

**Индивидуальный предприниматель  
Селендеева Оксана Николаевна**

---

**УТВЕРЖДАЮ**

**Индивидуальный предприниматель**

\_\_\_\_\_/Селендеева О.Н./

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА  
ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА  
«ВЕБ-ПРИЛОЖЕНИЯ НА RYTHON ПРИ ПАРТНЕРСТВЕ С ВМК МГУ  
ИМ. М.В. ЛОМОНОСОВА»**

**Москва, 2024**

## Оглавление

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА .....	3
1.1. Общая характеристика программы .....	3
1.2. Цели и задачи программы .....	4
2. УЧЕБНЫЙ ПЛАН ПРОГРАММЫ .....	7
3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК .....	10
4. РАБОЧАЯ ПРОГРАММА .....	13
5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ .....	18
5.1. Контроль знаний, умений и навыков .....	18
5.2. Критерии оценивания освоения программы при проведении различных форм контроля: .....	19
6. УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ (УЧЕБНИКИ, РАЗДАТОЧНЫЕ МАТЕРИАЛЫ, МЕТОДИЧЕСКИЕ УКАЗАНИЯ, ПЛАКАТЫ, СЛАЙДЫ, ИНТЕРНЕТ- РЕСУРСЫ) .....	20
6.1. Информационные и учебно-методические условия реализации программы .....	20
6.2. Рекомендованная литература для обучающихся: .....	20
6.3. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее- сеть "Интернет"), рекомендованных для освоения программы: .....	21
7. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ..	21
7.1. Материально-техническая и ресурсная база .....	21
7.2. Кадровое обеспечение программы .....	21
Приложения.....	23
Приложение 1. Примерные вопросы для промежуточного тестирования.....	23
Приложение 2. Примерные задания для оценки качества освоения учебного материала .....	25

# 1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

## 1.1. Общая характеристика программы

Данный документ описывает комплекс основных характеристик образования (объем, содержание, планируемые результаты) и организационно-педагогических условий, который представлен в виде учебного плана, календарного учебного графика, рабочих программ учебных предметов, модулей, иных компонентов, а также оценочных и методических материалов общеобразовательной общеразвивающей программы "Веб-приложения на Python при партнерстве с ВМК МГУ им. М.В. Ломоносова".

В ходе дополнительной общеобразовательной общеразвивающей программы "Веб-приложения на Python при партнерстве с ВМК МГУ им. М.В. Ломоносова" слушатели познакомятся с основами профессии Python-разработчика, освоят язык программирования Python, изучат основы объектно-ориентированного программирования.

Выдача обучающимся документов о дополнительном образовании (сертификат установленного образца) осуществляется при условии успешного прохождения итоговой аттестации.

Программа разработана на основе следующих **нормативных документов**:

- Федеральный закон от 29 декабря 2012 г. №273-ФЗ «Об образовании в Российской Федерации»;
- Приказ Министерства просвещения Российской Федерации от 27 июля 2022 г. №629 "Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам"
- Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи" (вместе с "СП 2.4.3648-20. Санитарные правила...")» (Зарегистрировано в Минюсте России 18.12.2020 № 61573).

**Направленность (профиль) программы:** техническая

**Актуальность программы, соответствие государственному, социальному заказу/запросам.**

Курс по изучению веб-разработки для подростков 13-16 лет обладает актуальностью как с точки зрения подготовки к цифровому будущему, так и с социальной и образовательной перспектив. В современном мире, который становится все более цифровым, веб-разработка и программирование становятся важными навыками не только для специалистов в области информационных технологий, но и для всех, кто стремится успешно адаптироваться к быстро меняющейся диджитализированной среде. Такой курс поможет подросткам развить компьютерные навыки, которые будут востребованы на рынке труда в будущем, а также овладеть пониманием принципов работы интернета и веб-приложений, что является ключевым для безопасного и эффективного использования онлайн-ресурсов. В рамках стратегии развития образования и цифровой экономики, многие страны интегрируют информационные технологии в образовательный процесс, и курс по веб-разработке отвечает

этому запросу, предоставляя подросткам необходимые навыки для успешной адаптации в цифровой эпохе. Таким образом, программа по изучению веб-разработки для подростков 13-16 лет соответствует как государственному, так и социальному заказу, обеспечивая молодежь необходимыми навыками для успешного развития в современном обществе.

**Отличительные особенности программы** заключаются в ее простоте и доступности, практической направленности и интерактивных проектах. Программа предоставляет возможность создавать собственные веб-приложения, сайты или игры, что способствует непосредственному применению полученных знаний. Индивидуализация обучения позволяет учитывать разнообразный уровень подготовки и интересы учащихся, а система поддержки и обратной связи обеспечивает помощь и консультации от опытных преподавателей. Программа также регулярно обновляется и дополняется новыми материалами, что позволяет подросткам быть в курсе последних тенденций веб-разработки и готовыми к будущим вызовам.

**Срок обучения:** программа реализуется в объеме 56 академических часов, 28 недель (7 месяцев).

**Режим занятий:** 3-4 академических часа в неделю

Продолжительность академического часа – 45 минут.

Занятия начинаются не ранее 9.00 часов утра и заканчиваются не позднее 20.00 часов. Для обучающихся в возрасте 16-18 лет допускается окончание занятий в 21.00 часов.

Продолжительность занятий в учебные дни - не более 3-х академических часов в день, в выходные и каникулярные дни - не более 4 академических часов в день. После 30-45 минут теоретических занятий организуется перерыв длительностью не менее 10 мин.

**Адресат программы и примерный портрет слушателя курсов:** программа разработана для учащихся от 13 до 16 лет, которых интересует программирование на языке Python. К освоению дополнительной общеобразовательной программы – дополнительной общеразвивающей программы допускаются: лица без предъявления требований к уровню образования.

**По завершении реализации программы,** как правило, проводится анкетирование обучающихся с целью изучения мнения по вопросу эффективности и информативности проведенного обучения, уровню организации учебного процесса, удовлетворенности учебно-методическим материалом, работниками образовательной организации проводится анализ высказанных предложений и пожеланий.

## 1.2. Цели и задачи программы

Цель программы: Подготовка подростков в возрасте от 13 до 16 лет к основам веб-разработки, обеспечивая им необходимые навыки для создания собственных веб-приложений и сайтов.

Задачи программы:

1. Освоить основные принципы технологий HTML/ CSS / JavaScript для создания веб-страниц.
2. Изучить стили и форматирование с помощью CSS.
3. Изучить основы серверной разработки с использованием Flask.
4. Овладеть основами работы с базами данных для хранения информации.
5. Понять принципы безопасности веб-приложений и сайтов.

### 1.3. Планируемые результаты обучения

По итогам освоения дополнительной общеобразовательной программы - дополнительной общеразвивающей программы "Веб-приложения на Python при партнерстве с ВМК МГУ им. М.В. Ломоносова" обучающиеся должны будут овладеть следующими знаниями, умениями и навыками:

#### Знания

- Основы языка программирования Python.
- Принципы работы веб-приложений и их архитектура.
- Работа с веб-фреймворком Flask для создания веб-приложений.
- Основы работы с базами данных и их интеграция в веб-приложения.
- Принципы безопасности веб-приложений и защита от распространенных атак.
- Использование HTML для структурирования содержимого веб-страниц.
- Изучение CSS для стилизации и визуализации веб-приложений.
- Понимание принципов клиент-серверной архитектуры и взаимодействия между клиентом и сервером.
- Основы адаптивной верстки для создания отзывчивого дизайна веб-приложений.
- Знание основных HTTP-методов и протокола.

#### Умения:

- Написание серверной части веб-приложения на Python с использованием Flask.
- Работа с базами данных на уровне CRUD операций (создание, чтение, обновление, удаление).
- Разработка пользовательских интерфейсов с использованием HTML и CSS.
- Обработка пользовательских запросов и форм веб-приложений.
- Развертывание и настройка веб-приложений на сервере.
- Отладка и тестирование веб-приложений для обнаружения и исправления ошибок.
- Использование системы контроля версий Git для управления кодовой базой проекта.
- Создание адаптивных и мобильно-оптимизированных веб-приложений.
- Применение базовых принципов защиты данных и обеспечения безопасности веб-приложений.
- Взаимодействие с API для интеграции сторонних сервисов в веб-приложения.

#### Навыки:

- Проектирование и разработка полнофункциональных веб-приложений на Python.
- Работа в команде над веб-проектами, распределение задач и совместная разработка.
- Управление и поддержка веб-приложений в процессе их эксплуатации.
- Анализ требований пользователей и преобразование их в функциональные возможности веб-приложений.
- Умение эффективно решать проблемы и искать нестандартные подходы к разработке веб-приложений.
- Креативность и умение придумывать новые идеи для улучшения функциональности и пользовательского опыта веб-приложений.

- Навыки оценки и анализа производительности веб-приложений и оптимизации их работы.
- Коммуникационные навыки для эффективного общения с командой, заказчиками и пользователями веб-приложений.
- Способность к самостоятельному изучению новых технологий и инструментов в сфере веб-разработки.
- Готовность к постоянному обучению и развитию в области веб-приложений и программирования.

## 2. УЧЕБНЫЙ ПЛАН ПРОГРАММЫ

В процессе преподавания курса «Веб-приложения на Python при партнерстве с ВМК МГУ им. М.В. Ломоносова» используются как классические методы обучения (лекции), так и различные виды практической работы обучающихся по заданию преподавателя, которые направлены на развитие навыков разработки программного обеспечения, креативных качеств и на поощрение интеллектуальных инициатив учащихся.

**Формы организации образовательного процесса** (индивидуальные, групповые и т.д.) и другие виды занятий по программе определяются содержанием программы. Образовательная деятельность обучающихся предусматривает следующие виды учебных занятий и учебных работ: лекции, практические и семинарские занятия, круглые столы, мастер-классы, тренинги, семинары по обмену опытом, проведение открытых занятий, консультации и другие виды учебных занятий и методической работы, определенные учебным планом.

№	Наименование дисциплины/раздела/темы	Количество академических часов				Форма аттестации /контроля
		Всего	в т.ч. аудиторных		СРС	
			теория	практич. занятия		
1	Повторение Python	2	1	1	0	Практическое задание
2	Готовимся стать web-программистами	2	0	2	0	Практическое задание
3	Объектно - ориентированное программирование	2	1	1	0	Практическое задание
4	Новый взгляд на ООП	2	0	1	1	Практическое задание, Контрольные вопросы
5	Операционная Система	2	2	0	0	Практическое задание
6	Мир web	2	0	2	0	Практическое задание
7	Настройка Flask	2	1	1	0	Практическое задание
8	Язык Web	2	0	1	1	Практическое задание, Контрольные вопросы
9	Шаблоны	2	1	1	0	Практическое задание

10	Формы Flask	2	1	1	0	Практическое задание
11	Базы данных	2	0	1	1	Практическое задание
12	Базы данных. Полное погружение	2	2	0	0	Практическое задание, Контрольные вопросы
13	Диалог с пользователем	2	0	2	0	Практическое задание
14	Неотъемлемая часть программирования - тесты	2	1	1	0	Практическое задание
15	Модульность проектов	2	0	1	1	Практическое задание
16	Командообразование	2	1	1	0	Практическое задание, Контрольные вопросы
17	Командная разработка. Практика	2	0	1	1	Практическое задание
18	Подробный HTML	2	2	0	0	Практическое задание
19	CSS. Глубокое погружение	2	0	2	0	Практическое задание
20	Адаптивная верстка	2	1	1	0	Практическое задание, Контрольные вопросы
21	Расширяем возможности Flask	2	1	1	0	Практическое задание
22	Сервер	2	0	1	1	Практическое задание
23	Серверные приложения	2	2	0	0	Практическое задание
24	Ораторское искусство	2	0	2	0	Практическое задание, Контрольные вопросы
25	Командная разработка	2	1	1	0	Практическое задание
26	Командная разработка	2	0	2	0	Практическое задание
27	На пути к MVP	2	0	1	1	Практическое



						задание
28	Подведение итогов и презентация проектов курса	<b>2</b>	<b>0</b>	<b>2</b>	<b>0</b>	Практическое задание. Контрольные вопросы. Защита проекта.
	<b>ИТОГО</b>	<b>56</b>	<b>18</b>	<b>31</b>	<b>7</b>	

### 3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Календарный учебный график формируется при осуществлении обучения в течение всего календарного года. По мере набора групп слушателей по программе составляется календарный график, учитывающий объемы лекций, практики, самоподготовки, другие формы организации занятий.

Темы / недели	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	Итого часов
Повторение Python	2																												2
Готовимся стать web-программистами		2																											2
Объектно - ориентированное программирование			2																										2
Новый взгляд на ООП				1,5																									1,5
Промежуточный контроль				0,5																									0,5
Операционная Система					2																								2
Мир web						2																							2
Настройка Flask							2																						2
Язык Web								1,5																					1,5





## 4. РАБОЧАЯ ПРОГРАММА

### 1. Повторение Python

- Типы и структуры данных
- Что в памяти нашего компьютера?
- Повтор конструкции ветвления и циклов

**Практическое задание:** создание программы для работы с различными типами данных и их вывода на экран.

### 2. Готовимся стать web-программистами

- Новые алгоритмы работы с данными
- Разбор нескольких реальных кейсов из мира web
- Проектирование

**Практическое задание:** разработка простой веб-страницы с использованием HTML и CSS.

### 3. Объектно - ориентированное программирование

- Паттерны программирования
- ООП
- Сравнение ООП с другими принципами на тестах

**Практическое задание:** Разработка класса, моделирующего объект из реального мира.

### 4. Новый взгляд на ООП

- Погружение в ООП
- Конструкторы и деструкторы классов
- Лексикон программиста

**Практическое задание:** Создание класса для работы с банковским счетом.

### 5. Операционная Система

- Узнаем, как работает ОС
- Поработаем с командной строкой
- Заглянем в глубины нашего компьютера

**Практическое задание:** Использование командной строки для управления файлами и директориями.

### 6. Мир web

- Интернет и классификация контента
- Как работает браузер
- Инфраструктура web-приложения

**Практическое задание:** Исследование структуры HTTP-запросов и ответов с использованием инструментов разработчика браузера.

### 7. Настройка Flask

- Что такое фреймворки?
- Настраиваем окружение
- Первые команды и начало работы с Flask

**Практическое задание:** Установка и настройка Flask, создание простого веб-приложения и его запуск на локальном сервере.

## 8. Язык Web

- Первое приложение Flask
- Основы Web – знакомимся с HTML
- Структура сайтов

**Практическое задание:** Разработка простой веб-страницы с использованием Flask и HTML.

## 9. Шаблоны

- Еще пару слов о проектировании
- Шаблоны Flask – зачем они нужны
- Собственные шаблоны

**Практическое задание:** Разработка нескольких шаблонов HTML для различных страниц веб-приложения.

## 10. Формы Flask

- Что такое формы?
- Классификация и работа с формами
- Получение данных и проверка полей

**Практическое задание:** Разработка веб-формы для ввода данных пользователем и их обработка в Flask-приложении.

## 11. Базы данных

- Куда и как сохранить данные
- Что такое Базы данных
- Изучим модели Баз данных

**Практическое задание:** Создание базы данных для хранения информации о пользователях веб-приложения и реализация CRUD операций.

## 12. Базы данных. Полное погружение

- Подводные камни
- Что такое связи?
- Обновление баз Данных

**Практическое задание:** Реализация сложных запросов к базе данных и обновление схемы базы данных.

## 13. Диалог с пользователем

- Смотрим на формы глубже
- Авторизация
- Условный рендер страниц

**Практическое задание:** Разработка веб-приложения с реализацией системы авторизации, форм для ввода данных и условным рендерингом страниц.

## 14. Неотъемлемая часть программирования - тесты

- Узнаем, какими бывают тесты
- Найдем логические ошибки
- Поймем, как тесно связаны ошибки и безопасность

**Практическое задание:** Написание тестов для веб-приложения, проверка их корректности и безопасности.

## **15. Модульность проектов**

- Понятие модульности
- Разбивка на логические части
- Модули – первый шаг к работе в команде

**Практическое задание:** Разработка модульной структуры веб-приложения с разделением на логические компоненты.

## **16. Командообразование**

- Распределение ролей в командах для разработки

**Практическое задание:** Формирование команды для работы над проектом.

## **17. Командная разработка. Практика**

- Система контроля версий
- Разбивка проекта
- Роль в команде разработчиков

**Практическое задание:** Создание репозитория проекта на платформе для управления версиями и настройка рабочего процесса с использованием Git.

## **18. Подробный HTML**

- Табличная/Блочная верстка
- Современные виды верстки и Perfect Pixel
- Возможности HTML 5

**Практическое задание:** Создание веб-страниц с использованием различных методов верстки и новых возможностей HTML 5.

## **19. CSS. Глубокое погружение**

- Приоритетность web-страниц
- CSS 3
- @Media запросы

**Практическое задание:** Разработка стилей для веб-приложения с использованием CSS 3 и адаптивной верстки.

## **20. Адаптивная верстка**

- Понимание ключевых принципов адаптивности
- Библиотека Bootstrap
- Визуализация форм

**Практическое задание:** Создание адаптивного дизайна для веб-приложения с использованием Bootstrap и CSS.

## **21. Расширяем возможности Flask**

- Библиотека сторонних компонентов
- Подключаем компоненты в наш проект
- Настраиваем работу с основным приложением

**Практическое задание:** Расширение функционала веб-приложения с использованием сторонних компонентов для Flask.

## 22. Сервер

- Что такое сервер? Взгляд изнутри
- Безопасность серверов
- Поднимем свой собственный сервер

**Практическое задание:** Настройка и обеспечение безопасности собственного сервера.

## 23. Серверные приложения

- Развернем приложение на сервере
- Тестируем доступы
- Учимся ловить запросы

**Практическое задание:** Развертывание веб-приложения на удаленном сервере и проверка его работоспособности.

## 24. Ораторское искусство

- Основы выступления на публике
- Правила работы с презентацией

**Практическое задание:** Подготовка и проведение ораторского выступления перед аудиторией.

## 25. Командная разработка

- Оценка идеи
- Определение ключевых задач
- Старт разработки

**Практическое задание:** Оценка идеи предполагаемого проекта, определение ключевых задач и начало разработки первых модулей.

## 26. Командная разработка. Продолжение

- Итоги 1-й недели
- Расстановка приоритетов
- Мотивация и упрощение процессов разработки

**Практическое задание:** Обсуждение и анализ проделанной работы за первую неделю, установка приоритетов и мотивационные мероприятия.

## 27. На пути к MVP

- Тест реализации на фокус-группе
- Корректировка курса разработки
- Ускоряем разработку грамотным тайм-менеджментом

**Практическое задание:** Разработка MVP и проведение тестирования на фокус-группе, корректировка плана разработки.



## **28. Подведение итогов и презентация проектов курса**

- Подготовка к презентации;
- Презентация проектов;
- Дальнейшее развитие и обучение в сфере Python-разработки.
- Подведение итогов курса.

**Практическое задание:** подготовка выступления и презентация своего проекта.

## 5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

### 5.1. Контроль знаний, умений и навыков

Формой подведения итогов реализации дополнительной образовательной программы "Веб-приложения на Python при партнерстве с ВМК МГУ им. М.В. Ломоносова" выступает текущая, промежуточная и итоговая аттестация.

Образовательный процесс осуществляется на основании учебного плана и регламентируется расписанием занятий для каждой учебной группы.

В целях оценки показателей знаний, умений и навыков обучающихся по дополнительной образовательной общеразвивающей программе "Создание чат-ботов на Python для ВК и Telegram" проводится текущий и промежуточный контроль знаний, а также итоговая аттестация.

#### Виды текущего контроля:

- устный ответ на поставленный вопрос;
- проверка результатов выполнения практических заданий

#### Виды промежуточного контроля:

- тестирование устное/письменное/с помощью электронных форм
- проверка результатов выполнения практических работ/проектов по итогам учебного модуля.

Тестирование - это форма измерения знаний обучающихся, основанная на применении тестов. Материалы для промежуточного и итогового тестирования предоставляются вместе с комплектом учебно-методических материалов к программе.

#### Итоговая аттестация

Итоговая аттестация проводится с целью установления уровня знаний обучающихся с учетом прогнозируемых результатов обучения и требований к результатам освоения образовательной программы.

Итоговая аттестация обучающихся осуществляется в форме презентации созданных в курсе проектов – ботов различной тематики, которая сопровождается демонстрацией проекта и презентацией. Презентация – это электронный документ, предназначенный для визуальной демонстрации выполненной работы. Как правило, презентация имеет сюжет, сценарий и структуру, созданную для удобного восприятия информации.

Выдача обучающимся документов о дополнительном образовании (сертификат о прохождении курса) осуществляется при условии успешного прохождения итоговой аттестации.

## 5.2. Критерии оценивания освоения программы при проведении различных форм контроля:

**Тестирование** (Приложение 1. Примерные вопросы для промежуточного тестирования).  
Процент результативности (правильных ответов при выполнении тестовых заданий):

<b>Выполнение теста</b>	<b>Итоговая оценка</b>
70% и более правильных ответов	"Зачтено"
Менее 70% правильных ответов	"Не зачтено"

**Проверка выполнения практических работ** (Приложение 2. Примерные задания для проверки усвоения качества учебного материала). Система оценивания:

"Зачтено" – необходимый уровень выполнения задания достигнут, обучающийся демонстрирует хорошее знание теоретической и практической части материала занятия/учебного модуля, достигнуты промежуточные и/или итоговые результаты работы над заданием.

"Не зачтено" - необходимый результат/уровень освоения не достигнут, обучающийся не усвоил теоретические основы и/или изученные практические приемы и методы разработки, инструменты создания чат-ботов, не достиг промежуточных и итоговых результатов при выполнении задания.

**Проверка результатов демонстрации проектов на итоговой аттестации:**

<b>Критерии оценки презентации проектов</b>	<b>БАЛЛЫ</b>
Обоснование выбора темы и ее актуальность	0-2 балла
Удобство использования, функциональность веб-приложений	0-3 балла
Применение изученных программных средств, техническая сложность веб-приложений	0-3 балла
Креативность идеи	0-3 балла
Самостоятельность работы над проектом	0-3 балла
Компетентность докладчика (ответы на вопросы)	0-2 балла
<b>Итоговая оценка:</b> «Не зачтено» «Зачтено»	<b>0-10 баллов</b> <b>11-16 баллов</b>

## **6. УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ (УЧЕБНИКИ, РАЗДАТОЧНЫЕ МАТЕРИАЛЫ, МЕТОДИЧЕСКИЕ УКАЗАНИЯ, ПЛАКАТЫ, СЛАЙДЫ, ИНТЕРНЕТ-РЕСУРСЫ)**

### **6.1. Информационные и учебно-методические условия реализации программы**

#### **Учебно-методический комплект**

Для реализации целей и задач обучения по общеобразовательной общеразвивающей программе "Веб-приложения на Python при партнерстве с ВМК МГУ им. М.В. Ломоносова" используется комплект материалов преподавателя, который включает:

1. Текстовое методическое пособие с описанием целей, результатов каждого занятия, теоретического материала и практических работ.
2. Видеоурок для преподавателя с методическими указаниями и порядком объяснения учебного материала.
3. Раздаточный материал для учащихся - описание дополнительной самостоятельной работы учащихся по каждому занятию с примерами и рекомендациями по выполнению.
4. Описание мероприятий по контролю знаний – тестовые вопросы, практические задания.
5. Рекомендации по проведению итоговой аттестации и защиты проектов.
6. Дополнительные материалы – примеры кодов реализованных ботов, скрипты.
7. Дополнительные материалы – инструкции по установке необходимого программного обеспечения, описание технических требований к компьютерному оборудованию.

Материалы преподавателя размещаются на учебном портале преподавателей, размещенном на сервере информационно-телекоммуникационной сети "Интернет", и доступны по ссылке для всех преподавателей курса. Материалы обучающихся раздаются в печатном виде или рассылаются преподавателем индивидуально каждому обучающемуся.

### **6.2. Рекомендованная литература для обучающихся:**

Основная:

1. Широков А.И., Пышняк М. Информатика. Разработка программ на языке программирования Питон. Часть 1. Базовые языковые конструкции / А. И. Широков, Марина Пышняк – Москва: ИД МИСиС, 2020. – 144 с.
2. Бриггс Джейсон. Python для детей. Самоучитель по программированию / Д.Бриггс. - Москва: Манн Иванов и Фербер (МИФ), 2016. – 289 с.
3. Грацианова Т.Ю. Информатика. Программирование в примерах и задачах / Т.Ю. Грацианова, Москва: ВМК МГУ (Лаборатория знаний), 2020. – 393 с.

Дополнительная:

4. Воронин, И. Программирование для детей. От основ к созданию роботов / И. Воронин, В. Воронина. - СПб.: Питер, 2019. - 304 с.
4. Олифер, В. Компьютерные сети. Принципы, технологии, протоколы / В. Олифер, Н. Олифер. – Санкт-Петербург: Питер, 2021. – 1005 с.
5. Брайсон Пейн. Программирование на Python для детей и родителей / Б. Пейн. – Москва: Эксмо, 2017 – 354 с.

### **6.3. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее - сеть "Интернет"), рекомендованных для освоения программы:**

- <https://pythonru.com/uroki/vvedenie-uroki-po-python-dlja-nachinajushhih> - PythonRu — Уроки по Python для начинающих
- <https://pythontutor.ru/> - проект Питонтьютор
- [https://ru.hexlet.io/courses/python\\_101](https://ru.hexlet.io/courses/python_101) – Хекслет. Курс «Введение в Python»
- <https://www.udemy.com/course/python-project/> - Python. Практика для начинающих
- [http://window.edu.ru/catalog/resources?p\\_rubr=2.1.6](http://window.edu.ru/catalog/resources?p_rubr=2.1.6) – единое окно доступа к образовательным ресурсам, раздел "Информатика и ОКТ"
- <https://www.python.org/downloads/windows/> - официальный сайт Python

## **7. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ**

### **7.1. Материально-техническая и ресурсная база**

Для реализации программы предполагается использование учебных аудитории для проведения занятий лекционного типа, занятий семинарского типа, выполнения проектных работ, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации:

- Компьютерные классы, обеспечивающие доступ в Интернет, и оснащенные мультимедиа проектором или иными средствами визуализации учебного материала, магнитной доской или флипчартом.
- Электронный информационно-образовательный портал, размещенный на сервере в информационно-телекоммуникационной сети "Интернет".
- Специальное программное обеспечение для Python-разработки, необходимое для реализации образовательных задач курса.
- Стандартное программное обеспечение для работы над разработкой учебно-методических материалов.
- Мастерские и аудитории для проведения открытых занятий.

Специальных помещений, предполагающих наличие какого-либо специального оборудования для реализации данной программы, не предусматривается.

### **7.2. Кадровое обеспечение программы**

Образовательный процесс по программе осуществляется педагогом дополнительного образования с профильным высшим или средним профессиональным образованием.

К занятию педагогической деятельностью по дополнительной общеобразовательной программе также допускаются лица, обучающиеся по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующим направленности дополнительных общеобразовательных программ, и успешно прошедшие промежуточную аттестацию не менее чем за два года обучения.

Реализация дополнительной общеобразовательной (общеразвивающей) программы обеспечивается руководящими и педагогическими работниками организации, а также лицами, привлекаемыми к реализации программы на условиях гражданско-правового договора.

У педагогического работника, реализующего дополнительную общеобразовательную программу, должны быть сформированы основные компетенции, необходимые для обеспечения успешного достижения обучающимися планируемых результатов освоения программы, в том числе умения:

- обеспечивать условия для успешной деятельности, позитивной мотивации, а также самомотивирования обучающихся;
- осуществлять самостоятельный поиск и анализ информации с помощью современных информационно-поисковых технологий;
- разрабатывать программы учебных предметов, выбирать учебники и учебно-методическую литературу, рекомендовать обучающимся дополнительные источники информации, в том числе Интернет-ресурсы;
- реализовывать педагогическое оценивание деятельности обучающихся;
- работать с текстовыми редакторами, электронными таблицами, электронной почтой и браузерами, мультимедийным оборудованием.

# Приложения

## Приложение 1. Примерные вопросы для промежуточного тестирования

1. Какой из перечисленных языков программирования является основным для создания веб-приложений на Python?

- Java
- **Python**
- C++
- JavaScript

2. Какой веб-фреймворк чаще всего используется для разработки веб-приложений на Python?

- **Django**
- Ruby on Rails
- Flask
- Express.js

3. Какая база данных чаще всего используется в веб-приложениях на Python?

- MongoDB
- SQLite
- MySQL
- **PostgreSQL**

4. Какая из перечисленных операционных систем не является подходящей для разработки веб-приложений на Python?

- Windows
- Linux
- MacOS
- **iOS**

5. Какой HTTP-метод обычно используется для получения данных с сервера?

- POST
- DELETE
- **GET**
- PUT

6. Какие языки используются для структурирования и стилизации веб-страниц?

- Python и JavaScript
- **HTML и CSS**
- Java и PHP
- Ruby и SQL

7. Какой командой можно создать новый проект Flask?

- flask new
- create-flask
- flask create
- **flask init**

8. Какой HTTP-код обычно означает успешное выполнение запроса?

- **200**
- 404
- 500
- 403

9. Какой метод базы данных используется для создания новых записей?

- UPDATE
- DELETE
- **INSERT**
- SELECT

10. Какая система контроля версий широко используется для управления исходным кодом проектов?

- Subversion (SVN)
- Mercurial
- **Git**
- CVS



## Приложение 2. Примерные задания для оценки качества освоения учебного материала

1. Создайте простое веб-приложение с использованием Flask, которое выводит на экран "Hello, World!" при открытии главной страницы.
2. Добавьте статический контент (например, изображения, стили CSS) к веб-приложению и корректно их отобразите на странице.
3. Разработайте форму для ввода данных пользователем (например, имя, электронная почта) с использованием HTML и их отправкой на сервер.
4. Настройте обработку данных из формы веб-приложением на сервере с помощью Flask, а затем вывод полученных данных на новой странице.
5. Подключите базу данных (например, PostgreSQL) к веб-приложению и реализуйте создание таблицы для хранения информации о пользователях.
6. Разработайте функционал регистрации новых пользователей, включая проверку на уникальность электронной почты и хэширование пароля.
7. Реализуйте механизма аутентификации и авторизации пользователей приложения с использованием Flask-Login.
8. Создайте страницу личного кабинета пользователя, где он может просматривать и изменять свои данные.
9. Разработайте функционал загрузки файлов на сервер и их сохранение в определенной директории.
10. Интегрируйте стороннее API (например, API карт Google) в веб-приложение для отображения интерактивной карты на странице.