

МБОУ «Рудничская СОШ» п.г.т. Тенишево  
Камско-Устьинского муниципального района  
Республики Татарстан

«Рассмотрено»

Руководитель МО

Естественно-математического  
цикла

Андрей /Миннигалеева А.Н.  
Протокол №1  
25 августа 2023 г.

«Согласовано»

Заместитель директора УР  
МБОУ «Рудничская СОШ»

п.г.т. Тенишево  
Илья /Шакирова А.Х./  
26 августа 2023 г.

«Утверждаю»

Директор МБОУ

МБОУ «Рудничская СОШ»  
п.г.т. Тенишево  
Г.И. Хасанова Г.И.  
26 августа 2023

*Рабочая программа  
по дополнительному образованию  
«Основы программирования PYTHON»  
для учащихся 7-8 классов*

Подготовила:  
Миннигалеева Э.И.,  
учитель информатики  
МБОУ «Рудничская СОШ»

п.г.т. Тенишево, 2023 г.

## СОДЕРЖАНИЕ

<b>Раздел №1. Комплекс основных характеристик дополнительной общеобразовательной общеразвивающей программы</b>	<b>3</b>
<b>1.1. Пояснительная записка</b>	<b>3</b>
направленность программы	3
уровень освоения	3
актуальность программы	3
педагогическая целесообразность	4
отличительные особенности программы	4
адресат программы	4
объем и сроки освоения программы	4
формы обучения	5
формы организации образовательного процесса	5
режим занятий	5
<b>1.2. Цель и задачи программы</b>	<b>5</b>
цель	5
задачи: воспитательные, развивающие, образовательные	6
<b>1.3. Содержание программы</b>	<b>7</b>
учебный план	7
содержание учебного плана	7
<b>1.4. Планируемые результаты: личностные, метапредметные, предметные</b>	<b>7</b>
<b>Раздел №2. Комплекс организационно-педагогических условий</b>	<b>9</b>
<b>2.1. Календарно-учебный график</b>	<b>9</b>
<b>2.2. Условия реализации программы: материально-техническое, информационное и кадровое обеспечение</b>	<b>13</b>
<b>2.3. Формы аттестации</b>	<b>13</b>
<b>2.4. Оценочные материалы</b>	<b>13</b>
<b>2.5. Методические материалы</b>	<b>13</b>
<b>2.6. Список литературы</b>	<b>14</b>

## **Раздел №1. «Комплекс основных характеристик дополнительной общеобразовательной общеразвивающей программы»**

### **1.1. Пояснительная записка**

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования. Python» - это программа **технической направленности**.

Дополнительная общеобразовательная программа составлена в соответствии с нормативно-правовыми документами:

- Федеральным законом «Об образовании в Российской Федерации» (№ 273-ФЗ от 29 декабря 2012 г.).
- Приказом министерства образования и науки №1008 от 29 августа 2013 г. «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».
- Концепцией развития дополнительного образования детей (распоряжение РФ от 4.09.14 №1726-р).
- Санитарно-эпидемиологическими требованиями к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей (Сан.ПиН 2.4.4. 3172-14).

#### **Уровень освоения программы - общекультурный.**

Программа представляет особый интерес в связи с широким распространением цифровой техники в обществе, в связи с возрастающей потребностью обучающихся в освоении цифровых технологий и повышением их интереса к современным языкам программирования.

Содержание программы предусматривает ознакомление воспитанников с основами работы в интегрированной среде программирования и основными возможностями языка программирования Python. Программа позволяет освоить практически все операторные конструкции и познакомиться с основными функционального и объектного программирования.

**Актуальность программы** заключается в том, что в настоящее время владение компьютерными технологиями рассматривается как важнейший компонент образования, играющий значимую роль в решении приоритетных задач образования – в формировании целостного мировоззрения, системно-информационной картины мира, учебных и коммуникативных навыков.

Воспитанники, прошедшие обучение по данной программе, получают знания, умения и навыки, необходимые для сознательного выбора в будущем профессии, связанной с программированием. Программа позволяет школьникам познакомиться и понять тонкости профессии программист и профессий связанных с разработкой ИТ-приложений, оценить себя в этих профессиях, выработать профессионально важные качества.

**Педагогическая целесообразность** этой программы заключается в том, что она является целостной и непрерывной в течении всего процесса обучения, и позволяет школьнику шаг за шагом раскрывать в себе творческие возможности и само реализоваться в современном мире. В процессе программирования дети получат дополнительное образование в области алгебры, геометрии и информатики.

## **Отличительные особенности программы.**

Особенностью программы является её направленность на формировании у воспитанников потребности в инженерном образовании научном исследовании. Инновационной является авторская методика, позволяющая сочетать изучение достаточно сложного языка программирования с исследовательской деятельностью и разработкой проектов самых разнообразных по содержанию, позволяющая ребенку в конце курса обучения почувствовать себя настоящим профессионалом.

**Адресат программы.** Программа рассчитана для учащихся 7-8 классов. Программа может корректироваться в процессе работы с учетом возможностей материально-технической базы, возрастных особенностей обучающихся, их способностей усваивать материал.

Обучающиеся, поступающие в объединение, проходят собеседование, направленное на выявление их индивидуальности и склонности к выбранной деятельности. Занятия проводятся в группах, подгруппах и индивидуально, сочетая принцип группового обучения с индивидуальным подходом. Условия набора детей в коллектив: принимаются все желающие. Наполняемость в группах составляет 20 человек.

**Объем и сроки освоения программы.** Программа рассчитана на 1 год обучения - 1 раз в неделю по 2 академических часа, итого 68 часов.

Занятия проводятся в компьютерном классе, оснащенном индивидуальными стационарными компьютерами и оборудованием для презентаций. В каждом занятии прослеживаются три части: теоретическая, практическая и исследовательская.

Учебные часы условно делятся на теорию и практику. Однако, теория включает в себя не только разбор конструкция языка программирования, алгоритмических приёмов, технологий решения задач, но и практические задания, выполнение которых непосредственно не связано с работой на персональном компьютере. На теории организуется обсуждение с учащимися сложных вопросов, проектных решений, применение тех или иных методов в различных профессиях.

Практические занятия проводятся непосредственно на персональном компьютере, и предполагают работу в интегрированной среде программирования Python с 10 минутным перерывом, во время которого учащиеся выполняют упражнений для снятия зрительного напряжения и снятия напряжения с костно-мышечного аппарата.

Исследовательская часть предполагает всестороннее изучение возможностей отдельных конструкций языка программирования Python и возможность их применения при решении нестандартных задач прикладного характера, разработку оптимальных нестандартных алгоритмов, создание компактного программного кода. Обучающиеся могут предложить собственный алгоритм решения задачи и протестировать работу соответствующей программы.

## **Формы обучения:**

- теоретическая форма, в которой преподаватель объясняет новый материал и консультирует обучающихся в процессе выполнения ими практических заданий на компьютере;

- практическая форма, в которой обучающиеся после занятий самостоятельно выполняют на компьютере практические задания.

**Формы организации образовательного процесса:** лекция, беседа, демонстрация, практические занятия, творческая работа; проектная деятельность.

#### **Режим занятий**

Занятия проходят 1 раза в неделю по 2 академических часа.

### **1.2. Цель и задачи программы**

**Цель:** Создание условий для достижения обучающимися результатов развития в личностном, предметном, метапредметном направлениях, обеспечивающих их социальную адаптацию в области программирования, ИКТ на профессиональном уровне.

Развить личность обучающегося, способного к творческому самовыражению через овладение основами программирования на одном из самых востребованных языков программирования Python.

Понять значение алгоритмизации как метода познания окружающего мира, принципы структурной алгоритмизации; научиться разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python.

#### **Задачи:**

#### **Обучающие:**

- Познакомить с возможностями и особенностями современного языка программирования Python;
- Сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- Изучить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных (строками, списками, кортежами, словарями, множествами); научить применять функции при написании программ на языке программирования Python;
- Научить отлаживать и тестировать программы, делать выводы о работе этих программ.
- Сформировать систему знаний, умений и навыков, необходимых в работе программиста;
- Обучить технологии работы со справочниками по языку программирования и поиску нужной информации на сайтах разработчиков программного обеспечения;
- Обучить приёмам предъявления результатов проделанной работы на конференциях и, конкурсах;
- Обучить коммуникативным навыкам при совместной работе над проектом.

#### **Развивающие:**

- Развить познавательные потребности и способности школьников
- Развивать творческие способности, алгоритмическое и логическое мышление;

- Развивать представления о возможностях применения современных компьютерных технологий в профессиональной деятельности;
- Развивать внимательность и наблюдательность, прививать навыки аккуратности и точности в работе;
- Повысить интеллектуальный уровень и расширить интеллектуальные навыки;
- Создать мотивацию к постоянному самообразованию.

#### **Воспитательные:**

- Воспитывать навыки взаимодействия при командной работе над проектом;
- Воспитывать чувство ответственности за результаты деятельности;
- Способствовать формированию культуры программирования;
- Способствовать формированию у воспитанников культуры создания и этики представления проектов на конференциях и в Интернете;
- Сформировать осознанное отношение к выбору будущей профессии.

### **1.3. Содержание программы**

#### **1. Основы охраны труда. Организация рабочего места.**

Валеологические аспекты и требования безопасности при работе за компьютером. Безопасность в Интернете.

Обучающиеся знакомятся с правилами поведения в компьютерном классе, правилами охраны труда, рекомендациями по организации рабочего места.

#### **2. Знакомство с языком Python. Установка программы Python.**

Общие сведения о языке Python, преимущества языка, сферы применения. Установка программы Python на компьютер. Режимы работы Python. Использование командной строки интерпретатора. Выбор редактора. Работа со средой программирования. Сохранение, открытие и запуск программ. Получение помощи. Знакомство с языком Python и написание простейших программ. Структура программы на языке Python Комментарии.

#### **3. Переменные и выражения. Операции. Организация ввода и вывода данных.**

Литеральные константы, числа, строки. Использование кавычек. Объединение строковых констант. Метод format.

Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова.

Объекты. Примеры использования переменных и констант. Логические и физические строки. Отступы. Операторы и их применение. Выражения. Операции. Порядок выполнения операций. Краткая запись математических операций и порядок вычисления.

Изменение порядка вычисления. Математические функции. Композиция. Ассоциативность. Элементарные действия с числами.

Ввод и вывод данных. Ввод данных с клавиатуры. Вывод данных на экран. Пример скрипта, использующего ввод и вывод данных.

#### **4. Программирование ветвящихся алгоритмов.**

Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Поток команд. Ветвящиеся алгоритмы. Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python. Решение задач на программирование ветвящихся алгоритмов.

#### **5. Циклы. Программирование циклических алгоритмов.**

Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием.

Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов.

Оператор цикла с параметром for. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Реализация циклических алгоритмов. Решение задач с использованием циклов. Операторы управления циклом. Оператор break. Оператор continue.

Случайные числа. Функция randrange. Функция random. Примеры решения задач с циклом.

#### **6. Функции**

Создание функций. Параметры функций. Локальные и глобальные переменные. Зарезервированные слова «global», «nonlocal». Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda. Значения аргументов по умолчанию. Ключевые аргументы. Переменное число параметров. Ключевые параметры. Оператор «return».

Строки документации. Аннотации. Решение задач с использованием функций. Рекурсивные функции.

Примеры решения задач с использованием функций. Рекурсивные функции. Вычисление факториала. Числа Фибоначчи.

#### **7. Структуры данных**

Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки. Срезы строк. Строки нельзя изменить. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки, списки, кортежи). Примеры решения задач со строками.

Списки. Объекты и классы. Тип список (list). Индексы. Обход списка. Проверка вхождения в список. Добавление в список. Суммирование или изменение списка. Операторы для списков.

Срезы списков. Удаление списка. Клонирование списков. Списочные параметры. Функция range. Списки: примеры решения задач. Матрицы. Вложенные списки. Матрицы. Строки и списки.

## 8. Генераторы списков в Python.

Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения. Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы. Последовательности.

9. **Множества** в языке Python. Множества. Множественный тип данных. Описание множеств. Операции, допустимые над множествами: объединение, пересечение, разность, включение. Оператор определения принадлежности элемента множеству.

10. Процесс разработки программного обеспечения. Стиль программирования. Отладка программ.

## 11. Работа над проектами

Требования к проектам. Выбор темы. Разработка алгоритма и написание программного кода. Работа над тестированием и отладкой программ. Оформление проектной документации. Защита проектов.

# УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№ разде ла	Тема занятий	Всего	Количество часов	
			теория	практика
1	Основы охраны труда. Организация рабочего места.	2	1	1
2	Знакомство с языком Python.	2	1	1
3	Установка программы Python.	2	1	1
4	Переменные и выражения. Организация ввода и вывода данных. Операции.	6	2	4
5	Программирование ветвящихся алгоритмов.	6	2	4
6	Циклы	8	3	5
7	Функции	8	3	5
8	Модули	4	2	2
9	Структуры данных	14	7	7
10	Стиль программирования и отладка программ	2	1	1
11	Работа над проектами	14	3	11
итого		68	23	45

#### **1.4. Планируемые результаты: личностные, метапредметные, предметные:**

##### Предметные:

- изучить принципы структурного, функционального и объектного программирования на примере языка программирования Python;
- научиться формулировать и анализировать алгоритмы, составлять и отлаживать программы.

##### Личностные результаты:

- сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
- гармоничное развитие творческих способностей и логического мышления учащихся;
- осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов.

##### Метапредметные результаты:

- владение навыками познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем;
- способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания.

## Раздел №2. «Комплекс организационно-педагогических условий»

### 2.1. Календарно-учебный график

№ п/п	план	факт	Тема занятия	Кол-во часов	Форма занятия	Место проведения	Форма контроля	
1			Инструктаж по технике безопасности. Организация рабочего места.	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение	
2			Знакомство с языком Python.	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение	
3			Установка программы Python.	2	Практическое занятие	Кабинет информатики	Беседа, наблюдение	
<b>Переменные и выражения. Организация ввода и вывода данных. Операции.</b>								
4			Переменные	1	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение	
5			Выражения	1	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение	
6			Операции. Элементарные действия с числами	2	Практическое занятие	Кабинет информатики	Беседа, наблюдение	
7			Ввод и вывод данных	2	Практическое занятие	Кабинет информатики	Беседа, наблюдение	
<b>Программирование ветвящихся алгоритмов.</b>								
8			Логические выражения и операторы. Условный оператор. Множественное ветвление.	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение	
9			Реализация ветвления на языке Python	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение	

10			Решение задач на программирование ветвящихся алгоритмов	2	Практическое занятие	Кабинет информатики	Беседа, наблюдение
<b>Циклы</b>							
11			Оператор цикла с условием Оператор While	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
12			Оператор цикла с параметром FOR	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
13			Вложенные циклы	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
14			Реализация циклических алгоритмов	2	Практическое занятие	Кабинет информатики	Беседа, наблюдение
<b>Функции</b>							
15			Создание функций	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
16			Локальные и глобальные переменные	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
17			Решение задач с использованием функций	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
18			Рекурсивные функции	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
<b>Модули</b>							
19			Модули в языке Python	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
20			Оформление собственных модулей	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
<b>Структуры данных</b>							

21			Строки. Срезу строк	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
22			Списки. Срезы списков	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
23			Кортежи	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
24			Словари	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
25			Последовательности	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
26			Матрицы. Множества	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
27			Ссылки	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение

#### Стиль программирования и отладка программ

28			Стиль программирования	1	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
29			Отладка программ	1	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение

#### Работа над проектами

30			Требования к проектам. Выбор темы	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
31			Разработка алгоритма и написание программного кода	2	Практическое занятие	Кабинет информатики	Беседа, наблюдение
			Разработка алгоритма и написание программного кода	2	Практическое занятие	Кабинет информатики	Беседа, наблюдение
32			Работа над тестированием и отладкой программ	2	Практическое занятие	Кабинет информатики	Беседа, наблюдение
33			Работа над тестированием и отладкой программ	2	Практическое занятие	Кабинет информатики	Беседа, наблюдение

34			Защита проектов	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение
35			Защита проектов	2	Комбинированное занятие	Кабинет информатики	Беседа, наблюдение

## **2.2 Условия реализации программы: материально-техническое, информационное и кадровое обеспечение**

### **Материально-техническое обеспечение программы:**

- кабинет Информатики
- комплект столов и стульев на 20 посадочных мест;
- стол для педагога;
- раздаточный материал
- ноутбуки с комплектом программ по изучению языка программирования;
- Интернет.

### **Информационное обеспечение:**

- методические и дидактические материалы
- презентации, подготовленные к каждому занятию.

### **Кадровое обеспечение программы.**

Реализацию программы обеспечивает учитель информатики, обладающий не только профессиональными знаниями, но и компетенциями в организации и ведении образовательной деятельности творческого объединения.

## **2.3. Формы аттестации/контроля**

Система оценивания - безотметочная. Используется только словесная оценка достижений учащихся.

Формы отслеживания и фиксации образовательных результатов: аналитическая справка, аналитический материал, видеозапись, журнал посещаемости.

Формы предъявления и демонстрации образовательных результатов: аналитическая справка, демонстрация программ, контрольная работа, презентация итогового проекта перед родителями и педагогами.

## **2.4. Оценочные материалы**

Для оценки результативности программы используются следующие методики и диагностики:

- наблюдение в ходе обучения с фиксацией результата;
- проведение проверочных работ;
- анализ, обобщение и обсуждение результатов обучения;
- проведение открытых занятий с их последующим обсуждением;
- участие в проектной деятельности школы, города;
- участие в соревнованиях муниципального, зонального и регионального уровней;
- оценка выполненных практических работ.

## **2.5. Методические материалы**

- Раздаточный материал контролирующего и обучающего характера по каждой теме.
- Задания на практические работы.
- Карточки с индивидуальными заданиями.
- Раздаточный материал справочного характера.
- Раздаточный материал теоретического характера.
- Демонстрационные материалы в электронном виде.

## **2.6. Список литературы**

**Литература для учителя:**

1. Домашняя страница Python [www.python.org](http://www.python.org) . Справочные материалы, официальная документация.
2. Сайт проекта Интуит: Национальный открытый университет, курс «Введение в программирование на Python», <http://www.intuit.ru/studies/courses/12179/1172/info>.
3. Сайт проекта Интуит: Национальный открытый университет. Курс «Язык программирования Python» <http://www.intuit.ru/studies/courses/49/49/info>.
4. Сайт проекта Open Book Project [openbookproject.net](http://openbookproject.net) содержит серию практических примеров на Python Криса Мейерса.
5. Python. Подробный справочник Дэвида М. Бизли — книга со справочной информацией о языке Python и модулях стандартной библиотеки.
6. Python. Справочник Марка Лутца. Справочник по наиболее часто использующимся функциям и модулям.

### **Литература для учащегося:**

1. Домашняя страница Python [www.python.org](http://www.python.org) . Справочные материалы, официальная документация.
2. Сайт проекта Интуит: Национальный открытый университет, курс «Введение в программирование на Python», <http://www.intuit.ru/studies/courses/12179/1172/info>.
3. Сайт проекта Интуит: Национальный открытый университет. Курс «Язык программирования Python» <http://www.intuit.ru/studies/courses/49/49/info>.
4. Сайт проекта Open Book Project [openbookproject.net](http://openbookproject.net) содержит серию практических примеров на Python Криса Мейерса.
5. Python. Подробный справочник Дэвида М. Бизли — книга со справочной информацией о языке Python и модулях стандартной библиотеки.
6. Python. Справочник Марка Лутца. Справочник по наиболее часто использующимся функциям и модулям.