

**Программа  
внеурочной деятельности «Поколение Python»  
для обучающихся 5-11 классов**

Составитель: Корепанова Э.П.,  
учитель информатики

**Раздел 1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**1.1. Актуальность программы**

По результатам исследования компании Jet Brains (<https://www.jetbrains.com/ru-ru/lp/devecosystem-2019/>) язык Python – самый изучаемый в 2019 году и один из самых востребованных на рынке труда. Python достаточно широко используется при изучении основ алгоритмизации и программирования в школьном курсе информатики. В частности, используется в качестве базового языка для изучения в УМК по информатике авторов Поляков К. Ю., Еремин Е. А., допущенном к использованию в общеобразовательных организациях Министерством просвещения РФ.

Изучение Python в школе откроет ученикам возможности дальнейшего развития в области IT и поможет профориентации в старших классах, пригодится в олимпиадах по программированию и решению заданий ЕГЭ.

Курс состоит из двух разделов:

- для начинающих;
- для продвинутых.

Курс предполагает смешанный формат обучения. Сочетание групповой работы с учителем в классе и индивидуальной работы в личном кабинете на онлайн-платформе Stepik.org позволяет ученикам выработать не только технические навыки программирования, но и навыки социального взаимодействия при работе над финальным проектом курса, а главное – научиться самостоятельно выстраивать свое профессиональное развитие.

**Цель реализации программы** – формирование у обучающихся навыков программирования на языке Python и самонаправленного обучения; изучение дополнительных возможностей этого языка и погружение в рабочие процессы команды программистов-разработчиков.

**Направленность дополнительной общеобразовательной программы:** техническая.

**Задачи реализации программы:**

**Обучающие**

1. Изучить основы программирования на языке Python;
2. Изучить дополнительные возможности языка Python

**Развивающие**

3. Научиться применять полученные знания для решения практических задач.

**Воспитательные**

4. Научиться применять полученные знания для решения практических задач и создания собственных проектов.
5. Повысить уровень самостоятельности в обучении.
6. Научиться декомпозировать задачи и работать над ними в небольших командах, с разделением ответственности и ролей.

#### **Методические материалы курса состоят из:**

1. Методических указаний для учителя в текстовом виде,
2. Презентации с иллюстративным изложением теоретического материала;
3. Упражнений на платформе Stepik с теоретическим и практическим материалом языка Python;
4. Интерактивных проверочных заданий в приложениях Kahoot и Learning Apps;
5. Ролевых игр, для закрепления знаний, полученных на занятии.
6. Инструкций для проведения рефлексии процесса обучения с учениками.

#### **Планируемые результаты обучения:**

##### **Личностные**

По окончании курса учащийся сможет:

- Программировать на языке Python, применяя дополнительные возможности языка.
- Использовать инструменты разработки среды Wing/PyCharm.
- Самостоятельно реализовывать проекты, связанные с разработкой игр.
- Самостоятельно реализовывать проекты с использованием сторонних API и простых графических интерфейсов.

##### **Метапредметные**

- Ставить учебные цели.
- Формулировать достигнутый результат.
- Планировать свою самостоятельную учебно-познавательную деятельность; выбирать индивидуальную траекторию достижения учебной цели.
- Определять подходы и методы для достижения поставленной цели.
- Отбирать необходимые средства для достижения поставленной цели.
- Оценивать промежуточные и финальные результаты своей самостоятельной учебно-познавательной деятельности.
- Распределять задачи в команде. Расставлять приоритеты, улучшать командную работу.

##### **Предметные**

Раздел 1. Практическим результатом работы служит финальный проект каждого ученика: 2D игра на движке Pygame, либо серия самостоятельно разработанных мини-проектов в консольном режиме языка Python.

Раздел 2:

- игра написанная с помощью черепашей графики (библиотека turtle);
- проект по созданию мемов (библиотека pillow);
- телеграмм-бот (библиотека telebot)

Основные формы организации занятий программы «Python для начинающих»:

- Практические занятия с использованием онлайн-платформы Stepik;
- Работа в IDE Wing, PyCharm;
- Прохождение опросов в приложениях Kahoot и Learning Apps;
- Домашние практические занятия с использованием онлайн-платформы Stepik, направленные на отработку навыков программирования на языке Python.

**Категория обучающихся:** ученики общеобразовательных школ от 12 до 17 лет в рамках внеурочной деятельности и дополнительного образования.

**Форма обучения:** смешанная: очное и онлайн-обучение. В очных занятиях в группах по 10-15 человек используются задания на онлайн-платформе с автоматизированной проверкой. Задания на платформе доступны ученикам для самостоятельного изучения в любое время.

В ходе курса запланирован плавный переход к формату “перевернутого класса”: на первых пяти занятиях учеников знакомят с онлайн-платформой и основными инструментами программирования на Python. Начиная с шестого урока предлагаются различные формы самостоятельной активности на уроке, с групповой взаимопроверкой. После десятого урока иногда ученикам предлагается самостоятельно изучить новую тему, опираясь на конспект онлайн-курса, а роль учителя сдвигается к консультации и модерации. Последние пять уроков отведены на почти самостоятельную работу над проектами.

Во второй части курса ученикам предложены два индивидуальных проекта, во время работы с которыми они познакомятся с системой трекинга подзадач в командах разработчиков. В финале курса их ожидает групповой проект - создание и наполнение собственного чат-бота с API Telegram.

**Срок освоения программы:** в течение одного учебного года, в объеме 102 часа.

Реализация программы предполагает проведение аудиторных занятий с использованием вариативных дистанционных ресурсов:

- авторских онлайн уроков, размещенных на платформе Stepik (Белых Роман, Гув Тимур, Скобина Ольга);
- авторских интерактивных заданий (опросов, викторин, дидактических игр и др.), размещенных на бесплатных специализированных сервисах Kahoot и Learning Apps;
- авторских проектных заданий для создания учащимися игр на языке Python (Белых Роман, Гув Тимур, Скобина Ольга).

## Раздел 2. Содержание программы

В рамках смешанного подхода к обучению каждая из тем, изложенных преподавателем, отрабатывается в уроках платформы Stepik и в практических заданиях.

(<https://stepik.org/course/58852/syllabus>, <https://stepik.org/course/68343/syllabus>).

## 2.1. Учебный (тематический) план

№	Название темы	Количество часов	Теоретические	Практические
1.	Знакомство с Python. Команды input() и print().	2	1	1
2.	Параметры sep, end. Переменные. Комментарии. PEP 8	2	1	1
3.	Работа с целыми числами	2	1	1
4.	Условный оператор. Логические операции and, or, not	2	1	1
5.	Вложенный и каскадный условный оператор	2	1	1
6.	Типы данных int, float, str. Встроенные функции min(), max(), abs(). Оператор in.	2	1	1
7.	Цикл for. Функция range().	2	1	1
8.	Частые сценарии при написании циклов. Расширенные операторы присваивания.	2	1	1
9.	Цикл с предусловием while	2	1	1
10.	Операторы break, continue, else.	2	1	1
11.	Вложенные циклы	2	1	1
12.	Строковый тип данных: индексация и срезы	2	1	1
13.	Методы строк	2	1	1
14.	Введение в списки.	2	1	1
15.	Основы работы со списками. Методы списков	2	1	1
16.	Вывод элементов списка. Строковые методы split() и join()	2	1	1
17.	Методы списков. Списочные выражения	2	1	1
18.	Функции	2	1	1
19.	Локальные и глобальные переменные. Функции возвращающие значения.	2	1	1
20.	Функции возвращающие значения.	2	1	1
21.	Работа над проектом	2	1	1
22.	Работа над проектом	4	0	4

23.	Тип данных bool и NoneType	2	1	1
24.	Вложенные списки	2	1	1
25.	Матрицы	2	1	1
26.	Итоговая работа на вложенные списки и матрицы	2	1	1
27.	Кортежи	2	1	1
28.	Множества в математике	2	1	1
29.	Множества в Python	2	1	1
30.	Методы работы с множествами. Часть 1	2	1	1
31.	Методы работы с множествами. Часть 2. Генераторы и замороженные множества	2	1	1
32.	Словари	2	1	1
33.	Методы работы со словарями	2	1	1
34.	Модуль random	2	1	1
35.	Модули decimal и fractions	2	1	1
36.	Модуль “Черепашка”. Часть 1	2	1	1
37.	Модуль “Черепашка”. Часть 2	2	1	1
38.	Проект с черепашьей графикой	2	0	2
39.	Функции с именованными аргументами и с произвольным числом аргументов	2	1	1
40.	Парадигмы программирования, введение в функциональное программирование. Функции как объекты	2	1	1
41.	Функции высшего порядка. Функции map(), filter(), reduce()	2	1	1
42.	Анонимные функции	2	1	1
43.	Встроенные функции any(), all(), zip(), enumerate()	2	0	2
44.	Проект с обработкой графики средствами библиотек “Мемогенератор”	2	0	2
45.	Файловый ввод и вывод, работа с текстовыми файлами	2	1	1
46.	Работа с текстовыми файлами	2	1	1

47.	Групповой проект “Чат-бот”: формирование команд	2	1	1
48.	Командная работа над проектом	4	0	4
49.	Презентация итогового проекта	2	0	2
	Итого	102		

## Раздел 3. Организационно-педагогические условия реализации программы

### 3.1. Литература для педагога

1. Васильев, А. Н. Python на примерах [Текст]: практ. курс /А. Н. Васильев - Наука и Техника, 2019 - 432 с.
2. Прохоренок, Н. А. Python 3: самое необходимое [Текст]: практ. курс / Н. А. Прохоренок, В. А. Дронов - БХВ-Петербург, 2019 - 608 с.
3. Гэддис, Т. Начинаем программировать на Python [Текст]: учебник/Т. Гэддис - БХВ-Петербург, 2019 - 768 с.
4. Седжвик, Р. Программирование на языке Python /Р. Седжвик, К. Уэйн, Р. Дондеро - Вильямс, 2017 - 736 с.
5. Харрисон, М. Как устроен Python. [Текст]: практ. курс / М. Харрисон - Питер, 2002 - 272 с.

Литература для обучающихся: Не предусмотрена

Электронные ресурсы:

1. Курс [Поколение Python: курс для начинающих](https://stepik.org/course/58852/syllabus) на платформе Stepik. (<https://stepik.org/course/58852/syllabus>)
2. Курс [Поколение Python: курс для продвинутых](https://stepik.org/course/68343/syllabus) на платформе Stepik. (<https://stepik.org/course/68343/syllabus>)

### 4.2. Материально-технические условия реализации программы

#### 1. Обязательные

- помещение;
- компьютеры;
- проектор, аудиоколонки;
- Интернет-соединение, скорость загрузки не менее 2 Мбит/сек;
- магнитно-маркерная доска или флипчарт;
- общие условия в соответствии с СанПиН 2.4.4.3172-14

Требования к ПО:

- Операционная система Windows 10 или моложе / MacOS / Unix-based системы с поддержкой протокола HTML5;
- Приложения Google Chrome, Gimp, Brackets;
- интерактивная оболочка (бесплатная IDE Wing, PyCharm).