

Муниципальное бюджетное общеобразовательное учреждение  
«Средняя общеобразовательная школа №2 г. Советский»

**Дополнительная общеобразовательная общеразвивающая программа «Основы  
программирования на языке Python»**

**Направленность: техническая**

Возраст обучающихся: 15 - 17 лет

Срок реализации: 1 год

Уровень: базовый

г. Советский  
2023 г.

## 1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования на языке Python» составлена с использованием учебно-методической и дополнительной (специальной) литературы по информатике, с учетом возрастных особенностей детей.

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования на языке Python» соответствует уровню дополнительного образования и имеет **техническую направленность**.

Программа представляет собой самостоятельный модуль, отвечает интересам детей, способствует углублению знаний, умений и навыков, приобретаемых в общеобразовательной школе на уроках информатики.

### **Актуальность программы.**

Компьютеры и компьютерные системы – неотъемлемая часть жизни нашего общества. Научившись программировать, мы можем быть не только пользователями информационных технологий, но и активными их создателями.

Умение составлять алгоритмы решения и навыки программирования являются элементами информационной компетенции. Умение находить решение, составлять алгоритм решения и реализовать его с помощью языков программирования — необходимое условие подготовки современных учащихся.

Учиться программировать очень интересно. Результат программирования виден сразу. Кроме того, создание компьютерных игр и обучающих программ способствует развитию логики и креативного мышления. Ещё одной значимой стороной обучения программированию является спрос на рынке труда на специалистов данного направления деятельности.

Программа служит средством внутрипрофильной специализации в области новых информационных технологий, что способствует созданию дополнительных условий для проявления индивидуальных образовательных интересов учащихся, их дальнейшей профессиональной ориентации.

### **Программа составлена в соответствии с**

- Федеральным Законом «Об образовании в Российской Федерации» от 29.12.2012 №273-ФЗ.
- Концепцией развития дополнительного образования детей на 2014-2020 гг. (Утверждена Распоряжением Правительства РФ № 1726-р 4 сентября 2014 г.).
- Санитарно-эпидемиологическими правилами и нормативами СанПиН 2.4.4.3172-14 (Зарегистрировано в Минюсте России 20 августа 2014 г. N 33660).
- Приказом Министерства просвещения РФ от 09 ноября 2018 г. № 196 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».
- Письмом Министерства образования и науки РФ от 18.11.2015 г. «О направлении Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)».
- Приказом Департамента образования и молодежной политики Ханты-Мансийского автономного округа – Югры от 26.05.2017 года № 871 «О внесении изменений в приказ Департамента образования и молодежной политики ХМАО-Югры от 04.08.2016 года № 1224 «Об утверждении Правил персонифицированного финансирования дополнительного образования детей в ХМАО-Югре».

### **Отличительные особенности программы:**

Отличительной особенностью Программы является её направленность на формирование у учащихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и его реализации с помощью языка программирования Python.

Основное количество часов отводится практическому написанию программ. В результате освоения программы, каждый обучающийся реализует индивидуальный проект, который имеет прикладной характер и может быть использован по необходимости.

Программирование мотивирует к занятиям в различных научных областях (физики, информатики, алгебры, геометрии и др.), развивает воображение и способствует ранней профориентации подростков. Учащиеся вспоминают свои знания по основам алгоритмизации и программирования и на их основе, углубляя их, учатся составлять простые и сложные 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.

**Цель Программы:** Способствовать формированию творческой личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python.

#### **Задачи Программы:**

##### *Образовательные:*

- обучить языку программирования Python и созданию программ на его основе;
- научить создавать прикладное программное обеспечение;
- расширять кругозор обучающихся в области программирования;
- научить дизайнерскому оформлению созданного ПО.

##### *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.

#### **Планируемые результаты освоения программы.**

В рамках программы «Основы программирования на языке Python» учащиеся достигают личностные, метапредметные и предметные результаты.

В основном формируются и получают развитие метапредметные результаты такие, как:

- умение самостоятельно планировать пути достижения целей, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;
- умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата;
- умение оценивать правильность выполнения учебной задачи, собственные возможности её решения;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;
- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;
- умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий (далее ИКТ-компетенции).

Вместе с тем делается существенный вклад в развитие личностных результатов, таких как:

- формирование ответственного отношения к учению;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками и взрослыми в процессе образовательной, творческой и других видов деятельности.

В части развития предметных результатов наибольшее влияние изучение курса оказывает:

- на формирование умений формализации и структурирования информации, умения выбирать способ представления данных в соответствии с поставленной задачей с использованием соответствующих программных средств обработки данных;
- формирование навыков и умений безопасного и целесообразного поведения при работе с компьютерными программами и в Интернете, умения соблюдать нормы информационной этики и права.

### **Раздел 1. Знакомство с языком Python**

*Учащиеся должны знать/понимать:*

- понятие программы;
- структуру программы на Python;
- режимы работы с Python.

*Учащиеся должны уметь:*

- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе.

### **Раздел 2. Переменные и выражения.**

*Учащиеся должны знать/понимать:*

- общую структуру программы;
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания. **Раздел 3. Условные предложения.**

*Учащиеся должны знать/понимать:*

- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;

*Учащиеся должны уметь:*

- использовать условный оператор;
- создавать сложные условия с помощью логических операторов.

### **Раздел 4. Циклы.**

*Учащиеся должны знать/понимать:*

- циклы с условием и их виды;
- правила записи циклов с условием;
- назначение и особенности использования цикла с параметром;
- формат записи цикла с параметром;
- примеры использования циклов различных типов.

*Учащиеся должны уметь:*

- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием;
- определять целесообразность применения и использовать цикл с параметром для решения поставленной задачи.

## **Раздел 5. Функции.**

*Учащиеся должны знать /понимать:*

- понятие функции; способы описания функции;
- принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- понятие формальных и фактических параметров подпрограмм;
- способ передачи параметров.

*Учащиеся должны уметь:*

- создавать и использовать функции;
- использовать механизм параметров для передачи значений.

## **Раздел 6. Строки - последовательности символов.**

*Учащиеся должны знать /понимать:*

- назначение строкового типа данных;
- операторы для работы со строками;
- процедуры и функции для работы со строками;
- операции со строками.

*Учащиеся должны уметь:*

- описывать строки; соединять строки;
- находить длину строки; вырезать часть строки;
- находить подстроку в строке;
- находить количество слов в строке.

## **Раздел 7. Сложные типы данных.**

*Учащиеся должны знать/понимать:*

- сложные типы данных;
- способ описания списка; способ доступа к элементам списка;
- способ описания кортежа; способ описания словаря;
- операции, выполняемые со списками, кортежами и словарями;
- понятие множества; способы описания множества; операторы работы с множествами.

*Учащиеся должны уметь:*

- описывать списки; вводить элементы списка; выводить элементы списка;
- выполнять поиск элемента в списке, поиск минимума и максимума, нахождение суммы элементов списка;
- использовать вложенные списки;
- приводить примеры использования вложенных списков (матриц);
- описывать множества;
- определять принадлежность элемента множеству;
- вводить элементы множества;
- выводить элементы множества.

## **Раздел 8. Стиль программирования и отладка программ (6 часов)**

*Учащиеся должны знать/понимать:*

- что такое стиль программирования;
- правила именования объектов;
- основные рекомендации при написании программ.

*Учащиеся должны уметь:*

- определять вид ошибок и находить ошибки в программе.
- выполнять тестирование и отладку программ.

## **Способы оценки результатов достижений учащихся.**

Предметом диагностики и контроля программы «Основы программирования на языке Python» являются внешние образовательные продукты учащихся (созданные блок-схемы, программы), а также их внутренние личностные качества (освоенные способы деятельности, знания, умения), которые относятся к целям и задачам Программы.

Качество внешней образовательной продукции оценивается по следующим параметрам:

- алгоритм должен быть оптимальным по скорости выполнения и максимально простым в реализации на языке программирования;
- программа должна выполнять поставленные задачи;
- по степени «читаемости кода» (должны быть соблюдены отступы, обязательное наличие комментариев к коду программы и т. д.).

Проверка достигаемых учащимися результатов производится в следующих формах:

- текущий рефлексивный самоанализ, контроль и самооценка учащимися выполняемых заданий;
- текущая диагностика и оценка педагогом деятельности учащихся;
- итоговая оценка деятельности и образовательной продукции ученика в соответствии с его индивидуальными и возрастными особенностями;
- итоговый контроль проводится в конце всего курса. Он состоит из тестирования и защиты творческого проекта.

## **2. Механизм реализации Программы.**

**Форма обучения** - очная.

**Срок реализации:** 1 модуль/1 год.

**Режим работы:** занятия по программе проводятся 1 раз в неделю по 2 академических часа/70 часов в год

**Форма проведения учебных занятий:** по группам.

**Характеристика групп:**

Программа рассчитана на детей 15–17 лет, проявляющих интерес к информационно-коммуникационным технологиям. Объединение комплектуется на основании заявлений законных представителей учащихся (самих учащихся с 14 лет). Группы формируются из школьников разного возраста на добровольной внеконкурсной основе.

В группе от 10 до 15 человек, состав группы постоянный.

Программа реализуется **без предъявления требований** к уровню образования и квалификации педагога.

## **Методическое обеспечение программы.**

**Методы обучения:**

1. Словесный: объяснение нового материала; рассказ обзорный для раскрытия новой темы; беседы с учащимися в процессе изучения темы.
2. Наглядный: применение демонстрационного материала, наглядных пособий, презентаций по теме.
3. Практический: индивидуальная и совместная продуктивная деятельность, выполнение учащимися определенных заданий, решение задач.
4. Интерактивный: создание специальных заданий, моделирующих реальную жизненную ситуацию, из которой учащимся предлагается найти выход.

**Технологии:**

1. Технология проблемного диалога. Учащимся не только сообщаются готовые знания, но и организуется такая их деятельность, в процессе которой они сами делают «открытия», узнают что-то новое и используют полученные знания и умения для решения жизненных задач.

2. Технология коллективного взаимообучения («организованный диалог», «сочетательный диалог», «коллективный способ обучения», «работа учащихся в парах сменного состава») позволяет плодотворно развивать у обучаемых самостоятельность и коммуникативные умения.
3. Игровая технология. Игровая форма в образовательном процессе создаётся при помощи игровых приёмов и ситуаций, выступающих как средство побуждения к деятельности. Способствует развитию творческих способностей, продуктивному сотрудничеству с другими учащимися. Приучает к коллективным действиям, принятию решений, учит руководить и подчиняться, стимулирует практические навыки, развивает воображение.
4. Элементы здоровьесберегающих технологий являются необходимым условием снижения утомляемости и перегрузки учащихся.
5. Проектная технология предлагает практические творческие задания, требующие от учащихся их применение для решения проблемных заданий, знания материала на данный исторический этап. Овладевая культурой проектирования, школьник приучается творчески мыслить, прогнозировать возможные варианты решения стоящих перед ним задач.
6. Информационно-коммуникационные технологии активизируют творческий потенциал учащихся; способствует развитию логики, внимания, речи, повышению качества знаний; формированию умения пользоваться информацией, выбирать из нее необходимое для принятия решения, работать со всеми видами информации, программным обеспечением, специальными программами и т.д.

В ходе реализации программы применяется дифференцированный, индивидуальный подход к каждому обучающемуся.

**Материально-техническое оснащение:**

- персональные компьютеры – 12 рабочих станций;
- рабочее место учителя – персональный компьютер, принтер, звуковые колонки;
- мультимедийный проектор;
- интерактивная доска;
- маркерная доска;
- канцтовары.

**Информационное обеспечение:**

- специальная и научно-популярная литература для педагога и учащихся;
- видеоматериалы разной тематики по программе;
- выход в сеть Internet.

**Аппаратное обеспечение:**

- Процессор не ниже Core2 Duo;
- Объем оперативной памяти не ниже 4 ГбDDR3;
- Дисковое пространство на менее 128 Гб;

**Программное обеспечение:**

- Операционная система Windows 7 Профессиональная или выше;
- Интерпретатор Python версии 3.7 и выше;
- Пакет офисных программ;
- AdobePhotoshop или другой растровый графический редактор;
- Любой браузер для интернет серфинга.

**Кадровое обеспечение:** педагог.

## 5. Учебный план.

№ п/п	Название раздела, темы	Количество часов.			Формы аттестации/контроля
		Всего	Теория	Практика	
1	Раздел 1. Знакомство с языком Python	4	2	2	Входной контроль (Тестирование)
2	Раздел 2. Переменные и выражения	8	4	4	Тестирование, решение практических задач.
3.	Раздел 3. Условные предложения	10	2	8	
4.	Раздел 4. Циклы	14	2	12	
5.	Раздел 5. Функции	10	2	8	
6.	Раздел 6. Строки - последовательности символов	6	2	4	
7.	Раздел 7. Сложные типы данных	12	4	8	
8	Раздел 8. Стиль программирования и отладка программ	6	2	4	Презентация проекта, рефлексия.
		<b>70</b>	<b>20</b>	<b>50</b>	

## 4. Содержание учебного плана.

### Раздел 1. Знакомство с языком Python(4 часов)

**Теория:** Инструктаж по ТБ. Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программы на языке Python. Комментарии.

**Практика:**

Практическая работа 1.1. Установка программы Python

Практическая работа 1.2. Режимы работы с Python

Тест № 1. Знакомство с языком Python

### Раздел 2. Переменные и выражения (8 часов)

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

Выражения. Операции.Порядок выполнения операций. Композиция.

Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.

**Практика:**

Практическая работа 2.1. Работа со справочной системой

Практическая работа 2.2. Переменные

Практическая работа 2.3. Выражения

Практическая работа 2.5. Задачи на элементарные действия с числами



Тест № 2. Выражения и операции.

### **Раздел 3. Условные предложения (10 часов)**

**Теория:** Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.

#### **Практика:**

Практическая работа 3.1. Логические выражения

Практическая работа 3.2. Условный оператор

Практическая работа 3.3. Множественное ветвление

Практическая работа 3.4. Условные операторы

Самостоятельная работа № 1. Решение задач по теме «Условные операторы».

Зачетная работа № 1. «Составление программ с ветвлением».

Тест № 3. "Условные операторы".

### **Раздел 4. Циклы (14 часов)**

**Теория:** 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. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Случайные числа. Функция randrange. Функция random. Примеры решения задач с циклом.

#### **Практика:**

Практическая работа 4.1. Числа Фибоначчи.

Практическая работа 4.2. Решение задачи с циклом for.

Практическая работа 4.3. Реализация циклических алгоритмов

Практическая работа 4.4. Случайные числа

Практическая работа 4.5. Решение задач с циклом.

Самостоятельная работа № 2 "Составление программ с циклом"

Тест № 4. Циклы

Творческая работа № 1. "Циклы"

### **Раздел 5. Функции (10 часов)**

**Теория:** Создание функций. Параметры и аргументы. Локальные и глобальные переменные. Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda. Примеры решения задач с использованием функций.

Рекурсивные функции. Вычисление факториала. Числа Фибоначчи.

#### **Практика:**

Практическая работа 5.1. Создание функций

Практическая работа 5.2. Локальные переменные

Практическая работа 5.3. Решение задач с использованием функций

Практическая работа 5.4. Рекурсивные функции

Самостоятельная работа № 3 по теме «Функции»

Тест № 5. Функции

### **Раздел 6. Строки - последовательности символов (6 часов)**

**Теория:** 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.

**Практика:**

Практическая работа 6.1. Строки

Практическая работа 6.2. Решение задач со строками.

**Раздел 7. Сложные типы данных (12 часов)**

**Теория:** Списки. 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. Генераторы списков в Python.

Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения

Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы.

Множества в языке Python. Множества. Множественный тип данных. Описание множеств.

Операции, допустимые над множествами: объединение, пересечение, разность, включение.

Оператор определения принадлежности элемента множеству.

**Практика:**

Практическая работа 7.1. Списки.

Практическая работа 7.2. Решение задач со списками.

Тест № 7. Списки

**Раздел 8. Стиль программирования и отладка программ (6 часов)**

**Теория:** Стиль программирования. Отладка программ.

**Практика:** Защита проекта «Программирование на языке Python»

### 5. Календарный учебный график.

№п/п	Месяц	Число	Время проведения занятия	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля
<b>Раздел 1. Знакомство с языком Python</b>								
1	сентябрь	Неделя 1	по расписанию	Комбинированная	2	Инструктаж по технике безопасности в компьютерном классе. Общие сведения о языке Практическая работа: «Установка программы Python»	Кабинет	
2	сентябрь	Неделя 2	по расписанию	Комбинированная	2	Режимы работы. Практическая работа: «Режимы работы с Python» Тест №1. Знакомство с языком Python	Кабинет	
<b>Раздел 2. Переменные выражения</b>								
3	сентябрь	Неделя 3	по расписанию	Комбинированная	2	Переменные Практическая работа: «Работа со справочной системой» Практическая работа: «Переменные»	Кабинет	
4	сентябрь	Неделя 4	по расписанию	Комбинированная	2	Выражения Практическая работа: «Выражения»	Кабинет	
5	октябрь	Неделя 1	по расписанию	Объяснение нового материала	2	Ввод и вывод	Кабинет	
6	октябрь	Неделя 2	по расписанию	Комбинированная	2	Задачи на элементарные	Кабинет	

				ная		действия с числами Практическая работа «Задачи на элементарные действия с числами» Тест № 2. Выражения и операции		
<b>Раздел 3. Условные предложения</b>								
7	октябрь	Неделя 3	по расписанию	Комбинированная	2	Логические выражения и операторы. Практическая работа: «Логические выражения»	Кабинет	
8	октябрь	Неделя 4	по расписанию	Комбинированная	2	Условный оператор Практическая работа: «Условный оператор»	Кабинет	
9	ноябрь	Неделя 1	по расписанию	Комбинированная	2	Множественное ветвление Практическая работа: «Множественное ветвление»	Кабинет	
10	ноябрь	Неделя 2	по расписанию	Комбинированная	2	Реализация ветвления в языке Python. Практическая работа: «Условные операторы» Самостоятельная работа №1 по теме «Условные операторы».	Кабинет	
11	ноябрь	Неделя 3	по расписанию	Комбинированная	2	Зачетная работа №1. "Составление программ ветвлением". Тест №3."Условные операторы".	Кабинет	
<b>Раздел 4. Циклы</b>								
12	ноябрь	Неделя 4	по расписанию	Комбинированная	2	Оператор цикла с	Кабинет	

				ная		условием. Практическая работа "Числа Фибоначчи"		
13	декабрь	Неделя 1	по расписанию	Комбинированная	2	Оператор цикла for Практическая работа «Решение задачи с циклом for.»	Кабинет	
14	декабрь	Неделя 2	по расписанию	Комбинированная	2	Вложенные циклы Практическая работа «Реализация циклических алгоритмов»	Кабинет	
15	декабрь	Неделя 3	по расписанию	Комбинированная	2	Случайные числа Практическая работа «Случайные числа»	Кабинет	
16	декабрь январь	Неделя 4 Неделя 1	по расписанию	Комбинированная	4	Примеры решения задач с циклом Практическая работа «Решение задач с циклом» Самостоятельная работа №2 "Составление программ с циклом". Тест №4. Циклы	Кабинет	
17	январь	Неделя 2	по расписанию	Практическая	2	Творческая работа №1. "Циклы"	Кабинет	
<b>Раздел 5. Функции</b>								
18	январь	Неделя 3	по расписанию	Комбинированная	2	Создание функций Практическая работа «Создание функций»	Кабинет	
19	январь	Неделя 4	по расписанию	Комбинированная	2	Локальные переменные Практическая работа «Локальные	Кабинет	

						переменные»		
20	февраль	Неделя 1 Неделя 2	по расписанию	Комбинированная	4	Примеры решения задач с использованием функций Практическая работа «Решение задач с использованием функций» Самостоятельная работа № 3 по теме "Функции"	Кабинет	
21	февраль	Неделя 3	по расписанию	Комбинированная	2	Рекурсивные функции Практическая работа 5.4. «Рекурсивные функции» Тест № 5. Функции	Кабинет	
<b>Раздел 6. Строки - последовательности символов</b>								
22	февраль	Неделя 4	по расписанию	Объяснение нового материала	2	Строки	Кабинет	
23	март	Неделя 1	по расписанию	Комбинированная	2	Срезы строк Практическая работа «Строки»	Кабинет	
24	март	Неделя 2	по расписанию	Комбинированная	2	Примеры решения задач со строками. Практическая работа «Решение задач со строками.»	Кабинет	
<b>Раздел 7. Сложные типы данных</b>								
25	март	Неделя 3	по расписанию	Объяснение нового материала	1	Списки	Кабинет	
26	март	Неделя 3 4	по расписанию	Комбинированная	2	Срезы списков. Практическая работа «Списки»	Кабинет	

27	Март апрель	Неделя 4 Неделя 1	по расписанию	Комбинированная	2	Списки: примеры решения задач. Практическая работа «Решение задач со списками»	Кабинет	
28	апрель	Неделя 12	по расписанию	Комбинированная	2	Матрицы. Тест №6 Списки	Кабинет	
29	апрель	Неделя 23	по расписанию	Объяснение нового материала	2	Кортежи	Кабинет	
30	апрель	Неделя 3	по расписанию	Объяснение нового материала	1	Введение в словари	Кабинет	
31	апрель	Неделя 4	по расписанию	Объяснение нового материала	2	Множества в языке Python	Кабинет	
<b>Раздел 8. Стил программирования и отладка программ</b>								
32	май	Неделя 1	по расписанию	Практическая	1	Стил программирования	Кабинет	
33	май	Неделя 1	по расписанию	Практическая	1	Отладка программ	Кабинет	
34	май	Неделя 2	по расписанию	Практическая	2	Зачет по курсу «Программирование на языке Python»	Кабинет	
35	май	Неделя 3	по расписанию	Комбинированная	2	Что дальше?	Кабинет	
					70			

## **6. Список литературы.**

### ***Нормативно-правовые акты и документы***

1. Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 № 273-ФЗ
2. Приказ Министерства образования и науки РФ от 29.08.2013 №1008 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».
3. Письмо Министерства образования Российской Федерации от 18 июня 2003 №28-02-484/16 «Требования к содержанию и оформлению образовательных программ дополнительного образования детей»,
4. Постановление Главного государственного санитарного врача Российской Федерации от 4.07.2014 г. «Об утверждении СанПиН 2.4.4.3172-14 «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей».
5. Стратегия инновационного развития Российской Федерации на период до 2020 года. Распоряжение правительства Российской Федерации от 08.12.2011 № 2227-р.
6. Концепция развития дополнительного образования детей. Распоряжение Правительства Российской Федерации от 24 апреля 2015 года № 729-р.
7. Стратегия развития воспитания в Российской Федерации на период до 2025 года». Распоряжение Правительства Российской Федерации от 29 мая 2015 года № 996-р.

### ***Литература для педагога***

1. Лутц, Марк Python. Справочник. – М.: Вильямс, 2015.
2. Официальный сайт программы [Электронный ресурс] – Режим доступа: <https://docs.python.org/>, свободный.
3. Сайт, среда разработки для языка Python. [Электронный ресурс] – Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu>, свободный.
4. Язык Python. [Электронный ресурс] – Режим доступа: <http://kpolyakov.spb.ru/school/probook/python.htm>, свободный.

### ***Литература для родителей и учащихся***

1. Сайт / справочные материалы [Электронный ресурс] – Режим доступа: <https://metanit.com/python/>, свободный.
2. Сайт / интерактивный сборник задач для практики программирования [Электронный ресурс] – Режим доступа: <http://pythontutor.ru/>, свободный.
3. Сайт / Адаптивный тренажер Python [Электронный ресурс] – Режим доступа: <https://stepik.org/course/431>, свободный.
4. Сайт / среда разработки для языка Python [Электронный ресурс] – Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu>, свободный.