

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
«ДОМ ЮНОШЕСКОГО ТЕХНИЧЕСКОГО ТВОРЧЕСТВА»

ПРИНЯТО

на заседании педагогического совета
ГБУ ДО «ДЮТТ Челябинской области»
Протокол № 3 от «15» мая 2025 г.

УТВЕРЖДАЮ
Руководитель СП «Отдел по проектному
управлению и внешним коммуникациям»

Штраух Н.А.
Приказ № 396 «25» июня 2025 г.



ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ
ПРОГРАММА

«Программирование на языке Python (Проектная группа)»

Направленность: техническая

Уровень программы: продвинутый

Срок освоения программы: 1 год (144 часа)

Возрастная категория обучающихся: 14 - 17 лет

Автор-составитель: Ахмедзянов Дамирьян
Закирьянович,
педагог дополнительного образования

Челябинск
2025 г.

Оглавление

Раздел1. Комплекс основных характеристик программы:

1.1.Пояснительная записка	3
1.2.Сведения о программе.....	3
1.3.Цели и задачи программы.....	4
1.4. Содержание программы.....	5
1.5 Учебный план.....	10
1.6. Планируемые результаты.....	12

Раздел2. Комплекс организационно-педагогических условий:

2.1 Календарный учебный график.....	13
2.2.Условия реализации программы.....	13
2.3.Формы аттестации	14
2.4.Оценочные материалы.....	15
2.5.Методические материалы.....	17
2.6.Воспитательный компонент.....	19
2.7. Информационные ресурсы и литература.....	20

Приложение.....	22
-----------------	----

Раздел 1. Комплекс основных характеристик программы.

1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Программирование на языке Python (Проектная группа)» технической направленности, модифицированная, очной формы обучения, сроком реализации 1 месяц, для детей в возрасте 14-17 лет, вводный уровень освоения.

Проект программы составлен в соответствии с государственными требованиями к образовательным программам системы дополнительного образования детей на основе следующих нормативных документов:

- Федеральный закон "Об образовании в Российской Федерации" от 29.12.2012 N 273-ФЗ (последняя редакция);
- Федеральный закон Российской Федерации от 14.07.2022 г. № 295-ФЗ «О внесении изменений в Федеральный закон «Об образовании в Российской Федерации»;
- Приказ Министерства просвещения Российской Федерации от 27.07.2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- Распоряжение Правительства ЧО № 901-рп от 20.09.2022 г. "Об утверждении регионального плана мероприятий на 2022 – 2024 годы по реализации Концепции развития дополнительного образования детей до 2030 года" ;
- Приказ Министерства образования и науки Российской Федерации от 23.08.2017 г. № 816 «Порядок применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;
- Приказ Министерства науки и высшего образования Российской Федерации и Министерства просвещения Российской Федерации от 5 августа 2020 г. № 882/391 «Порядок организации и осуществления образовательной деятельности при сетевой форме реализации образовательных программ»;
- Методические рекомендации по реализации адаптированных дополнительных общеобразовательных программ, способствующих социально-психологической реабилитации, профессиональному самоопределению детей с ограниченными возможностями здоровья, включая детей-инвалидов, с учетом их особых образовательных потребностей (письмо Министерства образования и науки РФ от 29 марта 2016 г. № ВК-641/09);
- Методические рекомендации «Создание современного инклюзивного образовательного пространства для детей с ограниченными возможностями здоровья и детей-инвалидов на базе образовательных организаций, реализующих дополнительные общеобразовательные программы в субъектах Российской Федерации» (письмо Министерства просвещения Российской Федерации от 30 декабря 2022 года № АБ – 3924/06);
- Методические рекомендации по проектированию дополнительных общеразвивающих программ нового поколения (включая разноуровневые программы в области физической культуры и спорта) (утвержденные приказом ФГБУ «Федеральный центр организационно-методического обеспечения физического воспитания, 2021 год);
- Письмо министерства просвещения Российской Федерации от 19.08.2022 г. «Об адаптированных дополнительных общеразвивающих программах»;
- Концепция развития дополнительного образования детей до 2030 года (распоряжение Правительства РФ от 31 марта 2022 г. № 678-р);

- Паспорт национального проекта «Образование» (утвержденный президиумом Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам, протокол от 24.12.2018г. №16);
- Постановление Главного государственного санитарного врача Российской Федерации от 28.09.2020г. № 28 «Об утверждении СанПиН 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- Приказ Министерства просвещения Российской Федерации от 3 сентября 2019 года № 467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей»;
- Разработка и реализация раздела о воспитании в составе дополнительной общеобразовательной общеразвивающей программы. Методические рекомендации ФГБНУ «Институт изучения детства, семьи и воспитания» // Москва: Институт изучения детства, семьи и воспитания РАО, 2023.;
- Приказ Министерства труда и социальной защиты РФ № 652-н от 21.09.2021 г «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»;
- Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы) / Письмо Минобрнауки РФ от 18.11.2015 г. № 09-3242/;
- Письмо Министерства образования и науки РФ от 25.07.2016 № 09-1790 «Рекомендации по совершенствованию дополнительных образовательных программ, созданию детских технопарков, центров молодежного инновационного творчества и внедрению иных форм подготовки детей и молодежи по программам инженерной направленности»;
- Практические рекомендации о реализации образовательных программ с использованием дистанционных технологий /Письмо Мин. Просвещения от 16 ноября 2020 г. № ГД-2072/03/;
- Государственная программа Челябинской области «Развитие образования в Челябинской области» на 2018–2025 годы. / Постановление Правительства ЧО от 28.12.2017 г. № 732 – П/;
- Устав ГБУ ДО ДЮТТ /утвержден приказом Министерства образования и науки Челябинской области 29.09.2015 № 01/2769/;
- Положение о проведении текущего контроля успеваемости, промежуточной и итоговой аттестации результатов освоения обучающимися ДООП в ГБОУ ДО ДЮТТ/утверждено приказом директора ГБУ ДО ДЮТТ от 09.01.2019 г/;
- Положение о порядке разработки и реализации ДООП в ГБОУ ДО ДЮТТ / утверждено приказом директора ГБОУ ДО ДЮТТ № 142А от 01.06.2022 г/;
- Положение о реализации ДООП с применением дистанционных образовательных технологий в ГБОУ ДО ДЮТТ / утверждено приказом директора №103 от 24.09.2018 г./.
- Программа воспитания ГБУДО «Дом юношеского технического творчества «Челябинской области на 2023-2026 учебные годы /утверждено приказом директора №125 от 18 мая 2023г./

Срок реализации: программа рассчитана на 1 год обучения.

Общее количество часов в год: 144 часа

Общее количество занятий в год: 72

Количество часов в неделю: 2 акад. часа

Режим занятий: 2 раз в неделю.

Продолжительность образовательной деятельности устанавливается в соответствии с требованиями по регламенту и не превышает 1,5 часа. Через 40 минут перерыв 10 минут, могут проводиться физкультурные минутки, они могут соответствовать теме образовательной деятельности, в образовательную деятельность включаются зрительная гимнастика, речевая разминка, пальчиковая гимнастика.

Каждое занятие состоит из 2-х академических часов (по 1,5 часа) и 10 минутного перерыва. Занятия проводятся 1 раз в неделю.

Форма обучения: очная

Формы организации: групповое и индивидуально-групповое. В группе до 12 человек.

Виды занятий: практические. На практических занятиях педагог дополнительного образования использует различные формы занятий: игра, конкурс, творческая работа, творческий отчет, соревнования.

Метод обучения: наглядный, практический, объяснительно- иллюстративный.

1.2. Сведения о программе

Название программы	«Программирование на языке Python (Проектная группа)»
Возраст обучающихся	14-17 лет
Длительность программы (в часах)	144 часа
Количество занятий в неделю	2 занятия в неделю по 2 часа
Цель, задачи	Цель: Развитие проектной деятельности, практическое применение python, подготовить свое портфолио, научиться работать с технологиями, получить опыт в разработке новых проектов, предложение новых проектов, Задачи. Обучающие: • Понимать и способы решения задач на алгоритмы • Понимать концепцию работы интернета • Делать выборку данных через специальные конструкции • Работать с базами данных • Создавать сложные архитектуры приложений • Использовать инструменты веб- разработки • Понимать общую концепцию back-end приложений на базе джанго • Создавать ML- конструкции • Интегрировать различные сервисы в веб-приложения Метапредметные: • развить творческие способности и логическое мышление; • выявить и развить у детей природные задатки и способности, помогающие достичь успеха в техническом творчестве; • стимулировать интерес к смежным областям знаний: математике, геометрии, физике, биологии; • содействовать повышению привлекательности науки, научно-технического творчества для подрастающего поколения; • развить творческую активность через индивидуальное раскрытие технических способностей каждого ребенка; • развить навыки совместной работы, умения работать самостоятельно, мобилизуя необходимые ресурсы, правильно оценивая смысл и последствия своих действий; • развить креативное мышление и пространственное воображение. Воспитательные:

	• формировать интерес к практическому применению знаний, умений и навыков в повседневной жизни и в дальнейшем обучении; • поощрять целеустремленность, усердие, настойчивость, оптимизм, трудолюбие, аккуратность; • воспитать у учащихся стремление к получению качественного законченного результата; • привить навыки работы в группе; • поддерживать представление учащихся о значимости общечеловеческих нравственных ценностей, доброжелательности, сотрудничества; • прививать культуру организации рабочего места, дисциплину обращения со сложными и опасными инструментами; • воспитывать бережливость и сознательное отношение к вверенным материальным ценностям.
Краткое описание программы	Программа направлена на развитие качеств, помогающих обучающимся ориентироваться в современном мире информационных технологий, выполнять задачи различной сложности, самореализоваться в выбранном направлении. В обучении используются проблемный метод обучения, кейс-метод, проектная деятельность. Большая часть занятий направлена на решение практических задач. Аттестация по итогам освоения программы по программе проходит в виде защиты проектов и является результатом проектной деятельности
Первичные знания, необходимые для освоения программы	Продвинутый уровень освоения программы
Результат освоения	К концу года обучения обучающиеся Будут знать: • способы решения задач на алгоритмы • концепцию работы интернета • общую концепцию back-end приложений на базе джанго • ML- конструкции Будут уметь: • Делать выборку данных через специальные конструкции • Работать с базами данных: • Создавать сложные архитектуры приложений • Интегрировать различные сервисы в веб-приложения У обучающихся будут сформированы: • представление о грамотном составлении технического задания • представления проекта; • навыки работы с интерфейсом; • навыки самостоятельной работы и работы в команде; • мастерство презентации готового продукта.

Перечень соревнований, в которых учащиеся смогут принять участие	Техноолимп, Технострелка, НТО, Шаг в будущее, ЮТИ, региональные соревнования
Перечень основного оборудования, необходимого для освоения программы	Компьютеры 16шт; Стулья 16 шт; Столы 16 шт; Камеры 16 шт; ПО (PowerPoint, Figma, Blender, Нейросети(ChatGPT, DeepSeek)); Доска
Преимущества данной программы (отличия от других подобных курсов)	На сегодняшний день навыки веб-разработки имеют высокий спрос в различных сферах деятельности. Программа " Программирование на языке Python (Проектная группа)" предоставляет следующие преимущества: Профессиональное становление: • Углубленное освоение востребованных технологий (асинхронное программирование, микросервисы, ML) • Формирование навыков уровня Senior-разработчика • Подготовка к реальным задачам в IT-компаниях Практико-ориентированный подход: • Работа с промышленными кейсами (highload, оптимизация, рефакторинг) • Разработка сложных распределенных систем • Решение нетривиальных алгоритмических задач Развитие профессиональных компетенций: • Архитектурное мышление и проектирование систем • Навыки performance-оптимизации • Умение работать с legacy-кодом • Командные практики (code review, CI/CD) Доступность технологий: • Использование open-source инструментов (Docker, Kafka, Redis) • Бесплатные облачные решения для обучения • Возможность дальнейшего самостоятельного развития Актуальность содержания: • Изучение современных фреймворков и подходов • Практика работы с актуальными технологическими стеками • Подготовка к требованиям IT-рынка 2024-2025 годов Гибкость обучения: • Возможность адаптации под индивидуальные проекты

	• Разнообразие практических кейсов (от веб-разработки до Data Science) • Поддержка наставников при реализации собственных идей
--	---

1.3 Цели и задачи программы

Цель: Развитие проектной деятельности, практическое применение python, подготовить свое портфолио, научиться работать с технологиями, получить опыт в разработке новых проектов, предложение новых проектов,

Задачи.

Обучающие:

- Понимать и способы решения задач на алгоритмы
- Понимать концепцию работы интернета
- Делать выборку данных через специальные конструкции
- Работать с базами данных
- Создавать сложные архитектуры приложений
- Использовать инструменты веб- разработки
- Понимать общую концепцию back-end приложений на базе джанго
- Создавать ML- конструкции
- Интегрировать различные сервисы в веб-приложения

Метапредметные:

- развить творческие способности и логическое мышление;
- выявить и развить у детей природные задатки и способности, помогающие достичь успеха в техническом творчестве;
- стимулировать интерес к смежным областям знаний: математике, геометрии, физике, биологии;
- содействовать повышению привлекательности науки, научно-технического творчества для подрастающего поколения;
- развить творческую активность через индивидуальное раскрытие технических способностей каждого ребенка;
- Развить креативное мышление и навыки командной работы.
- Научить структурировать информацию и эффективно представлять идеи.
- Развить навыки работы с современными онлайн-инструментами и платформами.развить креативное мышление и пространственное воображение.

Воспитательные:

- формировать интерес к практическому применению знаний, умений и навыков в повседневной жизни и в дальнейшем обучении;
- поощрять целеустремленность, усердие, настойчивость, оптимизм, трудолюбие, аккуратность;
- воспитать у учащихся стремление к получению качественного законченного результата;
- привить навыки работы в группе;
- поддерживать представление учащихся о значимости общечеловеческих нравственных ценностей, доброжелательности, сотрудничества;
- прививать культуру организации рабочего места, дисциплину обращения со сложными и опасными инструментами;
- воспитывать бережливость и сознательное отношение к вверенным материальным ценностям.

1.4 Содержание программы

Модуль 1. Введение в программу

Тема 1.1. Повторение пройденного материала

Теоретическая часть: Основные понятия и принципы работы с микроконтроллерами, изученные ранее: устройство Arduino Uno, виды датчиков и исполнительных механизмов, типы сигналов (аналоговые и цифровые), работа с библиотеками, программирование на языке C/C++ в среде Arduino IDE. Повторение принципов построения электрических цепей, назначения основных радиоэлементов (светодиоды, резисторы, кнопки, датчики и т.д.).

Практическая часть: Сборка простой электронной схемы на основе ранее изученных компонентов. Написание и загрузка программы на Arduino для управления светодиодом, считывания данных с кнопки или датчика. Проверка работоспособности устройства и отладка кода.

Модуль 2. Решение задач алгоритмов

Тема 2.1. Обзор платформы для решения задач

Теоретическая часть: Введение в онлайн-платформы для решения программистских задач (например, Stepik, Codewars, Яндекс.Контест, LeetCode). Интерфейс, структура заданий, система проверки решений, рейтинговая система.

Практическая часть: Регистрация на одной из платформ. Освоение интерфейса. Выполнение первого тестового задания.

Тема 2.2. Концепция решения задач

Теоретическая часть: Что такое алгоритмическая задача. Структура входных и выходных данных. Чтение условий. Анализ примеров. Подходы к разработке решения.

Практическая часть: Разбор и решение простой задачи на понимание условия и формата вывода.

Тема 2.3. Решение первых задач на ввод и вывод данных

Теоретическая часть: Форматы ввода и вывода данных. Способы чтения данных в Python, C++, Java и других языках. Особенности работы с потоками ввода/вывода.

Практическая часть: Решение задач на чтение чисел, строк, вывод результатов по заданному формату.

Тема 2.4. Решение задач на математические выражения

Теоретическая часть: Применение арифметических операций, приоритеты вычислений, использование математических функций.

Практическая часть: Решение задач на расчёт значений выражений, применение библиотек math, cmath.

Тема 2.5. Решение задач на циклы. Решение задач первого порядка

Теоретическая часть: Цикл for и while. Использование циклов для обработки последовательностей, подсчёта суммы, произведения и т. д.

Практическая часть: Решение задач на подсчёт количества элементов, сумму чисел, факториал.

Тема 2.6. Решение задач на циклы. Решение задач второго порядка

Теоретическая часть: Вложенные циклы. Алгоритмы перебора. Прерывание и продолжение цикла (break, continue).

Практическая часть: Решение задач на поиск делителей, проверку простоты числа, генерацию таблиц умножения и т. п.

Тема 2.7. Решение задач на массивы. Решение простых задач

Теоретическая часть: Что такое массив. Индексация. Одномерные массивы. Перебор элементов.

Практическая часть: Решение задач на поиск максимума, минимума, среднего значения, количество элементов по условию.

Тема 2.8. Решение задач на массивы. Решение задач повышенной сложности

Теоретическая часть: Изменение массива в процессе обработки. Удаление и вставка элементов. Поиск пар, подмассивов.

Практическая часть: Решение задач на сортировку, фильтрацию, модификацию массивов.

Тема 2.9. Решение задач на массивы. Многоуровневые массивы

Теоретическая часть: Двумерные и многомерные массивы. Их представление в памяти. Принципы работы.

Практическая часть: Решение задач на работу с матрицами: сумма, диагонали, поворот, транспонирование.

Тема 2.10. Дополнительные функции, помогающие решать задачи

Теоретическая часть: Полезные встроенные функции языка (например, map, filter, zip, enumerate в Python). Библиотечные функции.

Практическая часть: Использование встроенных функций для оптимизации кода и упрощения логики.

Тема 2.11. Решение задач на строки

Теоретическая часть: Строки как тип данных. Операции над строками: конкатенация, индексация, срезы, методы строк.

Практическая часть: Решение задач на подсчёт символов, замену частей строки, анализ содержания.

Тема 2.12. Решение задач на строки: запись, перезапись

Теоретическая часть: Работа с регистром. Преобразование типов. Поиск подстрок.

Практическая часть: Решение задач на изменение регистра, форматирование текста, создание новых строк на основе старых.

Тема 2.13. Функциональное программирование

Теоретическая часть: Понятие функционального программирования. Чистые функции. Лямбда-выражения. Каррирование.

Практическая часть: Реализация задач с использованием функций высшего порядка и лямбд.

Тема 2.14. Паттерны программирования: DRY, KISS, SOLID и другие

Теоретическая часть: Основные принципы проектирования кода: DRY (Don't Repeat Yourself), KISS (Keep It Simple, Stupid), SOLID.

Практическая часть: Рефакторинг существующего кода под соблюдение принципов. Анализ качества кода.

Тема 2.15. Повторение ООП

Теоретическая часть: Основные понятия объектно-ориентированного программирования: класс, объект, наследование, инкапсуляция, полиморфизм.

Практическая часть: Создание простых классов и работа с ними.

Тема 2.16. Решение задач на ООП

Теоретическая часть: Применение ООП в реальных задачах. Связь между классами. Наследование и композиция.

Практическая часть: Решение задач, требующих создания нескольких взаимодействующих классов.

Тема 2.17. Операторы перегрузки

Теоретическая часть: Что такое перегрузка операторов. Возможности изменения поведения стандартных операций для пользовательских типов.

Практическая часть: Реализация классов с перегруженными операторами (например, +, -, ==, <<).

Тема 2.18. Функциональное программирование с использованием ООП

Теоретическая часть: Комбинация функционального подхода и ООП. Использование функций как атрибутов классов.

Практическая часть: Реализация проектов или задач, где используются оба парадигма.

Тема 2.19. Решение задач на платформе повышенной сложности

Теоретическая часть: Подходы к решению олимпиадных задач. Алгоритмы и структуры данных. Оценка сложности.

Практическая часть: Решение задач повышенной сложности на выбранной платформе (Codewars, LeetCode, Codeforces и др.).

Модуль 3. Функциональное программирование

Тема 3.1. Принцип работы ОС

Теоретическая часть: Что такое операционная система. Основные функции ОС: управление процессами, памятью, устройствами и файловой системой. Виды операционных систем (Windows, Linux, macOS). Роль ядра и драйверов.

Практическая часть: Изучение информации о системе через интерфейс операционной системы. Обзор дисков, процессора, памяти. Запуск и завершение процессов.

Тема 3.2. Работа с терминалом

Теоретическая часть: Что такое терминал и командная строка. Базовые команды для навигации по файловой системе, управления файлами и процессами. Различия между оболочками (bash, cmd, PowerShell).

Практическая часть: Выполнение базовых команд в терминале: создание, копирование, удаление файлов и папок. Перемещение по директориям. Автодополнение и история команд.

Тема 3.3. Чтение и запись файлов

Теоретическая часть: Файловая система. Типы файлов: текстовые, бинарные. Открытие, чтение и запись файлов в различных языках программирования. Режимы открытия файла.

Практическая часть: Написание программы, которая считывает данные из одного файла и записывает их в другой с определённой обработкой (например, подсчёт строк или слов).

Тема 3.4. Работа с браузером

Теоретическая часть: Устройство веб-браузера. Как происходит загрузка страниц. Консоль разработчика, элементы страницы (HTML, CSS, JS).

Практическая часть: Использование инструментов разработчика (DevTools) для просмотра структуры сайта, изменения стилей и выполнения JavaScript-скриптов на лету.

Модуль 4. Парсинг данных

Тема 4.1. Принцип работы браузера и его инструменты

Теоретическая часть: Как работает веб-браузер: обработка HTML, CSS и JavaScript. Жизненный цикл загрузки страницы. Инструменты разработчика (DevTools): элементы, сеть, консоль, Sources.

Практическая часть: Изучение структуры сайта с помощью DevTools. Анализ сетевых запросов. Изменение стилей и отслеживание загрузки ресурсов.

Тема 4.2. Программы-пауки, обзор и принцип работы

Теоретическая часть: Что такое веб-скрепинг и программы-пауки. Как происходит сбор данных с сайтов. Этические и юридические аспекты парсинга.

Практическая часть: Запуск простого скрипта-паука, который собирает ссылки или текст с заданной страницы.

Тема 4.3. Добавление библиотеки BeautifulSoup. Первый парсинг

Теоретическая часть: Обзор библиотеки BeautifulSoup. Возможности по анализу и извлечению данных из HTML и XML.

Практическая часть: Установка и подключение BeautifulSoup. Написание первого парсера, извлекающего заголовки или текст со страницы.

Тема 4.4. API. Понятие, функционал

Теоретическая часть: Что такое API. Виды API (REST, SOAP, GraphQL). Как клиент взаимодействует с сервером. Форматы передачи данных (JSON, XML).

Практическая часть: Отправка GET-запроса к публичному API. Анализ ответа. Использование Postman или Python для тестирования.

Тема 4.5. Создание парсера через API

Теоретическая часть: Преимущества использования API вместо парсинга HTML. Как найти и использовать документацию к API. Работа с ключами и токенами.

Практическая часть: Создание программы, получающей данные через API и обрабатывающей их (например, прогноз погоды, курсы валют, товары с площадки).

Тема 4.6. Проект: Отбор информации с торговых площадок

Теоретическая часть: Цели и задачи проекта. Выбор источников данных. Сравнение методов парсинга и работы через API. Подготовка к сбору данных.

Практическая часть: Реализация парсера или API-интерфейса для получения данных о товарах с популярных торговых площадок. Сохранение результатов в файл или базу данных.

Модуль 5. Введение базы данных

Тема 5.1. Введение в базы данных. Основная информация

Теоретическая часть: Что такое база данных. Виды баз данных: реляционные и нереляционные. Основные понятия: таблицы, записи, поля, ключи. Применение баз данных в реальных задачах.

Практическая часть: Обзор примеров использования баз данных (магазины, соцсети, системы учета). Обсуждение структуры простой БД.

Тема 5.2. Установка и обзор интерфейса

Теоретическая часть: Обзор СУБД: MySQL, SQLite, PostgreSQL, phpMyAdmin. Установка и настройка локальной среды. Интерфейсы работы с БД: графические и командная строка.

Практическая часть: Установка и запуск СУБД (например, SQLite или MySQL). Освоение основных функций интерфейса.

Тема 5.3. Создание, удаление, обновление таблиц. Команды CREATE, DELETE, UPDATE

Теоретическая часть: Язык SQL. Основные команды для управления структурой и данными: CREATE TABLE, DROP TABLE, DELETE, UPDATE. Типы данных в таблицах.

Практическая часть: Создание нескольких таблиц, добавление полей. Выполнение операций удаления и обновления записей.

Тема 5.4. Вывод информации. SELECT, WHERE, ORDER BY и другие

Теоретическая часть: Команда SELECT. Фильтрация данных с помощью WHERE. Сортировка (ORDER BY), ограничение вывода (LIMIT).

Практическая часть: Выполнение запросов к таблицам для получения нужной информации. Формирование выборки по заданным условиям.

Тема 5.5. Создание проекта «Программа-паук»

Теоретическая часть: Цель проекта: сбор данных из интернета и сохранение их в базу данных. Обзор этапов: парсинг страниц, обработка данных, загрузка в БД.

Практическая часть: Реализация программы-паука, которая собирает данные с сайта и сохраняет их в созданную ранее базу данных.

Тема 5.6. Промежуточная аттестация

Теоретическая часть: Обобщение пройденного материала. Критерии оценки знаний: работа с SQL, создание таблиц, выполнение запросов, использование БД в проектах.

Практическая часть: Выполнение практического задания по созданию базы данных и выполнению SQL-запросов. Защита проекта «Программа-паук».

Модуль 6. Архитектура данных

Тема 6.1. Понятие архитектуры, и её виды

Теоретическая часть: Что такое архитектура программной системы. Виды архитектур: монолит, микросервисы, клиент-сервер, MVC, MVP, MVVM. Преимущества и недостатки каждой.

Практическая часть: Примеры реальных проектов с разными типами архитектур. Обсуждение выбора архитектуры для будущего проекта.

Тема 6.2. Способы создания архитектуры

Теоретическая часть: Методы проектирования архитектуры: диаграммы UML, DFD, ERD. Определение модулей, связей между ними, взаимодействие компонентов.

Практическая часть: Создание диаграммы архитектуры будущего приложения (например, интернет-магазина) в графическом редакторе или на бумаге.

Тема 6.3. Понятие FOREIGN KEY. Создание взаимосвязей между таблицами

Теоретическая часть: Что такое внешний ключ (FOREIGN KEY). Типы связей: один к одному, один ко многим, многие ко многим. Целостность данных.

Практическая часть: Создание нескольких связанных таблиц в БД и установка между ними отношений с помощью FOREIGN KEY.

Тема 6.4. Создание таблиц под приложение. Архитектура приложений

Теоретическая часть: Как структурировать базу данных под конкретное приложение. Соответствие таблиц бизнес-логике. Проектирование с учетом масштабируемости.

Практическая часть: Создание таблиц для будущего интернет-магазина: пользователи, товары, категории, заказы.

Тема 6.5. Создание проекта «Интернет-магазин». Концепция архитектуры

Теоретическая часть: Формирование общей концепции интернет-магазина: модули, функционал, роли пользователей, источники данных.

Практическая часть: Создание технического задания и архитектурной схемы проекта.

Тема 6.6. Создание проекта «Интернет-магазин». Создание таблиц и взаимосвязей

Теоретическая часть: Реализация логической модели в виде физических таблиц. Назначение полей, типы данных, индексы, связи.

Практическая часть: Подключение СУБД и создание всех необходимых таблиц с правильными связями.

Тема 6.7. Создание проекта «Интернет-магазин». Описание логики

Теоретическая часть: Логика работы интернет-магазина: регистрация, авторизация, добавление товаров, оформление заказа, управление администратором.

Практическая часть: Описание и документирование алгоритмов работы основных функций проекта.

Тема 6.8. Создание проекта «Интернет-магазин». Создание аккаунтов: user, admin

Теоретическая часть: Роли пользователей в системе. Управление доступом. Хранение паролей. Разграничение прав.

Практическая часть: Реализация регистрации и авторизации пользователей с разными уровнями доступа (user, admin).

Тема 6.9. Создание таблицы «Заказов»

Теоретическая часть: Структура заказа: поля, статусы, связи с пользователями и товарами.

Практическая часть: Создание таблицы orders с необходимыми полями и внешними ключами.

Тема 6.10. Проверка обработки заказов

Теоретическая часть: Жизненный цикл заказа. Обработка, изменение статуса, отмена.

Практическая часть: Тестирование работы с заказами: добавление, просмотр, изменение статуса через интерфейс или SQL-запросы.

Тема 6.11. Логика системы защиты пользователей платформы

Теоретическая часть: Безопасность веб-приложений. Защита паролей (хэширование), защита от SQL-инъекций, XSS-атак. Использование сессий и токенов.

Практическая часть: Добавление хэширования паролей, проверка безопасности формы входа и регистрации.

Тема 6.12. Оптимизация, асинхронные действия программ

Теоретическая часть: Асинхронное программирование. Зачем нужны асинхронные задачи. Использование очередей, фоновых процессов.

Практическая часть: Реализация асинхронного импорта данных или отправки уведомлений в проекте интернет-магазина.

Модуль 7. Создание Web-приложений

Тема 7.1. Принципы создания веб-приложений

Теоретическая часть: Что такое веб-приложение. Структура клиент-сервер. Работа HTTP-запросов и ответов. Разница между статическими и динамическими сайтами.

Практическая часть: Обзор примеров веб-приложений. Обсуждение этапов разработки от дизайна до запуска.

Тема 7.2. Шаблоны HTML и CSS

Теоретическая часть: Основы HTML и CSS. Структура HTML-документа. Использование тегов, классов, id. Верстка страниц с использованием CSS.

Практическая часть: Создание простой HTML-страницы с подключенным CSS-файлом и базовым оформлением.

Тема 7.3. Шаблоны HTML и CSS. Оживленные анимации

Теоретическая часть: Введение в CSS-анимацию. Ключевые кадры (@keyframes), переходы и эффекты. Адаптивность и отзывчивый дизайн.

Практическая часть: Добавление анимаций на страницу: кнопки, заголовки, элементы интерфейса.

Тема 7.4. Django. Обзор фреймворка. Первая программа

Теоретическая часть: Что такое Django. MVC/MVT-архитектура. Возможности фреймворка: ORM, админка, маршрутизация, шаблонизатор.

Практическая часть: Установка Django. Создание первого проекта и приложения. Запуск сервера и вывод "Hello World".

Тема 7.5. Маршрутизация

Теоретическая часть: Как работает маршрутизация в Django. URL-паттерны. Подключение представлений к адресам.

Практическая часть: Настройка маршрутов для нескольких страниц сайта. Реализация динамических URL.

Тема 7.6. Шаблонизатор Jinja

Теоретическая часть: Что такое шаблонизация. Принцип работы шаблонизатора в Django (Jinja). Передача данных из views в шаблоны.

Практическая часть: Создание HTML-шаблона и передача в него переменных из view-функции.

Тема 7.7. ORM Django. Создание первой модели

Теоретическая часть: Что такое ORM. Преимущества использования ORM вместо SQL. Определение моделей в Django.

Практическая часть: Создание первой модели (например, Product) и миграции. Работа с базой через Python-код.

Тема 7.8. Модели. Взаимодействие с моделями. CRUD-команды

Теоретическая часть: Основные операции с данными: Create, Read, Update, Delete. Фильтрация и запросы к БД через ORM.

Практическая часть: Реализация CRUD-операций в Django: добавление, просмотр, редактирование и удаление записей.

Тема 7.9. Работа с API. Подключение таблиц в PostgreSQL

Теоретическая часть: Интеграция Django с PostgreSQL. Настройка базы данных. Работа с REST API в Django (DRF — Django REST Framework).

Практическая часть: Подключение PostgreSQL. Создание API для одной из моделей интернет-магазина.

Тема 7.10. Создание сайта интернет-магазина по готовой архитектуре

Теоретическая часть: Реализация интернет-магазина на основе ранее спроектированной архитектуры. Подключение всех модулей: пользователи, товары, заказы.

Практическая часть: Создание основных страниц сайта: главная, каталог товаров, корзина, профиль пользователя.

Тема 7.11. Создание сайта интернет-магазина. Добавление шаблона

Теоретическая часть: Интеграция HTML-шаблонов в Django. Использование базового шаблона и наследования.

Практическая часть: Подключение созданного шаблона ко всем страницам интернет-магазина.

Тема 7.12. Хостинг сайта

Теоретическая часть: Что такое хостинг и домен. Виды хостинга. Подготовка проекта к публикации: настройка баз данных, статика, безопасности.

Практическая часть: Заливка сайта на хостинг (например, PythonAnywhere, Render, Heroku). Проверка работоспособности сайта в сети.

Модуль 8. Инструменты для разработки

Тема 8.1. Понятие Git, основные команды

Теоретическая часть: Что такое система контроля версий Git. Основные понятия: репозиторий, коммит, ветка, слияние. Основные команды: init, add, commit, push, pull, branch, merge.

Практическая часть: Создание локального репозитория. Выполнение базовых операций с ветками и коммитами. Загрузка проекта на GitHub.

Тема 8.2. Docker. Деплой сайта. Контейнеризация

Теоретическая часть: Что такое контейнеризация. Принцип работы Docker. Образы и контейнеры. Преимущества использования Docker для разработки и деплоя.

Практическая часть: Установка Docker. Создание Dockerfile для интернет-магазина. Сборка образа и запуск контейнера.

Тема 8.3. Дополнительные инструменты в веб-разработке

Теоретическая часть: Полезные инструменты для веб-разработки: Postman, Figma, Swagger, Chrome DevTools, VSCode Extensions, CI/CD.

Практическая часть: Работа с одним или несколькими инструментами: тестирование API в Postman, создание прототипа в Figma, отладка кода через DevTools.

Модуль 9. Машинное обучение

Тема 9.1. Введение в машинное обучение

Теоретическая часть: Что такое машинное обучение. Виды обучения: обучение с учителем, без учителя, обучение с подкреплением. Примеры задач и приложений.

Практическая часть: Обзор библиотек (scikit-learn, TensorFlow). Первый пример обучения модели на простом наборе данных.

Тема 9.2. Работа с изображениями: фильтры, обнаружение границ

Теоретическая часть: Основы компьютерного зрения. Форматы изображений. Фильтрация изображений. Алгоритмы обнаружения границ (Canny, Sobel).

Практическая часть: Реализация фильтров и обнаружения границ на изображении с помощью OpenCV.

Тема 9.3. Распознавание несложных объектов

Теоретическая часть: Методы распознавания объектов: цветовые диапазоны, формы, шаблоны. Использование пороговой обработки и контуров.

Практическая часть: Написание программы, которая распознаёт геометрические фигуры или объекты определённого цвета на изображении.

Тема 9.4. Детектирование объектов через YOLO

Теоретическая часть: Что такое YOLO (You Only Look Once). Принцип работы алгоритма детекции объектов в реальном времени.

Практическая часть: Установка и запуск YOLOv5/v8. Детектирование объектов на изображениях и видео.

Тема 9.5. Работа с камерой. Обработка машинным зрением в реальном времени

Теоретическая часть: Работа с видеопотоком. Получение кадров с камеры. Обработка в реальном времени.

Практическая часть: Подключение веб-камеры. Обнаружение и отслеживание объектов в реальном времени с использованием OpenCV и YOLO.

Тема 9.6. Реализация проекта «Сайт с обработкой фото» (1 часть)

Теоретическая часть: Цель проекта: создать сайт, принимающий изображения и обрабатывающий их с помощью машинного зрения. Построение архитектуры.

Практическая часть: Создание интерфейса загрузки изображения. Настройка серверной части для получения файла.

Тема 9.7. Реализация проекта «Сайт с обработкой фото» (2 часть)

Теоретическая часть: Интеграция модели машинного обучения в веб-приложение. Отправка результата на клиентскую часть.

Практическая часть: Подключение модели к сайту. Обработка изображения и вывод результата пользователю.

Тема 9.8. Внедрение правок в проекте

Теоретическая часть: Анализ обратной связи. Поиск и исправление ошибок. Тестирование и оптимизация производительности.

Практическая часть: Внесение доработок в сайт: улучшение интерфейса, коррекция логики обработки, добавление новых функций.

Тема 9.9. Итоговая аттестация

Теоретическая часть: Обобщение всего пройденного материала. Критерии оценки: работа с Git, использование Docker, реализация веб-приложения, применение машинного обучения.

Практическая часть: Защита итогового проекта. Демонстрация работы сайта с обработкой изображений. Ответы на вопросы.

1.5 Учебный план

№ п/п	Название модуля, темы	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
Модуль 1. Введение в программу		2	1	1	
1	Тема 1.1 Повторение пройденного материала	2	1	1	Педагогическое наблюдение, решение задач
Модуль 2. Решение задач алгоритмов		38	18	20	
2	Тема 2.1 Обзор платформы для решения задач	2	1	1	Текущий: Педагогическое наблюдение, решение задач

3	Тема 2.2 Концепция решения задач	2	1	1	Текущий: Педагогическое наблюдение, решение задач
4	Тема 2.3 Решение первых задач на ввод и вывод данных	2	1	1	Текущий: Педагогическое наблюдение, решение задач
5	Тема 2.4 Решение задач на математические выражения	2	1	1	Текущий: Педагогическое наблюдение, решение задач
6	Тема 2.5 Решение задач на циклы. Решение задач первого порядка	2	1	1	Текущий: Педагогическое наблюдение, решение задач
7	Тема 2.6 Решение задач на циклы. Решение задач второго порядка	2	1	1	Текущий: Педагогическое наблюдение, решение задач
8	Тема 2.7 Решение задач на массивы. Решение простых задач.	2	1	1	Текущий: Педагогическое наблюдение, решение задач
9	Тема 2.8 Решение задач на массивы. Решение задач повышенной сложности.	2	1	1	Текущий: Педагогическое наблюдение, решение задач
10	Тема 2.9 Решение задач на массивы. Многоуровневые массивы	2	1	1	Текущий: Педагогическое наблюдение, решение задач
11	Тема 2.10 Дополнительные функции помогающие решать задачи	2	1	1	Текущий: Педагогическое наблюдение, решение задач
12	Тема 2.11 Решение задач на строки	2	1	1	Текущий: Педагогическое наблюдение, решение задач
13	Тема 2.12 Решение задач на строки запись, перезапись	2	1	1	Текущий: Педагогическое наблюдение, решение задач
14	Тема 2.13 Функциональное программирование	2	1	1	Текущий: Педагогическое наблюдение, решение задач
15	Тема 2.14 Паттерны программирования SOAP, DRY итд	2	1	1	Текущий: Педагогическое наблюдение, решение задач
16	Тема 2.15 Повторение ООП	2	1	1	Текущий: Педагогическое наблюдение, решение задач
17	Тема 2.16 Решение задач на ООП	2	1	1	Текущий: Педагогическое наблюдение, решение задач
18	Тема 2.17 Операторы перегрузки	2	1	1	Текущий: Педагогическое наблюдение, решение задач
19	Тема 2.18 Функциональное программирование с использованием ООП	2	1	1	Текущий: Педагогическое наблюдение, решение задач

20	Тема 2.19 Решение задач на платформе повышенной сложности	2	-	2	Текущий: Педагогическое наблюдение, решение задач
Модуль 3. Функциональное программирование		8	4	4	
21	Тема 3.1 Принцип работы ОС	2	1	1	Текущий: Педагогическое наблюдение, решение задач
22	Тема 3.2 Работа с терминалом	2	1	1	Текущий: Педагогическое наблюдение, решение задач
23	Тема 3.3 Чтение и запись файлов	2	1	1	Текущий: Педагогическое наблюдение, решение задач
24	Тема 3.4 Работа с браузером	2	1	1	Текущий: Педагогическое наблюдение, решение задач
Модуль 4. Парсинг данных		12	5	7	
25	Тема 4.1 Принцип работы браузера и его инструменты	2	1	1	Текущий: Педагогическое наблюдение, решение задач
26	Тема 4.2 Программы пауки, обзор и принцип работы	2	1	1	Текущий: Педагогическое наблюдение, решение задач
27	Тема 4.3 Добавление библиотеки beautifulsoupпервый парсинг	2	1	1	Педагогическое наблюдение, решение задач
28	Тема 4.4 API понятие, функционал	2	1	1	Педагогическое наблюдение, решение задач
29	Тема 4.5 Создание парсера через API	2	1	1	Педагогическое наблюдение, решение задач
30	Тема 4.6 Проект отбор информации с торговых площадок	2	-	2	Педагогическое наблюдение, решение задач
Модуль 5. Введение базы данных		12	5	7	Педагогическое наблюдение, решение задач
31	Тема 5.1 Введение в базы данных основная информация	2	1	1	Педагогическое наблюдение, решение задач
32	Тема 5.2 Установка и обзор интерфейса	2	1	1	Педагогическое наблюдение, решение задач
33	Тема 5.3 Создание, удаление обновление таблиц команды Create delete update	2	1	1	Текущий: Педагогическое наблюдение, решение задач
34	Тема 5.4 Вывод информации select, while, desc итд	2	1	1	Педагогическое наблюдение, решение задач
35	Тема 5.5 Создание проекта «Программа пауков»	2	1	1	Педагогическое наблюдение, решение задач

36	Тема 5.6 Промежуточная аттестация	2	-	2	Промежуточный: защита проектов
Модуль 6. Архитектура данных		24	12	12	
37	Тема 6.1 Понятие архитектуры, и ее виды	2	1	1	Педагогическое наблюдение, решение задач
38	Тема 6.2 Способы создания архитектуры	2	1	1	Педагогическое наблюдение, решение задач
39	Тема 6.3 Понятие FOREIGN KEY Создание взаимосвязей между таблиц	2	1	1	Педагогическое наблюдение, решение задач
40	Тема 6.4 Создание таблиц под приложения, архитектура приложений.	2	1	1	Педагогическое наблюдение, решение задач
41	Тема 6.5 Создание проекта интернет-магазина Концепция архитектуры	2	1	1	Педагогическое наблюдение, решение задач
42	Тема 6.6 Создание проекта интернет-магазина Создание таблиц и взаимосвязей	2	1	1	Текущий: Педагогическое наблюдение
43	Тема 6.7 Создание проекта интернет-магазина Описание логики	2	1	1	Текущий: Педагогическое наблюдение, решение задач
44	Тема 6.8 Создание проекта интернет-магазина Создание аккаунтов user, admin	2	1	1	Текущий: Педагогическое наблюдение, решение задач
45	Тема 6.9 Создание таблицы Заказов	2	1	1	Текущий: Педагогическое наблюдение, решение задач
46	Тема 6.10 Проверка обработки заказов	2	1	1	Текущий: Педагогическое наблюдение, решение задач
47	Тема 6.11 Логика системы защиты пользователей платформы	2	1	1	Текущий: Педагогическое наблюдение, решение задач
48	Тема 6.12 Оптимизация, асинхронные действия программ	2	1	1	Текущий: Педагогическое наблюдение, решение задач
Модуль 7. Создание Web-приложений		24	12	12	
49	Тема 7.1 Принципы создания Веб-приложений	2	1	1	Педагогическое наблюдение, решение задач
50	Тема 7.2 Шаблоны HTML и CSS	2	1	1	Педагогическое наблюдение, решение задач
51	Тема 7.3 Шаблоны HTML и CSS	2	1	1	Педагогическое

	оживленные анимации				наблюдение, решение задач
52	Тема 7.4 Django, обзор фреймворка первая программа	2	1	1	Педагогическое наблюдение, решение задач
53	Тема 7.5 Маршрутизация	2	1	1	Педагогическое наблюдение, решение задач
54	Тема 7.6 Шаблонизатор Jinja	2	1	1	Педагогическое наблюдение, решение задач
55	Тема 7.7 ORM джанго, создание первой	2	1	1	Педагогическое наблюдение, решение задач
56	Тема 7.8 Модели, взаимодействие с моделями CRUD команды	2	1	1	Педагогическое наблюдение, решение задач
57	Тема 7.9 Работа с API Подключение таблиц в postgres	2	1	1	Педагогическое наблюдение, решение задач
58	Тема 7.10 Создание сайта интернет-магазина по готовой архитектуры	2	1	1	Педагогическое наблюдение, решение задач
59	Тема 7.11 Создание сайта интернет-магазина. Добавление шаблона	2	1	1	Педагогическое наблюдение, решение задач
60	Тема 7.12 Хостинг сайта	2	1	1	Педагогическое наблюдение, решение задач
Модуль 8. Инструменты для разработки		6	3	3	
61	Тема 8.1 Понятие GIT, основные команды	2	1	1	Педагогическое наблюдение, решение задач
62	Тема 8.2 Docker деплой сайта контейнеризация	2	1	1	Педагогическое наблюдение, решение задач
63	Тема 8.3 Дополнительные инструменты в веб-разработке	2	1	1	Педагогическое наблюдение, решение задач
Модуль 9. Машинное обучение		18	8	10	
64	Тема 9.1 Введение в машинное обучение	2	1	1	Педагогическое наблюдение, решение задач
65	Тема 9.2 Работа с изображениями: фильтры, обнаружение границ	2	1	1	Педагогическое наблюдение, решение задач
66	Тема 9.3 Распознавание несложных объектов	2	1	1	Педагогическое наблюдение, решение задач
67	Тема 9.4 Детектирование объектов через Yolo	2	1	1	Педагогическое наблюдение, решение задач

68	Тема 9.5 Работа с камерой обработка машинным зрением в реальном времени	2	1	1	Педагогическое наблюдение, решение задач
69	Тема 9.6 Реализация проекта сайт с обработкой фото (1 часть)	2	1	1	Педагогическое наблюдение, решение задач
70	Тема 9.7 Реализация проекта сайт с обработкой фото (2 часть)	2	1	1	Педагогическое наблюдение, решение задач
71	Тема 9.8 Внедрение правок в проекте	2	1	1	Педагогическое наблюдение, решение задач
72	Тема 9.9 Итоговая аттестация	2	-	2	Итоговый: Защита проектов
Итого		144	68	76	

1.6. Планируемые результаты

Обучающие:

- Понимать и способы решения задач на алгоритмы
- Понимать концепцию работы интернета
- Делать выборку данных через специальные конструкции
- Работать с базами данных
- Создавать сложные архитектуры приложений
- Использовать инструменты веб- разработки
- Понимать общую концепцию back-end приложений на базе джанго
- Создавать ML- конструкции
- Интегрировать различные сервисы в веб-приложения

Метапредметные:

- развить творческие способности и логическое мышление;
- выявить и развить у детей природные задатки и способности, помогающие достичь успеха в техническом творчестве;
- стимулировать интерес к смежным областям знаний: математике, геометрии, физике, биологии;
- содействовать повышению привлекательности науки, научно-технического творчества для подрастающего поколения;
- развить творческую активность через индивидуальное раскрытие технических способностей каждого ребенка;
- Развить креативное мышление и навыки командной работы.
- Научить структурировать информацию и эффективно представлять идеи.
- Развить навыки работы с современными онлайн-инструментами и платформами.развить креативное мышление и пространственное воображение.

Воспитательные:

- формировать интерес к практическому применению знаний, умений и навыков в повседневной жизни и в дальнейшем обучении;
- поощрять целеустремленность, усердие, настойчивость, оптимизм, трудолюбие, аккуратность;
- воспитать у учащихся стремление к получению качественного законченного результата;
- привить навыки работы в группе;

- поддерживать представление учащихся о значимости общечеловеческих нравственных ценностей, доброжелательности, сотрудничества;
- прививать культуру организации рабочего места, дисциплину обращения со сложными и опасными инструментами;
- воспитывать бережливость и сознательное отношение к вверенным материальным ценностям.

К концу года обучения обучающиеся

Будут знать:

- способы решения задач на алгоритмы
- концепцию работы интернета
- общую концепцию back-end приложений на базе джанго
- ML- конструкции

Будут уметь:

- Делать выборку данных через специальные конструкции
- Работать с базами данных:
- Создавать сложные архитектуры приложений
- Интегрировать различные сервисы в веб-приложения

У обучающихся будут сформированы:

- представление о грамотном составлении технического задания
- представления проекта;
- навыки работы с веб-ресурсом;
- навыки самостоятельной работы и работы в команде;
- мастерство презентации готового продукта.

Раздел 2. Комплекс организационно-педагогических условий. Условия реализации программы.

2.1. Календарный учебный график

Режим организации занятий по данной дополнительной общеобразовательной общеразвивающей программы определяется календарным учебным графиком и соответствует нормам, утвержденным «СанПин к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей» Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4. 3648-20

Год обучения	Всего учебных недель	Количество учебных часов	Режим занятий
1 год	72	144	2 раза в неделю по 2 часа.

2.2 Условия реализации программы

Для проведения занятий необходимо светлое помещение с хорошей вентиляцией. Помещение имеет хорошее освещение, т.к. недостаток света вызывает перенапряжение зрения и быстрое утомление учащихся. Поэтому рабочие места размещены так, чтобы при естественном освещении не было недостатков в дополнительных источниках света.

В процессе освоения программы используется оборудование:

Наименование	Количество (из расчета на 10-12 учащихся), шт.
Столы для учащихся, двухместные	16 (17)
Стол педагога	1
Стулья	16 (17)
Классная доска	1
Персональный компьютер	12 (13)
Доска	1
ПО (PowerPoint, Figma, Pycharm, PGAdmin, python 3.7>, Нейросети(ChatGPT, DeepSeek));	16

2.3 Формы аттестации

Для определения результатов освоения общеобразовательной общеразвивающей программы разработана система диагностического контроля, который предусматривает проверку уровня подготовки обучающихся на всех этапах обучения.

Виды контроля:

1. Входной контроль – цель оценка общего уровня подготовки каждого ребенка.

Для входного контроля используется анкетирование и/или собеседование.

Результаты входного тестирования используется для вывода о целесообразности редактирования планирования в части качества и/или введения дополнительных тем занятий для корректировки знаний и умений.

2. Промежуточная аттестация – осуществляется 3 раза в течении учебного года (по истечении 3 месяцев обучения по программе).

Формы промежуточного контроля и анализа результатов освоения программы, виды оценочных работ – в зависимости от уровня подготовки учащегося, от года обучения – это зачетные работы, аттестационные занятия, опрос и тестирование на усвоение теоретических знаний, обсуждение результатов выполнения определенных операций, самооценка и общий анализ выполненных конструкций.

3. Итоговая аттестация – завершающий этап обучения.

Основные методы контроля:

- Опрос – устный или письменный;
- Анализ процесса работы;
- Анализ готовой модели.

2.4 Оценочные материалы

Для определения результатов освоения общеобразовательной общеразвивающей программы разработана система диагностического контроля, который предусматривает проверку уровня подготовки обучающихся на всех этапах обучения.

Виды контроля:

4. Входной контроль – цель оценка общего уровня подготовки каждого ребенка. Для входного контроля используется анкетирование и/или собеседование.

Результаты входного тестирования используется для вывода о целесообразности редактирования планирования в части качества и/или введения дополнительных тем занятий для корректировки знаний и умений.

5. Промежуточная аттестация – осуществляется 3 раза в течении учебного года (по истечении 3 месяцев обучения по программе).

Формы промежуточного контроля и анализа результатов освоения программы, виды оценочных работ – в зависимости от уровня подготовки учащегося, от года обучения – это зачетные работы, аттестационные занятия, опрос и тестирование на усвоение теоретических знаний, обсуждение результатов выполнения определенных операций, самооценка и общий анализ выполненных конструкций.

6. Итоговая аттестация – завершающий этап обучения.

Основные методы контроля:

- Опрос – устный или письменный;
- Анализ процесса работы;
- Анализ готовой модели.

Для оценки уровня и степени усвоения материала используется рейтинговая (многобалльная) система оценивания. Каждый вид работы в зависимости от сложности оценивается определенным количеством баллов, до сведения обучаемых доводится информация о максимальном количестве баллов, которые они могут набрать. Во время промежуточной аттестации обучаемые предъявляют свою работу, педагог сам или совместно с другими обучающимися оценивают этапы работы, обсуждают положительные и отрицательные результаты по итогам. Что нового усвоили на занятиях, что было интересно, как они сами оценивают результаты своей деятельности.

Для оценивая используется многобалльная система (20 баллов и более) с выделением уровней достижений, уровни определяются таким образом: 85% полученных баллов и более – «высокий», 65-84% - «средний», менее 65% - «низкий:

- «элементарный»
- «низкий»
- «средний»
- «высокий»

Организация системы контроля

Вид контроля по этапам	Форма контроля	Тема диагностическо й работы	Что предлагается выявить
Входной	Анкетирование/ Собеседование	Комплектование групп	Базовые знания Python (ООП, работа с файлами, исключения) Опыт работы с Git и виртуальными окружениями Умение решать алгоритмические задачи (сложность LeetCode Medium) Понимание принципов работы баз данных (SQL/NoSQL)
1 промежуточная аттестация	Педагогическое наблюдение Практическая работа	Основы web-программирован ие	Понимание терминов веб-программирования, принципы прогроаммирования beck и front end разработки Сложные SQL-запросы (оконные функции, CTE) Оптимизация ORM (SQLAlchemy/Django ORM) Кэширование (Redis) API: Реализация REST/gRPC Документирование (Swagger/OpenAPI) Обработка ошибок и валидация Инфраструктура: Docker-контейнеризация Настройка CI/CD (GitHub Actions)
2 промежуточная аттестация	Практическая диагностическая работа Соревнования	Телеграм-боты	Понимание основных терминов (кнопки, виджеты, связь с каналами) Умение создавать простые телеграм ботами. Способность работать с интерфейсом телеграмом

			Уровень самостоятельности при выполнении заданий 3.Уровень и степень владения методами и приемами постановки и проведения экспериментов. 3.Умение работать индивидуально, в малых группах и принимать участие в коллективных проектах;
3 промежуточная аттестация	Диагностическая практическая работа, Соревнования	Работа с серверами	1.Понимание смысла терминов по программированию и умение использовать их в процессе работы над проектом. 2. Внутренняя организованность и умение довести работу до конца, устранить ошибки. 3.Степень самостоятельности при выполнении диагностических заданий. 4.Умение выполнять работу по инструкции.
Итоговая аттестация	Соревнования	Готовый рабочий шаблон	1.Уровень и степень владения основными понятиями и технологиями по работе с кейсом. 2. Степень и уровень подготовки по программированию и степень использования приемов обработки данных, полученных в ходе проведенных испытаниях. 4.Понимание отличия между разными типами алгоритмов и умение их программировать для разных действий. 5. Умение оценить свою работу и работу своих товарищей по предложенным критериям педагога.

2.5 Методические материалы

На занятиях в проектной группе используются словесные и наглядные методы. Учебные занятия организуются в форме: лекции, рассказа, беседы, презентации и практических занятий. В ходе реализации программы используется системно- деятельный подход.

– **Приёмы использования технических средств, специальных приборов и оборудования** для облегчения восприятия, и формирования полных представлений об объектах.

Дифференциация и индивидуализация обучения

Дифференциация обучения – объединение в группу детей по принципу учета состояния здоровья. Заключается в организации работы различной по содержанию, объёму, сложности, методам, приёмам и средствам в зависимости от психофизических возможностей ребенка (Л. А. Дружинина).

Индивидуальный подход – гибкое использование педагогом различных форм и методов педагогического воздействия с целью достижения оптимальных результатов образовательного процесса по отношению к каждому ребенку.

Индивидуальный подход в воспитании необходим в двух отношениях: во-первых, он обеспечивает развитие индивидуального своеобразия, давая возможность максимального проявления имеющихся у ребенка способностей; во-вторых, без учета индивидуальных особенностей ребенка любое педагогическое воздействие не может быть эффективным. Вот почему для осуществления индивидуального подхода, как в обучении, так и в воспитании, необходимо изучение психологических особенностей детей.

Использование информационно-коммуникационных технологий

Применения компьютерных технологий позволяет разработать новые «обходные пути» обучения, возможные только на базе этих технологий; создать компьютерно-опосредованные педагогические технологии, позволяющие выявить и преодолеть дисбаланс между развитием и обучением применительно к разным содержательным моментам развития ребенка, так как именно в компьютерной форме они становятся наиболее легко воспринимаемыми и тиражируемыми технологиями.

Технологии на основе активизации и интенсификации деятельности

Игровые технологии

Концептуальные идеи и принципы:

- игра – ведущий вид деятельности и форма организации процесса обучения;
- игровые методы и приёмы - средство побуждения, стимулирования обучающихся детей к познавательной деятельности;
- постепенное усложнение правил и содержания игры обеспечивает активность действий;
- игра как социально-культурное явление реализуется в общении. Через общение она передается, общением она организуется, в общении она функционирует;
- использование игровых форм занятий ведет к повышению творческого потенциала обучаемых и, таким образом, к более глубокому, осмысленному и быстрому освоению изучаемой дисциплины;
- цель игры – учебная (усвоение знаний, умений и т.д.). Результат прогнозируется заранее, игра заканчивается, когда результат достигнут;
- механизмы игровой деятельности опираются на фундаментальные потребности личности в самовыражении, самоутверждении, саморегуляции, самореализации.

Технологии проблемного обучения

Концептуальные идеи и принципы:

- создание проблемных ситуаций под руководством педагога и активная самостоятельная деятельность обучающихся по их разрешению, в результате чего и осуществляется развитие мыслительных и творческих способностей, овладение знаниями, умениями и навыками;
- целью проблемной технологии выступает приобретение ЗУН, усвоение способов самостоятельной деятельности, развитие умственных и творческих способностей;
- проблемное обучение основано на создании проблемной мотивации;
- проблемные ситуации могут быть различными по уровню проблемности, по содержанию неизвестного, по виду рассогласования информации, по другим методическим особенностям;
- проблемные методы — это методы, основанные на создании проблемных ситуаций, активной познавательной деятельности учащихся, требующей актуализации знаний, анализа, состоящей в поиске и решении сложных вопросов, умения видеть за отдельными фактами явление, закон.

Технологии, основанные на коллективном способе обучения

Технологии сотрудничества

Концептуальные идеи и принципы:

- позиция взрослого как непосредственного партнера детей, включенного в их деятельность;
- уникальность партнеров и их принципиальное равенство друг другу, различие и оригинальность точек зрения, ориентация каждого на понимание и активную интерпретацию его точки зрения партнером, ожидание ответа и его предвосхищение в собственном высказывании, взаимная дополнительность позиций участников совместной деятельности;
- неотъемлемой составляющей субъект-субъектного взаимодействия является диалоговое общение, в процессе и результате которого происходит не просто обмен идеями или вещами, а взаиморазвитие всех участников совместной деятельности;
- диалоговые ситуации возникают в разных формах взаимодействия: педагог - ребенок; ребенок - ребенок; ребенок - средства обучения; ребенок – родители;
- сотрудничество непосредственно связано с понятием – активность. Заинтересованность со стороны педагога отношением ребёнка к познаваемой действительности, активизирует его познавательную деятельность, стремление подтвердить свои предположения и высказывания в практике;
- сотрудничество и общение взрослого с детьми, основанное на диалоге - фактор развития дошкольников, поскольку именно в диалоге дети проявляют себя равными, свободными, раскованными, учатся самоорганизации, самостоятельности, самоконтролю.

Проектная технология

Концептуальные идеи и принципы:

- развитие свободной творческой личности, которое определяется задачами развития и задачами исследовательской деятельности детей, динамичностью предметно-пространственной среды;
- особые функции взрослого, побуждающего ребёнка обнаруживать проблему, проговаривать противоречия, приведшие к её возникновению, включение ребёнка в обсуждение путей решения поставленной проблемы;
- способ достижения дидактической цели в проектной технологии осуществляется через детальную разработку проблемы (технология);
- интеграция образовательных содержаний и видов деятельности в рамках единого проекта совместная интеллектуально – творческая деятельность;
- завершение процесса овладения определенной областью практического или теоретического знания, той или иной деятельности, реальным, осязаемым практическим результатом, оформленным тем или иным образом.

Здоровьесберегающие технологии:

Концептуальные идеи и принципы:

- физкультурно-оздоровительная деятельность на занятиях в виде зрительных гимнастик, физкультминуток, динамических пауз и пр.;
- обеспечение эмоционального комфорта и позитивного психологического самочувствия ребенка в процессе общения со сверстниками и взрослыми в детском саду, семье.

Используются дидактические материалы в виде инструкций, схем, шаблонов, тесты с возможностью самоконтроля, карточек с заданиями, поля для соревнований, видеофильмов, готовых роботов для анализа их работы.

2.6 Воспитательный компонент

Цель: развитие личности; создание условий для самоопределения, в том числе и для профессионального самоопределения, на основе социокультурных, духовно-нравственных ценностей и принятых в российском обществе правил и норм поведения.

Возможность воспитания обучающихся, реализующейся программы дополнительного образования посредством знакомства с производством своего края, путем организации экскурсий на предприятия своего города, решения кейсов. По завершению курса проектная группа представляет проект в соответствии с заданием предприятия, что способствует ранней профориентации

школьников и патриотическому воспитанию обучающихся. Деятельность подбирается в соответствии с возрастными особенностями, профессиональным интересом, потребностями группы, а также актуальности проблем, стоящих в данный период перед обществом.

Задачи воспитания:

1. Развивать коммуникативные умения, навыки сотрудничества при организации совместной деятельности (обсуждение, планирование, совместный поиск решения проблемы, аргументация точки зрения, работа в парах, группах).

2. Поддержка детской инициативы, развитие способности аргументировано высказывать свою точку зрения.

3. Воспитывать ценностное отношение к собственному труду, труду других людей и его результатам.

Основными формами воспитания являются: беседа, практическое занятие, защита проектов и другие формы взаимодействия обучающихся.

Методики, технологии воспитания, обучения и развития детей.

В работе с детьми используются традиционные методы:

- словесные: беседа, рассказ, монолог, диалог;
- наглядные: демонстрация иллюстраций, рисунков, макетов, моделей, презентаций и т.д.;
- практические: решение творческих заданий, изготовление моделей, и др.;
- проблемно-поисковые: изготовление изделий по образцу, по собственному замыслу, решение творческих задач;
- индивидуальные: задания в зависимости от достигнутого уровня развития, учащегося;
- игровые.

Условия воспитания:

Воспитательный процесс осуществляется в условиях организации деятельности детского коллектива на основной учебной базе реализации программы в организации дополнительного образования детей в соответствии с нормами и правилами работы организации.

Запланированы мероприятия по взаимодействию с родителями. Проведение родительских собраний, совместных праздников, мастер-классов.

В конце учебного года будут проведены внутренние соревнования.

Детям предоставляется возможность участия в конкурсах и выставках.

Примерный перечень мероприятий

Сроки	Уровень проведения соревнований, конкурсов, мероприятий	Название соревнований, конкурсов, мероприятий
май	региональный	Техноолимп
ноябрь	региональный	НТО
май	региональный	ЮТИ
февраль	всероссийский	Технострелка
апрель	всероссийский	Шаг в будущее

Краткосрочная программа каникулярного периода не предусмотрена.

Анализ **результатов** воспитания проводится в процессе педагогического наблюдения за поведением детей, их общением, отношениями детей друг с другом, в коллективе, их отношением к педагогам, к выполнению своих заданий по программе. Методами оценки результативности реализации программы в части воспитания является педагогическое наблюдение, педагогический анализ результатов тестирования, опросы.

2.7 Информационные ресурсы и литература

Литература для педагога

1. Методическое пособие: «Алгоритмы и структуры данных» / Автор: Сорокина Е.В., 2023.
2. Методические рекомендации: «Работа с Git и GitHub в образовательной среде» / Автор: Федотов А.А., 2022.
3. Учебная тетрадь: «Основы SQL и работы с базами данных» / Автор: Кузнецов А.И., 2021.
4. Сборник задач по программированию на Python / Автор: Роганов Е.В., 2022.
5. Методический материал: «Введение в компьютерное зрение» / Автор: Михайлов А.А., 2023.

Литература для обучающихся

1. Дейтел, Х.М., Дейтел, П.Д. Как программировать на C++ . — Санкт-Петербург: БХВ-Петербург, 2021.
2. Лутц М. Изучаем Python , 5-е издание. — Санкт-Петербург: Символ-Плюс, 2021.
3. (Подходит для модулей 4–9, особенно машинное обучение и парсинг данных)
4. Беликов А.В. Основы алгоритмизации и программирования: Учебное пособие . — М.: Инфра-М, 2020.
5. Конев А.А. Работа с базами данных в Python . — М.: ДМК Пресс, 2022.
6. Громов Ю.Ю. Введение в веб-программирование на Django . — М.: Нолидж, 2021.
7. (Подходит для модуля 7)
8. Немченко К.Э. Создание веб-приложений с использованием HTML и CSS . — М.: Вильямс, 2020.
9. Зубарев А.Е. Git и GitHub на практике . — СПб.: Питер, 2021.
10. Орлова И.В. Docker: контейнеризация приложений от новичка к профессионалу . — М.: Эксмо, 2022.
11. Багратуни С.Г. Машинное зрение и обработка изображений . — М.: СОЛОН-Пресс, 2020.
12. Чернышов Г.В. Введение в машинное обучение с использованием Python . — М.: Наука и Технологии, 2021.

Интернет-ресурсы для обучающихся

1. OpenCV Documentation & Tutorials
2. https://docs.opencv.org/master/d6/d00/tutorial_py_root.html
3. Real Python – статьи и уроки по Python
4. <https://realpython.com>
5. W3Schools – HTML, CSS, JS
6. <https://www.w3schools.com>
7. MDN Web Docs – официальная документация по веб-технологиям
8. <https://developer.mozilla.org/ru/>
9. Django Project Official Documentation
10. <https://docs.djangoproject.com/en/stable/>
11. YOLOv8 Documentation by Ultralytics
12. <https://docs.ultralytics.com/>

итоговой аттестации по программе «3д-моделирование»

Теоретическая часть:

A1

Что такое декоратор в Python?
 - Функция, изменяющая поведение других функций
 - Способ объявления переменных
 - Тип данных для хранения коллекций
 - Метод обработки исключений

A2

Для чего используется async/await?
 - Для асинхронного выполнения кода
 - Для создания классов
 - Для работы с файлами
 - Для математических вычислений

A3

Что такое метакласс?
 - Класс, создающий другие классы
 - Способ хранения данных
 - Модуль для тестирования
 - Декоратор функций

A4

Какой ORM чаще используется в Python?
 - SQLAlchemy
 - Pandas
 - NumPy
 - Matplotlib

A5

Что такое GIL?
 - Глобальная блокировка интерпретатора
 - Генератор списков
 - Графическая библиотека
 - Группировка исключений

A6

Какой инструмент используется для виртуальных окружений?
 - venv
 - pip
 - PyCharm
 - Django

A7

Что такое WSGI?
 - Интерфейс для веб-серверов
 - Библиотека для машинного обучения
 - Средство работы с базами данных
 - Фреймворк для тестирования

A8

Для чего используется pytest?
 - Для тестирования кода
 - Для создания веб-приложений
 - Для анализа данных
 - Для работы с API

A9

Что такое DRY принцип?
 - Don't Repeat Yourself (Не повторяйся)
 - Data Recovery Yield
 - Dynamic Resource Yielding
 - Digital Response Yield

A10

Какой формат чаще используется для API?
 - JSON
 - CSV
 - XML
 - TXT

2. Практическая часть

Б1. Разработка микросервиса

Создан REST API (FastAPI/Flask)

Реализован CRUD для модели

Настроена JWT-аутентификация

Настроено логирование и обработка ошибок

Б2. Оптимизация производительности

Использован cProfile для анализа

Выявлены узкие места

Оптимизирован алгоритм (улучшена сложность Big O)

Реализовано кэширование через Redis

Б3. Работа с асинхронностью (максимум – 10 баллов)

Создано асинхронное веб-приложение

Реализовано взаимодействие с внешними API

Настроен Celery/RQ для задач

Протестирована нагрузочная устойчивость

Б4. Итоговый проект — Защита работы (максимум – 10 баллов)

Подготовка презентации:

Архитектурное решение

Использованные технологии

Проблемы и решения

Метрики производительности

Демонстрация:

Рабочий код на GitHub

Документация API

Тестовое покрытие

Примеры запросов

Ответы на вопросы:

Обоснование выбора технологий

Trade-off принятых решений

Понимание принципов работы системы