

Министерство образования и науки Республики Дагестан  
Государственное бюджетное профессионально образовательное учреждение РД  
«Профессионально-педагогический колледж имени М.М.Меджидова»

Принята на заседании  
методического совета  
ГБПОУ РД «ПК им.М.М.Меджидова»  
От 23.05.2024 г.  
Протокол № 7



УТВЕРЖДАЮ:  
Директор ГБПОУ РД  
«ПК им.М.М.Меджидова»  
С.М.Алдиева

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ  
(ОБЩЕРАЗВИВАЮЩАЯ) ПРОГРАММА  
технической направленности  
«Программирование на PYTHON»**

Уровень программы: базовый

Возраст учащихся: 14-18 лет

Срок реализации: 1 год

Педагог дополнительного образования:  
Абдуллаева Зарема Курбанмагомедовна

г.Избербаш,

2024г.

## СОДЕРЖАНИЕ

### РАЗДЕЛ 1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ

1.1. Пояснительная записка .....	3
1.2. Цель и задачи программы .....	5
1.3. Учебно-тематический план и содержание программы .....	6
1.4. Планируемые результаты .....	9

### РАЗДЕЛ 2. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

2.1. Календарный учебный график .....	10
2.2. Условия реализации программы .....	10
2.3. Формы контроля .....	12
2.4. Оценочные материалы .....	12
2.5. Методические материалы .....	12
2.6. Список литературы .....	13
<b>ПРИЛОЖЕНИЯ</b> .....	14

## Раздел 1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ

### Пояснительная записка

#### Направленность программы

Дополнительная общеразвивающая программа «Язык программирования Python» имеет техническую направленность и предназначена для организации внеурочной деятельности по нескольким взаимосвязанным направлениям развития личности, таким как общеинтеллектуальное, общекультурное, социальное. Программа предлагает ее реализацию в факультативной или кружковой форме.

В современную жизнь человека все шире внедряются компьютеры и информационные технологии. Поэтому все большее значение приобретает компьютерная грамотность.

Курс изучения компьютерной грамотности состоит из двух разделов: пользовательского курса и программирования. Раздел «Программирование» в школьном курсе представлен языком программирования Pascal, а многим учащимся хочется познакомиться с другими языками программирования, самим попробовать разработать программы, которые можно использовать на уроках и во внеурочной деятельности.

Данная Программа позволяет реализовать эти желания, так как уделяется большое внимание практической работе учащихся на компьютере, самостоятельной разработке программ для решения практических задач.

Программа ориентирована на обобщение и углубление знаний и умений по курсу информатики основной школы.

**Нормативно-правовое обоснование.** В настоящее время содержание, роль, назначение и условия реализации программ дополнительного образования закреплены в следующих нормативных документах:

- Федеральный Закон Российской Федерации от 29.12.2012 № 273 «Об образовании в Российской Федерации» (с изменениями и дополнениями);

- Изменения в Федеральный закон «Об образовании в Российской Федерации» 273-ФЗ в части определения содержания воспитания в образовательном процессе с 01.09.2020;

- Указ Президента Российской Федерации «О национальных целях развития Российской Федерации на период до 2030 года», определяющего одной из национальных целей развития Российской Федерации предоставление возможности для самореализации и развития талантов;

- Приказ Министерства просвещения РФ от 9 ноября 2018 г. № 196 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;

- Приказ Министерства образования и науки Российской Федерации от 23 августа 2017 г. № 816 «Порядок применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ» (зарегистрирован Министерством юстиции Российской Федерации от 18 сентября 2017 г., регистрационный № 48226);

- Письмо Минобрнауки России от 18.11.2015 №09-3242. «Методические рекомендации по проектированию дополнительных общеразвивающих программ» (включая разноуровневые программы);

- Постановление Главного государственного санитарного врача РФ от 28 сентября 2020 г. № 28 «Об утверждении санитарных правил СП 2.4. 3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;

- Устав ГБПОУ РД «Профессионально-педагогический колледж имени М.М.Меджидова»;

- Локальные акты, регламентирующие образовательную деятельность Центра

цифрового образования детей «IT-куб» ГБПОУ РД «Профессионально-педагогический колледж имени М.М.Меджидова».

**Направленность** программы – техническая.

**Новизна программы.** Новизна Программы заключается в том, что Python дает более широкие возможности в области программирования, чем Pascal, который входит в школьный курс информатики. На языке Python можно легко и быстро создавать простые компьютерные игры, трёхмерные модели и программировать роботов. Этот язык быстрее и легче усваивается, чем Pascal. Многие мировые компании такие, как Intel, Cisco, Hewlett-Packard, используют этот язык при реализации своих проектов. Крупнейшие интернет-ресурсы такие, как Google, YouTube, также разработаны с помощью языка программирования Python.

**Актуальность программы**

Python – это язык программирования общего назначения, распространяемый с открытыми исходными текстами. Он оптимизирован для создания качественного программного обеспечения. Язык Python используется сотнями тысяч разработчиков по всему миру в таких областях, как создание веб-сценариев, системное программирование, создание пользовательских интерфейсов, настройка программных продуктов под пользователя, численное программирование и в других. Как считают многие, один из самых используемых языков программирования в мире.

**Педагогическая целесообразность**

Данная дополнительная общеразвивающая программа направлена на развитие логического и пространственного мышления слушателя, способствует раскрытию творческого потенциала личности, формированию усидчивости и трудолюбия, приобретению практических умений и навыков в области компьютерных технологий, способствует интеллектуальному развитию ребенка.

**Отличительные особенности данной программы**

Основное количество часов отводится практическому написанию программ. Каждый обучающийся реализует индивидуальный проект в результате освоения программы. Продукт, полученный в результате освоения программы, имеет прикладной характер и может быть использован по необходимости.

**Уровень освоения программы** - базовый

**Возраст детей**, участвующих в реализации данной программы от 14 до 18 лет.

**Объем и срок освоения программы.**

Срок реализации программы – 1 год.

Общий объем - 144 часа.

**Наполняемость групп и особенности набора учащихся.**

Наполняемость группы составляет 12-15 человек (по количеству компьютеров).

Программа охватывает теоретический и практический блоки содержания.

В процессе реализации программы используются различные формы Занятий: традиционные, комбинированные, практические занятия, игры, праздники, конкурсы, и другие.

*Методы обучения*

Для достижения поставленной цели и реализации задач предмета используются следующие методы обучения:

- словесный (объяснение, беседа, рассказ);
- наглядный (показ, наблюдение, демонстрация приемов работы);
- практический;

- эмоциональный (подбор ассоциаций, образов, художественные впечатления).

При определении режима занятий учтены санитарно-эпидемиологические требования к организациям дополнительного образования детей. Структура каждого занятия зависит от конкретной темы и решаемых задач.

### **Режим занятий, периодичность и продолжительность.**

Занятия проводятся группой 2 раза в неделю, продолжительностью 2 часа.

Занятия в соответствии с требованиями СанПиНа 2.4.4.3172-14 разделены на академические часы (45 минут) с перерывами между ними по 10 минут.

В случае необходимости программа может быть реализована с применением электронного обучения и использованием дистанционных образовательных технологий.

### **Особенности организации образовательного процесса.**

Очная форма обучения предполагает организацию теоретических и практических занятий.

При дистанционном обучении педагог взаимодействует с обучающимися в следующих режимах:

- on-line (онлайн): педагог использует средства коммуникации, одновременно взаимодействуя с обучающимся (обучающимися);
- off-line (офлайн): самостоятельная работа обучающегося (обучающихся), с последующим оцениванием и определением рекомендаций.

## **1.2. Цель и задачи программы**

### **Цель программы**

Основной целью данного учебного курса является ознакомление слушателя с объектно-ориентированным языком программирования Python, с возможностями, синтаксисом языка, технологией и методами программирования в среде Python, обучение практическим навыкам программирования на языке Python для решения типовых задач математики и информатики, а также при разработке простейших игр.

Реализация поставленной цели предусматривает решение **следующих задач**:

*Образовательные:*

- развитие интереса учащихся к изучению программирования;
- знакомство учащихся с основами программирования в среде Python;
- овладение базовыми понятиями теории алгоритмов при решении математических задач;
- формирование навыков работы в системе программирования Python;
- формирование самостоятельности и творческого подхода к решению задач с помощью средств современной вычислительной техники;
- формирование навыков алгоритмического и логического мышления;
- формирование навыков грамотной разработки программ;
- формирование практических навыков решения прикладных задач;
- формирование практических навыков разработки игр.

*Развивающие:*

- развивать память и внимание, познавательную и творческую активность;
- развивать творческие способности, эстетическое и эргономическое восприятие объектов труда;
- развивать логическое мышление.

*Воспитательные:*

- прививать интерес к активному творческому самовыражению, культуре труда;
- воспитывать упорство в достижении желаемого результата;
- воспитывать эстетический вкус;
- воспитывать чувство взаимопомощи, доверия, коллективизма.

### 1.3. Учебно-тематический план и содержание программы

№ п/п	Наименование раздела, темы	Количество часов			Формы аттестации/контроля
		всего	теория	практика	
<b>1.</b>	<b>Введение в python</b>	<b>14</b>	<b>6</b>	<b>8</b>	
1.1.	Язык python. Среда Программирования. Особенности ввода-вывода	4	2	2	Практическое задание
1.2.	Типы данных, операции. Оператор присваивания	4	2	2	Выполнение упражнений
1.3.	Числа. Стандартные операции	6	2	4	Практическое задание
<b>2.</b>	<b>Алгоритмические инструкции</b>	<b>22</b>	<b>6</b>	<b>16</b>	
2.1.	Условный оператор	2	2	-	Опрос
2.2.	Цикл while. Теория чисел	10	2	8	Практическое задание
2.3.	Цикл for	10	2	8	
<b>3.</b>	<b>Строки</b>	<b>20</b>	<b>8</b>	<b>12</b>	
3.1.	Литералы строк	4	2	2	Практическое задание
3.2.	Срезы строк	4	2	2	Практическое задание
3.3.	Методы строк	12	4	8	Практическое задание
<b>4.</b>	<b>Функции</b>	<b>20</b>	<b>8</b>	<b>12</b>	
4.1.	Парадигма и преимущества структурного программирования	2	2	-	Опрос
4.2.	Граф и стек вызовов функций. области видимости переменных	4	2	2	Практическое задание
4.3.	Прямая рекурсия	10	2	8	Практическое задание
4.4.	Косвенная рекурсия	4	2	2	Решение задач повышенной трудности
<b>5.</b>	<b>Списки и кортежи</b>	<b>20</b>	<b>8</b>	<b>12</b>	
5.1.	Списки и кортежи в Python. сходства и различия	2	2	-	Опрос
5.2.	Операции со списками	10	2	8	Практическое задание
5.3.	Срезы списков	4	2	2	Практическое задание
5.4.	Матрицы. Операции над матрицами	4	2	2	Решение задач повышенной трудности
<b>6.</b>	<b>Словари и множества</b>	<b>8</b>	<b>4</b>	<b>4</b>	
6.1.	Словари	4	2	2	Практическое задание
6.2.	Множества	4	2	2	Практическое задание
<b>7.</b>	<b>Объектно- ориентированное</b>	<b>38</b>	<b>4</b>	<b>34</b>	

	<b>программирование</b>				
7.1.	Классы в Python.	2	2	-	Опрос
7.2.	Разработка собственного класса	6	2	4	Практическое задание
7.3.	Разработка и программирование собственного проекта	30	-	30	
<b>8</b>	<b>Заключительное занятие</b>	<b>2</b>		<b>2</b>	
8.1.	Подведение итогов. Индивидуальный проект	2		2	Защита проекта
		144	44	100	

## СОДЕРЖАНИЕ УЧЕБНОГО (ТЕМАТИЧЕСКОГО) ПЛАНА

### 1. Введение в Python

*Основные понятия:* трансляция, интерпретация, компиляция, синтаксис, семантика, прагматика, переменная, динамическая типизация, служебные слова, идентификаторы, простые типы данных, приоритеты операций, литералы чисел, операция присваивания, PEP 8.

**Язык Python. Среда программирования. Особенности ввода- вывода Теория.** Язык программирования Python. Достоинства и недостатки. Области применения. Интерактивный режим работы программы.

*Практика.* Установка языка программирования Python 3.5 и среды программирования WingIDE 100. Регистрация на Интернет-ресурсах.

#### **Типы данных, операции. Оператор присваивания**

*Теория.* Ввод и вывод числовой информации.

*Практика.* Тренировочное задание на ввод и вывод числовой информации. **Числа.**

#### **Стандартные операции**

*Теория.* Стандартные операции с целыми и действительными числами.

Стиль программирования Python (PEP 8).

*Практика.* Решение простых задач в интерактивном режиме.

### 2. Алгоритмические инструкции

*Основные понятия:* логический тип данных, логические операции (and, or, not,<sup>A</sup>), условный оператор, условное и альтернативное исполнение алгоритма, операторы сравнения, вложенность операторов, оператор цикла, переменная-флаг, генерация псевдослучайной последовательности, инструкции break, continue и pass.

#### **Условный оператор**

*Теория.* Условная и циклическая инструкции. Каскадность и вложенность алгоритмических инструкций.

#### **Цикл while**

*Теория.* Теория чисел. Фильтрация потока чисел. НОД и НОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума.

*Практика.* Решение задач на анализ чисел потока и целочисленной арифметики.

#### **Цикл for**

*Теория.* Теория чисел. Фильтрация потока чисел. НОД и НОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума.

*Практика.* Решение задач на анализ чисел потока и целочисленной арифметики повышенной трудности.

### 3. Строки

*Основные понятия:* символ, строка, литерал, таблицы кодов ASCII, UTF-8, отладка кода, неизменяемый объект, формат вывода строки, экранированные escape-

последовательности, положительная и отрицательная строке, срез, конкатенация, длина строки.

нумерация символов в

### **Литералы строк**

*Теория.* Понятие «литералы строк».

*Практика.* Ввод-вывод строки. Решение задач на ввод строки, поиск подстроки.

### **Срезы строк**

*Теория.* Форматирование строки.

*Практика.* Преобразование строки. Решение упражнений.

### **Методы строк**

*Теория.* Методы работы со строкой.

*Практика.* Применение методов строки. Решение задач.

## **4. Функции**

*Основные понятия:* подпрограмма, функция, процедура, рекурсия, глубина рекурсии, объявление, определение и вызов функции, возврат значений, глобальные и локальные переменные, передача параметров, работас памятью, граф вызовов, стек вызовов, полиморфизм функций, утиная типизация, lambda- функции.

### **Парадигма и преимущества структурного программирования**

*Теория.* Обзор парадигм программирования. Особенности применения языков программирования.

### **Граф и стек вызовов функций. Области видимости переменных**

*Теория.* Применение стек и граф вызовов функций.

*Практика.* Выполнение тренировочных упражнений.

### **Прямая рекурсия**

*Теория.* Понятие прямой рекурсии и ее применение.

*Практика.* Выполнение тренировочных упражнений.

### **Косвенная рекурсия**

*Теория.* Понятие косвенной рекурсии и ее применение.

*Практика.* Выполнение тренировочных упражнений.

*Практические занятия к темам 4.2.,4.3.,4.4.:* Нахождение суммы чисел. Числа Фибоначчи. Вычисление степени. Ханойские башни. Использование библиотеки математических функций. Решение задач повышенной трудности.

## **5. Списки и кортежи**

*Основные понятия:* список, кортеж, элемент списка и кортежа, индекс, срез списка, матрица, многомерный список, сортировка, сложность алгоритма, устойчивость сортировки, квадратичная, быстрая, синхронная, поразрядная сортировки списка, случайное перемешивание.

### **Списки и кортежи в Python. Сходства и различия**

*Теория.* Представление списка и кортежа в памяти компьютера, сходства и различия.

### **Операции со списками**

*Теория.* Способы заполнения списка (с клавиатуры, из файла, случайным образом, по формуле).

*Практика.* Решение задач на ввод-вывод элементов одномерного и многомерного списка.

### **Срезы списков**

*Теория.* Методы работы со списком и кортежем. Методы сортировки списка.

*Практика.* Решение задач на ввод-вывод элементов одномерного и многомерного списка и кортежа, преобразование, поиск, замену, подсчет.

### **Матрицы. Операции над матрицами**

*Теория.* Вычисление сложности алгоритма. Многомерные списки.

*Практика.* Решение задач повышенной трудности.

## **6. Словари и множества**

Основные понятия: *словарь, множество, ключ, кодирование.*

### **Словари**

**Теория.** Понятие словаря. Способы создания. Словарь, преимущества и недостатки, методы работы со словарем. Словари со смешанными значениями. Кодирование и декодирование текста.

**Практика.** Решение задач на заполнение, преобразование, поиск, замену, подсчет, вывод элементов словаря.

### **Множества**

**Теория.** Понятие множества. Создание множеств. Множество, преимущества и недостатки, методы работы с множеством.

**Практика.** Решение задач повышенной трудности.

## **7. Объектно-ориентированное программирование (ООП)**

**Основные понятия:** ООП, класс, метод INIT, экземпляр, наследование, полиморфизм, исключения, виджет, интерфейс, событие, техническое задание, проект, проектная деятельность, виды проектов.

### **Классы в Python**

**Теория.** Понятия «класс», «метод INIT», «экземпляр», «наследование», «полиморфизм», «исключения», «виджет», «интерфейс», «интерфейс», «событие».

### **Разработка собственного класса**

**Теория.** Принципы разработки собственного класса. Обработка и генерация исключений. Виджет, методы виджета. Графическая библиотека tkinter, класс Tk. Системные методы. События.

**Практика.** Создание собственного класса.

### **Разработка и программирование собственного проекта**

**Практика.** Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.

## **8. Подведение итогов**

### **Защита индивидуального проекта**

## **1.4. Планируемые результаты**

### **Ожидаемые результаты и способы определения их результативности**

По итогам реализации программы, учащиеся

#### **будут знать:**

- принципы программирования на языке Python;
- основы дизайнерского оформления созданных программ.

По итогам реализации программы, учащиеся

#### **будут уметь:**

- производить чтение и запись программ на языке Python;
- запускать и отлаживать программу.

## РАЗДЕЛ 2. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

### 2.1. Календарный учебный график

Год обучения	Объем учебных часов	Всего учебных недель	Режим работы	Кол-во учебных дней	Даты начала и окончания учебных периодов/ этапов	Продолжительность каникул
Первый	144	36	2 раза в неделю по 2 часа	72	С 01 сентября по 31 мая	С 1 по 10 января (10 дней)

### 2.2. Условия реализации программы

#### Методическое обеспечение.

##### Методы обучения:

- ✓ словесные методы (лекция, объяснение);
- ✓ демонстративно-наглядные (демонстрация работы в программе, схем, скриптов, таблиц);
- ✓ исследовательские методы;
- ✓ *активные* ( программированного и интерактивного обучения, метод проектов, игровой метод);
- ✓ работа в парах;
- ✓ работа в малых группах;
- ✓ проектные методы (разработка проекта по спирали творчества, моделирование, планирование деятельности),
- ✓ работа с Интернет-сообществом (публикация проектов в Интернет-сообществе скретчеров)

Практическая часть работы – работа в среде программирования со скриптами и проектирование информационных продуктов. Для наилучшего усвоения материала практические задания рекомендуется выполнять каждому за компьютером. При выполнении глобальных проектов рекомендуется объединять школьников в пары. Текущий контроль уровня усвоения материала осуществляется по результатам выполнения рефлексивных упражнений и практических заданий. Итоговый контроль осуществляется по результатам разработки проектов. Формы подведения итогов: презентация проекта, испытание квеста, игры.

Для успешной реализации программы используются следующие методические материалы:

- ✓ учебно-тематический план;
- ✓ календарно-тематический план;
- ✓ теоретический материал по изучаемым темам;
- ✓ инструкции по технике безопасности и правилам поведения в учреждении;
- ✓ справочники и переводчики в электронном виде;
- ✓ методическая литература для педагогов дополнительного образования.
- ✓ Дидактическое обеспечение программы включает в себя следующие материалы:
- ✓ учебные презентации по темам;
- ✓ материалы для практических и самостоятельных заданий;
- ✓ материалы для проведения конкурсных мероприятий.

**Форма организации образовательного процесса – групповая**

### **Формы проведения занятий:**

- ✓ занятие с использованием игровых технологий;
- ✓ занятие-игра;
- ✓ занятие-исследование;
- ✓ творческие практикумы (сбор скриптов с нуля);
- ✓ занятие-испытание игры;
- ✓ занятие-презентация проектов;
- ✓ занятие с использованием тренинговых технологий (работа на редактирование готового скрипта в соответствии с поставленной задачей).

### **Формы организации учебного занятия:**

Форма организации деятельности групповая, при этом отдельные вопросы и ошибки рассматриваются в индивидуальном порядке с каждым обучающимся, исходя из особенностей каждого обучающегося в усвоении пройденного материала.

Первая часть занятия предполагает получение обучающимся нового материала. Во время второй части занятия обучающийся пытается самостоятельно реализовать полученную теоретическую базу в рамках собственного проекта. Оценка результатов производится коллективно всей группой.

Также предусматривается самостоятельная работа обучающихся с литературой и в сети Интернет.

Некоторые занятия полностью отведены на реализацию проектной работы.

Общение на занятии ведётся в свободной форме — каждый обучающийся в любой момент может задать интересующий его вопрос без поднятия руки. Данный момент очень важен в процессе обучения, так как любой невыясненный вопрос, может превратиться в препятствие для получения обучающимся последующих знаний и реализации им собственных проектов.

### **Педагогические технологии, используемые в образовательной деятельности.**

- технология группового обучения;
- технология проблемного обучения;
- технология игровой деятельности;
- ИКТ;
- здоровьесберегающая технология.

### **Алгоритм учебного занятия.**

1. Мотивационный этап.
2. Объявление прогнозируемых (учебных) результатов.
3. Подготовка к работе ПО Python 3.5
4. Работа в среде Python 3.5.
5. Физкультминутка, которая проводится после 30 мин работы за ноутбуком
6. Редактирование скрипта.
7. Рефлексия.

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

- оборудованный кабинет на базе колледжа;
- столы (парты) – 12 шт.;
- стулья – 24 шт.;
- шкафы и стеллажи для хранения учебных пособий и дидактического материала – 1 шт.;
- интерактивный комплекс;
- ноутбуки – ученические 12 шт. + 1 ноутбук учителя;
- интернет.

### **Организационно-педагогические условия реализации программы**

Занятия проводятся в следующих формах: лекции, семинары, практические задания.

*Дидактический материал*, необходимый для проведения занятий:

- краткие конспекты материалов для лекций;
- распечатки заданий для практикумов;
- презентационные материалы для объяснения;
- карточки с индивидуальными заданиями.

*Техническое оснащение занятий:*

- компьютер для демонстрации презентаций;
- проектор;
- рабочие компьютеры учащихся для работы с доступом в Интернет;
- принтер для распечатки заданий

### **2.3. Формы контроля**

#### ***Формы отслеживания и фиксации образовательных результатов***

Диагностика результатов обучения осуществляется в несколько этапов:

- *вводный контроль* проводится с целью выявления образовательного уровня учащихся на начало обучения и способствует подбору эффективных форм и методов деятельности для учащихся;

- *промежуточный* - проводится в конце первого года обучения с целью отслеживания уровня освоения программного материала;

- *тематический контроль* позволяет определить степень сформированности знаний, умений и навыков по темам программы, это дает возможность своевременно выявлять пробелы в знаниях и оказывать учащимся помощь в усвоении содержания материала;

- *итоговый контроль* призван определить конечные результаты обучения

#### **Формы подведения итогов реализации программы**

Результаты обучения по программе выявляются по итогам проведения олимпиад, соревнований по программированию, защиты проекта.

### **2.4. Оценочные материалы**

Форма аттестации обучающихся по данной программе итоговая проектная работа.

Запланированы участия в конкурсах, результаты которых также являются оценочной единицей

По завершении изучения крупных тем или в конце учебного года целесообразно проведение нескольких занятий в форме конференции, где бы каждый ученик или группа учеников могли представить свою работу, по заинтересовавшей их тематике.

Оценочные материалы:

- сохраненный готовый проект в Python.
- готовый проект, удовлетворяющий требованиям текущей темы.

### **2.5. Методические материалы**

#### ***Дидактические и наглядные материалы***

- учебные пособия;
- видео-уроки;
- лабораторные работы;
- виртуальная доска
- готовые проекты, сделанные другими детьми ранее;

## СПИСОК ЛИТЕРАТУРЫ

1. Доусен М. Програмуем на Python / М. Доусен - СПб.: Питер, 2016. - 416с.
2. Лутц М. Изучаем Python, 4 издание / М. Лутц - СПб.: Символ- Плюс, 2011. - 1280 с.
3. Любанович Б. Простой Python. Современный стиль программирования / Б. Любанович. - СПб.: Питер, 2016. - 480с.
4. Прохоренок Н.А., Дронов В.А. Python 3 и PyQt 5. Разработка приложений / Н.А. Прохоренок, В.А. Дронов - СПб.: «БХВ- Петербург», 2016.
5. Саммерфильд М. Python на практике / М. Саммерфильд, пер. А.А. Слинкин – М.: ДМК-Пресс, 2014. - 338с.

### **Цифровые образовательные ресурсы**

1. Россум Г., Дж. Дрейк Ф.Л., Откидач Д.С., Задка М., Левис М., Монтаро С., Реймонд Э.С., Кучлинг А.М., Лембург М.-А., Йи К.-П., Ксиллаг Д., Петрилли Х.Г., Варсав Б.А., Ахлстром Дж.К., Роскинд Дж., Шеменор Н., Мулендер С. Язык программирования Python: [Электронный ресурс]. 2001. URL: <https://goo.gl/8TzY8w>.  
(Дата обращения: 27.08.2018).

### Диагностическая карта освоения обучающимся раздела программы

Название раздела, кол-во часов Ф.И.О. обучающегося

№ п/п	Показатели результативности освоения раздела	Оценка результативности освоения раздела		
		1 балл (низкий уровень)	2 балла (средний уровень)	3 балла (высокий уровень)
1.	Теоретические знания			
2.	Практические умения и навыки			
3.	Самостоятельность в познавательной деятельности			
4.	Потребность в самообразовании и саморазвитии			
5.	Применение знаний и умений в социально-значимой деятельности			
Общая сумма баллов:				

После оценки каждого параметра результативности освоения раздела, все баллы суммируются. На основе общей суммы баллов определяется общий уровень освоения модуля в соответствии с нижеприведенной шкалой:

- 1 – 4 балла – раздел освоен на низком уровне;
- 5 – 10 баллов – раздел освоен на среднем уровне;
- 11 – 15 баллов – раздел освоен на высоком уровне.

Информационная карта освоения разделов программы заполняется на основе результатов педагогического наблюдения, бесед, выполнения учащимися заданий на занятиях. Применение данной методики в долгосрочном периоде позволяет определить динамику личностного развития каждого учащегося.