

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ, НАУКИ И МОЛОДЕЖНОЙ ПОЛИТИКИ
НИЖЕГОРОДСКОЙ ОБЛАСТИ**

**Государственное бюджетное профессиональное образовательное учреждение
«АРЗАМАССКИЙ ТЕХНИКУМ СТРОИТЕЛЬСТВА И ПРЕДПРИНИМАТЕЛЬСТВА»**

Центр цифрового образования детей «IT-куб»

СОГЛАСОВАНО:

Начальник ЦЦОД «IT-Куб»

Г.А. Волгунова

(подпись)

(Ф.И.О)

« 16 » июля 20 21 г.

УТВЕРЖДАЮ:

Директор ГБПОУ АТСП

О.Н. Бабушкин

(Ф.И.О)

Приказ № 294 от « 03 » 08 20 21 г.



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА**

«Основы промышленного программирования»

Направленность – техническая

Возраст обучающихся: 13-18 лет

Срок реализации: 1 год

Объем: 144 часа

Авторы-составители:

Дятлов Антон Сергеевич,

педагог дополнительного образования,

Лазарева Анастасия Михайловна,

методист

Арзамас

2021

Содержание

1. Комплекс основных характеристик программы	3
1.1 Пояснительная записка.....	3
1.2 Цели и задачи курса	11
1.3. Содержание общеразвивающей программы	13
1.3.1 Учебный план	13
1.3.2 Содержание учебного плана	25
1.4 Перечень планируемых результатов обучения.....	32
2. Комплекс организационно-педагогических условий реализации общеразвивающей программы.....	35
2.1 Календарный учебный график на 2021-2022 учебный год.....	35
2.2 Условия реализации программы	36
2.3 Формы аттестации и оценочные материалы	38
2.4 Методические материалы.....	40
Список литературы	43
Приложение	46

1. Комплекс основных характеристик программы

1.1 Пояснительная записка

В настоящее время мы переживаем большие изменения в развитии общества. В современную жизнь человека все больше внедряются компьютеры и информационные технологии. Все большее значение приобретает умение человека грамотно обращаться с компьютером, причем зачастую не на пользовательском уровне, а на уровне начинающего программиста.

В обязательном школьном курсе информатики программирование нередко представлено лишь на элементарном уровне, на это выделяется недостаточное количество часов. Лишь немногие школы могут себе позволить преподавать программирование на достойном уровне. Следствием этого является формальное восприятие обучающимися основ современного программирования и неумение применять полученные знания на практике.

Программа составлена опираясь на уникальный опыт преподавания в Школе анализа данных Яндекса (АНО ДПО «ШАД»). В ней большое внимание уделяется практической и самостоятельной работе.

Программа «Основы промышленного программирования» имеет **техническую направленность**, в ее основу заложены принципы практической направленности, что обеспечит вариативность обучения. Содержание учебных модулей направлено на детальное изучение алгоритмизации, реализацию межпредметных связей, организацию проектной и исследовательской деятельности обучающихся.

Основанием для проектирования и реализации данной общеразвивающей программы служит **перечень следующих нормативных правовых актов и государственных программных документов:**

– Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 N 273-ФЗ;

– Стратегия развития воспитания в Российской Федерации на период до 2025 года. Распоряжение Правительства Российской Федерации от 29 мая 2015 г. № 996-р;

– Письмо Министерства образования и науки РФ от 18.11.2015г. № 09-3242. «О направлении Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)»;

– Распоряжение правительства РФ от 04.09. 2014 № 1726-р «Об утверждении Концепции развития дополнительного образования детей»;

– «Основы законодательств РФ об охране здоровья граждан», утвержденные Верховным советом РФ от 22.07.1993 № 5487 - (ред. от 25.11.2009);

– Федеральный закон от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребенка в РФ»; Федеральный закон от «Об основах охраны здоровья граждан в Российской Федерации», 2011г.;

– Приказ Министерства просвещения России от 09.11.2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (Приказ №1008 отменен).

Актуальность программы обусловлена потребностью общества в технически грамотных специалистах и полностью отвечает социальному заказу по подготовке квалифицированных кадров в области программирования, а также высоким интересом обучающихся к ИТ-сфере. Важнейшей характеристикой подрастающего поколения является активность в информационном пространстве, интернет-коммуникации.

В рамках изучения программы обучающиеся постоянно будут сталкиваться с необходимостью самостоятельной работы над заданиями: обучающиеся учатся решать задачи без помощи педагога. Для этого в содержании курса фигурируют задания, в которых для решения задачи необходимо найти информацию в сети Интернет; может потребоваться

устранение ошибки, которую не так просто обнаружить; условие сформулировано недостаточно прозрачно и обучающемуся необходимо самостоятельно формализовать его (или задать правильные вопросы педагогу). Все эти знания, умения и практические навыки решения актуальных задач, полученные на занятиях, готовят обучающихся к самостоятельной проектно-исследовательской деятельности с применением современных технологий.

Прогностичность программы «Основы промышленного программирования» заключается в том, что она отражает требования и актуальные тенденции не только сегодняшнего, но и завтрашнего дня и имеет междисциплинарный характер, что полностью отражает современные тенденции построения как дополнительных общеобразовательных программ, так и образования в целом.

Практическая значимость курса заключается в том, что он способствует более успешному овладению знаниями и умениями по направлению «Основы промышленного программирования» через развитие самостоятельности обучающихся и оптимизацию средств и методов обучения.

Отличительная особенность программы «Основы промышленного программирования» в том, что она является практико-ориентированной. Освоение обучающимися IT-навыков происходит в процессе практической и самостоятельной работы. Это позволяет им получать не только теоретические знания в области программирования, но и уверенно овладевать IT-технологиями, что поможет им самоопределиться и выстроить траекторию личностного роста в современном информационном обществе.

Изучение основных принципов программирования невозможно без регулярной практики написания программ на каком-либо языке. Для обучения был выбран язык Python. Данный выбор обусловлен тем, что синтаксис языка достаточно прост и интуитивно понятен, а это понижает порог вхождения и позволяет сосредоточиться на логических и

алгоритмических аспектах программирования, а не на выучивании тонкостей синтаксиса. При этом Python является очень востребованным языком; он отлично подходит для знакомства с различными современными парадигмами программирования и активно применяется в самых разных областях от разработки веб-приложений до машинного обучения.

Научившись программировать на языке Python, обучающиеся получают мощный и удобный инструмент для решения как учебных, так и прикладных задач. Вместе с тем чистота и ясность его конструкций позволит обучающимся потом с легкостью выучить любой другой язык программирования.

Знания и умения, приобретенные в результате освоения курса, могут быть использованы обучающимися при участии в олимпиадах по программированию, при решении задач по физике, химии, биологии, лингвистике и другим наукам, а также они являются фундаментом для дальнейшего совершенствования мастерства программирования.

Прием на программу производится по итогам окончания программы «Основы программирования на языке Python». Курс программы заканчивается получением сертификата после успешного прохождения итогового контроля. Итоговый контроль заключается в создании проекта и последующей его защите.

Адресат программы: дополнительная общеобразовательная общеразвивающая программа «Основы промышленного программирования» предназначена для обучающихся в возрасте 13-18 лет, не имеющих ограниченных возможностей здоровья, проявляющих интерес к программированию.

Группа формируется *по возрасту* 13-18 лет. **Формы занятий** групповые, количество обучающихся в группе – 8-12 человек.

Место проведения занятий: 607220, Нижегородская обл., г. Арзамас, ул. Мира, д. 1.

Возрастные особенности группы

Содержание программы учитывает возрастные и психологические особенности обучающихся в возрасте 13-18 лет, указанные в ДООП и определяющие выбор форм проведения занятий с ними. Выделенные нами возрастные периоды при формировании групп 13-18 лет базируются на психологических особенностях развития старшего подросткового возраста.

Содержание программы учитывает возрастные и психологические особенности обучающихся в возрасте 13-18 лет, которые определяют выбор форм проведения занятий с ними. Обучающиеся этого возраста отличаются открытием своего внутреннего мира, внутреннего «Я». Главным измерением времени в самосознании является будущее, к которому они себя готовят. Ведущая деятельность в этом возрасте - учебно-профессиональная, в процессе которой формируются такие новообразования, как мировоззрение, профессиональные интересы, самосознание, мечта и идеалы, поэтому основной формой проведения занятий выбраны практические занятия.

Также следует отметить, что обучающиеся данной возрастной группы характеризуются такими психическими процессами, как стремление углубленно понять себя, разобраться в своих чувствах, настроениях, мнениях, отношениях. Это порождает у обучающихся стремление к самоутверждению, самовыражению (проявления себя в тех качествах, которые они считают наиболее ценными) и самовоспитанию. Эти процессы позволяют положить начало созданию начального профессионального самоопределения.

Режим занятий, объем общеразвивающей программы: длительность одного занятия составляет 2 академических часа (по 45 минут) с перерывом (переменой) в 10 минут, периодичность занятий – 2 раза в неделю.

Срок освоения общеразвивающей программы определяется содержанием программы и составляет 1 год (144 часа).

Формы обучения: очная с применением дистанционных образовательных технологий (Закон №273-ФЗ, гл.2, ст.17, п.2.).

Основной вид занятий – комбинированный, сочетающий в себе элементы теории и практики. Большинство заданий курса выполняется самостоятельно с помощью персонального компьютера и необходимых программных средств.

Единицей учебного процесса является блок уроков (раздел). Каждый такой раздел охватывает отдельную информационную технологию или ее часть. Внутри раздела разбивка по времени изучения производится педагогом самостоятельно, но с учетом рекомендованного календарно-тематического плана. С учетом регулярного повторения ранее изученных тем темп изучения отдельных разделов блока определяется субъективными и объективными факторами.

Каждая тема курса начинается с постановки задачи - характеристики предметной области или конкретной программы на языке Python, которую предстоит изучить. С этой целью педагог проводит демонстрацию презентации или показывает саму программу, а также готовые работы, выполненные в ней. Закрепление знаний проводится с помощью практики отработки умений самостоятельно решать поставленные задачи, соответствующие минимальному уровню планируемых результатов обучения.

Основные задания являются обязательными для выполнения всеми обучающимися группы. Задания выполняются на компьютере с использованием интегрированной среды разработки. При этом обучающиеся не только формируют новые теоретические и практические знания, но и приобретают новые технологические навыки.

Методика обучения ориентирована на индивидуальный подход. Для того чтобы каждый обучающийся получил наилучший результат обучения, программой предусмотрены индивидуальные задания для самостоятельной работы на компьютере. Такая форма организации обучения стимулирует интерес обучающегося к предмету, активность и самостоятельность

обучающихся, способствует объективному контролю глубины и широты знаний, повышению качества усвоения материала обучающимися, позволяет педагогу получить объективную оценку выбранной им тактики и стратегии работы, методики индивидуального обучения и обучения в группе, выбора предметного содержания.

Для самостоятельной работы используются разные по уровню сложности задания, которые носят репродуктивный и творческий характер. Количество таких заданий в работе может варьироваться.

По типу организации взаимодействия педагогов с обучающимися при реализации программы используются личностно ориентированные технологии, технологии сотрудничества.

Реализация программы предполагает использование здоровьесберегающих технологий.

Здоровьесберегающая деятельность реализуется:

- через создание безопасных материально-технических условий;
- включением в занятие динамических пауз, периодической смены деятельности обучающихся;
- контролем соблюдения обучающимися правил работы на ПК;
- через создание благоприятного психологического климата в учебной группе в целом.

Объем общеразвивающей программы: 144 часа.

Педагогическая целесообразность программы

Программа «Основы промышленного программирования» является целостной и непрерывной в течение всего процесса обучения, позволяет обучающемуся шаг за шагом раскрывать в себе творческие возможности и самореализоваться в современном мире, развить компьютерную грамотность и предполагает использование форм организации материала, обеспечивающих доступ к сложным и нетривиальным разделам в рамках содержательно-тематического направления общеразвивающей программы.

Также предполагает углубленное изучение содержания общеразвивающей программы и доступ к предпрофессиональным и профессиональным знаниям в рамках содержательно-тематического направления общеразвивающей программы.

Осваивая данную программу, обучающиеся будут овладевать навыками востребованных на рынке труда. Практически для каждой перспективной профессии будут полезны знания и навыки, рассматриваемые в данной программе. Программа помогает решать проблемы личностного и профессионального самоопределения, самореализации.

1.2 Цели и задачи курса

Цель программы: Сформировать у обучающихся начальные профессиональные компетенции в области IT-технологий посредством овладения навыками разработки эффективных алгоритмов, для реализации их в виде программы, написанной на языке программирования Python.

Для успешной реализации поставленной цели необходимо решить ряд педагогических, развивающих и воспитательных **задач**:

Обучающие:

- изучение конструкций языка программирования Python;
- изучение принципов и методов функционального и объектно-ориентированного программирования;
- приобретение навыков работы в интегрированной среде разработки на языке Python;
- формирование навыков разработки эффективных алгоритмов и программ на основе изучения языка программирования Python.

Развивающие:

- формирование и развитие навыков алгоритмического и логического мышления, грамотной разработки программ;
- приобретение навыков поиска информации в сети Интернет, анализ выбранной информации на соответствие запросу, использование информации при решении задач;
- развитие у обучающихся интереса к программированию, самостоятельности и творческого подхода к решению задач с использованием средств вычислительной техники;
- формирование и развитие навыков работы с различными источниками информации, необходимой для решения учебных задач; формирование умений планировать свои действия с учетом фактора времени, в обстановке с элементами конкуренции, предвидеть результат и достигать его, при

необходимости вносить коррективы в первоначальный замысел.

Воспитательные:

- воспитание этики групповой работы, отношений делового сотрудничества, взаимоуважения;
- развитие основ коммуникативных отношений внутри проектных групп и в коллективе в целом;
- воспитание упорства в достижении результата;
- формирование целеустремленности, организованности, равнодушия, ответственного отношения к труду, толерантности и уважительного отношения к окружающим.

1.3. Содержание общеразвивающей программы

1.3.1 Учебный план

Таблица 1

№, п/п	Название раздела, темы	Описание тем	Кол-во часов		
			Всего	Теория	Практика
Раздел 1. Программирование на Python			42	19	23
1-2	Повторение. Решение задач на основные конструкции и структуры данных	Инструктажи по технике безопасности с отметкой в журнале. Введение в программу 2-го года обучения. Общая характеристика программы. Цели и задачи. Тематические модули.	2	1	1
3-6	Повторение. Решение задач на классы	Повторение классов и способов работы с ними	4	3	1

7-8	QT1. Что такое QT и PyQt. Знакомство	Обзор рынка фреймворков для разработки настольных приложений на Python. История, предназначение, преимущества и недостатки фреймворка Qt. Подключение и первые шаги с PyQt.	2	2	-
9-12	QT2. QtDesigner, pyuic, два способа подключения uic-файла	Изучение основных элементов графического интерфейса и настроек QtDesigner, способы интеграции графического и текстового стиля создания настольных приложений.	4	2	2
13-16	QT3. Обработка исключений. Создание собственных исключений.	Изучение механизма работы и создания исключений	4	2	2
17-20	QT4. Файлы в Python. Типы файлов и работа с ними. Внутреннее устройство файлов	Принципы работы с файловой системой средствами Python. Чтение и запись информации в файл приложением, созданным при помощи QtDesigner.	4	2	2

21-24	QT 5 Диалоги, работа с изображениями	Изучение набора встроенных в QtDesigner диалоговых окон. Способы чтения, модификации и внедрения графических файлов в настольное приложение.	4	2	2
25-26	Самостоятельная работа на файлы	Выполнение самостоятельной работы.	2	-	2
27-30	QT6. Работа с простыми таблицами (csv). Фильтрация, сортировка, вывод. Работа с табличными данными в PyQT	Формат данных csv, его преимущества и недостатки. Способы чтения и записи файлов в данном формате. Работа с табличными данными в PyQT в рамках собственного настольного приложения.	4	2	2
31-34	QT 7 Введение в БД, работа с SQL-таблицами и отображение данных в PyQT Часть 1	Предназначение и разновидности баз данных. Реляционные базы данных и язык запросов SQL. Обращение к базе данных из создаваемого при помощи PyQT настольного приложения.	4	2	2

35-36	QT 8 Введение в БД, работа с SQL-таблицами и отображение данных в PyQT Часть 2	Технология написания более сложных SQL запросов. Объектно-реляционное отображение.	2	-	2
37-40	QT9. Проект PyQT. Обработка клавиатуры и курсора. Сборка независимого приложения	Петля событий настольного приложения. Обработка событий стандартных устройств ввода (мышь, клавиатура). Обработка событий, генерируемых модулями программы. Сборка интерактивного приложения.	4	1	3
41-42	Самостоятельная работа на SQL-запросы	Выполнение самостоятельной работы.	2	-	2
Раздел 2. PyGame			42	17	25
43-46	Введение в репозитории Подключение PyCharm Работа с удаленными репозиториями	Разновидности и предназначение систем контроля версий. Репозитории для хранения и обмена данными. Использование среды разработки PyCharm с удаленными репозиториями.	4	1	3

47-48	PyGame. Введение	Что такое PyGame, его место в экосистеме и какие игры можно создавать с его помощью.	2	1	1
49-52	PyGame. Игровой цикл События	Создание игрового цикла, обработка событий стандартных устройств ввода.	4	2	2
53-56	Основные команды при одиночной работе с Git	Система контроля версий Git. Использование git при разработке приложения в одиночку - графический и консольный интерфейс, основные команды.	4	2	2
57-60	PyGame. Клеточное поле	Реализация клеточного поля в PyGame. Способы хранения глобальных настроек и обмена данными между игровыми агентами.	4	1	3

61-64	PyGame. Классические игры на клеточном поле	Обзор классических игр на клеточном поле и вариантов их реализации при помощи PyGame.	4	2	2
65-66	Совместная работа над проектом, основные понятия и команды. Работа с репозиториями в среде разработки.	Изучение основных команд для работы с репозиториями в среде разработки	2	-	2
67-68	PyGame. Изображения Спрайты	Графические возможности PyGame, чтение, изменение и встраивание изображений. Техника работы со спрайтами.	2	1	1
69-72	PyGame. Столкновения и другие взаимодействия	Обсуждение основных видов взаимодействия между игроками и другими сущностями игры, а также способов программной реализации такого рода взаимодействий.	4	2	2

73-76	PyGame. Игра в целом Теория	Знакомство с архитектурой игрового приложения. Проектирование архитектуры игрового приложения.	4	2	2
77-80	PyGame. Украшения игр	Нюансы создания интерфейса игры, визуального оформления игровой механики, подключение звуковых эффектов.	4	2	2
81-84	Цели и подходы к тестированию Создание "самодельных" тестов (без библиотек)	Как и зачем осуществляют тестирование в промышленной разработке. Подходы к тестированию программных продуктов. Технология создания тестов стандартными средствами Python.	4	1	3
Раздел 3. Проекты WEB и API			46	16	30
85-88	WEB. Работа с популярными форматами файлов (json, zip)	Структура форматов json и zip. Python-модули для работы с ними. Основные сферы применения, практика применения в WEB.	4	2	2

89-90	WEB. Знакомство с API	Программный интерфейс приложения в разработке для основных существующих платформ. Особенности API в WEB-разработке.	2	1	1
91-94	WEB. Работа с командной строкой (скрипты, аргументы) Периодические задачи (модуль shedule)	Интерфейс командной строки основных операционных систем. Его применение в backend-разработке и администрировании.	4	2	2
95-96	WEB. Библиотека argparse Задачи на создание скриптов с ее помощью	Парсинг аргументов командной строки средствами argparse.	2	-	2
97-100	WEB. Работа с протоколом HTTP	Протокол HTTP и клиент-серверная архитектура. Обмен данными с использованием данного протокола.	4	2	2

101-104	WEB. Решение задач на API Яндекс-карт	Картографические сервисы Яндекса для сайтов, мобильных приложений и логистики. Работа с ними средствами Python применительно к WEB-проектам.	4	2	2
105-106	Самостоятельная работа на http, json и командную строку	Выполнение самостоятельной работы.	2	-	2
107-108	WEB. Введение во flask Обработка HTML-форм	Фреймворк для создания WEB-приложений Flask, структура и базовая механика. Создание и обработка HTML-форм с его помощью.	2	1	1
109-112	WEB. Шаблоны. Flask-wtf	Расширения фреймвора Flask. Предназначение и использование расширения Flask-WTF.	4	2	2

113-116	WEB. Знакомство с SQLalchemy	Предназначение и использование расширения (фреймворка Flask) Flask-SQLAlchemy.	4	2	2
117-118	WEB. Flask-sqlalchemy	Нюансы Flask-SQLAlchemy.	2	-	2
119-122	WEB. REST-API. Понятие. Делаем простое Rest-api	REST подход к организации взаимодействия компонентов распределенного приложения. REST-API.	4	2	2
123-124	WEB. REST-API. Flask-restfull	Расширение (фреймворка Flask) Flask-RESTful для быстрого построения REST-API.	2	-	2

125-128	WEB. Разворачиваем проект в облаке. Дорешка	Размещение web-проекта в сети интернет.	4	-	4
129-130	Итоговая контрольная работа	Выполнение контрольной работы.	2	-	2
Раздел 4. Проектная деятельность			14	1	13
131-132	Работа над индивидуальным проектом	Выбор темы индивидуального или группового проекта. Планирование индивидуального или группового проекта. Реализация индивидуального или группового проекта.	2	1	1
133-140	Оформление проекта	Оформление индивидуального проекта. Технические требования к оформлению проектных работ. Изучение требований к составлению презентаций для защиты индивидуального проекта	8	-	8

141-142	Консультации	Подготовка к защите индивидуального проекта	2	-	2
143-144	Итоговая аттестация		2	-	2
Итого			144	54	90

1.3.2 Содержание учебного плана

Раздел 1. Программирование на Python

1.1. Повторение. Решение задач на основные конструкции и структуры данных

Теория: Введение в программу 2-го года обучения. Общая характеристика программы. Цели и задачи. Тематические модули.

Практика: решение задач.

1.2. Повторение. Решение задач на классы

Теория: Повторение классов и способов работы с ними.

Практика: решение задач.

1.3. Повторение. Проектирование классов

Теория: Повторение способов проектирования классов.

Практика: решение задач.

1.4. QT1. Что такое QT и PyQt

Теория: Обзор рынка фреймворков для разработки настольных приложений на Python. История, предназначение, преимущества и недостатки фреймворка Qt. Подключение и первые шаги с PyQt.

Практика: решение задач.

1.5. QT2. QtDesigner, pyuis, два способа подключения uic-файла

Теория: Изучение основных элементов графического интерфейса и настроек QtDesigner, способы интеграции графического и текстового стиля создания настольных приложений.

Практика: решение задач, связанных с созданием простейшего однооконного приложения.

1.5. QT3. Обработка исключений. Создание собственных исключений.

Теория: Изучение механизма работы и создания исключений.

Практика: решение задач с использованием исключений.

1.6. QT4. Файлы в Python. Типы файлов и работа с ними.

Внутреннее устройство файлов

Теория: Принципы работы с файловой системой средствами Python. Чтение и запись информации в файл приложением, созданным при помощи QtDesigner.

Практика: решение задач, связанных с работой с файлами из настольного приложения.

1.7. QT5. Диалоги, работа с изображениями

Теория: Изучение набора встроенных в QtDesigner диалоговых окон. Способы чтения, модификации и внедрения графических файлов в настольное приложение.

Практика: решение задач по обработке изображений.

1.8. Самостоятельная работа на файлы

Практика: выполнение самостоятельной работы.

1.9. QT6. Работа с простыми таблицами (csv). Фильтрация, сортировка, вывод. Работа с табличными данными в PyQT

Теория: Формат данных csv, его преимущества и недостатки. Способы чтения и записи файлов в данном формате. Работа с табличными данными в PyQT в рамках собственного настольного приложения.

Практика: решение задач по работе с таблицами.

1.10. QT7. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 1

Теория: Предназначение и разновидности баз данных. Реляционные базы данных и язык запросов SQL. Обращение к базе данных из создаваемого при помощи PyQT настольного приложения.

Практика: решение задач по работе с БД и SQL.

1.10. QT8. Введение в БД, работа с SQL-таблицами и отображение данных в PyQT. Часть 2

Теория: Технология написания более сложных SQL запросов. Объектнореляционное отображение.

Практика: решение задач на работу с БД из PyQT.

1.11. QT9. Проект PyQT. Обработка клавиатуры и курсора. Сборка независимого приложения

Теория: Петля событий настольного приложения. Обработка событий стандартных устройств ввода (мышь, клавиатура). Обработка событий, генерируемых модулями программы. Сборка интерактивного приложения.

Практика: работа над проектом.

1.12. Самостоятельная работа на SQL-запросы

Практика: выполнение самостоятельной работы.

Раздел 2. PyGame

2.1. Введение в репозитории. Подключение PyCharm. Работа с удаленными репозиториями

Теория: Разновидности и предназначение систем контроля версий. Репозитории для хранения и обмена данными. Использование среды разработки PyCharm с удаленными репозиториями.

Практика: решение задач по работе с репозиториум.

2.2. PyGame. Введение

Теория: Что такое PyGame, его место в экосистеме и какие игры можно создавать с его помощью.

Практика: решение задач.

2.3. PyGame. Игровой цикл. События

Теория: Создание игрового цикла, обработка событий стандартных устройств ввода.

Практика: решение задач.

2.4. Основные команды при одиночной работе с Git

Теория: Система контроля версий Git. Использование git при разработке приложения в одиночку — графический и консольный интерфейс, основные команды.

Практика: решение задач.

2.5. PyGame. Клеточное поле

Теория: Реализация клеточного поля в PyGame. Способы хранения глобальных настроек и обмена данными между игровыми агентами.

Практика: решение задач.

2.6. PyGame. Классические игры на клеточном поле

Теория: Обзор классических игр на клеточном поле и вариантов их реализации при помощи PyGame.

Практика: решение задач по реализации игры на клеточном поле.

2.7. Совместная работа над проектом, основные понятия и команды. Работа с репозиториями в среде разработки

Теория: Изучение основных команд для работы с репозиториями в среде разработки.

Практика: работа с репозиториями.

2.8. PyGame. Изображения. Спрайты

Теория: Графические возможности PyGame, чтение, изменение и встраивание изображений. Техника работы со спрайтами.

Практика: решение задач по работе с изображениями.

2.9. PyGame. Столкновения и другие взаимодействия

Теория: Обсуждение основных видов взаимодействия между игроками и другими сущностями игры, а также способов программной реализации такого рода взаимодействий.

Практика: решение задач.

2.10. PyGame. Игра в целом

Теория: Проектирование архитектуры игрового приложения.

Практика: создание своей игры.

2.11. PyGame. Украшения игр

Теория: Нюансы создания интерфейса игры, визуального оформления игровой механики, подключение звуковых эффектов.

Практика: украшение своей игры.

2.12. Цели и подходы к тестированию. Создание «самодельных» тестов (без библиотек)

Теория: Как и зачем осуществляют тестирование в промышленной разработке. Подходы к тестированию программных продуктов. Технология создания тестов стандартными средствами Python.

Практика: тестирование своего игрового приложения.

Раздел 3. Проекты WEB и API

3.1. WEB. Работа с популярными форматами файлов (json, zip)

Теория: Структура форматов json и zip. Python-модули для работы с ними. Основные сферы применения, практика применения в WEB.

Практика: решение задач.

3.2. WEB. Знакомство с API

Теория: Программный интерфейс приложения в разработке для основных существующих платформ. Особенности API в WEB-разработке.

Практика: решение задач.

3.3. WEB. Работа с командной строкой (скрипты, аргументы). Периодические задачи (модуль shedule)

Теория: Интерфейс командной строки основных операционных систем. Его применение в backend-разработке и администрировании.

Практика: решение задач.

3.4. WEB. Библиотека argparse. Задачи на создание скриптов с ее помощью

Теория: Парсинг аргументов командной строки средствами argparse.
Практика: решение задач.

3.5. WEB. Работа с протоколом HTTP

Теория: Протокол HTTP и клиент-серверная архитектура. Обмен данными с использованием данного протокола.

Практика: решение задач - самостоятельная работа.

3.6. WEB. Решение задач на API Яндекс-карт

Теория: Картографические сервисы Яндекса для сайтов, мобильных приложений и логистики. Работа с ними средствами Python применительно к WEB-проектам.

Практика: решение задач.

3.7. Самостоятельная работа на http, json и командную строку

Практика: выполнение самостоятельной работы.

3.8. WEB. Введение во flask. Обработка html-форм

Теория: Фреймворк для создания WEB-приложений Flask, структура и базовая механика. Создание и обработка HTML-форм с его помощью.

Практика: решение задач - самостоятельная работа.

3.9. WEB. Шаблоны. flask--wtf

Теория: Расширения фреймвора Flask. Предназначение и использование расширения Flask-WTF.

Практика: решение задач.

3.10. WEB. Знакомство с SQLAlchemy

Теория: Предназначение и использование расширения (фреймворка Flask) Flask-SQLAlchemy.

Практика: решение задач.

3.11. WEB. Flask-sqlalchemy

Теория: Нюансы Flask-SQLAlchemy.

Практика: решение задач — контрольная работа.

3.12. WEB. REST-API. Понятие. Делаем простое Rest-api

Теория: REST подход к организации взаимодействия компонентов распределенного приложения. REST-API.

Практика: решение задач.

3.13. WEB. REST-API. Flask-restful

Теория: Расширение (фреймворка Flask) Flask-RESTful для быстрого построения REST-API.

Практика: работа над проектом — создание своего web-сервиса.

3.14. WEB. Разворачиваем проект в облаке. Дорешка

Теория: Размещение web-проекта в сети Интернет.

Практика: работа над проектом.

3.15. Итоговая контрольная работа

Практика: выполнение контрольной работы.

Раздел 4. Проектная деятельность

4.1 Работа над индивидуальным проектом

Практика: Выбор темы индивидуального или группового проекта.

Планирование индивидуального или группового проекта. Реализация индивидуального или группового проекта.

4.2 Оформление проекта

Теория: Технические требования к оформлению проектных работ.

Изучение требований к составлению презентаций для защиты индивидуального проекта

Практика: Оформление индивидуального проекта.

4.3 Консультации

Практика: Подготовка к защите индивидуального проекта.

1.4 Перечень планируемых результатов обучения

Предметные:

- умение определять результат выполнения алгоритма при заданных исходных данных, узнавать изученные алгоритмы обработки чисел и числовых последовательностей, создавать на их основе несложные программы анализа данных, читать и понимать несложные программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня;
- понимание основных предметных понятий («информация», «алгоритм», «исполнитель», «модель») и их свойств;
- развитие логических способностей и алгоритмического мышления, умения составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами - линейной, условной и циклической;
- умение выполнять пошагово (с использованием компьютера или вручную) алгоритмы управления исполнителями и анализа числовых и текстовых данных;
- развитие навыков и опыта разработки программ в выбранной среде программирования, включая тестирование и отладку программ;
- умение использовать основные управляющие конструкции объектно-ориентированного программирования и библиотеки прикладных программ, выполнять созданные программы;
- умение разрабатывать и использовать компьютерно-математические модели, оценивать числовые параметры моделируемых объектов и процессов, интерпретировать результаты, получаемые в ходе моделирования реальных процессов, анализировать готовые модели на предмет соответствия реальному объекту или процессу;
- умение соблюдать нормы информационной этики и права.

Личностные:

- формирование ответственного отношения к обучению, способности довести до конца начатое дело аналогично завершенным творческим учебным проектам;
- формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретенной благодаря иллюстрированной среде программирования мотивации к обучению и познанию;
- развитие опыта участия в социально значимых проектах, повышение уровня самооценки благодаря реализованным проектам;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;
- формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;
- формирование осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;
- формирование ценности здорового и безопасного образа жизни;
- усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Метапредметные:

- умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата;
- умение критически оценивать правильность решения учебно-исследовательской задачи;
- умение корректировать свои действия, вносить изменения в программу и отлаживать ее в соответствии с изменяющимися условиями;

- владение основами самоконтроля, способность к принятию решений;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий (ИКТ-компетенция);
- умение организовывать учебное сотрудничество и совместную деятельность с педагогом и обучающимися в процессе проектной и учебно-исследовательской деятельности.

2. Комплекс организационно-педагогических условий реализации общеразвивающей программы

2.1 Календарный учебный график на 2021-2022 учебный год

Таблица 2

Год обучения	сентябрь					октябрь				ноябрь				декабрь				январь				февраль				март				апрель				май				Всего учебных недель/ часов	Всего часов по программе																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																	
	даты	1	2	3	4	даты	5	6	7	даты	8	9	10	даты	11	12	13	даты	14	15	16	даты	17	18	19	даты	20	21	даты	22	23	даты	24	25	даты	26	27			даты	28	29	даты	30	31	даты	32	33	даты	34	35	даты	36	37	даты	38	39																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															
1 год	01.09 - 05.09	06.09 - 12.09	13.09 - 19.09	20.09 - 26.09	27.09 – 03.10	04.10 – 10.10	11.10 – 17.10	18.10 – 24.10	25.10 – 31.11	01.11 – 07.11	08.11 – 14.11	15.11 – 21.11	22.11 – 28.11	29.11 – 05.12	06.12 – 12.12	13.12 – 19.12	20.12 – 26.12	27.12 – 02.01	03.01 - 10.01	10.01 – 17.01	17.01 – 24.01	24.01 – 31.01	31.01– 06.02	07.02 – 13.02	14.02 – 20.02	21.02 – 27.02	28.02 – 06.03	07.03 – 13.03	14.03 – 20.03	21.03 – 27.03	28.03 – 03.04	04.04 – 10.04	11.04 – 17.04	18.04 – 24.04	25.04-01.05	02.05 – 08.05	09.05 – 15.05	16.05 – 22.05	23.05 – 29.05																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																	

Условные обозначения:		
	Занятия по расписанию	Каникулярный период
	Итоговая аттестация обучающихся	

2.2 Условия реализации программы

Основной организационной формой обучения в ходе реализации данной образовательной программы является занятие. Это форма обеспечивает организационную четкость и непрерывность процесса обучения. Знание педагогом индивидуальных особенностей воспитанников позволяет эффективно использовать стимулирующее влияние коллектива на учебную деятельность каждого обучающегося.

Неоспоримым преимуществом занятия, является возможность соединения фронтальных, групповых и индивидуальных форм обучения.

Формы занятий: беседы, обсуждения, теоретические занятия, практические занятия, метод проектов. Также программа курса включает групповые и индивидуальные формы работы обучающихся (в зависимости от темы занятия).

Материально-техническое обеспечение:

Требования к помещению:

- помещение для занятий, отвечающие требованиям СанПин для учреждений дополнительного образования;
- качественное освещение;
- столы, стулья по количеству обучающихся и 1 рабочим местом для педагога.

Оборудование:

- ноутбуки на каждого обучающегося и педагога;
- Wi-Fi для поддержания online доступа к системе обучения;
- мультимедийный проектор либо интерактивная доска для показа презентаций.

Расходные материалы:

- маркеры;
- бумага писчая;
- шариковые ручки.

Информационное обеспечение:

- операционная система Windows;
- поддерживаемые браузеры (для работы LMS): Yandex Browser, Chrome, Chrome Mobile, Firefox, Opera, Safari, Mobile Safari, Edge, Python;
- среда Wing IDE 101 (версии 6 или выше);
- среда PyCharm Community Edition;
- пакет PyQt4 (на Qt5);
- пакет библиотек со SciPy: numpy, scipy, matplotlib, ipython + ipythonnotebook, sympy, pandas;
- рекомендуется установить ПО Anaconda.

Кадровое обеспечение:

Программа реализуется педагогом дополнительного образования - Дятловым Антоном Сергеевичем.

2.3 Формы аттестации и оценочные материалы

Система отслеживания результатов обучающихся выстроена следующим образом:

- текущий контроль;
- итоговая аттестация.

Текущий контроль осуществляется путем наблюдения, определения качества выполнения заданий, отслеживания динамики развития обучающегося. Способы проверки уровня освоения тем: опрос, выполнение упражнений, наблюдение. Формы проведения итогов по каждой теме и каждому разделу общеразвивающей программы соответствуют целям и задачам ДООП.

В конце освоения дополнительной общеобразовательной общеразвивающей программы обучающиеся реализуется итоговая аттестация в виде защиты индивидуальных/групповых проектов. Темы для проектов обучающиеся выбирают на свое усмотрение при согласовании с педагогом или получают одну из тем проекта, подготовленных педагогом. Список предлагаемых тем проектов для выполнения обучающимися согласуется с учебной частью не позднее чем за 3 месяца до проведения итоговой аттестации по дополнительной общеобразовательной общеразвивающей программе «Основы промышленного программирования».

Индивидуальный/групповой проект оценивается формируемой комиссией по 100-балльной шкале, которая переводится в один из уровней освоения образовательной программы согласно таблице 3:

Таблица 3

Баллы, набранные обучающимся	Уровень освоения
0–30 баллов	низкий
31–70 баллов	средний
71–100 баллов	высокий

Критерии оценивания индивидуального проекта представлены в Приложении.

Состав комиссии (не менее 3 человек): в обязательном порядке входит педагог; приветствуется привлечение ИТ - профессионалов, представителей высших и других учебных заведений, администрации учебной организации.

Компонентами оценки индивидуального проекта являются (по мере убывания значимости): качество ИП, отзыв педагога, уровень презентации и защиты проекта. Если проект выполнен группой учащихся, то при оценивании учитывается не только уровень исполнения проекта в целом, но и личный вклад каждого из авторов. Решение принимается коллегиально.

2.4 Методические материалы

Образовательный процесс осуществляется в очной форме с применением дистанционных технологий.

Выбор методов обучения осуществляется исходя из анализа уровня готовности обучающихся к освоению содержания раздела, степени сложности материала, типа учебного занятия. На выбор методов обучения значительно влияет персональный состав группы, индивидуальные особенности, возможности и запросы обучающихся.

Используются следующие педагогические технологии:

- технология группового обучения;
- технология коллективно-взаимного обучения;
- технология работы с аудио- и видеоматериалами.

Дидактические материалы:

- дидактические материалы по теме занятия, распечатанные на листе формата А4 для выдачи каждому обучающемуся.

Формы обучения:

фронтальная - предполагает работу педагога сразу со всеми обучающимися в едином темпе и с общими задачами. Для реализации обучения используется компьютер педагога с мультимедиа проектором, посредством которых учебный материал демонстрируется на общий экран. Активно используются Интернет-ресурсы;

коллективная - это форма сотрудничества, при котором коллектив обучает каждого обучающегося и каждый из них активно участвует в обучении по совместной учебной работе;

групповая - предполагает, что занятия проводятся с подгруппой. Для этого группа распределяется на подгруппы не более 6 человек, работа в которых регулируется педагогом;

индивидуальная - подразумевает взаимодействие педагога с одним обучающимся. Как правило данная форма используется в сочетании с

фронтальной. Часть занятия (объяснение новой темы) проводится фронтально, затем обучающийся выполняют индивидуальные задания или общие задания в индивидуальном темпе;

Формы организации учебного занятия:

В образовательном процессе помимо традиционного учебного занятия используются многообразные формы, которые несут учебную нагрузку и могут использоваться как активные способы освоения обучающимися образовательной программы, в соответствии с возрастом, составом группы, содержанием учебного модуля: беседа, лекция, мастер-класс, практическое занятие, защита проектов, конкурс, викторина, диспут, круглый стол, «мозговой штурм».

Методы: мотивация, убеждение, поощрение, упражнение, стимулирование, создание ситуации успеха и др.

Педагогические технологии:

- индивидуализации обучения;
- группового обучения;
- коллективного взаимообучения;
- дифференцированного обучения;
- разноуровневого обучения;
- проблемного обучения;
- развивающего обучения;
- дистанционного обучения;
- игровой деятельности;
- коммуникативная технология обучения;
- коллективной творческой деятельности;
- решения изобретательских задач;
- здоровьесберегающая технология.

Дидактические материалы:

Методические пособия, разработанные педагогом с учетом конкретных

задач, варианты демонстрационных программ, материалы по терминологии ПО, инструкции по настройке оборудования, учебная и техническая литература. Используются педагогические технологии индивидуализации обучения и коллективной деятельности.

Список литературы

Нормативные документы

- Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 N 273-ФЗ;
- Стратегия развития воспитания в Российской Федерации на период до 2025 года. Распоряжение Правительства Российской Федерации от 29 мая 2015 г. № 996-р;
- Письмо Министерства образования и науки РФ от 18.11.2015г. № 09-3242. «О направлении Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)»;
- Распоряжение правительства РФ от 04.09. 2014 № 1726-р «Об утверждении Концепции развития дополнительного образования детей»;
- «Основы законодательств РФ об охране здоровья граждан», утвержденные Верховным советом РФ от 22.07.1993 № 5487 - (ред. от 25.11.2009);
- Федеральный закон от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребенка в РФ»;
- Федеральный закон «Об основах охраны здоровья граждан в Российской Федерации», 2011г.;
- Приказ Министерства просвещения России от 09.11.2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (Приказ №1008 отменен);
- Санитарно-эпидемиологические правила и нормативы СанПиН 2.4.4.3172-14.

Перечень учебной литературы, необходимой для освоения курса

Основные источники

1. Лутц М. Изучаем Python, том 1, 5-е изд. / М. Лутц. – СПб.: ООО «Диалектика», 2019. – 832 с.
2. Поляков К.Ю. Информатика. Углубленный уровень. Учебник для 10 класса в 2 частях / К.Ю. Поляков, Е.А. Еремин. – М.: БИНОМ. Лаборатория знаний, 2019. – 352 с.
3. Окулов С.М. Задачи по программированию. 2-е изд. / С.М. Окулов. – М.: БИНОМ. Лаборатория знаний, 2019. – 823 с.

Дополнительные источники

1. Вейдт В.П. О воспитании подростков: в помощь классному руководителю: Методическое пособие / В.П. Вейдт. – Калининград: Издательство Калининградского областного института развития образования, 2019. – 140 с.
2. Сэнд У. Hello World! Занимательное программирование / У. Сэнд, К. Сэнд. – СПб.: Питер, 2016. – 400 с.
3. Джексон Н. Классный учитель: Как работать с трудными учениками, сложными родителями и получать удовольствие от профессии / Н. Джексон. – «Альпина Диджитал», 2015. – 332 с.

*Перечень ресурсов информационно-телекоммуникационной сети
«Интернет»*

1. Материалы и презентации к урокам в LMS Академии Яндекс Лицея
2. Python. Введение в программирование:
<https://younglinux.info/python.php>
3. Питонтьютор. Бесплатный курс по программированию с нуля.
Работает прямо в браузере: <https://pythontutor.ru/>
4. Python 3 для начинающих: <https://pythonworld.ru/samouchitel-python>

Контрольно-измерительные материалы

к дополнительной общеобразовательной общеразвивающей программе

«Основы промышленного программирования»

Арзамас
2021

Дополнительной общеобразовательной общеразвивающей программой «Основы промышленного программирования» предусмотрено проведение:

– Промежуточного контроля – выполнение заданий и прохождение контрольных и самостоятельных работ в системе LMS.

– Итоговой аттестации – по итогам прохождения программы будут выдаваться дипломы в зависимости от рейтинга, набранного в системе LMS.

1. Материалы промежуточного контроля

Промежуточный контроль проводится путем проведения контрольных и самостоятельных работ в системе LMS. Пробные варианты тестирующих работ становятся доступны за неделю до работы по эталонному КТП. Это помогает понять, как лучше подготовить учеников, какие темы с ними повторить, какие акценты сделать, а также понять, какие проблемы могут возникнуть у группы во время проверочной работы.

Сами варианты проверочных работ доступны по ссылке в системе LMS в соответствующем уроке самостоятельной или контрольной работы. Пример: <https://lyceum.yandex.ru/courses/353/groups/3306/lessons/2079>

2. Материалы итоговой аттестации

Итоговая аттестация производится по баллам, набранным обучающимся в системе LMS к концу обучения. В зависимости от количества баллов обучающемуся будет выдан диплом.