


**МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ
ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ «УЖУРСКАЯ СРЕДНЯЯ
ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА №6 ИМЕНИ ГЕРОЯ СОВЕТСКОГО
СОЮЗА Ю.Н. ПЕТЛИНА»**

«Рассмотрено»
на методическом совете
МБОУ «Ужурская СОШ №6 им. ГСС
Ю.Н. Петелина»
Протокол № 1
От «25» августа 2023 г.

«Согласовано»
 /Хлебникова О.А.
Заместитель директора по ВР
От «28» августа 2023г



«Утверждено»
Карелина Т.Б.
Директор МБОУ «Ужурская СОШ №6 им.
ГСС Ю.Н. Петелина»
Приказ № 01-18-57 от «30» августа 2023 г

РАБОЧАЯ ПРОГРАММА

к дополнительной общеобразовательной общеразвивающей программе
«КОД БУДУЩЕГО»

на **2023- 2024** учебный год

год обучения: 1

возраст учащихся: 14-16 лет _____

Разработчик:

Ротарь Снежанна Федоровна ,
педагог дополнительного
образования

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Аннотация

Программа «КОД будущего» предназначена для учащихся 8-11 классов. Этот курс будет полезен тем, кто интересуется областью разработки и поддержки десктопных приложений. Используя язык можно создать различное программное обеспечение, в том числе разрабатывать клиенты для мессенджеров, организовывать соединения с базами данных (при помощи языка запросов SQL).

Курс не требует специальной дополнительной подготовки, достаточно знаний предметной области информатики на базовом уровне.

Формирование знаний и навыков на высоком уровне достигается за счет того, что программа носит практико-ориентированный характер. Все знания, во время лекций отрабатываются практико-ориентированно.

АКТУАЛЬНОСТЬ

Язык программирования Питон в настоящее время является одним из самых востребованных. Он используется во многих проектах и в различных качествах: как как особый язык программирования так и для разработки приложений.

Цель программы

Формирование знаний, умений и навыков достаточных для разработки приложений с использованием языка программирования питон.

Задачи программы

Программа направлена на реализацию следующих задач обучения, развития и воспитания.

Задачи обучения

- Формировать и разбивать компетентность в области разработки приложений.
- Формировать представление о процессах разработки.
- Формировать представление об архитектуре компьютеров и сети.
- Формировать представление о правилах безопасной работы с информацией.
- Формировать ИКТ-компетенции.
- Формировать представление об уровне технического прогресса.

Задачи развития

- Формировать умение планировать и регулировать свою деятельность.
- Способность формировать навыки самооценки.
- Способствовать формированию навыков сотрудничества со взрослыми и сверстниками.

Задачи воспитания

- Способствовать формированию нравственных норм и ценностей в поведении и сознании.
- Способствовать принятию и освоению социальной роли обучающегося, развитию мотивов учебной деятельности и формированию личностного смысла образования.
- Способствовать развитию самостоятельности и личной ответственности за свои поступки, в том числе в информационной деятельности, на основе представлений о нравственных нормах, социальной справедливости и свободе. Развивать рефлексивную деятельность учащихся.
- Способствовать развитию эстетических чувств, доброжелательности и эмоционально-нравственной отзывчивости, понимания чувств других людей и сопереживания им.
- Способствовать формированию установки на безопасный, здоровый образ жизни, наличие мотивации к творческому труду, работе на результат, бережному отношению к материальным и духовным ценностям.

Планируемые результаты обучения

Сформировать у обучающихся школ базовые знания и умения в области решения задач разработки приложений.

По окончании обучения по программе обучающийся будет знать:

- Понятие алгоритма и алгоритмического языка программирования.
- Назначение Python и его сфера применения.
- Основы синтаксиса языка программирования Python.
- Основы программирования на Python:
- Базовые операторы, переменные и их типы, константы, массивы, управляющие конструкции, условные операторы, циклы, функции, принципы создание функций,

- аргументы функции, синхронные и асинхронные функции, область видимости переменных, стандартные библиотеки языка программирования Python.
- Структуру и элементы приложений. Компоненты, из которых состоит приложение. Основные принципы УБ/UI-дизайна.
 - Реляционные базы данных. СУБД. Модели данных, основные операции и ограничения. Технология установки и настройки сервера баз данных. Язык запросов SQL. Настройка окружения и PИstPrOSQL. Транзакции, понятие данных и метаданных. Особенности системы управления базами данных ДВОКЯОг.
 - Систему контроля версий.
 - Возможности используемой системы контроля версий и вспомогательных инструментальных программных средств для обработки исходного текста программного кода.
 - Основные принципы отладки и тестирования программных продуктов. Типы тестирования.
 - Методы и приемы отладки программного кода. Типы и форматы сообщений об ошибках, предупреждениях.
 - Принципы работы современных компиляторов, отладчиков и оптимизаторов программного кода.

По окончании обучения по программе обучающийся будет уметь:

- Устанавливать и настраивать среду разработки Python.
- Применять язык программирования Python для написания программного кода для решения учебных и практических задач.
- Создавать структуру кода, размещающую элементы приложения.
- Использовать возможности языка программирования Python для разработки десктопного приложения.
- Проектировать и создавать базы данных.
- Создавать сценарии исполнения запросов на создание, изменение, выборку и удаление данных на языке SQL.
- Работать с системой контроля версий.
- Использовать инструментальные средства для отладки и тестирования.
- Выполнять отладку и тестирование программы на уровне модуля.
- Выявлять ошибки в программном коде.
- Интерпретировать сообщения об ошибках, предупреждениях.

Категория обучающихся (возраст) по программе

Учащиеся, осваивающие образовательные программы основного общего и среднего общего образования (с 8 по 11 классы)

Форма обучения:

Очная форма обучения без применения дистанционных образовательных технологий.

Срок реализации программы:

1 год

Трудоемкость программы:

78 академических часов.

(ТЕМАТИЧЕСКИЙ) ПЛАН

№ п/п	Наименование модулей/тем программы	Всего, час	Виды учебных занятий			Формы контроля
			Лекции	Практические занятия	Самостоятельная работа	
1	Тема 1.1. Архитектура компьютера и сетей	8	2	2	4	
2	Тема 1.2. Введение в разработку. Знакомство с Python	28	4	10	14	
3	Тема 2.1 Введение в работу с функциями на языке Python	16	3	7	6	
4	Тема 2.2 Работа с файлами	6	2	2	2	
5	Тема 2.3 Основы работы с базами данных	14	3	5	6	
6	Тема 3.1 Работа с Git. Версионность	9	2	3	4	
7	Тема 3.2 Введение в объектно-ориентированное программирование	22	6	8	8	
8	Тема 3.3 Интерфейс приложения	5	1	2	2	
9	Тема 4.1 Архитектура приложений и шаблоны проектирования	6	1	2	3	
10	Тема 4.2 Разработка полноценного приложения	25	4	9	12	
11	Тема 4.3 Тестирование приложений	5	1	2	2	
13	Промежуточная аттестация	4				Тестирование
14	Итоговая аттестация	2				Защита проекта
15	Итого	150	29	52	63	

Модуль 1. Введение в разработку. Знакомство с Python

Модуль знакомит с основами и особенностями языка программирования Python, его назначения и его сфера применения, архитектурой компьютера и сетей. Во время обучения, обучающиеся подготовят свое рабочее место и приступят к изучению синтаксиса и базовых конструкций языка программирования Python. Научатся использовать язык программирования Python для решения учебных и практических задач.

Название темы	Содержание		
	Лекций	Практических занятий	Самостоятельной работы
Тема 1.1. Архитектура компьютера и сетей	Архитектура компьютера Принцип работы процессора, устройства памяти. Операционные системы. Различия и сходства в языках программирования разных уровней. Архитектура сетей.	Решение учебных и практических задач. Сборка идеального компьютера «на бумаге». Разработка алгоритма для работы одной единицы техники	Повторение материалов по теме. Сборка идеального компьютера «на бумаге». Разработка алгоритма для работы одной единицы техники
Тема 1.2. Введение в разработку. Знакомство с Python	Понятие алгоритма и алгоритмического языка программирования. Синтаксис языка программирования Python. Базовые операторы. Переменные и их типы. Константы. Массивы. Управляющие конструкции. Условные операторы. Операторы инкремента, декремента. Циклы. Работа с операторами Отладка кода.	Решение алгоритмических задач с помощью блок-схем и описание объектов. Решение практических задач с помощью языка программирования Python. Работа с условиями, циклами, массивами, операторами. Создание веб-приложения.	Повторение материалов по теме. Решение алгоритмических задач с помощью блок-схем и описание объектов. Решение практических задач с помощью языка программирования Python.
Промежуточная аттестация			

Модуль 2. Введение в работу с функциями на языке Python. Работа с данными.

Модуль знакомит с функциями, научатся создавать собственные функции и использовать библиотеки. Научатся работать с файлами для чтения и записи данных. Познакомятся с базами данных и языком запросов SQL. По окончании модулю спроектируют базу данных собственного сайта, смогут создавать сценарии исполнения запросов на создание, изменение, выборку и удаление данных, получать результата запроса и вывод его на страницу сайта.

Название темы	Содержание		
	Лекций	Практических занятий	Самостоятельной работы
Тема 2.1 Введение в работу с функциями на языке Python	Функции. Создание функции. Аргументы функции и аргументы по умолчанию. Область	Создание собственных функций. Подключение библиотек. Оптимизация кода. Использование	Повторение материалов по теме. Использование функций для решения практических задач. Использование

	видимости переменных. Статические переменные. Локальные и глобальные переменные. Обзор встроенных функций. Библиотеки. Полезные функции. Синхронные и асинхронные функции. Сортировка.	функций для решения практических задач.	продвинутых возможности функций для написания кода.
Тема 2.2 Работа с файлами	Функции для работы с файлами. Режимы работы. Чтение и запись. Функции для работы с директориями.	Работа с файлами. Организация чтения и записи. Решение учебных и практических задач	Повторение материалов по теме. Работа с файлами. Организация чтения и записи.
Тема 2.3 Основы работы с базами данных	Общие сведения о системах управления базами данных (СУБД). PostgreSQL. Программа Python для работы с СУБД MySQL. Язык запросов SQL. Настройка окружения и PostgreSQL. Транзакции, понятие данных и метаданных. Создание базы данных и таблицы. Создание запросов на языке SQL. Соединение с сервером баз данных MySQL и выбор базы данных. Создание сценариев исполнения запросов на создание, изменение, выборку и удаление данных. Получение результата запроса и вывод его на страницу сайта.	Создание простых таблиц, заполнение их данными и обновление. Разработка кода на языке базы данных PostgreSQL — PL/SQL. Отправка данных на сервер. Создание схемы базы данных проекта.	Повторение материалов по теме. Работа с запросами на языке SQL.
Промежуточная аттестация			

Модуль 3. Основы объектно-ориентированного программирования. Работа с системой контроля версий.

Модуль знакомит с возможностями принципов объектно-ориентированного программирования для разработки приложений, учит работать с версионностью и применять системы контроля версий для обработки исходного текста программного кода, разрабатывать элементарный интерфейс приложений.

Название темы	Содержание		
	Лекций	Практических занятий	Самостоятельной работы
Тема 3.1 Работа с Git. Версионность	Система контроля версий. Интерфейс Git. Возможности используемой системы	Применение системы контроля версий для обработки исходного текста программного	Повторение материалов по теме.

	контроля версий и вспомогательных инструментальных программных средств для обработки исходного текста программного кода.	кода. Регистрация изменений исходного текста программного кода в системе контроля версий	
Тема 3.2 Введение в объектно-ориентированное программирование	Классы и объекты. Конструктор класса. Магические методы. Наследование классов. Инкапсуляция. Полиморфизм. СтКлТМ, соХП, и ПЧКХ. Абстрактные классы. Анонимные классы. Перегрузка. Пространство имен. Классы ДКлОТЦО. Замыкания. Генераторы. Итераторы.	Решение практических задач с использованием классов, объектов. Использование объектов для реализации программной логики. Защита состояния объектов от внешнего воздействия с помощью модификаторов доступа. Использование исключений.	Повторение материалов по теме. Решение практических задач.
Тема 3.3 Интерфейс приложения	Компоненты, из которых состоит приложение. Введение в УБ/UI-дизайн. Генерация идей приложения.	Разработка прототипа для одной из выдвинутых идей приложения.	Повторение материалов по теме. Разработка прототипа для одной из выдвинутых идей приложения.
Промежуточная аттестация			

Модуль 4. Архитектура приложений. Тестирование

Модуль знакомит с шаблонами проектирования программного обеспечения, конфигурацией, стандартами кодирования и интерфейсами. Во время обучения, обучающиеся научатся изменять базовую конфигурацию без создания опасных ситуаций, разрабатывать полноценное приложение и выполнять его тестирование.

Название темы	Содержание		
	Лекций	Практических занятий	Самостоятельной работы
Тема 4.1 Архитектура приложений и шаблоны проектирования	Понятие MVC, F#ШЧг Controller и настройка веб-сервера. Шаблоны проектирования программного обеспечения. Конфигурация.	Применение шаблонов проектирования при разработке веб-приложения. Реализация шаблона декоратор. Реализация произвольного шаблона. Переопределение базовой конфигурации без создания опасных ситуаций	Повторение материалов по теме. Реализация шаблонов проектирования.
Тема 4.2 Разработка полноценного приложения	Этапы разработки приложения. Разработка полноценного приложения. Повторение.	Разработка интерфейса и исходного кода приложения.	Повторение материалов по теме. Разработка приложения.
Тема 4.3 Тестирование приложений	Тестирование приложения: моки и стабы. Непрерывная доставка Pning. JMeter	Тестирование ошибок. Подготовка данных для тестов. Тестирование кода, взаимодействующего с	Повторение материалов по теме. Тестирование приложения.

	нагрузочные испытания. Автотесты	файлами. Работа с тестами.	
Промежуточная аттестация			
Итоговая аттестация			

Методическое обеспечение. Методы, формы и технологии, применяемые при реализации программы

Обучение на курсе организовано в групповой и индивидуально-групповой форме.

Курс разработан с использованием следующих образовательных технологий:

- технология развивающего обучения,
- технология проблемного обучения,
- проектная технология,
- здоровьесберегающие технологии.

При разработке материалов курса учитываются различные типы и методы.

Ко всем необходимым материалам доступ обучающихся организуется через платформу обучения: презентациям, заданиям, записям прошедших вебинаров.

Все знания, полученные во время онлайн-лекций, отрабатываются вовремя онлайн практических занятий. Система заданий во время практики организована таким образом, что сначала обучающимся предлагаются задания на формирование и отработку конкретных умений через решение алгоритмических задач. После чего учащимся предлагается выполнение учебного проекта, например, создание сайта-визитки, новостной страницы, калькулятора, игры, создания формы обратной связи, формы регистрации, справочника, создания схемы базы данных собственного проекта и реализации работы веб-приложения с базой данных.

Материально-техническое обеспечение

- Персональный компьютер, который соответствует минимальным требованиям:
Для ПК на АТЧНШas: процессор I-серии (I3, I5, I7) или аналогичный от AMD (от 4 ядер), от 4G В RAM.
Для ПК на МКМ OS: конфигурация IЧтOX T7-2720QM/RAM 4 GB/HDD 750 GB., ОС АТЧНШas 10 (версии НШЦО достаточно) либо МКМ OS Б;
- Высокоскоростной стабильный доступ в интернет.

УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

В процессе обучения по образовательной программе система заданий организована таким образом, что сначала обучающимся предлагаются задания на формирование и отработку конкретных умений через решение алгоритмических задач. После чего учащимся предлагается выполнение учебного проекта.

ФОРМЫ АТТЕСТАЦИИ

Аттестация проводится в двух формах: промежуточная итоговая.

Промежуточная аттестация проходит после изучения каждого модуля.

Форма контроля – тестирование.. Тест считается пройденным если выполнено более 50 %

Итоговая аттестация. Данный вид аттестации проводится по итогам освоения всей Образовательной программы, включая освоение ее последнего модуля.

Форма контроля – защита проектов.

Показатели и критерии оценивания:

Проекты проверяются в соответствии с заданиями, описанными в материалах.

За каждое выполненное условие ставится от 0 до 2 баллов согласно критериям оценивания. Критерии оценивания:

0 — условие не выполнено

1 — условие выполнено, частично

2 — условие выполнено, полностью

Шкала оценивания

Нижнее значение	0
Верхнее значение	10
Минимальный проходной балл для успешной сдачи	5

ПЕРЕЧЕНЬ ИСТОЧНИКОВ ИНФОРМАЦИОННОГО СОПРОВОЖДЕНИЯ

- Эрик Мэтиз Изучаем Python. Программирование игр, визуализация данных, веб-приложения. – СПб.: Питер, 2017. – 496 с: ил – (Серия “Библиотека программиста”)
- Марк Лутц Изучаем Python, том 1, 5-е изд.: Пер. с англ. — СПб.: ООО “Диалектика”, 2019. — 832 с.: ил. — Паралл. тит. англ.
- Джеймс Р. Грофф, Пол Н. Вайнберг, Эндрю Дж. Оппель. SQL. Полное руководство. - Диалектика, 2019. – 960 с.
- Дэниэл Кляйн, Брэнд Хант, Кевин Кляйн, SQL. Справочник. – 3-е изд. – СПб.: Символ-Плюс, 2010. – 656 с.

