Департамент образования города Москвы

Государственное бюджетное профессиональное

образовательное учреждение города Москвы

«Колледж малого бизнеса № 4»

|  |  |
| --- | --- |
|  | Дисциплина, профессиональный модуль, междисциплинарный курс:  ПМ.02. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности |

|  |  |  |
| --- | --- | --- |
|  |  | Допущен(а) к защите  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «\_\_\_»\_\_\_\_\_\_\_ 2022г. |

КУРСОВАЯ РАБОТА

Тема: «Раработка веб-платформы для заполонения DisDoc»

Специальность 09.02.05 Прикладная информатика (по отраслям)

Фамилия, инициалы обучающегося \_\_\_\_\_\_\_\_\_ Рыбаков А.С,

(подпись)

Группа ПС-31.19

Руководитель \_\_\_\_\_\_\_\_\_\_ А.Д. Шпатович

(подпись)

2022 г.

Департамент образования города Москвы

Государственное бюджетное профессиональное

образовательное учреждение города Москвы

«Колледж малого бизнеса № 4»

|  |  |
| --- | --- |
|  | Дисциплина, профессиональный модуль, междисциплинарный курс:  ПМ.02. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности |

|  |  |  |
| --- | --- | --- |
|  |  | Допущен(а) к защите  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «\_\_\_»\_\_\_\_\_\_\_ 2022г. |

ЗАДАНИЕ

на выполнение курсовой работы

Обучающемуся (ФИО) Рыбаков Александр Сергеевич

Курс 3 группы ПС-31.19 специальности 09.02.05 Прикладная информатика (по отраслям)

1. Тема работы: «Раработка веб-платформы для заполонения DisDoc
2. Дата выдачи задания « \_\_\_ » \_\_\_\_\_\_\_ 2022 г.
3. Исходные данные.

Цель: разработать сервис для заполнение DisDoc.

Задачи:

1.Изучение разработке веб сайтов на Python;

2.Верстка сайта;

3.Разработка серверной части для сайта.

4.Изучение библиотек python для работы с word.

1. Этапы выполнения и срок сдачи обучающимся завершенной работы

В ходе работы я познакомился со сложностями при поиске необходимой литературы, с работой на языке python и версткой сайтов.

каждого элемента игровой физики, правильным оформлением научных работ. Сдача была произведена до 25 февраля 2022 г.

Ф.И.О., должность преподавателя-руководителя: Шпатович А.Д., преподаватель

Руководитель (подпись) \_\_\_\_\_\_\_\_\_\_\_ « \_\_\_ » \_\_\_\_\_\_\_ 2022 г.

Обучающийся (подпись) \_\_\_\_\_\_\_\_\_\_\_ « \_\_\_ » \_\_\_\_\_\_\_ 2022 г.

Департамент образования города Москвы

Государственное бюджетное профессиональное

образовательное учреждение города Москвы

«Колледж малого бизнеса № 4»

РЕЦЕНЗИЯ

На курсовую работу по теме:

«Раработка веб-платформы для заполонения DisDoc»

Дисциплины: МДК 02.01 Разработка, внедрение и адаптация программного обеспечения отраслевой направленности.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Группа ПС-31.19

Дата защиты\_\_\_\_\_\_\_\_

Оценка\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ дата\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оглавление

[ВВЕДЕНИЕ 6](#_Toc96354701)

[Глава 1. Роль Веб-платформ в мире 7](#_Toc96354702)

[Часть 1. Веб-платформа 7](#_Toc96354703)

[Часть 2. Пользовалский интерфейс 8](#_Toc96354704)

[Часть 3. Редактор кода Visual Studio Code 9](#_Toc96354705)

[Часть 4.Выбор языка програмирования 11](#_Toc96354706)

[Часть 5. Язык программирования Python 11](#_Toc96354706)

[Часть 6. Библиотека Python flask 11](#_Toc96354706)

[Часть 7. Библиотека Python docxtpl 11](#_Toc96354706)

[Часть 8. Word 11](#_Toc96354706)

[Глава 2. Разработка 12](#_Toc96354707)

[Часть 1.Верстка Web страниц . 12](#_Toc96354708)

[Часть 2.Разработка серверной части сервиса. 14](#_Toc96354709)

[Часть 3.Тестирование. 16](#_Toc96354710)

[ЗАКЛЮЧЕНИЕ 31](#_Toc96354714)

[Список используемой литературы 32](#_Toc96354715)

# ВВЕДЕНИЕ

Актуальным на сегодняшний день являются различные веб-платформы, которые стали популярными в наше время из-за их развития технологий. В

Объектом исследования является процесс создания веб платформы для заполнение DisDoca.

Предметом исследования является разработка серверной часть вкб приложения и верстка сайта.

Выполненная работа состоит из введения, основной части из трёх глав (Юнити, физических основ игры и практических результатов), заключения и списка литературы.

Этот веб-платформа - это набор технологий, разработанных в качестве открытых стандартов с помощью Консорциума Всемирной паутины и других организаций по стандартизации, таких как веб-гипертекстовая технология применения рабочей группы, в кодировке Unicode, консорциум, в инженерной проектной группы интернета, и послужила основой международного. это общий термин введен Консорциума Всемирной паутины, и в 2011 году он был определен как "платформу для инноваций, консолидации и эффективности затрат" консорциумом W3C генеральный директор Джефф Джаффе. строится на Evergreen Web (где происходят быстрые, автоматические обновления программного обеспечения, сотрудничество поставщиков, стандартизация и конкуренция) позволила добавить новые возможности при одновременном устранении рисков безопасности и конфиденциальности. Кроме того, разработчики могут создавать совместимый контент на единой платформе.

Цель данной курсовой работы состоит в разработке веб приложения. В связи с поставленной целью необходимо решить следующие задачи: рассмотреть предметную область, т.е. что потребовалось для разработки физики; выбрать инструментальные средства для разработки программного модуля; провести анализ аналогов подобных программных модулей; разработать полноценную физику. Для достижения данной цели в работе решаются следующие задачи:

Изучение python;

Разработка html страниц;

Разработка Бекенда.

# Глава 1. Роль Веб-платформ в мире

Часть 1. Веб-платформа

Этот веб-платформа - это набор технологий, разработанных в качестве открытых стандартов с помощью Консорциума Всемирной паутины и других организаций по стандартизации, таких как веб-гипертекстовая технология применения рабочей группы, в кодировке Unicode, консорциум, в инженерной проектной группы интернета, и послужила основой международного. это общий термин введен Консорциума Всемирной паутины, и в 2011 году он был определен как "платформу для инноваций, консолидации и эффективности затрат" консорциумом W3C генеральный директор Джефф Джаффе. строится на Evergreen Web (где происходят быстрые, автоматические обновления программного обеспечения, сотрудничество поставщиков, стандартизация и конкуренция) позволила добавить новые возможности при одновременном устранении рисков безопасности и конфиденциальности. Кроме того, разработчики могут создавать совместимый контент на единой платформе.

Веб—платформа включает в себя технологии—компьютерные языки и API-интерфейсы, которые первоначально были созданы в связи с публикацией веб-страниц. Это включает в себя HTML-код, как CSS, SVG и, в MathML, вай-Ария, с ECMAScript, в WebGL, веб-хранилище, проиндексированных API базы данных, веб-компоненты, WebAssembly, веб-работников, с WebSocket, API-интерфейс Geolocation, сервер-отправлено событий, событий DOM, медиа фрагментов, с XMLHttpRequest, Кросс-происхождения совместное использование ресурсов, файл API, Поддержка RDFa, вуф, протокол HTTP, протокол TLS 1.2, и ИРИ.

**Часть 2. Пользовательский интерфейс**

Современное программное обеспечение совершило революцию во взаимодействии рядового пользователя с новыми технологиями. На момент 2022-го года утвердилось мнение, что для рядового пользователя программное обеспечение должно предоставлять максимально удобное и интуитивно понятное управление имеющимися функциями. Профессиональное программное обеспечение для использования предприятиями и индивидуальными предпринимателями должно содержать максимальное количество полезных в деле функций, позволять кроссплатформенную работу, быстро исполнять поставленные задачи (быть оптимизированным).

Интерфейс – это полный спектр возможностей взаимодействия пользователя и компьютера. В этот термин включается аппаратное и программное обеспечение персонального компьютера или иного используемого устройства, дающее информацию пользователю и позволяющее ему производить эффективную работу с его компьютером.

К аппаратному обеспечению компьютерного интерфейса относится огромное количество известных нам устройств, но бывают и узконаправленные устройства, неизвестные широкому спектру рядовых пользователей. Наиболее известными и используемыми устройствами, относящимися к аппаратному обеспечению компьютерного интерфейса, являются: компьютерная мышь, компьютерная клавиатура, полноразмерные наушники или наушники-вкладыши, Bluetooth-адаптеры для подключения Bluetooth-устройств, внешняя звуковая карта, микрофон, монитор, графический планшет, системный блок с его компонентами, web-камера. Программным обеспечением компьютера называется всё то, что обеспечивает пользователю взаимодействие с компьютером на экране монитора, то есть всё то, что он делает в интерфейсе компьютера. В программное обеспечение также входят различные руководства и обучающие материалы. Их можно найти как внутри программы в виде текстового файла, так и в виде ссылки, которая отправит пользователя на сайт производителя этой программы, на котором уже, в свою очередь, будет представлено руководство, видео или иной способ донести информацию. Также стоит отметить, что современное профессиональное программное обеспечение, как правило, не продаётся в готовом виде. На момент 2022-го года тенденция «подписок» и версий захватывает всё больше фирм-производителей программного обеспечения. По разным тарифам и на разных условиях можно получить доступ к программному обеспечению. Это может быть как полный доступ навсегда (за полную цену), так и ограниченная бесплатная версия, которая через определённое установленное время прекратит работу, если не внести оплату.

Немаловажным является и тот факт, что в настоящее время количество пользователей, взаимодействующих с интернетом через мобильные устройства (смартфон, планшет) превышает количество пользователей, взаимодействующих с интернетом через персональные компьютеры. Вследствие этого в разработке приложений и сайтов появился принцип “Mobile First”, означающий, что интерфейс и функционал приложения или веб-сайта изначально разрабатывается для мобильных устройств, а далее адаптируется под персональные компьютеры. Несмотря на это, доля персональных компьютеров в мире огромна, что невозможно игнорировать. Для подавляющего большинства профессиональных задач экрана и мощности аппаратной части мобильных устройств не хватает, поэтому персональные компьютеры до сих пор являются неотъемлемой частью жизни и ещё долгое время не утратят своей популярности, по крайней мере до того момента, как мобильные устройства не сравнятся с ними по вычислительной мощности.

Дизайн программного обеспечения может напрямую влиять и влияет на его коммерческий успех. Если программное обеспечение не имеет аналогов и жизненно необходимо для работы предприятий, то, конечно, можно пренебречь дизайном и продавать приложение в том виде, в каком вам хочется, но если на рынке имеются конкуренты (если их не имеется, то они появятся), дизайну стоит уделить внимание. Никому не захочется пользоваться приложением с интерфейсом зеленого цвета и текстом желтого цвета с непонятным управлением. Дизайн должен быть принят пользователем, должен быть ему приятен, дополнять возможности технической части приложения и невидимой нитью объединять функции между собой.

**Часть 3. Редактор кода VISUAL STUDIO CODE**

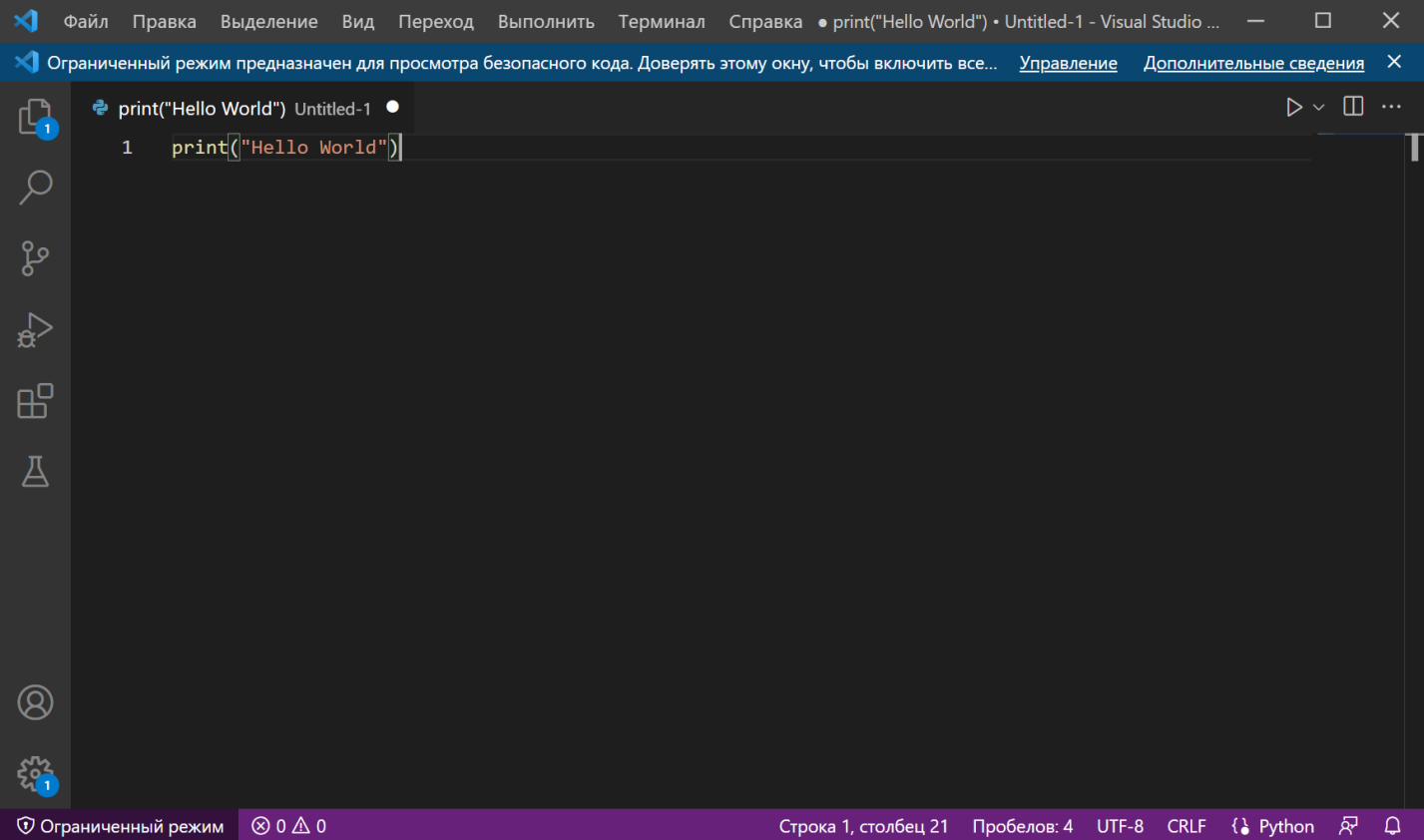
Любая программа представляет собой текст, который преобразуется в машинный язык и мы видим исполнение нашего кода. Программу можно написать в «Блокноте», но если она состоит больше, чем из пяти строк, это будет проблематично. Поэтому существуют специальные среды разработки, текстовые редакторы кода, которые посредством искусственного интеллекта подсказывают, что нужно написать, находят ошибки в коде, а также подсвечивают ключевые слова в определенном коде.

Существует множество таких программ, самыми популярными и часто используемыми являются: Sublime Text, Atom, Visual Studio Code.

Atom – представляет собой по сути, простой текстовый редактор, который лишь определяет, что введенный текст является кодом определенного языка программирования и подсвечивает его определенным цветом. Его функционал весьма ограничен, поэтому выбирать его для разработки крупного проекта не стоит.

Мной был сделан выбор между Sublime Text и Vusial Studio Code. Поскольку по функционалу и по популярности можно сказать, что они примерно равны, основываясь на своих ощущениях, я выбрал Visual Studio Code. Мне он более привычен и я имею больше опыта использования этой программы.

VISUAL STUDIO CODE



Visual Studio Code в своём интерфейсе на верхней части имеет такие вкладки как: «Файл», «Правка», «Выделение», «Вид», «Переход», «Выполнить», «Терминал», «Справка». Рассмотрим их плотнее, их функции и возможности использования.

Вкладка «Файл» – позволяет создавать, сохранять, открывать настраивать файлы, имеющиеся на компьютере пользователя, либо созданные в программе.

Вкладка «Правка» – позволяет отменять или возвращать совершенное в программе действие, копировать, вырезать и вставлять текст.

Вкладка «Выделение» – позволяет выделять нужный для выделения участок кода.

Вкладка «Вид» – позволяет настраивать внешний вид (Дизайн) программы. Практического применения для кода данная вкладка не имеет, можно работать и вовсе ни разу не открыв её. Она важна только для получения определенных ощущений от работы с программой, получения, так сказать, эстетического удовольствия. Очень приятно, когда пользователь волен сам менять в зависимости от настроения вид программы, в которой работает, либо выбрать любимое оформление.

Вкладка «Переход» – позволяет осуществлять переходы между редакторами.

Вкладка «Выполнить» – позволяет сразу выполнить ваш код.

Вкладка «Терминал» – позволяет взаимодействовать с файлами на компьютере, ставить задачи.

Вкладка «Справка» – позволяет получить ответы на вопросы, возникшие вследствие использования программы, решить проблему, с которой вы столкнулись, получить информацию, научиться работать с программой и многое другое, что может понадобиться человеку, который только начинает своё знакомство с Visual Studio Code.

Внизу на фиолетовой строке на представленном скриншоте мы можем видеть некоторую информацию и кнопки для взаимодействия. Слева – направо.

Плашка «Ограниченный режим» - говорит нам о том, что режим нашего взаимодействия с программой ограничен. Чем? Почему? Потому, что некоторые функции приложения ограничены, поскольку конкретное окно не является достоверным.

Далее мы видим счетчик проблем, что очень удобно. Если программа не работает, вследствие неправильного написания кода или допущения синтаксических ошибок, то мы сможем увидеть, во-первых: сколько этих ошибок было допущено, а во-вторых: получить справку, где эти ошибки и как их устранить на том языке программирования, на котором мы пишем код.

Строка, столбец – тут мы можем получить информацию о том, на какой строчке и в каком столбце мы в данный момент времени находимся. Кратко говоря, если мы видим следующую надпись «Строка 5, столбец 6», то мы можем сделать вывод, что наш курсор находится на пятой строке кода, на шестом столбце и этот вывод будет достоверным.

Далее мы видим счётчик количества пробелов в нашем коде.

Далее мы видим кодировку, она отвечает за то, в какой кодировке нужно читать наши символы, если выбрать неправильную кодировку, то мы увидим лишь непонятные символы, вместо текста, который имели в виду. В случае, показанном на скриншоте, кодировка выбрана UTF-8.

Далее на скриншоте мы видим “CRLF”. На этой вкладке мы можем выбрать последовательность конца строки.

Далее мы видим выбранный язык программирования, в случае, показанном на скриншоте, этот язык – Python, позже объясню почему. Язык, который мы выберем для написания программы будет отображаться в этом блоке.

Далее мы видим символьное обозначение блока, оно означает возможность отправить твит с отзывом.

И последним блоком, находящимся на нижней строке является блок уведомлений. Если мы получим какое-либо уведомление от приложения, оно будет отображено в этом блоке, поскольку на момент создания скриншота уведомлений не было, оно пустое.

Слева мы видим панель, на которой также присутствуют несколько блоков. азберём, что это за блоки и зачем они нужны. Сверху – вниз.

Самым верхним является блок «Проводник», «Проводник» показывает нам, что мы сейчас работаем в определенном файле, даёт информацию сохранена ли текущая версия файла или нет, что очень важно, поскольку если дописать что-то в коде, но не сохранить, исполняться будет последняя сохраненная версия.

Ниже мы видим блок «Поиск», который позволяет использовать наши текстовые запросы в этот блок для поиска внутри программы нужного ответа. Очень важный блок, который должен быть в любой профессиональной программе, для того, чтобы её освоение и использование пользователем, не являющимся профессионалом и не имеющим обширного опыта использования конкретной программы, становилось проще.

Ниже мы видим «Систему управления версиями», которая позволяет сохранять версии программы, выбирать между ними, выкладывать в систему Git.

Ниже мы видим блок «Запуск и отладка», этот блок позволяет запустить код, проверить его работоспособность, найти ошибки для их последующего исправления.

Ниже мы видим блок «Расширения», который позволяет устанавливать расширения, которые повышают удобство использования программы, позволяют решить особые задачи.

Ниже мы видим блок «Тестирования», которые позволяет протестировать нашу программу.

Предпоследним блоком является блок «Учетные записи», которые показывает учетную запись, под которой в данный момент мы сейчас работаем.

Последним блоком является блок «Настройки». С помощью этого блока можно настроить программу под себя.

В самом интерфейсе, мы можем видеть текстовый фрейм, открытую вкладку, аналогия приходит с браузером, можно быстро запустить программу или открыть расширенное взаимодействие, посредством нажатия на блок, представленный в виде символа трёх точек.

Каждую из этих вкладок можно открыть посредством «Горячих клавиш», то есть определенные значения на кнопках клавиатуры, которые нужно одновременно нажать, чтобы исполнить тот или иной функционал программы. Требует продвинутого знания программы.

Исходя из всего вышесказанного, можно сделать вывод, что редактор кода Visual Studio Code является удобным, простым. Имеет большой функционал и спектр возможностей, поэтому выбор редактора кода я остановил на нём.

**Часть 4. Выбор языка програмирования**

Для написания собственного веб-браузера, можно использовать многие языки программирования, самые популярные из них: Java, Python, C, C++, C#.

Рассмотрим каждый более подробно.

Язык программирования JAVA.

Java – строго типизированный объектно-ориентированный язык программирования. Используется для задач общего назначения. Разработан компанией Sun Microsystems, которую в последствии прибрела компания Oracle.

Приложения Java обычно транслируются в специальный байт-код, поэтому они могут работать на любой компьютерной архитектуре, для которой существует реализация виртуальной Java-машины. В рейтингах популярности языков программирования заняла 2-е место в 2020-м и 2021-м годах.

Плюсы языка программирования Java:

- простота изучения;

- является ООП;

- язык Java не зависит от платформы;

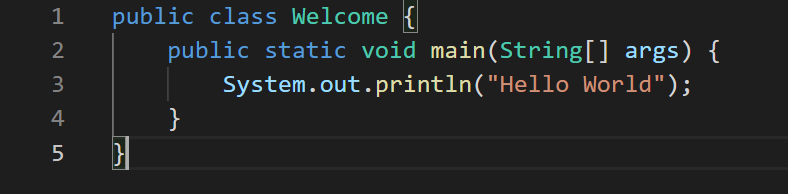
Минусы языка программирования Java:

- плохая производительность;

- код многословен;

- платные обновления для бизнеса и коммерческого использования;

Пример простейшей программы (вывод сообщения “Hello World”) на языке программирования Java.



Язык программирования C.

Язык программирования С – компилируемый сатически типизированный язык программирования общего назначения, разработанный в 1969 - 1973-х годах сотрудником Bell Labs Деннисом Ритчи как развитие языка Би. Первоначально был разработан для реализации операционно системы UNIX, но впоследствии был перенесён на множество других платформ. Согласно дизайну языка, его конструкции близко сопоставляются типичным машинным инструкциям, благодаря чему он нашёл применение в проектах, для которых был свойственен язык ассемблера, в том числе как в операционных системах, так и в различном прикладном программном обеспечении для множества устройств — от суперкомпьютеров до встраиваемых систем. Язык программирования Си оказал существенное влияние на развитие индустрии программного обеспечения, а его синтаксис стал основой для таких языков программирования, как C++, C#, Java и Objective-C.

Плюсы языка программирования С:

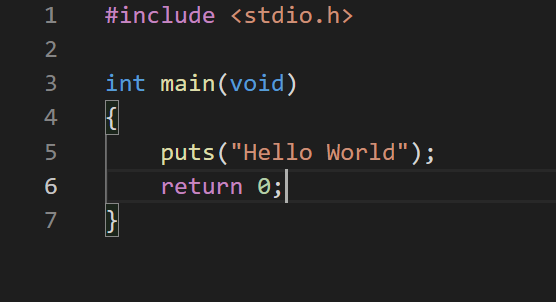
- является самым быстрым высокоуровневым языком программирования;

- лаконичен;

Минусы языка программирования С:

- устаревание;

Пример простейшей программы (вывод сообщения “Hello World”) на языке программирования C.



Язык программирования С++.

С++ - компилируемый, статически типизированный язык программирования общего назначения.

Поддерживает такие парадигмы программирования, как процедурное программирование, объектно-ориентированное программирование, обобщённое программирование. Язык имеет богатую стандартную библиотеку, которая включает в себя распространённые контейнеры и алгоритмы, ввод-вывод, регулярные выражения, поддержку многопоточности и другие возможности. C++ сочетает свойства как высокоуровневых, так и низкоуровневых языков. В сравнении с его предшественником — языком C — наибольшее внимание уделено поддержке объектно-ориентированного и обобщённого программирования.

C++ широко используется для разработки программного обеспечения, являясь одним из самых популярных языков программирования. Область его применения включает создание операционных систем, разнообразных прикладных программ, драйверов устройств, приложений для встраиваемых систем, высокопроизводительных серверов, а также игр. Существует множество реализаций языка C++, как бесплатных, так и коммерческих и для различных платформ. Например, на платформе x86 это GCC, Visual C++, Intel C++ Compiler, Embarcadero (Borland) C++ Builder и другие. C++ оказал огромное влияние на другие языки программирования, в первую очередь на Java и C#.

Синтаксис C++ унаследован от языка C. Изначально одним из принципов разработки было сохранение совместимости с C. Тем не менее C++ не является в строгом смысле надмножеством C; множество программ, которые могут одинаково успешно транслироваться как компиляторами C, так и компиляторами C++, довольно велико, но не включает все возможные программы на C.

Плюсы языка программирования C++:

- огромная библиотека;

- возможность транслирования как компиляторами С, так и С++;

- безопасность;

- возможность писать обобщённый код с помощью шаблонов;

- возможность использовать объектно-ориентированный подход;

- управление ресурсами;

- упрощение код за счет перегрузки функций и операторов;

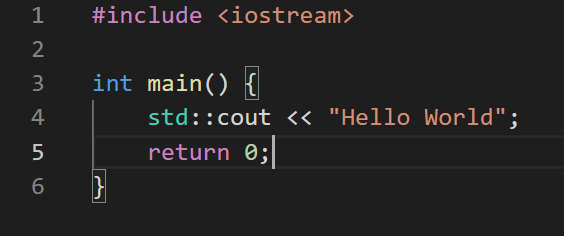
- более простая обработка ошибок за счет исключений;

Минусы языка программирования С++:

- строгие требования к написанию кода;

- очень сложно написать кроссплатформенный код;

Пример простейшей программы (вывод сообщения “Hello World”) на языке программирования C++.



Язык программирования C#.

С# - объектно-ориентированный язык программирования. Разработан в 1998 - 2001 годах группой инженеров компании Microsoft под руководством Андерса Хейлсберга и Скотта Вильтаумота как язык разработки приложений для платформы Microsoft .NET Framework и .NET Core. Впоследствии был стандартизирован как ECMA-334 и [ISO](https://ru.wikipedia.org/wiki/ISO)/IEC 23270.

C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java. Язык имеет статическую типизацию, поддерживает полиморфизм, перегрузку операторов (в том числе операторов явного и неявного приведения типа), делегаты, атрибуты, события, переменные, свойства, обобщённые типы и методы, итераторы, анонимные функции с поддержкой замыканий, LINQ, исключения, комментарии в формате XML.

Переняв многое от своих предшественников - языков C++, Delphi, Модула, Smalltalk и, в особенности, Java - С#, опираясь на практику их использования, исключает некоторые модели, зарекомендовавшие себя как проблематичные при разработке программных систем, например, C# в отличие от C++ не поддерживает множественное наследование классов (между тем допускается множественная реализация интерфейсов).

Плюсы языка программирования C#:

- поддержка подавляющего большинства продуктов Microsoft;

- бесплатность ряда инструментов для небольших компаний;

- простота освоения;

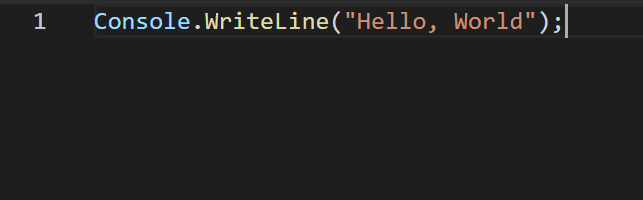
Минусы языка программирования C#:

- небезопасность;

- возможны проблемы с производительностью;

- не распространен повсеместно;

Пример простейшей программы (вывод сообщения “Hello World”) на языке программирования C#.



**Часть 5. Язык программирования Python.**

Python -  высокоуровневый язык программирования общего назначения с динамической строгой типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ. Язык является полностью объектно-ориентированным в том плане, что всё является объектами. Необычной особенностью языка является выделение блоков кода пробельными отступами. Синтаксис ядра языка минималистичен, за счёт чего на практике редко возникает необходимость обращаться к документации. Сам же язык известен как интерпретируемый и используется в том числе для написания скриптов. Недостатками языка являются зачастую более низкая скорость работы и более высокое потребление памяти написанных на нём программ по сравнению с аналогичным кодом, написанным на компилируемых языках, таких как C или C++.

Python является мультипарадигмальным языком программирования, поддерживающим императивное, процедурное, структурное, объектно-ориентированное программирование, метапрограммирование и функциональное программирование. Задачи обобщённого программирования решаются за счёт динамической типизации. Аспектно-ориентированное программирование частично поддерживается через декораторы, более полноценная поддержка обеспечивается дополнительными фреймворками. Такие методики как контрактное и логическое программирование можно реализовать с помощью библиотек или расширений. Основные архитектурные черты - динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений с глобальной блокировкой интерпретатора (GIL), высокоуровневые структуры данных. Поддерживается разбиение программ на модули, которые, в свою очередь, могут объединяться в пакеты.

Для написания веб-браузера я выбрал язык программирования Python, поскольку он является одним из самых широкофункциональных языков программирования, также является самым простым в использовании. Используется для написания огромного количества программ. Python используется для написания автоматизирующих скриптов, например для работы с большим количеством информации (Big Data). Python имеет огромную библиотеку, можно найти решение практически для любой задачи. Также активно используется для web-разработки.

Подводя итог, можно сказать, что я выбрал Python, поскольку этот язык очень прост в освоении, прост в использовании, является самым популярным на сегодняшний день и самым перспективным. Также стоит отметить быструю скорость работы, наличие готовых библиотек на любой случай жизни.

**Часть 6. Библиотека Python flask.**

Flask — фреймворк для создания веб-приложений на языке программирования Python, использующий набор инструментов Werkzeug, а также шаблонизатор Jinja2. Относится к категории так называемых микрофреймворков— минималистичных каркасов веб-приложений, сознательно предоставляющих лишь самые базовые возможности.

Поддерживается установка посредством пакетного менеджера PyPI, версия 1.0 совместима с Python 2.7, Python 3.3 и выше.

Создатель и основной автор — австрийский программист Армин Ронахер, начал работу над проектом в 2010 году.

Пример веб-приложения, которое показывает «Hello World!»

**Часть 7. Библиотека Python docxtpl.**

**Docxtpl** - шаблон слов. Python использует **docxtpl** для обработки изображений текстовых документов, не отображающих проблемы. Используйте Python для создания текстовых документов с указанными данными в пакетах на основе шаблонов (с использованием **docxtpl**). Реагировать на нативный бой - Navigation Redux (mac).

**Часть 8. Word .**

Microsoft Word (часто — MS Word, WinWord или просто Word) — текстовый процессор, предназначенный для создания, просмотра, редактирования и форматирования текстов статей, деловых бумаг, а также иных документов, с локальным применением простейших форм таблично-матричных алгоритмов. Выпускается корпорацией Microsoft в составе пакета Microsoft Office. Первая версия была написана Ричардом Броди (Richard Brodie) для IBM PC, использующих DOS, в 1983 году. Позднее выпускались версии для Apple Macintosh (1984), SCO UNIX и Microsoft Windows (1989). Текущей версией является Microsoft Office Word 2021 для Windows и macOS, а также веб-версия Word Online (Office Online), не требующая установки программы на компьютер.

Microsoft Word многим обязан Bravo — текстовому редактору с оригинальным графическим интерфейсом, разработанному в исследовательском центре «Xerox PARC». Создатель Bravo Чарльз Симони (Charles Simonyi) покинул PARC в 1981 году. Тем же летом Симони переманил Броди, с которым вместе работал над Bravo.

Первый выпуск Word для MS-DOS состоялся в конце 1983 года. Он был плохо принят рынком, продажи снижало наличие конкурирующего продукта — WordPerfect.

Однако версия для «макинтоша», выпущенная в 1985 году, получила широкое распространение. Через два года «Word 3.01 для Macintosh» усилил позиции (версия 3.0 изобиловала ошибками и быстро была заменена). Как и прочее программное обеспечение для «макинтоша», Word был полностью WYSIWYG-редактором (принцип «What You See Is What You Get» — «получаешь то, что видишь»).

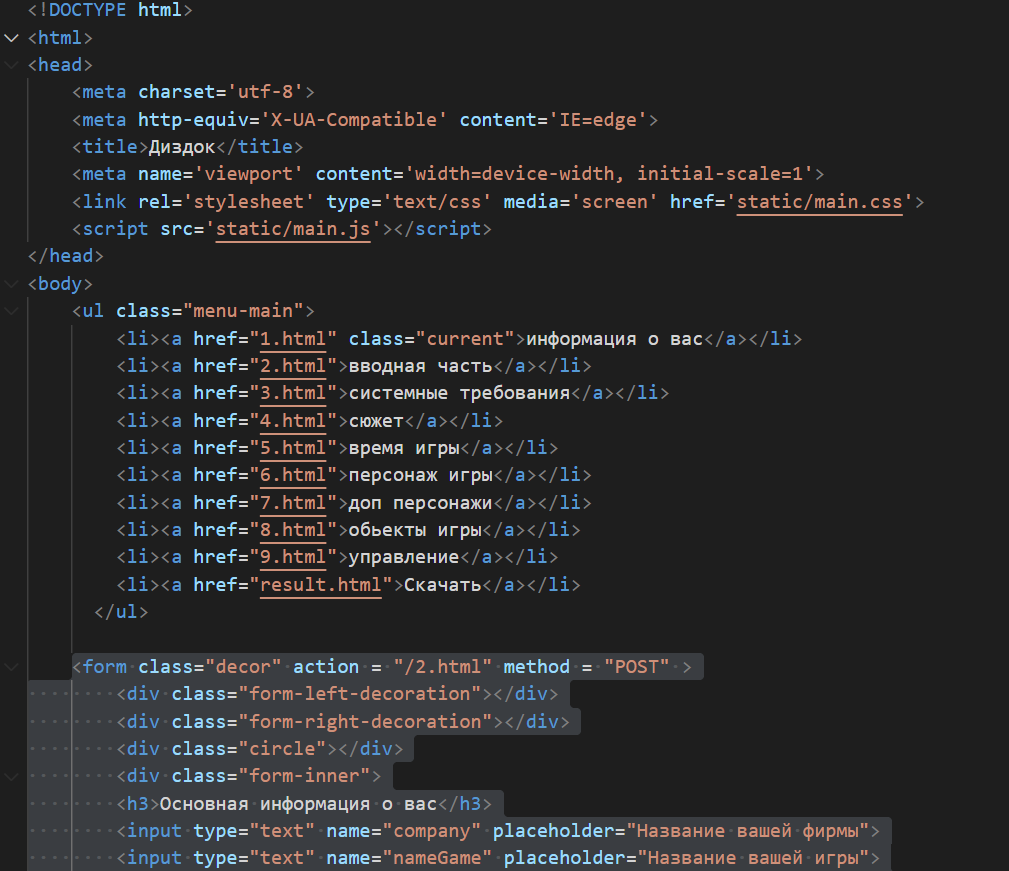
Хотя MS-DOS и являлась текстовой операционной системой, лишённой графической оболочки, Word для DOS был первым текстовым процессором для IBM PC, который был способен отображать разметку текста, например, полужирный или курсивный текст в процессе редактирования. Однако он всё же не являлся в полном смысле WYSIWYG-редактором. Другие же текстовые процессоры, такие как WordStar и WordPerfect, использовали простой текстовый экран с кодами разметки, иногда текст был цветным.

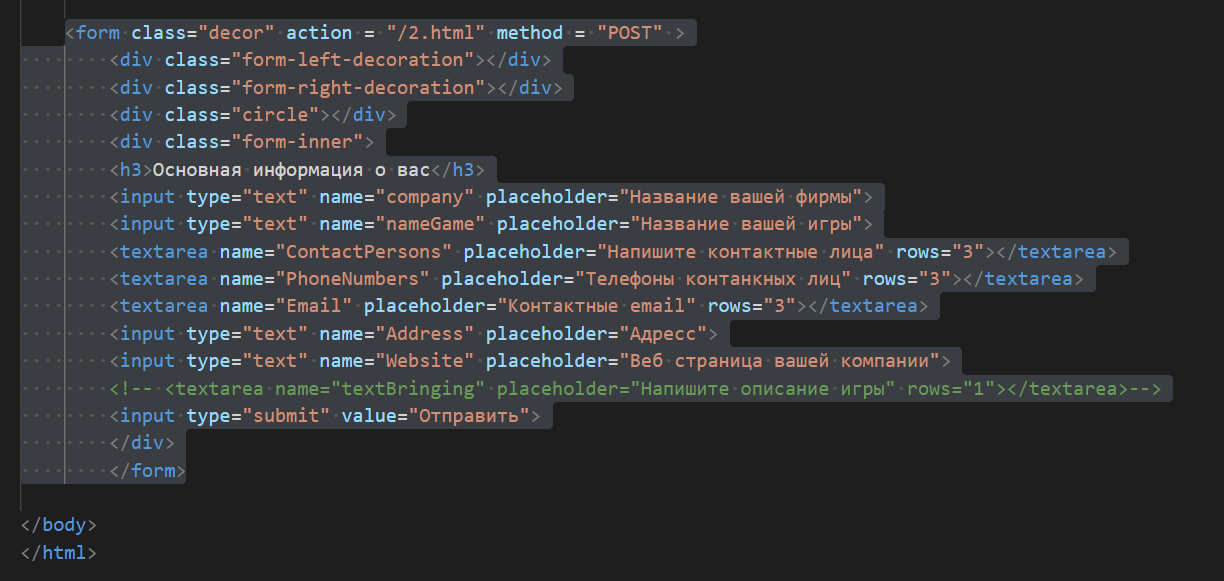
Однако, поскольку в большинстве программного обеспечения под DOS применялись собственные трудно-запоминаемые комбинации «горячих клавиш» для каждой команды (например, в Word’е для DOS сохранение файла выполнялось по комбинации ESC-T-S) и большинство секретарей умели пользоваться только WordPerfect, компании-пользователи WordPerfect весьма неохотно переходили на конкурирующие с ним продукты, имеющие сравнительно небольшие преимущества.

# Глава 2. Разработка

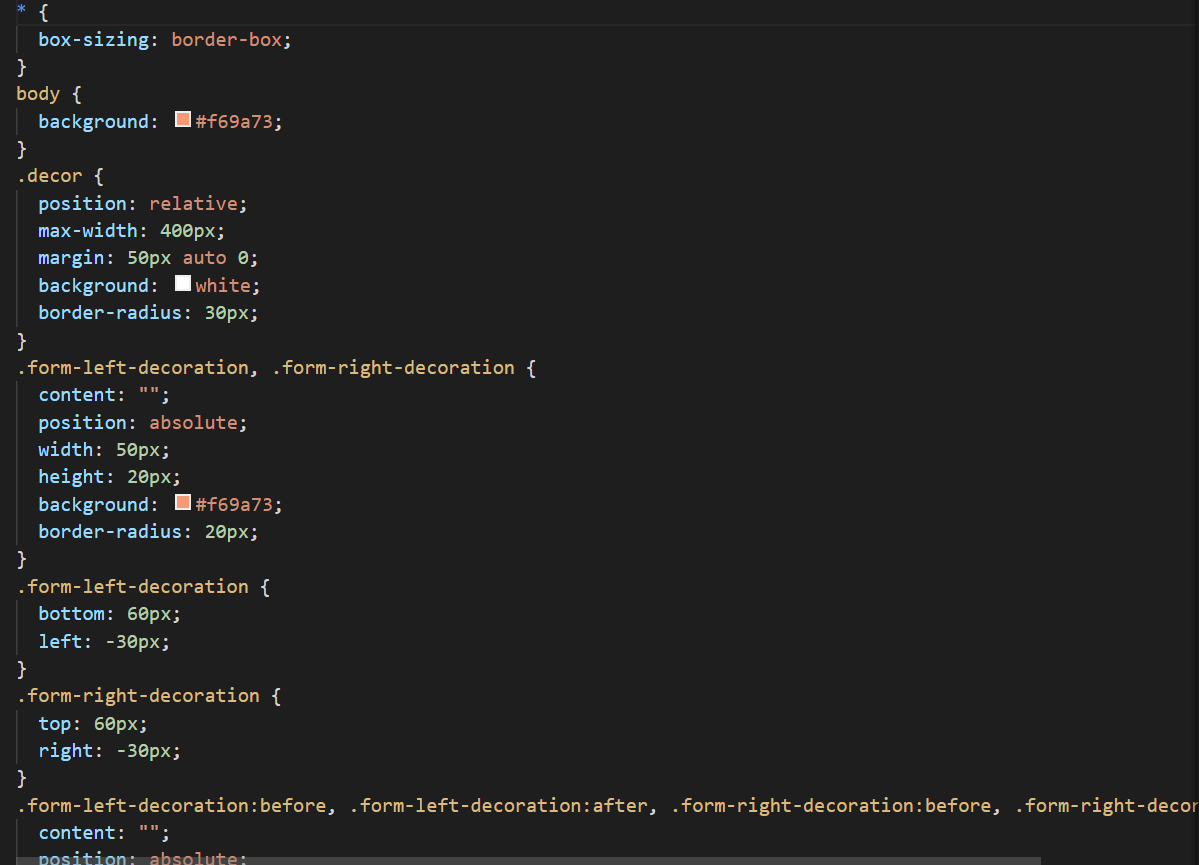
**Часть 1. Верстка Web страниц.**

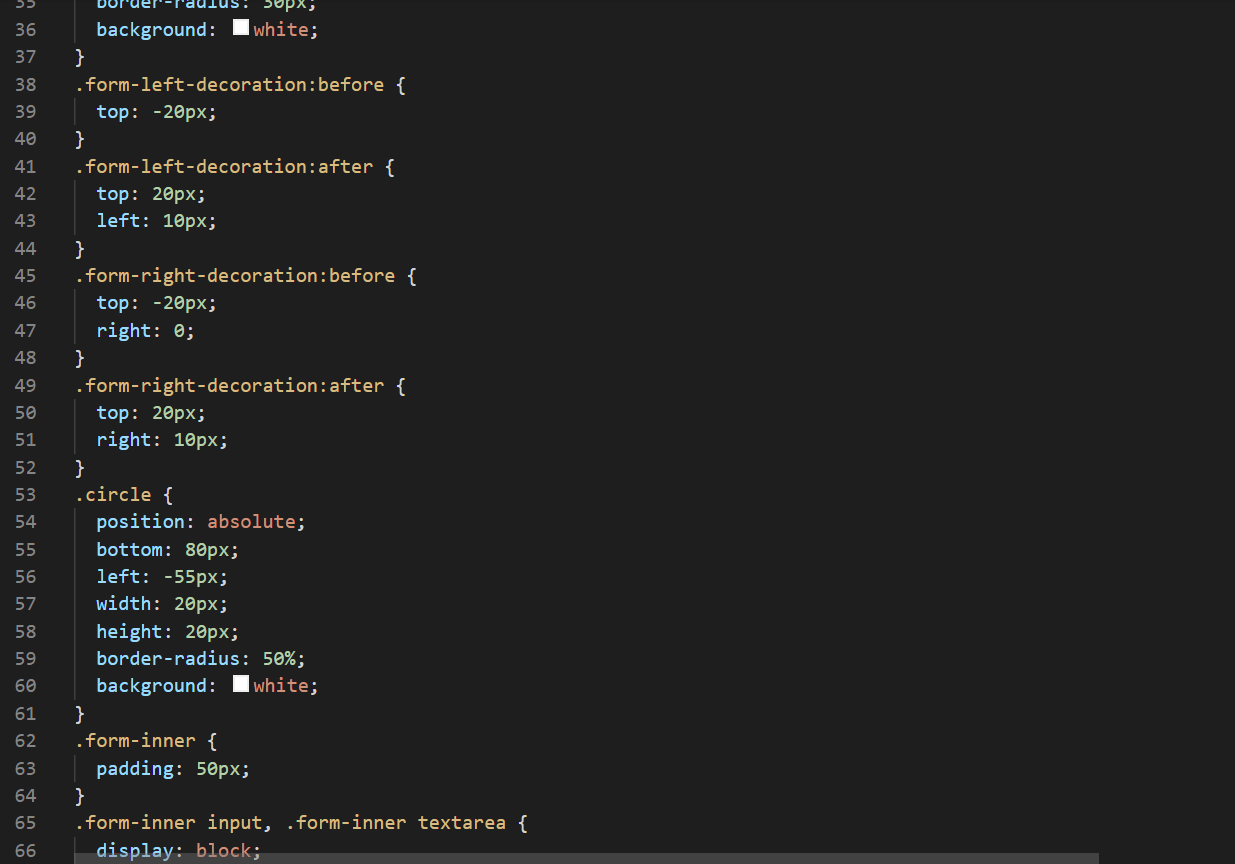
**Html –** предлагаю рассмотреть все страницы на примере одной, дальше они строятся по аналогии. В head я подключаю css стили, пишу название сайта. При помощи ul создаётся меню. И самое интересное это form где пользователь и вбивает данные.

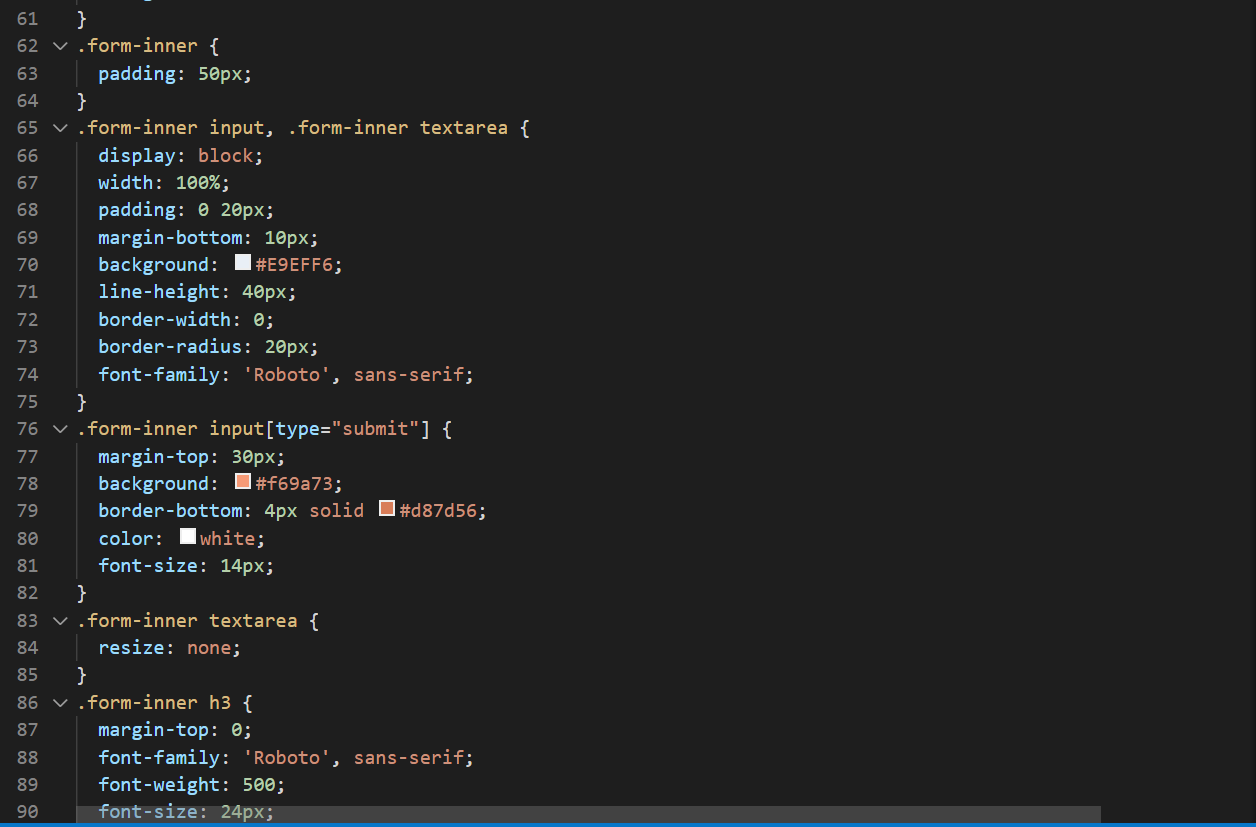
****

****

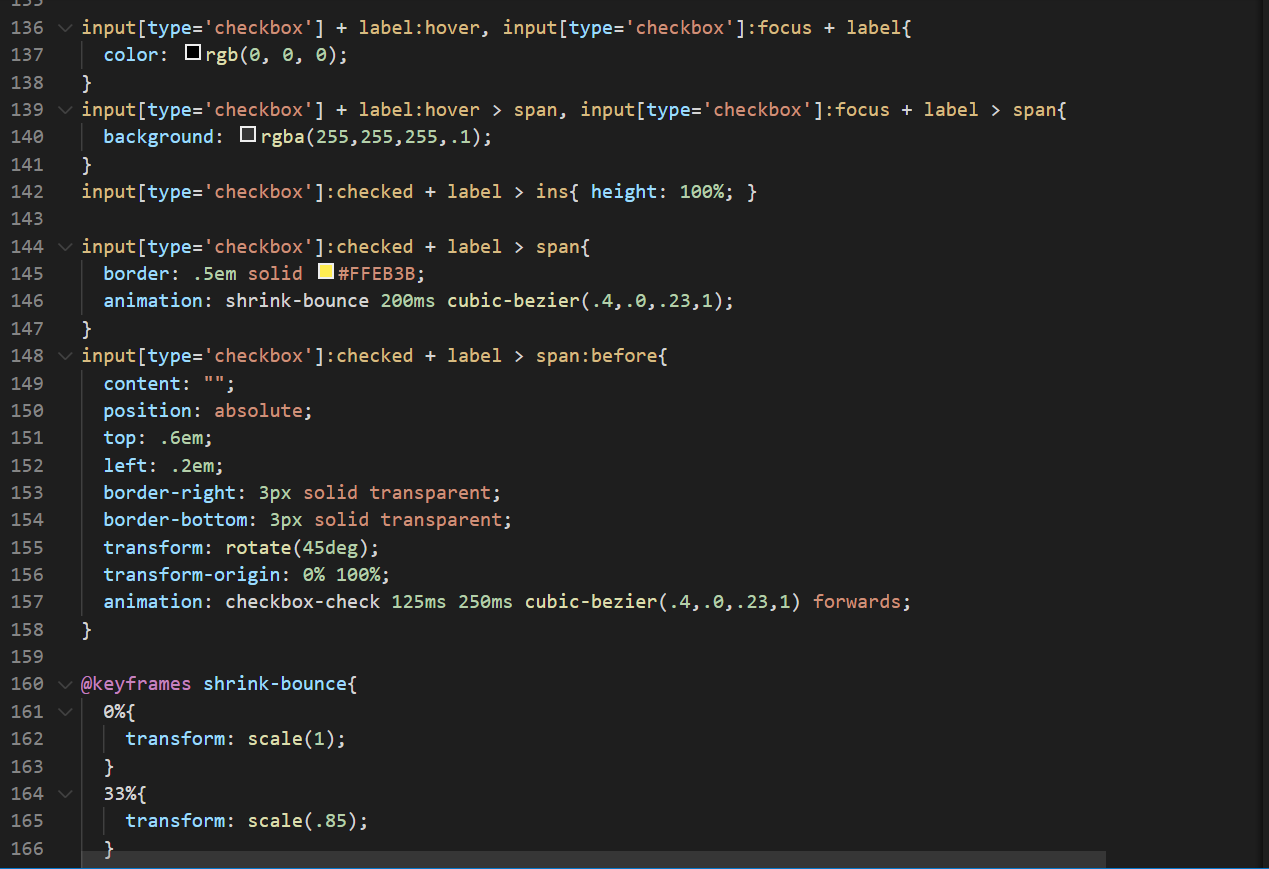
**css**

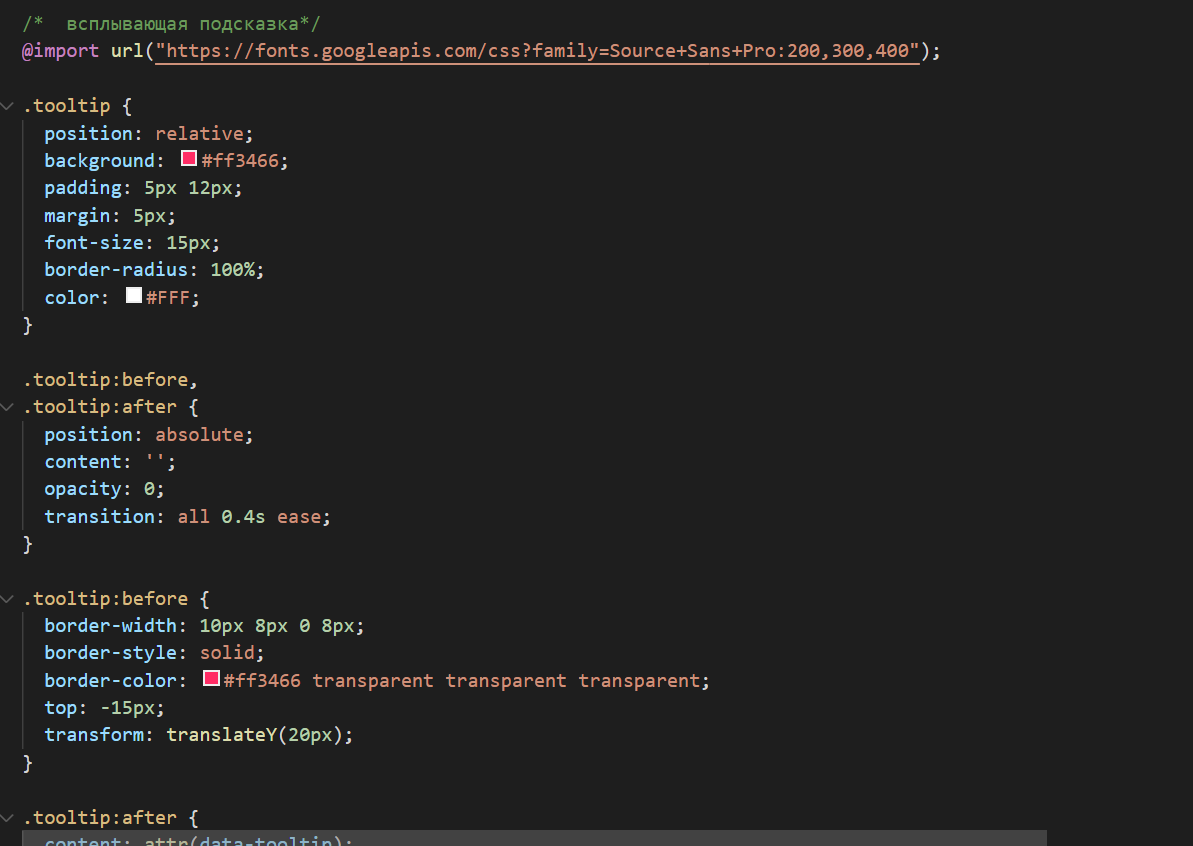
****

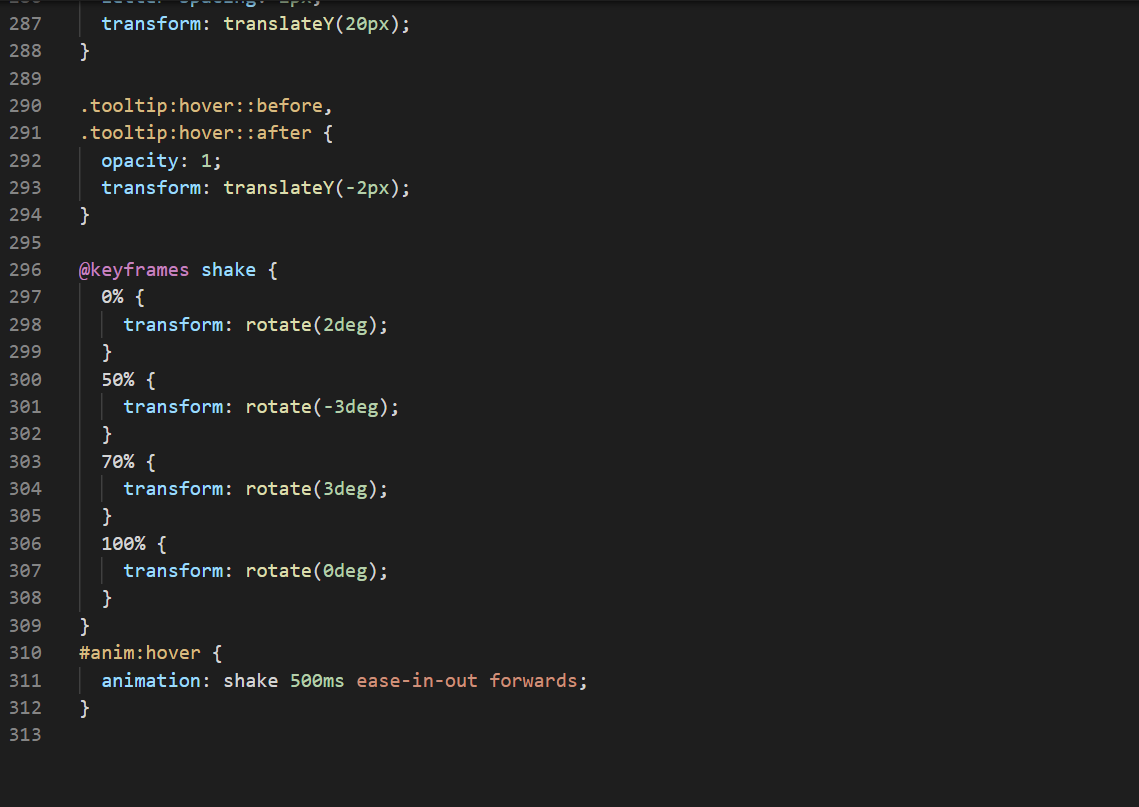
****

****

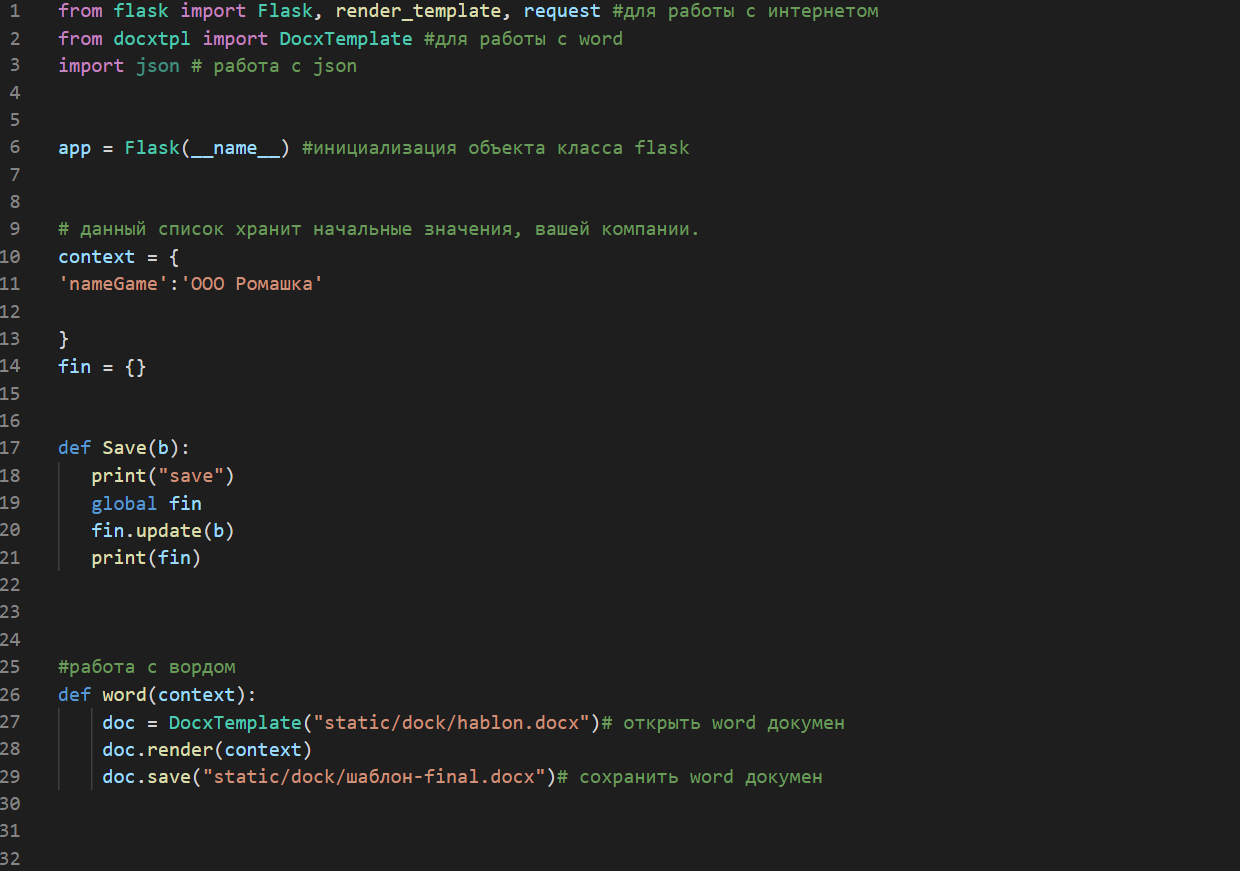
****

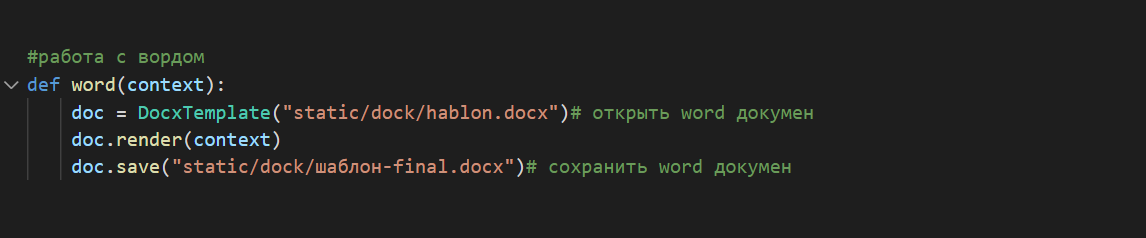
****

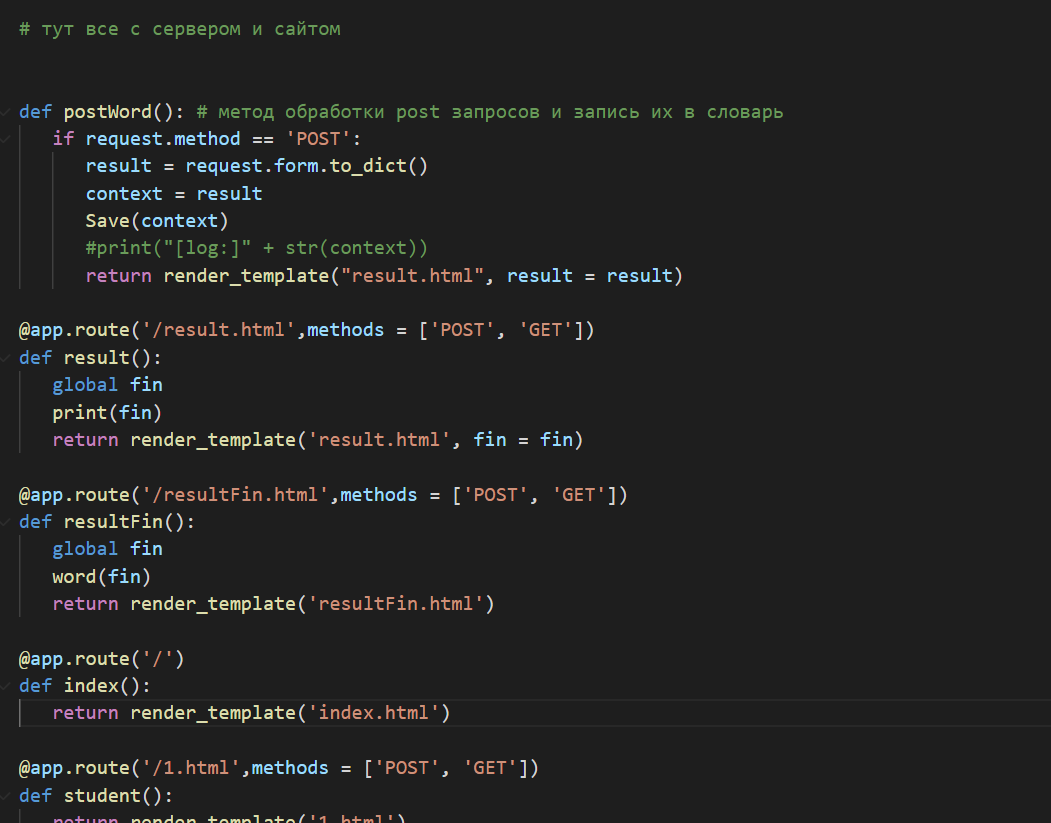
****

****

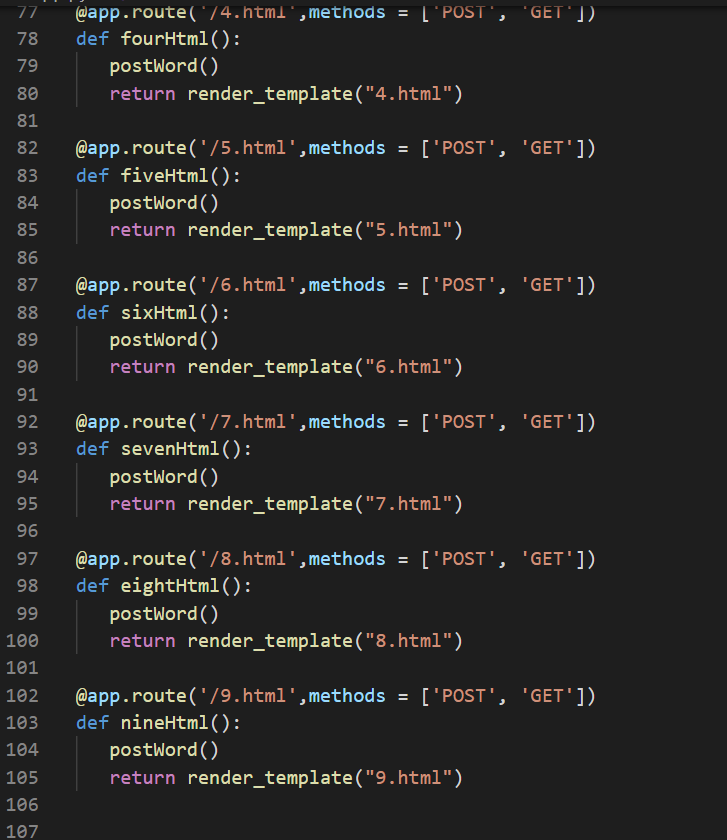
**Часть 2. Разработка серверной части сервиса.**

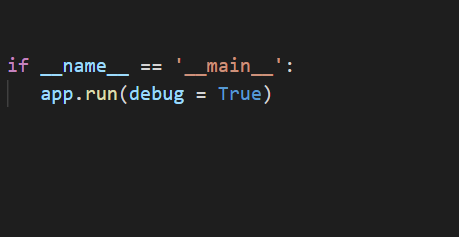
****

****

****

****

****

****

**Часть 3. Тестирование.**

Из за нехватки времени, тестирование не было проведено.

Оно продузомивало провести тестирование при использовании живых людей, для выявление багов.

# Заключение

Цель данной курсовой работы, состоящая в разработке веб-платформы для заполнения DisDoc, была успешно достигнута. Конечно, про возможности использовать данной веб платформ, можно говорить ещё долго, но поставленная цель, как для цели курсовой работы, так и для цели проекта, в котором разработанная веб платформа, успешно проработана и выполнена.

В ходе выполнения, было изучена информация об python, веб приложений, верстке сайтов и соединение всех связующих была достигнута. Также в ходе выполнения, я научился работе с python и его фремворками, с html, а также приблизил к важному для меня выпуску полноценного веб приложения.

# Список используемой литературы

1. Марк Лутц. Изучаем Python.
2. Зед Шоу. Лёгкий способ выучить Python.
3. Берт Бейтс и Кэти Сьерра. Изучаем Java.
4. Бьёрн Страуструп. Язык программирования C ++.
5. Microsoft.com
6. JetBrains.com
7. Code.visualstudio.com
8. Ibm.com
9. Limba.ru
10. studfile.net
11. Gb.ru
12. Wikipedia.org
13. Яндекс Дзен
14. YouTube.com
15. Build-system manual
16. ZetCode.com