

СОГЛАСОВАНО
Педагогическим советом
МАОУ гимназии № 99
(протокол от 29 августа 2025 г. № 1)

УТВЕРЖДЕНО
приказом МАОУ гимназии № 99
от 30 августа 2025 г. № 86-од

**Дополнительная общеобразовательная
общеразвивающая программа
«Основы программирования на Python»
Возраст обучающихся: 13-14 лет
Срок реализации: 1 год**

Автор-составитель:
Коржавина Анна Денисовна,
педагог дополнительного образования

**Екатеринбург
2025**

1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа разработана на основе следующих нормативных документов:

- Федеральный закон от 29.12.2012 г. №273-ФЗ «Об образовании в Российской Федерации», статьи: 2 (п.9, п.14, п.15, п.22), 12 (п.1 ч.4; ч.5), 13 (ч.1,2), 28 (п.6 ч.3), 33 (ч.2), 55 (п.2 ч.1), 58 (ч.1), 75 (ч.1, ч.4);
- Приказ Министерства просвещения РФ от 09.11.2018 г. №196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- СанПин 2.4.4.3172-14, утверждённый Постановлением Главного государственного санитарного врача РФ от 04.07.2014 г. №41;
- Концепция развития дополнительного образования детей (Распоряжение Правительства Российской Федерации от 4 сентября 2014 года №1726-р);
- Концепция развития и воспитания в Российской Федерации на период до 2025 года (утв. Распоряжением Правительства РФ от 28.05.2015 г. №996-р);
- Федеральный проект «Цифровая экономика Российской Федерации» (01.01.2019 – 31.12.2024);
- Концепция развития регулирования отношений в сфере технологий искусственного интеллекта и робототехники на период до 2024 года (Распоряжение Правительства Российской Федерации от 19 августа 2020 г. № 2129-р).

Направленность общеразвивающей программы: техническая.

Актуальность

Вовлечение населения Свердловской области в систематическое изучение программирования является ключевым вопросом, от успешности решения которого зависит эффективность реализации государственной политики в сфере информационных технологий и достижение запланированных показателей социально-экономического развития нашего региона на долгосрочную перспективу.

Огромное значение для профессионального роста имеют систематические занятия программированием. Общеизвестно, что это - верный путь к успешной карьере, интеллектуальному развитию, источнику жизненной энергии, творческой активности и высокой работоспособности.

Python — один из популярных языков программирования в образовательных программах. Он включён в учебные занятия, широко практикуется во внеклассной и внешкольной работе — это занятия в кружках по программированию, хакатоны, конкурсы и другие мероприятия.

Чтобы программировать на Python, необходимо уметь логически мыслить, решать задачи, разрабатывать алгоритмы и писать код. Занятия программированием улучшают работу мозга, развивают аналитическое мышление, укрепляют навыки решения проблем и повышают креативность. Постоянное взаимодействие с кодом способствует улучшению

концентрации, точности и внимательности. Развивается способность к быстрому принятию решений и адаптации к новым задачам. Программирование требует от занимающихся максимального проявления интеллектуальных возможностей, волевых усилий и умения пользоваться приобретёнными знаниями. Проявляются положительные эмоции: удовлетворение от решения задач, желание учиться и развиваться. Развивается чувство ответственности, коллективизма, скорость принятия решений. Благодаря своей интеллектуальной насыщенности программирование представляет собой средство не только профессионального развития, но и активного умственного отдыха.

В программе представлены доступные для обучающихся упражнения, способствующие овладению элементами синтаксиса и семантики языка Python, а также развитию навыков программирования.

Дополнительная общеобразовательная (общеразвивающая) программа «Основы программирования на Python» может рассматриваться как одна из ступеней к формированию культуры информационной грамотности и неотъемлемой частью всего воспитательно-образовательного процесса. Основная идея программы заключается в мотивации обучающихся на ведение интеллектуально насыщенной жизни, в формировании потребности в постоянном обучении и развитии как необходимого условия социального благополучия и успешности человека.

Адресат программы:

Программа «Основы программирования на Python» рассчитана на обучающихся 14-15 лет. Количество обучающихся в группе от 15 до 25 человек.

Режим занятий:

Занятия проводятся 1 раз в неделю по 2 часа.

Объем программы – 76 часов.

Срок освоения программы – 1 год.

Программа предполагает углубленный уровень освоения основ программирования на языке Python. Она направлена на развитие логического мышления, формирование навыков программирования и решение задач с использованием Python.

Перечень форм обучения: индивидуальные и групповые занятия.

Перечень видов занятий: мастер-классы, групповые занятия.

Перечень форм подведения итогов реализации программы: проектные работы, тестирование.

Цель программы: развитие навыков программирования и логического мышления у обучающихся посредством изучения языка Python.

Задачи программы:

Образовательные:

- Ознакомиться с основами программирования на языке Python;
- Научиться писать простые программы и решать задачи с использованием Python.

Развивающие:

- Развивать логическое мышление и навыки решения задач;
- Способствовать развитию аналитических способностей и креативного мышления.

Воспитательные:

- Воспитывать чувство ответственности и самостоятельности;
- Прививать интерес к программированию и информационным технологиям;
- Пропагандировать культуру использования современных технологий.

2. Учебный план

№ п/п	Название раздела, темы	Количество часов			Формы аттестации /контроля
		Всего	Тео рия	Прак тика	
	Тема занятия				
	Введение в программирование	6	4	2	
1.1.	Основные понятия программирования	3	2	1	Тест
1.2.	Установка и настройка среды разработки	3	2	1	Практическая работа
	Основы Python	13	6	7	
2.1.	Синтаксис и структуры данных	4	2	2	Практическая работа
2.2.	Переменные и типы данных	4	2	2	Практическая работа
2.3.	Операторы и выражения	5	2	3	Практическая работа
	Управляющие конструкции	10	4	6	
3.1.	Условные операторы	5	2	3	Тест
3.2.	Циклы	5	2	3	Практическая работа
	Функции и модули	10	4	6	
4.1.	Определение и использование функций	5	2	3	Практическая работа
4.2.	Модули и библиотеки	5	2	3	Практическая работа
	Работа с файлами	9	4	5	
5.1.	Чтение файлов	5	2	3	Практическая работа
5.2.	Запись файлов	4	2	2	Практическая работа
	Обработка исключений	6	2	4	
6.1.	Исключения и их обработка	6	2	4	Тест

Основы ООП		10	4	6	
7.1.	Классы и объекты	5	2	3	Практическая работа
7.2.	Наследование и полиморфизм	5	2	3	Практическая работа
Разработка проекта		10	4	8	Проектная работа
Итого		76	32	44	

3.Содержание учебного плана

1. Введение в программирование (6 ч.)

1.1.Основные понятия программирования (3 ч.)

Теория:

Введение в программирование: что такое программирование и зачем оно нужно. Основные понятия: переменные, типы данных (целые числа, строки, булевы значения), операторы (арифметические, логические). Примеры простых программ.

Практика:

Написание простых программ для закрепления понятий. Создание и использование переменных. Выполнение арифметических и логических операций.

1.2.Установка и настройка среды разработки (3 ч.)

Теория:

Обзор популярных сред разработки для Python (IDLE, PyCharm, VS Code). Пошаговая инструкция по установке среды разработки. Настройка среды для удобной работы.

Практика:

Установка выбранной среды разработки на компьютере. Настройка среды: выбор темы, настройка шрифтов, установка необходимых плагинов. Написание и выполнение первой программы "Hello, World!".

2. Основы Python (13 ч.)

2.1.Синтаксис и структуры данных (4 ч.)

Теория:

Основы синтаксиса Python: отступы, комментарии, строки. Структуры данных: списки, кортежи, множества, словари. Операции над структурами данных: добавление, удаление, изменение элементов.

Практика:

Написание программ с использованием различных структур данных. Создание и изменение списков, кортежей, множеств и словарей. Выполнение операций над элементами структур данных.

2.2.Переменные и типы данных (4 ч.)

Теория:

Переменные: объявление, присваивание значений. Типы данных: целые числа, вещественные числа, строки, булевы значения. Преобразование типов данных.

Практика:

Создание и использование переменных различных типов. Выполнение арифметических операций с целыми и вещественными числами. Работа со строками: конкатенация, форматирование, методы строк.

2.3.Операторы и выражения (5 ч.)

Теория:

Операторы: арифметические, логические, сравнения. Выражения: составление и выполнение выражений. Приоритет операторов.

Практика:

Написание программ с использованием различных операторов. Составление сложных выражений. Решение задач с использованием операторов и выражений.

3. Управляющие конструкции (10 ч.)

3.1.Условные операторы (5 ч.)

Теория:

Условные операторы: if, elif, else. Вложенные условия. Логические выражения в условиях.

Практика:

Написание программ с использованием условных операторов. Решение задач с использованием условий. Создание программ с вложенными условиями.

3.2.Циклы (5 ч.)

Теория:

Циклы: for, while. Вложенные циклы. Управление циклом: break, continue.

Практика:

Написание программ с использованием циклов. Решение задач с использованием циклов. Создание программ с вложенными циклами.

4. Функции и модули (10 ч.)

4.1.Определение и использование функций (5 ч.)

Теория:

Функции: определение, вызов, параметры, возвращаемое значение. Локальные и глобальные переменные. Рекурсия.

Практика:

Написание программ с использованием функций. Создание и использование собственных функций. Решение задач с использованием рекурсии.

4.2.Модули и библиотеки (5 ч.)

Теория:

Модули: импорт, использование стандартных модулей. Создание собственных модулей. Обзор популярных библиотек Python.

Практика:

Написание программ с использованием стандартных модулей. Создание и использование собственных модулей. Написание программ с использованием популярных библиотек.

5. Работа с файлами (9 ч.)

5.1. Чтение файлов (5 ч.)

Теория:

Открытие и закрытие файлов. Чтение данных из файлов. Обработка ошибок при работе с файлами.

Практика:

Написание программ для чтения данных из файлов. Обработка данных, прочитанных из файлов. Решение задач с использованием чтения файлов.

5.2. Запись файлов (4 ч.)

Теория:

Запись данных в файлы. Форматы файлов: текстовые, бинарные. Обработка ошибок при записи файлов.

Практика:

Написание программ для записи данных в файлы. Создание и запись данных в текстовые и бинарные файлы. Решение задач с использованием записи файлов.

6. Обработка исключений (6 ч.)

6.1. Исключения и их обработка (6 ч.)

Теория:

Исключения: определение, виды. Обработка исключений: try, except, finally. Создание собственных исключений.

Практика:

Написание программ с обработкой исключений. Решение задач с использованием обработки исключений. Создание и использование собственных исключений.

7. Основы ООП (10 ч.)

7.1. Классы и объекты (5 ч.)

Теория:

Основы объектно-ориентированного программирования. Классы и объекты: определение, создание, использование. Атрибуты и методы классов.

Практика:

Написание программ с использованием классов и объектов. Создание и использование собственных классов. Решение задач с использованием ООП.

7.2. Наследование и полиморфизм (5 ч.)

Теория:

Наследование: определение, использование. Полиморфизм: определение, использование. Переопределение методов.

Практика:

Написание программ с использованием наследования и полиморфизма. Создание и использование наследуемых классов. Решение задач с использованием наследования и полиморфизма.

8. Итоговый проект (12 ч.)

8.1. Разработка проекта (12 ч.)

Теория:

Обзор требований к проекту. Планирование и разработка проекта. Презентация проекта.

Практика:

Разработка и презентация итогового проекта. Работа в команде над проектом. Решение реальных задач с использованием полученных знаний.

4. Планируемые результаты освоения программы

Личностные результаты:

- устойчивый интерес и мотивация к изучению программирования и информационных технологий;
- развитие морально-этических и волевых качеств, таких как честность, ответственность и настойчивость;
- дисциплинированность, трудолюбие и упорство в достижении поставленных целей;
- умение управлять своими эмоциями в различных ситуациях, особенно при решении сложных задач;
- умение оказывать помощь своим сверстникам, работать в команде и поддерживать других.

Метапредметные результаты:

- умение находить ошибки в коде и исправлять их, анализировать и оптимизировать программы;
- умение рационально распределять своё время при выполнении заданий, планировать и организовывать учебный процесс;
- умение вести наблюдение за своим прогрессом в изучении программирования, анализировать свои достижения и ставить новые цели.

Предметные результаты:

- умение самостоятельно разрабатывать программы на Python, а также работать над проектами в команде;
- умение организовывать и проводить код-ревью, тестирование и отладку программ;
- умение применять полученные знания для решения реальных задач, разработки приложений и автоматизации процессов.

5. Условия реализации программы

Материально-техническое обеспечение программы

- Компьютерный класс с доступом в интернет;
- Компьютеры (не менее 10 шт.);
- Программное обеспечение для программирования на Python;
- Проектор и экран для презентаций;
- Учебные материалы и пособия по программированию.

Кадровое обеспечение

Педагог дополнительного образования с высшим педагогическим образованием и опытом работы в области программирования, отсутствием ограничений на занятие педагогической деятельностью, установленных законодательством Российской Федерации.

Методические материалы

Методическое обеспечение программы направлено на формирование навыков программирования и логического мышления. Включает учебные пособия, схемы и плакаты по основам программирования на Python, а также примеры кода и задачи для самостоятельного решения.

Программа основана на педагогических технологиях индивидуализации, проектного и группового обучения. Индивидуализация обучения предполагает дифференциацию учебного материала, разработку заданий различного уровня трудности и объёма, разработку системы мероприятий по организации процесса обучения в конкретных группах, учитывающих индивидуальные особенности каждого обучающегося. Образовательная технология разноуровневого обучения направлена на обеспечение равных возможностей получения качественного образования всем обучающимся с учётом разного уровня подготовки и развития.

Оценочные материалы включают различные диагностические материалы (карты, тесты, индивидуальные и групповые задания и т.д.), разработанные критерии оценки.

Формы аттестации/контроля и оценочные материалы

Способы определения результативности образовательной программы

Контрольные работы и проекты. Промежуточная и итоговая аттестация обучающихся проводится два раза в год: в начале учебного года (октябрь) и в конце учебного года (май).

Нормативы общей и специальной подготовки на этапе ознакомительного уровня освоения программы

- Разработка простых программ на Python: «Hello World», изображения, циклы и условия.
- Решение задач на логическое мышление и алгоритмизацию.

6.Список литературы

Литература для педагога

1. Лутц М. Изучаем Python. 5-е издание. М.: Вильямс, 2019. 1600 с.
2. Свиридов В. Основы программирования на Python. М.: БХВ-Петербург, 2020. 320 с.
3. Зед Шоу. Изучаем Python. М.: Питер, 2017. 464 с.
4. Беккер Д. Python для детей. Самоучитель по программированию. М.: Манн, Иванов и Фербер, 2015. 352 с.
5. Рамальо Л. Python. К вершинам мастерства. М.: ДМК Пресс, 2016. 624 с.

Литература для обучающихся

1. Свиридов В. Основы программирования на Python. М.: БХВ-Петербург, 2020. 320 с.
2. Зед Шоу. Изучаем Python. М.: Питер, 2017. 464 с.
3. Лутц М. Изучаем Python. 5-е издание. М.: Вильямс, 2019. 1600 с.
4. Матейчук В. Python. Программирование для детей. М.: Питер, 2019. 256 с.
5. Сведруп С. Python для начинающих. М.: Эксмо, 2021. 288 с.

**ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ**

СВЕДЕНИЯ О СЕРТИФИКАТЕ ЭП

Сертификат 698875933354843316134420126408267428494147114481

Владелец Селукова Татьяна Алексеевна

Действителен с 19.04.2025 по 19.04.2026