

**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 2»**

(трудоемкость 121 ак. ч.)

Разработчик:
Гуртовцев Никита Валерьевич
Преподаватель дополнительного образования

Возраст: дети (от 11 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.....	4
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.....	19
2.8.1. Электронные ресурсы.....	19

1. ОБЩАЯ ХАРАКТЕРИСТИКА

1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Курс по программированию на Python. Ступень 2» (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 разработки приложений, позволяет учащимся развивать практические навыки через создание программ и игр. Курс выделяется акцентом на работу с файлами, объектно-ориентированным программированием и разработкой графических интерфейсов, что углубляет понимание более сложных концепций программирования и способствует всестороннему развитию специалистов.

1.2. Направленность: техническая.

1.3. Актуальность программы:

Программа актуальна в свете растущей важности цифровых навыков в современном обществе. Знания языка Python становятся неотъемлемой частью образования, формируя у учащихся способности логического мышления и аналитического подхода к решению задач. Раннее освоение технологий программирования не только расширяет горизонты учеников, но и открывает новые возможности для их будущей карьеры в сферах науки и технологий.

1.4. Цели и задачи Программы:

Цель программы: углубить знания и навыки учащихся в программировании на языке Python, познакомить их с более сложными концепциями и инструментами, необходимыми для разработки приложений и игр.

Задачи программы:

- изучить работу с файлами: чтение, запись и сериализация данных;
- освоить обработку исключений для повышения надежности программ;
- научить созданию собственных модулей для организации кода;
- познакомить с основами объектно-ориентированного программирования (ООП);
- изучить основы многопоточности для повышения эффективности приложений;
- освоить библиотеку Tkinter для разработки графических интерфейсов;
- начать изучение библиотеки Pygame для создания игр с анимацией и звуковыми эффектами.

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

В результате изучения курса обучающиеся должны

знать:

- принципы работы с файлами и обработки исключений в Python;
- основы объектно-ориентированного программирования;
- как создавать и использовать собственные модули;
- основы работы с графическими интерфейсами с помощью Tkinter;
- базовые возможности библиотеки Pygame для разработки игр.

уметь:

- разрабатывать программы, использующие работу с файлами;
- обрабатывать исключения и писать надежный код;
- создавать собственные модули для организации кода;
- применять принципы ООП в своих проектах;
- разрабатывать графические интерфейсы для приложений с помощью Tkinter;

- создавать простые игры с анимацией и звуковыми эффектами в Pygame.

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: Углубленное изучение Python	14	7	7	ТК, ЕТ
2	Модуль №2: Объектно-ориентированное программирование (ООП) в Python	26	13	13	ТК, ЕТ

3	Модуль №3: Разработка графических приложений с использованием Tkinter	34	17	17	ТК, ЕТ
4	Модуль №4: Разработка игр с использованием Pygame	34	17	17	ТК, ЕТ
5	Модуль №5: Финальный проект и публикация	12	6	6	ТК, ЕТ
	Итоговая аттестация	1	0	1	Тестирование
	ИТОГО	121	60	61	

ТК – текущий контроль

ЕТ – ежемесячное тестирование

2.3. Рабочая программа

Модуль №1: Углубленное изучение Python

Урок №1: Инструменты Python-разработчика

Во время урока слушатели изучают: основные инструменты и среды разработки, используемые в Python, включая IDE и отладчики. По окончании урока слушатели выполняют домашнее задание.

Урок №2: Работа с библиотеками

Во время урока слушатели изучают: работу с библиотекой datetime, включая создание, форматирование и арифметику дат и времени. По окончании урока слушатели выполняют домашнее задание.

Урок №3: Работа с файлами

Во время урока слушатели изучают: методы чтения и записи данных в файлы, а также работу с текстовыми и бинарными файлами. По окончании урока слушатели выполняют домашнее задание.

Урок №4: Обработка исключений

Во время урока слушатели изучают: механизмы обработки исключений в Python, включая использование блоков try, except, finally и создание собственных исключений. По окончании урока слушатели выполняют домашнее задание.

Модуль №2: Объектно-ориентированное программирование (ООП) в Python

Урок №5: Введение в ООП

Во время урока слушатели изучают: основные концепции объектно-ориентированного программирования, такие как классы, методы и объекты. По окончании урока слушатели выполняют домашнее задание.

Урок №6: Принципы ООП. Часть 1

Во время урока слушатели изучают: инкапсуляция, наследование и полиморфизм как ключевые принципы ООП, а также их применение на практике. По окончании урока слушатели выполняют домашнее задание.

Урок №7: Принципы ООП. Часть 2

Во время урока слушатели изучают: геттеры, сеттеры и декораторы, а также как эти принципы помогают в проектировании классов и объектов. По окончании урока слушатели выполняют домашнее задание.

Урок №8: Модули и пакеты

Во время урока слушатели изучают: как создавать и использовать модули и пакеты в Python для организации кода и повторного использования функций. По окончании урока слушатели выполняют домашнее задание.

Урок №9: Многопоточность

Во время урока слушатели изучают: основы многопоточности в Python, включая создание потоков и управление ими с помощью модуля threading. По окончании урока слушатели выполняют домашнее задание.

Урок №10: Асинхронность

Во время урока слушатели изучают: асинхронное программирование с использованием ключевых слов async и await, а также работу с библиотекой asyncio. По окончании урока слушатели выполняют домашнее задание.

Урок №11: Разработка RPG. Часть 1

Во время урока слушатели изучают: основы проектирования RPG-игры, включая создание игрового мира и персонажей. По окончании урока слушатели выполняют домашнее задание.

Урок №12: Разработка RPG. Часть 2

Во время урока слушатели изучают: реализацию игровых механик, таких как бои и квесты, в RPG-игре. По окончании урока слушатели выполняют домашнее задание.

Модуль №3: Разработка графических приложений с использованием Tkinter

Урок №13: Введение в Tkinter

Во время урока слушатели изучают: основы работы с библиотекой Tkinter для создания графических интерфейсов, включая создание простого окна приложения. По окончании урока слушатели выполняют домашнее задание.

Урок №14: Работа с виджетами Tkinter

Во время урока слушатели изучают: различные виджеты Tkinter (кнопки, метки, поля ввода) и их свойства для создания интерактивных интерфейсов. По окончании урока слушатели выполняют домашнее задание.

Урок №15: Обработка событий в Tkinter

Во время урока слушатели изучают: как обрабатывать события (например, нажатия кнопок) в Tkinter для создания динамичных приложений. По окончании урока слушатели выполняют домашнее задание.

Урок №16: Разработка калькулятора

Во время урока слушатели изучают: процесс разработки простого калькулятора с использованием Tkinter, включая обработку пользовательского ввода. По окончании урока слушатели выполняют домашнее задание.

Урок №17: Работа с меню и диалоговыми окнами

Во время урока слушатели изучают: создание меню и диалоговых окон в приложениях Tkinter для улучшения пользовательского интерфейса. По окончании урока слушатели выполняют домашнее задание.

Урок №18: Разработка приложения "Список дел"

Во время урока слушатели изучают: создание приложения "Список дел" с использованием Tkinter, включая добавление, удаление и отображение задач. По окончании урока слушатели выполняют домашнее задание.

Урок №19: Работа с графикой в Tkinter

Во время урока слушатели изучают: основы работы с графикой в Tkinter, включая рисование фигур и загрузку изображений. По окончании урока слушатели выполняют домашнее задание.

Урок №20: Разработка графического редактора

Во время урока слушатели изучают: создание простого графического редактора с использованием Tkinter, включая инструменты рисования и изменения цвета. По окончании урока слушатели выполняют домашнее задание.

Урок №21: Работа с API

Во время урока слушатели изучают: основы работы с API, включая отправку запросов и получение данных из веб-сервисов с использованием библиотеки requests. По окончании урока слушатели выполняют домашнее задание.

Урок №22: Разработка приложения "Переводчик"

Во время урока слушатели изучают: создание приложения "Переводчик" с использованием API для перевода текста между языками. По окончании урока слушатели выполняют домашнее задание.

Модуль №4: Разработка игр с использованием Pygame

Урок №23: Введение в Pygame

Во время урока слушатели изучают: основы работы с библиотекой Pygame для создания игр, включая настройку игрового окна и обработку событий. По окончании урока слушатели выполняют домашнее задание.

Урок №24: Работа с изображениями в Pygame

Во время урока слушатели изучают: загрузку и отображение изображений в Pygame, а также использование спрайтов для создания игровых объектов. По окончании урока слушатели выполняют домашнее задание.

Урок №25: События в Pygame

Во время урока слушатели изучают: обработку событий (например, нажатия клавиш) в Pygame для управления игровым процессом. По окончании урока слушатели выполняют домашнее задание.

Урок №26: Анимация спрайтов

Во время урока слушатели изучают: создание анимации спрайтов в Pygame, включая использование кадров для плавного движения объектов. По окончании урока слушатели выполняют домашнее задание.

Урок №27: Обработка столкновений

Во время урока слушатели изучают: методы обработки столкновений между игровыми объектами в Pygame для реализации игрового процесса. По окончании урока слушатели выполняют домашнее задание.

Урок №28: Звуки и музыка

Во время урока слушатели изучают: добавление звуковых эффектов и музыки в игры на Pygame для улучшения игрового опыта. По окончании урока слушатели выполняют домашнее задание.

Урок №29: Создание игры "Пинг-понг". Часть 1

Во время урока слушатели изучают: разработку первой части игры "Пинг-понг", включая создание игрового поля и логики движения ракеток. По окончании урока слушатели выполняют домашнее задание.

Урок №30: Создание игры "Пинг-понг". Часть 2

Во время урока слушатели изучают: завершение разработки игры "Пинг-понг", включая реализацию логики счета и завершение игры. По окончании урока слушатели выполняют домашнее задание.

Урок №31: Создание игры "Змейка". Часть 1

Во время урока слушатели изучают: разработку первой части игры "Змейка", включая создание игрового поля и логики движения змейки. По окончании урока слушатели выполняют домашнее задание.

Урок №32: Создание игры "Змейка". Часть 2

Во время урока слушатели изучают: завершение разработки игры "Змейка", включая реализацию механики роста змейки и условий окончания игры. По окончании урока слушатели выполняют домашнее задание.

Модуль №5: Финальный проект и публикация

Урок №33: Проектирование игры "Платформер"

Во время урока слушатели изучают: основы проектирования платформенной игры, включая создание уровней и игровых механик. По окончании урока слушатели выполняют домашнее задание.

Урок №34: Разработка игры "Платформер"

Во время урока слушатели изучают: реализацию основных элементов платформенной игры, таких как управление персонажем и взаимодействие с окружающей средой. По окончании урока слушатели выполняют домашнее задание.

Урок №35: Доработка и публикация игры "Платформер"

Во время урока слушатели изучают: финализацию проекта платформенной игры, включая тестирование, исправление ошибок и подготовку к публикации. По окончании урока слушатели выполняют домашнее задание.

Урок №36: Функциональное программирование

Во время урока слушатели изучают: основы функционального программирования в Python, включая использование декораторов, генераторов и итераторов. 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">✓ Стол компьютерный/письменный 120x60 - 1 шт;✓ Кресло ARDOR GAMING Gun Shield 10B - 1 шт;✓ ИБП Dexp IEC Plus LCD 3000VA - 1 шт;✓ Переходник от ИБП к сетевому фильтру - 1 шт;✓ Сетевой фильтр с USB разъемами - 1 шт;✓ Софтбокс 50x70 600BT - 1 шт;✓ Лампа студийная для софтбокса 135W E27 5500K, Холодный белый свет, 135 Вт, Люминесцентная (энергосберегающая) - 4шт (запасные);✓ Предохранители в софтбокс 3- 5А, 250V - 5 штук;✓ Монитор 21.45" DEXP DF22N2 черный - 1 шт;✓ Микрофон петличный VOYA BY-M1 - 1 шт;✓ Apple iPad 10.2" (2021) Wi-Fi 64Gb Silver - 1 шт;✓ Apple MacBook Pro 16", M1, 16ГБ, 2021г - 1 шт;✓ Стилус WiWU Pencil Pro - 1 шт;✓ Хаб Baseus (4K@60Hz 8-in-1 STARJOY 8-PORT TYPE-C) - 1 шт;✓ Патч-корд RJ 45 - 1 шт;✓ Футболки (мерч) "Точка знаний" - 2 шт.
----------------------------	---

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
Зачтено	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.
Не зачтено	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 3. Работа с файлами

Задание

У тебя есть текстовый файл, например, с названием `input.txt`, в котором написаны разные предложения (строки). Нужно написать программу, которая посчитает, сколько строк в этом файле и запишет результат в новый файл `output.txt`.

Пошаговая инструкция

1. Создай файл `input.txt`. Создай текстовый файл с именем `input.txt` и напиши в него несколько строк (предложений, слов). Например:

Привет, мир!

Как дела?

Всё отлично.

2. Напиши программу на Python, которая:

Открывает файл `input.txt`. Программа должна открыть файл `input.txt` для чтения.

Читает все строки из файла. Прочитай все строки из файла и сохрани их в памяти.

Считает количество строк. Посчитай, сколько строк (предложений) содержится в файле.

Открывает файл `output.txt`. Открой новый файл с именем `output.txt` для записи.

Записывает результат в файл `output.txt`. Запиши в файл `output.txt` строчку с результатом, например, "Количество строк: 3".

Важно! Чтобы у тебя получилось, как надо, результат в твоём файле `output.txt` должен быть

записан в виде одной строки.

Закрывает оба файла. Запомни, что после работы с файлами нужно закрыть их.

3. Проверь работу программы. После того, как ты запустишь свою программу, проверь, создался ли файл `output.txt` и содержится ли в нём правильный результат. Программа также должна вывести сообщение в консоль о том, сколько строк было в файле `input.txt` и куда она записала результат.

Примеры оценочных материалов промежуточного контроля

Вопрос 1. Какой стиль именования переменных соответствует PEP-8?

Ответы:

```
myVariableName
MyVariableName
my_variable_name
myvariablename
MYVARIABLENAME
```

Вопрос 2. Какой из следующих вариантов является правильным комментарием в Python?

Ответы:

```
// This is a comment
# This is a comment
/* This is a comment */
-- This is a comment
<!-- This is a comment -->
```

Вопрос 3. Какой метод из библиотеки `math` возвращает квадратный корень числа?

Ответы:

```
math.sqrt()
math.square()
math.root()
math.pow()
math.cube()
```

Вопрос 4. Какой метод возвращает текущую дату и время в библиотеке `datetime`?

Ответы:

```
datetime.current()
datetime.now()
datetime.today()
datetime.get_time()
datetime.date_now()
```

Вопрос 5. Какой метод используется для открытия файла в режиме записи?

Ответы:

```
open('file.txt', 'r')
open('file.txt', 'w')
open('file.txt', 'a')
open('file.txt', 'x')
open('file.txt', 'b')
```

Вопрос 6. Какой метод позволяет записать список строк в файл?

Ответы:

```
write()
```

```
writeall()
writelines()
append()
addlines()
```

Вопрос 7. Какой метод считывает одну строку из файла?

Ответы:

```
read()
readline()
readlines()
get_line()
fetch_line()
```

Вопрос 8. Какой блок кода выполняется, если в блоке конструкции исключения try произошла ошибка?

Ответы:

```
try
catch
except
else
finally
```

Вопрос 9. Какой блок кода в конструкции исключения try выполняется всегда, независимо от того, произошла ошибка или нет?

Ответы:

```
try
catch
except
else
finally
```

Вопрос 10. Что произойдет при выполнении следующего кода, если переменная x равна 0?

```
try:
    result = 10 / x
except ZeroDivisionError:
    print("Деление на ноль!")
else:
    print("Результат:", result)
```

Ответы:

Выведет "Результат: ..."

Выведет "Деление на ноль!"

Выдаст ошибку TypeError

Выдаст ошибку ValueError

Ничего не выведет

Примеры оценочных материалов итоговой аттестации

Задание 1. Какой модуль используется для работы с математическими функциями в Python?

Ответы:

1. math
2. numbers
3. statistics

4. random
5. geometry

Задание 2. Какой метод из модуля time позволяет приостановить выполнение программы на заданное количество секунд?

Ответы:

1. time.sleep()
2. time.pause()
3. time.wait()
4. time.halt()
5. time.stop()

Задание 3. Какой метод используется для получения текущей даты и времени из модуля datetime?

Ответы:

1. datetime.now()
2. datetime.today()
3. datetime.current()
4. datetime.get()
5. datetime.time()

Задание 4. Какой оператор используется для обработки исключений в Python?

Ответы:

1. try...except
2. catch...finally
3. handle...error
4. except...try
5. error...catch

Задание 5. Какой метод открывает файл для чтения в Python?

Ответы:

1. open("filename", "r")
2. open("filename", "read")
3. open("filename", "w")
4. open("filename", "a")
5. open("filename", "rb")

Задание 6. Какой из следующих вариантов является корректным способом создания класса в Python?

Ответы:

1. class MyClass:
2. MyClass class:
3. define MyClass:
4. create class MyClass:
5. class: MyClass

Задание 7. Какой метод используется для добавления атрибута к объекту в Python?

Ответы:

1. setattr()
2. addattr()
3. appendattr()
4. insertattr()
5. updateattr()

Задание 8. Какой метод позволяет запускать несколько потоков одновременно в Python?

Ответы:

1. `threading.Thread()`
2. `concurrent.Thread()`
3. `process.Thread()`
4. `async.Thread()`
5. `multi.Thread()`

Задание 9. Какой ключевой слово используется для определения асинхронной функции в Python?

Ответы:

1. `async def`
2. `def async`
3. `await def`
4. `define async`
5. `async function`

Задание 10. Какой метод используется для создания окна в Tkinter?

Ответы:

1. `Tk()`
2. `Window()`
3. `Frame()`
4. `Canvas()`
5. `App()`

Задание 11. Какой метод позволяет загрузить изображение в Pygame?

Ответы:

1. `pygame.image.load("filename")`
2. `pygame.load.image("filename")`
3. `pygame.image.open("filename")`
4. `pygame.image.import("filename")`
5. `pygame.image.get("filename")`

Задание 12. Какой метод используется для обновления экрана в Pygame?

Ответы:

1. `pygame.display.update()`
2. `pygame.refresh()`
3. `pygame.display.flip()`
4. `pygame.update.display()`
5. `pygame.redraw()`

Задание 13. Какой метод используется для обработки событий в Pygame?

Ответы:

1. `pygame.event.get()`
2. `pygame.events.poll()`
3. `pygame.event.handle()`
4. `pygame.event.process()`
5. `pygame.event.listen()`

Задание 14. Какой метод устанавливает цвет фона в Tkinter?

Ответы:

1. `window.config(bg="color")`
2. `window.set_background("color")`
3. `window.bgcolor("color")`

4. `window.color("color")`
5. `window.background("color")`

Задание 15. Какой метод запускает основной цикл обработки событий в Tkinter?

Ответы:

1. `mainloop()`
2. `startloop()`
3. `runloop()`
4. `eventloop()`
5. `processloop()`

Задание 16. Какой модуль используется для работы с датами и временем в Python?

Ответы:

1. `datetime`
2. `time`
3. `calendar`
4. `dateutil`
5. `timespan`

Задание 17. Какой метод закрывает файл после работы с ним?

Ответы:

1. `file.close()`
2. `file.end()`
3. `file.quit()`
4. `file.exit()`
5. `file.stop()`

Задание 18. Какой метод позволяет создать новый поток выполнения в Python?

Ответы:

1. `threading.Thread(target=function)`
2. `threading.start_thread(function)`
3. `threading.new_thread(function)`
4. `threading.create_thread(function)`
5. `threading.run_thread(function)`

Задание 19. Какой метод используется для установки таймера в Pygame?

Ответы:

1. `pygame.time.set_timer(event, milliseconds)`
2. `pygame.timer.set(event, milliseconds)`
3. `pygame.set_timer(event, milliseconds)`
4. `pygame.time.delay(milliseconds)`
5. `pygame.timer.delay(event, milliseconds)`

Задание 20. Какой метод позволяет получить список всех файлов в директории?

Ответы:

1. `os.listdir(path)`
2. `os.files(path)`
3. `os.getfiles(path)`
4. `os.dir(path)`
5. `os.file_list(path)`

2.7. Список рекомендованной литературы

Основная:

1. Документация Python <https://www.python.org/doc/>
2. Руководство по началу работы с PyCharm <https://www.jetbrains.com/help/pycharm/getting-started.html>

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. PyCharm (интегрированная среда разработки) <https://www.jetbrains.com/pycharm/>