

**ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ  
ОНЛАЙН-ШКОЛА «ТОЧКА ЗНАНИЙ»**

---

УТВЕРЖДЕНА  
Приказом Генерального  
директора  
ООО «Точка знаний»  
№ 19 от «29» августа 2025 г.

**Дополнительная общеразвивающая программа  
«Курс по программированию на Python. Ступень 1»**

**(трудоемкость 121 ак. ч.)**

Разработчик:  
Гуртовцев Никита Валерьевич  
Преподаватель дополнительного образования

Возраст: дети (от 10 лет)  
Срок обучения: 9 месяцев

Краснодар, 2025 г.

## СОДЕРЖАНИЕ

1. Общая характеристика.....	3
1.1. Пояснительная записка.....	3
1.2. Направленность.....	3
1.3. Актуальность программы.....	3
1.4. Цели и задачи программы.....	3
1.5. Категория обучающихся.....	3
1.6. Форма обучения и сроки освоения.....	3
1.7. Форма организации образовательной деятельности.....	4
1.8. Документ, выдаваемый после завершения обучения.....	4
2. Содержание программы.....	5
2.1. Календарный учебный график.....	5
2.2. Учебный план.....	5
2.3. Рабочая программа.....	7
2.4. Кадровое обеспечение.....	12
2.5. Материально-техническое обеспечение реализации программы.....	12
2.6. Форма аттестации и оценочные материалы.....	13
2.7. Список рекомендованной литературы.....	18
2.8. Список использованной литература.....	18
2.8.1. Электронные ресурсы.....	19

# 1. ОБЩАЯ ХАРАКТЕРИСТИКА

## 1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Курс по программированию на Python. Ступень 1» (далее – Программа) разработана в соответствии с Федеральным законом от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации», Приказом Министерства просвещения РФ от 27.07.2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам», Письмом Министерства образования и науки РФ от 18.11.2015 № 09-3242 «О направлении Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)».

Изучение основ программирования на языке Python, интегрирующего знания логического мышления и решения задач, позволяет учащимся развивать практические навыки через создание приложений и игр. Курс акцентирует внимание на работе с библиотеками, что углубляет понимание программирования и его концепций.

**1.2. Направленность:** техническая.

**1.3. Актуальность программы:**

Программа актуальна из-за возрастающей значимости навыков программирования в современном мире. Знания языка Python помогают учащимся развивать логические и аналитические способности, что важно для их учёбы и карьеры. Раннее знакомство с технологиями формирует интерес к науке и способствует дальнейшему развитию в этой области.

**1.4. Цели и задачи Программы:**

**Цель программы:** обеспечить учащихся необходимыми знаниями и навыками для освоения основ программирования на языке Python, а также развить логическое мышление и навыки решения задач.

**Задачи программы:**

- ознакомить учащихся с основами языка Python и его экосистемой;
- научить работать с основными типами данных и структурами данных;
- научить работать с основными алгоритмическими конструкциями;
- развить навыки написания алгоритмов и программ на Python;
- развить навыки работы с библиотеками и внешними модулями;
- обучить созданию простых приложений и игр.

**1.5. Категория обучающихся:** К освоению дополнительной общеобразовательной общеразвивающей программы допускаются учащиеся 5-11-х классов, в возрасте 10-17 лет.

**В результате изучения курса обучающиеся должны**

**знать:**

- основные концепции программирования и синтаксис языка Python;
- различные типы данных и структуры данных в Python;
- основы работы алгоритмических конструкций;
- принципы работы с функциями и библиотеками.

**уметь:**

- писать простые программы на Python;
- использовать условные операторы и циклы для решения задач;
- создавать функции для организации кода;
- работать с различными структурами данных (списки, кортежи, словари, множества);
- разрабатывать простые графические приложения и игры.

**1.6. Форма обучения и сроки освоения:**

Программа реализуется исключительно с применением дистанционных образовательных технологий и электронного обучения заочной форме.

Сроки освоения программы – **9 месяцев.**

Лекционные занятия онлайн (видеоурок/вебинар) – **60 ак.ч.**

Самостоятельная работа – **60 ак.ч.**

Итоговая аттестация (тестирование) - **1 ак.ч.**

**Период обучения и режим занятий:**

Занятия проводятся 1 раз в неделю по **1,67 ак.ч.**

**1.7. Форма организации образовательной деятельности:** групповая.

**1.8. Документ, выдаваемый после завершения обучения:** сертификат об обучении.

## 2. СОДЕРЖАНИЕ ПРОГРАММЫ

**Программа включает:** 36 уроков и итоговый тест.

Перечень, трудоемкость, последовательность и распределение уроков, формы промежуточной аттестации слушателей определяются учебным (тематическим) планом.

Содержание каждого урока включает лекционный и практический материал.

### 2.1 Календарный учебный график

Общая учебная нагрузка – 36 дней																	
1 мес яц		2 ме ся ц		3 мес яц		4 мес яц		5 мес яц		6 месяц		7 месяц		8 месяц		9 меся ц	
ЛЗ, ЕТ	СР,	ЛЗ, СР, ЕТ		ЛЗ, ЕТ	СР,	ЛЗ, ЕТ	СР,	ЛЗ, ЕТ	СР,	ЛЗ, ЕТ	СР,	ЛЗ, СР, ЕТ		ЛЗ, ЕТ	СР,	ЛЗ, СР, ЕТ ИА	

ЛЗ – лекционные занятия

СР – самостоятельная работа

ЕТ – ежемесячное тестирование

ИА – итоговая аттестация

### 2.2. Учебный план

№ п / п	Название модулей	Количество часов			
		Всего ак.ч.	Лекционн ые занятия онлайн (видеоурок/ вебинар)  ак.ч.	Самостоя- тельная работа  ак.ч.	Форма проверки
1	Модуль №1: Основы Python	24	12	12	ТК, ЕТ
2	Модуль №2: Работа со строками	16	8	8	ТК, ЕТ

3	Модуль №3: Структуры данных	20	10	10	ТК, ЕТ
4	Модуль №4: Функции	10	5	5	ТК, ЕТ
5	Модуль №5: Введение в графику	20	10	10	ТК, ЕТ
6	Модуль №6: Объекты и файлы	14	7	7	ТК, ЕТ
7	Модуль №7: Разработка игр	16	8	8	ТК, ЕТ
	<b>Итоговая аттестация</b>	1	0	1	Тестирование
	<b>ИТОГО</b>	<b>121</b>	<b>60</b>	<b>61</b>	

ТК – текущий контроль

ЕТ – ежемесячное тестирование

## **2.3. Рабочая программа**

### **Модуль №1: Основы Python**

#### **Урок №1: Введение в Python**

Во время урока слушатели изучают:

Что такое Python и его основные особенности. Настройка среды разработки. Основы работы с интерпретатором Python.

По окончании урока слушатели выполняют домашнее задание.

#### **Урок №2: Переменные и типы данных**

Во время урока слушатели изучают:

Что такое переменные и как их использовать. Основные типы данных. Преобразование типов данных.

По окончании урока слушатели выполняют домашнее задание.

#### **Урок №3: Ввод данных и операторы**

Во время урока слушатели изучают:

Арифметические операторы и операции с числами. Функция input() для ввода данных от пользователя.

По окончании урока слушатели выполняют домашнее задание.

#### **Урок №4: Условия и сравнения**

Во время урока слушатели изучают:

Условные конструкции: if, elif, else. Практические примеры использования условий.

По окончании урока слушатели выполняют домашнее задание.

#### **Урок №5: Логические операторы**

Во время урока слушатели изучают:

Логические операторы not, and, or. Сложные условия. Практические примеры использования логических операторов.

По окончании урока слушатели выполняют домашнее задание.

#### **Урок №6: Циклы for и while**

Во время урока слушатели изучают:

Структура цикла for и его применение. Структура цикла while и его применение.

По окончании урока слушатели выполняют домашнее задание.

#### **Урок №7: Случайные числа**

Во время урока слушатели изучают:

Модуль random и генерация случайных чисел. Примеры использования случайных чисел в программах.

По окончании урока слушатели выполняют домашнее задание.

#### **Урок №8: Разработка игры "Угадай число"**

Во время урока слушатели изучают:

Применение всех изученных тем для создания игры. Логика игры, условия победы/поражения.

По окончании урока слушатели выполняют домашнее задание.

## **Модуль №2: Работа со строками**

### **Урок №9: Строки и срезы**

Во время урока слушатели изучают:

Основные операции со строками. Использование срезов для работы со строками. Примеры манипуляций со строками.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №10: Методы работы со строками**

Во время урока слушатели изучают:

Изучение встроенных методов строк (например, `.upper()`, `.lower()`).

По окончании урока слушатели выполняют домашнее задание.

### **Урок №11: Форматирование строк**

Во время урока слушатели изучают:

Форматирование с помощью оператора `%`. Использование метода `.format()`.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №12: Работа с символами**

Во время урока слушатели изучают:

Изучение символов и их кодов (Unicode). Примеры работы с отдельными символами в строках.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №13: Разработка приложения "Шифр Цезаря"**

Во время урока слушатели изучают:

Логика шифрования и дешифрования текста. Реализация приложения на Python.

По окончании урока слушатели выполняют домашнее задание.

## **Модуль №3: Структуры данных**

### **Урок №14: Знакомство с кортежами**

Во время урока слушатели изучают:

Что такое кортежи и как их использовать. Основные операции с кортежами.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №15: Знакомство со списками**

Во время урока слушатели изучают:

Что такое списки и как их использовать. Основные операции со списками. Индексация и срезы списков.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №16: Работа со списками**

Во время урока слушатели изучают:

Методы списков (`split`, `join` и др.). Поиск в списке. Сортировка.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №17: Знакомство со словарями**

Во время урока слушатели изучают:

Что такое словари и как они работают. Основные операции со словарями.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №18: Множества и их применение**

Во время урока слушатели изучают:

Что такое множества и их свойства. Операции над множествами (объединение, пересечение и другие).

По окончании урока слушатели выполняют домашнее задание.

### **Урок №19: Разработка игры "Угадай слово"**

Во время урока слушатели изучают:

Повторение основ работы со структурами данных. Создание мини-игры в консоли.

По окончании урока слушатели выполняют домашнее задание.

## **Модуль №4: Функции**

### **Урок №20: Знакомство с функциями**

Во время урока слушатели изучают:

Что такое функции и зачем они нужны. Объявление и вызов функций. Параметры и аргументы функций.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №21: Работа с функциями**

Во время урока слушатели изучают:

Аргументы переменной длины. Возврат значений из функций и рекурсии. Анонимные функции.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №22: Разработка текстового квеста. Часть 1**

Во время урока слушатели изучают:

Повторение основ языка, случайные события, структуры данных и функции. Логика текстового квеста на Python.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №23: Разработка текстового квеста. Часть 2**

Во время урока слушатели изучают:

Использование функций, списков и словарей в проекте. Разработка текстового квеста.

По окончании урока слушатели выполняют домашнее задание.

## **Модуль №5: Введение в графику**

### **Урок №24: Введение в библиотеку Turtle**

Во время урока слушатели изучают:

Основы работы с библиотекой Turtle для графики. Простые примеры рисования.  
По окончании урока слушатели выполняют домашнее задание.

### **Урок №25: Цвета и заливка**

Во время урока слушатели изучают:  
Работа с цветами в Turtle. Заливка фигур цветом.  
По окончании урока слушатели выполняют домашнее задание.

### **Урок №26: Система координат**

Во время урока слушатели изучают:  
Понимание системы координат в Turtle. Перемещение черепашки по экрану.  
По окончании урока слушатели выполняют домашнее задание.

### **Урок №27: Рисование картины**

Во время урока слушатели изучают:  
Проект по рисованию домика с использованием Turtle. Комбинирование различных форм и цветов.  
По окончании урока слушатели выполняют домашнее задание.

### **Урок №28: Создание фракталов**

Во время урока слушатели изучают:  
Что такое фракталы. Рисование фракталов Кривая Коха и Фрактальное дерево.  
По окончании урока слушатели выполняют домашнее задание.

## **Модуль №6: Объекты и файлы**

### **Урок №29: Знакомство с объектами**

Во время урока слушатели изучают:  
Основные понятия объектно-ориентированного программирования (ООП). Создание простых объектов в Python.  
По окончании урока слушатели выполняют домашнее задание.

### **Урок №30: Работа с событиями**

Во время урока слушатели изучают:  
Введение в обработку событий. Примеры использования событий в приложениях.  
По окончании урока слушатели выполняют домашнее задание.

### **Урок №31: Игра "Черепашьи гонки"**

Во время урока слушатели изучают:  
Создание черепашек. Запуск гонки. Отображение победителя.  
По окончании урока слушатели выполняют домашнее задание.

### **Урок №32: Игра "Побег черепашки"**

Во время урока слушатели изучают:  
Настройка игры. Создание черепашки. Создание соперников. Движение черепашек. Игровой цикл.  
Завершение игры.  
По окончании урока слушатели выполняют домашнее задание.

## **Модуль №7: Разработка игр**

### **Урок №33: Загрузка изображений**

Во время урока слушатели изучают:

Подготовка к работе с изображениям. Добавление изображения. Разработка игры с изображениями.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №34: Создание анимации**

Во время урока слушатели изучают:

Основы создания анимации с использованием библиотеки Turtle. Примеры простых анимаций.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №35: Игра "Битва с инопланетянами"**

Во время урока слушатели изучают:

Разработка финального проекта по созданию простой игры на Python. Применение всех изученных тем.

По окончании урока слушатели выполняют домашнее задание.

### **Урок №36: Возможности циклов и условий**

Во время урока слушатели изучают:

Конструкция цикла for - else и тернарные операторы для упрощения условий.

По окончании урока слушатели выполняют домашнее задание.

### **Итоговая аттестация**

Тестирование

## 2.4. Кадровое обеспечение

Реализация программы обеспечивается педагогическими кадрами, имеющими высшее профессиональное образование и отвечающими квалификационным требованиям, указанным в квалификационных справочниках, и (или) профессиональным стандартам, в рамках изучаемого цикла.

## 2.5. Материально-техническое и программное обеспечение реализации программы

ООО «ТОЧКА ЗНАНИЙ» располагает материально-технической базой, обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, итоговой аттестации слушателей, предусмотренных учебным планом:

Место работы преподавателя	<ul style="list-style-type: none"><li>✓ Стол компьютерный/письменный 120x60 - 1 шт;</li><li>✓ Кресло ARDOR GAMING Gun Shield 10B - 1 шт;</li><li>✓ ИБП Dexp IEC Plus LCD 3000VA - 1 шт;</li><li>✓ Переходник от ИБП к сетевому фильтру - 1 шт;</li><li>✓ Сетевой фильтр с USB разъемами - 1 шт;</li><li>✓ Софтбокс 50x70 600BT - 1 шт;</li><li>✓ Лампа студийная для софтбокса 135W E27 5500K, Холодный белый свет, 135 Вт, Люминесцентная (энергосберегающая) - 4шт (запасные);</li><li>✓ Предохранители в софтбокс 3- 5А, 250V - 5 штук;</li><li>✓ Монитор 21.45" DEXP DF22N2 черный - 1 шт;</li><li>✓ Микрофон петличный VOYA BY-M1 - 1 шт;</li><li>✓ Apple iPad 10.2" (2021) Wi-Fi 64Gb Silver - 1 шт;</li><li>✓ Apple MacBook Pro 16", M1, 16ГБ, 2021г - 1 шт;</li><li>✓ Стилус WiWU Pencil Pro - 1 шт;</li><li>✓ Хаб Baseus (4K@60Hz 8-in-1 STARJOY 8-PORT TYPE-C) - 1 шт;</li><li>✓ Патч-корд RJ 45 - 1 шт;</li><li>✓ Футболки (мерч) "Точка знаний" - 2 шт.</li></ul>
----------------------------	---

**Программное обеспечение:** лицензионные системные программы, обеспечивающие взаимодействие всех других программ с оборудованием и взаимодействие пользователя персонального компьютера с программами. Универсальные офисные прикладные программы и средства ИКТ, например, программа подготовки презентаций; использование Интернета, электронной почты; использование автоматизированных поисковых систем Интернета.

Информационно-образовательная среда включает в себя образовательную LMS «Точка Знаний».

Образовательная LMS «Точка Знаний» обеспечивает через Интернет доступ к:

- электронным информационным и образовательным ресурсам ООО «ТОЧКА ЗНАНИЙ»;
- доступ к нормативным и организационно-методическим документам, регламентирующим образовательный процесс в ООО «ТОЧКА ЗНАНИЙ»;

- систему электронного учёта слушателей; (см. [Положение](#))
- взаимодействие слушателей с преподавателями, организаторами образовательного процесса и администрацией ООО «ТОЧКА ЗНАНИЙ». (см. [Положение](#))

## 2.6. Форма аттестации и оценочные материалы

Программой предусмотрен текущий контроль в виде домашнего задания и тестов, размещенных в уроках на образовательной LMS «Точка Знаний», промежуточный контроль в виде тестов, согласно учебному плану.

Итоговая аттестация проводится в виде теста. Итоговый тест включает в себя 20 вопросов. Для успешного прохождения теста необходимо правильно выполнить не менее 85 % заданий.

Обучающиеся допускаются к итоговой аттестации после изучения Модулей программы в объеме, предусмотренном учебным планом программы.

Оценка качества освоения учебной программы проводится в процессе итоговой аттестации в форме тестирования.

Оценка	Критерии
<b>Зачтено</b>	Оценка «Зачтено» выставляется обучающемуся, продемонстрировавшему твердые и всесторонние знания материалы, умение применять полученные в рамках занятий практические навыки и умения. Достижения за период обучения и результаты текущей аттестации демонстрировали отличный уровень знаний и умений обучающегося. Не менее 85% правильных ответов при решении теста.
<b>Не зачтено</b>	Оценка «Не зачтено» выставляется обучающемуся, который в недостаточной мере овладел теоретическим материалом по дисциплине, допустил ряд грубых ошибок при выполнении практических заданий, а также не выполнил требований, предъявляемых к промежуточной аттестации. Достижения за период обучения и результаты текущей аттестации демонстрировали неудовлетворительный уровень знаний и умений обучающегося. Менее 80% правильных ответов при решении теста.

### Примеры оценочных материалов текущего контроля

Домашнее задание 1. Введение в Python

Задание:

Сегодня ты станешь волшебником, который использует Python для выполнения математических фокусов! Тебе предстоит создать программу на Python, которая вычисляет разные математические операции и показывает результаты.

Пошаговая инструкция:

1. Вычисли сумму чисел 1234 и 4321. Выведи результат в консоль.
2. Вычисли разность чисел 9876 и 5432. Выведи результат в консоль.
3. Вычисли произведение чисел 303 и 11. Выведи результат в консоль.
4. Вычисли частное от деления чисел 33330 и 15. Выведи результат в консоль.

### Примеры оценочных материалов промежуточного контроля

Вопрос 1. Какое из следующих утверждений верно описывает переменную в Python?

Ответы:

1. Переменная — это функция, которая выполняет код
2. Переменная — это место для хранения данных
3. Переменная — это ключевое слово в Python
4. Переменная — это тип данных

5. Переменная — это условная конструкция

Вопрос 2. Как правильно создать переменную age, которая хранит значение 25?

Ответы:

1. `25 = age`
2. `age = 25`
3. `age := 25`
4. `age == 25`
5. `age 25`

Вопрос 3. Какой из следующих кодов правильно выведет значение переменной name?

Ответы:

1. `print name`
2. `print(name)`
3. `print('name')`
4. `print("name")`
5. `print(name())`

Вопрос 4. Какие из следующих имен переменных являются допустимыми? Выберите все подходящие варианты

Ответы:

1. `name1`
2. `2d_shape`
3. `user_name`
4. `_age`
5. `if`

Вопрос 5. Какие из следующих кодов увеличат значение переменной count на 3? Выберите все подходящие варианты

Ответы:

1. `count = 3`
2. `count += 3`
3. `count = count + 3`
4. `count + 3 = count`
5. `count+++`

Вопрос 6. Какой из следующих вариантов правильно считывает ввод пользователя в Python?

Ответы:

1. `input()`
2. `read()`
3. `scan()`
4. `prompt()`
5. `get_input()`

Вопрос 7. Какой тип данных используется для хранения целых чисел?

Ответы:

1. `str`
2. `int`
3. `float`
4. `list`
5. `dict`

Вопрос 8. Какие из следующих кодов вернут значение True? Выберите все подходящие

варианты

Ответы:

1. `5 == 5`
2. `10 != 10`
3. `7 > 8`
4. `3 >= 3`
5. `4 < 5`

Вопрос 9. Какой из следующих фрагментов кода правильно проверяет, если переменная `coins` больше 10?

Ответы:

1. 

```
if coins > 10:  
    print("У тебя много монет!")
```
2. 

```
if (coins > 10)  
    print("У тебя много монет!")
```
3. 

```
if coins > 10  
    print("У тебя много монет!")
```
4. 

```
if (coins > 10):  
    print("У тебя много монет!")
```
5. 

```
if coins > 10  
    print("У тебя много монет!")
```

Вопрос 10. Какой будет результат выполнения следующего кода, если `temperature` равно 25?

```
temperature = 25  
if temperature < 20:  
    print("Холодно")  
elif temperature < 30:  
    print("Тепло")  
else:  
    print("Жарко")
```

Ответы:

1. "Холодно"
2. "Тепло"
3. "Жарко"
4. Ничего не выведется
5. Будет ошибка

### Примеры оценочных материалов итоговой аттестации

Задание 1. Какой из следующих вариантов является правильным способом для создания переменной в Python?

Ответы:

1. `1variable = 5`
2. `variable-1 = 5`
3. `variable_1 = 5`
4. `variable 1 = 5`
5. `variable@1 = 5`

Задание 2. Какой из следующих операторов используется для проверки равенства в Python?

Ответы:

1. `=`
2. `==`
3. `!=`

4. >
5. >=

Задание 3. Какой модуль используется для генерации случайных чисел в Python?

Ответы:

1. random
2. math
3. statistics
4. os
5. time

Задание 4. Какой результат будет при выполнении следующего кода: "Hello"[1:4]?

Ответы:

1. "Hel"
2. "ell"
3. "llo"
4. "lo"
5. "Hello"

Задание 5. Какой метод используется для преобразования строки в верхний регистр?

Ответы:

1. upper()
2. toUpperCase()
3. capitalize()
4. makeUpper()
5. upperCase()

Задание 6. Какой из следующих вариантов является корректным созданием кортежа?

Ответы:

1. my\_tuple = (1, 2, 3)
2. my\_tuple = [1, 2, 3]
3. my\_tuple = {1, 2, 3}
4. my\_tuple = (1. 2. 3)
5. my\_tuple = [1; 2; 3]

Задание 7. Какой результат будет у этого кода со списком?

```
my_list = [1, 2, 3]
```

```
print(my_list[1])
```

Ответы:

1. 1
2. 2
3. 3
4. Ошибка индексации
5. None

Задание 8. Какой метод добавляет один элемент по умолчанию в конец списка?

Ответы:

1. add()
2. push()
3. insert()
4. extend()
5. append()

Задание 9. Какой из следующих способов правильно создает словарь?

Ответы:

1. `my_dict = (key1: value1, key2: value2)`
2. `my_dict = [key1 => value1, key2 => value2]`
3. `my_dict = {key1: value1, key2: value2}`
4. `my_dict = <key1, value1>, <key2, value2>`
5. `my_dict = {key1; value1; key2; value2}`

Задание 10. Как объявить функцию в Python?

Ответы:

1. `func my_function():`
2. `function my_function():`
3. `define my_function():`
4. `def my_function():`
5. `create my_function():`

Задание 11. Что делает метод `turtle.penup()`?

Ответы:

1. Поднимает перо черепашки
2. Убирает цвет пера
3. Останавливает движение черепашки
4. Начинает заливку
5. Скрывает черепашку

Задание 12. Какой метод позволяет изменить цвет черепашки в `turtle`?

Ответы:

1. `turtle.fillcolor()`
2. `turtle.color()`
3. `turtle.turtlecolor()`
4. `turtle.setcolor()`
5. `turtle.linecolor()`

Задание 13. Какой метод перемещает черепашку в заданные координаты в `turtle`?

Ответы:

1. `turtle.goto(x, y)`
2. `turtle.move(x, y)`
3. `turtle.setpos(x, y)`
4. `turtle.jump(x, y)`
5. `turtle.position(x, y)`

Задание 14. Какой метод используется для рисования круга в `turtle`?

Ответы:

1. `turtle.draw_circle(radius)`
2. `turtle.make_circle(radius)`
3. `turtle.circle(radius)`
4. `turtle.arc(radius)`
5. `turtle.round(radius)`

Задание 15. Какой класс используется для инициализации объекта `turtle`?

Ответы:

1. `turtle.Turtle()`
2. `turtle.Start()`
3. `turtle.Begin()`

4. turtle.Init()
5. turtle.Setup()

Задание 16. Какой метод позволяет обрабатывать события мыши в turtle?

Ответы:

1. turtle.mouseclick()
2. turtle.eventclick()
3. turtle.listenclick()
4. turtle.onclick()
5. turtle.clickevent()

Задание 17. Какой метод запускает основной цикл программы в turtle?

Ответы:

1. turtle.run()
2. turtle.mainloop()
3. turtle.start()
4. turtle.execute()
5. turtle.loop()

Задание 18. Что происходит при использовании метода turtle.hideturtle()?

Ответы:

1. Черепашка исчезает с экрана
2. Черепашка становится прозрачной
3. Черепашка меняет цвет
4. Черепашка становится видимо
5. Черепашка останавливается

Задание 19. Какой метод позволяет загрузить изображение в качестве фона?

Ответы:

1. turtle.loadimage("filename")
2. turtle.bgpic("filename")
3. turtle.setbackground("filename")
4. turtle.image("filename")
5. turtle.background("filename")

Задание 20. Какой метод позволяет установить скорость черепашки?

Ответы:

1. turtle.speed(speed)
2. turtle.getspeed(speed)
3. turtle.changespeed(speed)
4. turtle.velocity(speed)
5. turtle.move\_speed(speed)

## 2.7. Список рекомендованной литературы

### Основная:

1. Документация Python <https://www.python.org/doc/>
2. Уроки Python <https://hourofpython.com/>

### Дополнительная:

1. Документация Python 3 <https://docs.python.org/3/>

## 2.8. Список использованной литературы

### 2.8.1. Электронные ресурсы

1. Электронная библиотека <https://elibrary.ru/>
2. Онлайнтестпад-онлайн конструктор тестов <https://onlinetestpad.com/>
3. Онлайн-платформа для программирования <https://trinket.io/>