** – это язык сценариев, который в основном используется для создания интерактивных элементов на WEB-сайте. Его можно использовать для создания анимаций, построения меню, проверки правильности заполнения форм. осле того, как вы изучите **HTML** и **CSS** и захотите двигаться дальше, логичным выбором станет изучение **JS**. Во-первых, это полноценный язык программирования. Он достаточно легкий для новичков, но при этом с него потом можно без проблем перейти почти на любой другой современный язык. Во-вторых, он позволит вам делать в браузере полноценные приложения и может полностью управлять всей страницей. С помощью него можно сделать, как и легкие эффекты, в духе анимации переходов, выпадающие меню и различные галереи, так и более сложные приложения, такие как **Google Drive**, **YouTube**, **Discord** и куча других сайтов, которыми вы с удовольствием пользуетесь.

Подключение скриптов JS происходит следующим образом. В конце тега body в коде страницы HTML мы прописываем некую команду подключения:



<script src="путь к файлу"></script>

**Пример сайта, написанном на JavaScript**



**Практическая часть.**

Я провел опрос сред учащихся 10го класса и получил 23 анкеты. Моя анкета состоит из 9 вопросов. Далее я распишу подробнее о вопросах и ответов учащихся на них

**Вопросы анкеты:**

* На сколько хорошо вы можете переводить большие текста?
* Вы предпочитаете работать в группе или одни?
* Знакомы ли вы с сайто-строением?
* Можете ли вы организовать группу людей?
* Видите ли в себя лидером группы или компании в будущем?
* Вы творческая личность?
* Вы можете принять быстрое решение, от которого зависит дальнейшая жизнь вас самих или вашей команды?
* Знакомы ли вы с каким-либо языком программирования? Если да, то с каким?
* Если бы вы начали заниматься написанием кода (программированием), то в какую сферу углубились бы?

**Ответы учащихся:**

* *На сколько хорошо вы можете переводить большие текста?*

**15 человек** ответили, что угадывают значение незнакомого слова по контексту.

**6 человек** ответили, что пользуются словарём всегда.

**2 человека** ответили, что вообще не пользуются словарём.

* *Вы предпочитаете работать в группе или одни*?

**11 человек** ответили, что проще работать одному

**12 человек** ответили, что работать в группе лучше

* *Знакомы ли вы с сайтостроением?*

**17 человек** ответили, что знакомы с сайтостроением.

**6 человек** ответили, что не знакомы с сайтостроением.

* *Можете ли вы организовать группу людей?*

**19 человек** ответили, что могут

**4 человека** ответили, что не могут

* *Видите ли в себя лидером группы или компании в будущем?*

**17 человек** ответили, видят себя лидером группы или компании

**6 человек** ответили, что не видят себя лидером.

* *Вы творческая личность?*

**10 человек** ответили, что считают себя творческой личностью

**4 человека** ответили, что не считаю себя творческой личностью

**9 человек** ответили, что считают себя творческой личностью лишь на половину.

* *Вы можете принять быстрое решение, от которого зависит дальнейшая жизнь вас самих или вашей команды?*

**17 человек** ответили, что могут.

**6 человек** ответили, что не могут.

* *Знакомы ли вы с каким-либо языком программирования? Если да, то с каким?*

**18 человек** ответили, что вообще не знакомы с каким-либо языком программирования

**5 человек** ответили, что знакомы. Среди них самыми популярными оказались С++ и JavaScript

* *Если бы вы начали заниматься написанием кода (программированием), то в какую сферу углубились бы?*

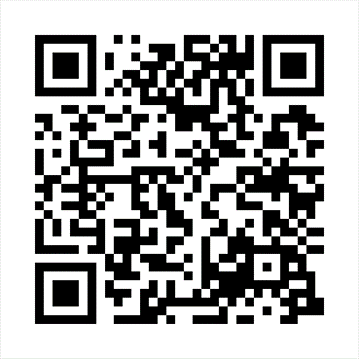
**10 человек** ответили, что хотели бы разрабатывать игры

**9 человек** ответили, что хотели бы создавать приложения

**7 человек** ответили, что хотели бы создавать сайты.

вывод

В ходе исследования я изучил такие языки, как, HTML, CSS, JavaScript. На основе полученных знаний я написал свой сайт и выложил его в интернет. Что бы на него зайти, нужно перейти по ссылке: [project.petrovich2.ru](https://project.petrovich2.ru/) или отсканировать **QR-код**:



Заключение

Как видно из исследования, на данный момент существуtт много языков web-программирования. Каждый из них имеет свои плюсы и минусы. Каждый из них позволяет улучшить интерфейс своего сайта и его взаимодействие с пользователем. Для своего дальнейшего изучения я выбрал язык JavaScript, так как он наиболее прост в изучении, по сравнению с другими языками.

Список используемых источников

<https://ru.wikipedia.org/wiki/JavaScript>

<https://ru.wikipedia.org/wiki/CSS>

<https://ru.wikipedia.org/wiki/HTML>

<https://learn.javascript.ru/>

https://vertex-academy.com/tutorials/ru/samouchitel-po-htmlcss-s-nulya/