

Министерство образования и молодежной политики Владимирской области
Управление образования Администрации муниципального образования
Гусь-Хрустальный район Владимирской области
Муниципальное бюджетное учреждение дополнительного образования
«Центр дополнительного образования детей» Гусь – Хрустального района

Принята на заседании
методического совета
Протокол № 51
от 15.06.2024 года

Утверждена приказом
№ 02-02\154

от 15.06.2024 года
Директор МБУ ДО «ЦДОД»
Гусь-Хрустального района
Пшенова Е.В.



Дополнительная общеобразовательная общеразвивающая программа

«Программирование на языке Python»

технической направленности

Возраст детей: 13-18 лет

Срок реализации: 1 год

Стартовый уровень

Автор-составитель:
Кубылина Наталья Николаевна,
педагог дополнительного образования

г. Курлово
2024 г.

Нормативные документы

Программа соответствует действующим нормативным документам:

- ФЗ № 273 от 29.12.2012 «Об образовании в Российской Федерации»,
- Концепция развития дополнительного образования детей до 2030 года (Распоряжение Правительства РФ от 31 марта 2022 года № 06-1172)
- Распоряжение Правительства РФ от 29.05.2015 № 996-р «Стратегия развития воспитания в РФ на период до 2025 года».
- Концепция духовно-нравственного развития и воспитания личности гражданина России (ФГОСООО)
- Приказ Министерства просвещения Российской Федерации от 27 июля 2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»
- Примерные требования к программам дополнительного образования детей в приложении к письму Департамента молодежной политики, воспитания и социальной поддержки детей Минобрнауки России от 11.12.2006 г. № 06-1844
- Письмо Министерства образования и науки РФ N 09-3242 от 18 ноября 2015 г. «Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)»
- Постановление Главного государственного санитарного врача Российской Федерации от 28.09.2020 №28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи».
- Положение о дополнительной общеобразовательной общеразвивающей программе педагога МБУ ДО «ЦДОД» Гусь-Хрустального района.
- Устав МБУ ДО «ЦДОД» Гусь-Хрустального района.

I. Комплекс основных характеристик дополнительной общеразвивающей программы «Программирование на языке Python»

1.1. Пояснительная записка

Направленность программы

Дополнительная общеобразовательная общеразвивающая программа «Программирование на языке Python» имеет техническую направленность.

Актуальность программы обусловлена широким распространением информационно-коммуникационных технологий в обществе и необходимостью обеспечивать связанную с этим инфраструктуру специалистами. Всё большее значение приобретает умение человека грамотно обращаться с компьютером, причём на уровне начинающего программиста.

Изучение основных принципов программирование невозможно без регулярной практики написания программ на каком-либо языке программирования. Для обучения в рамках программы выбран язык Python, который является достаточно эффективным и доступным инструментом достижения задач в области создания программных продуктов.

Синтаксис языка достаточно прост и интуитивно понятен, что понижает порог вхождения и позволяет сосредоточиться на алгоритмических аспектах программирования.

Отличительная особенность программы, новизна: Ключевой особенностью курса является его направленность на формирование у учащихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и его реализации с помощью средств программирования.

Новизна программы заключается в том, что Python дает более широкие возможности в области программирования, чем Pascal, который входит в школьный курс информатики. На языке Python можно легко и быстро создавать простые компьютерные игры, трёхмерные модели и программировать роботов. Этот язык быстрее и легче усваивается, чем Pascal. Многие мировые компании такие, как Intel, Cisco, Hewlett-Packard, используют этот язык при реализации своих проектов. Крупнейшие интернет-ресурсы такие, как Google, YouTube, также разработаны с помощью языка программирования Python.

Адресат программы:

Дополнительная общеразвивающая программа «Программирование на языке Python» рассчитана на детей 14–18 лет, проявляющих интерес к информационно-коммуникационным технологиям. Учащиеся объединения являются разными по возрасту и социальному статусу. Объединение комплектуется на основании заявлений законных представителей учащихся (самих учащихся с 13 лет). Группы формируются из школьников разного возраста на добровольной внеконкурсной основе.

Содержание программы разработано с учетом принятия всех желающих, не имеющих противопоказаний по здоровью.

Программа составлена с учётом индивидуальных и возрастных особенностей учащихся. Психолого-педагогические особенности учащихся определяют и методы индивидуальной работы педагога с каждым из них.

Группы постоянного состава. Набор обучающихся свободный по 12-15 человек в группу.

Объём и срок освоения программы.

Программа рассчитана на 1 год обучения, 72 часа

Форма обучения.

Очная

Особенности организации образовательного процесса.

Занятия проводятся со всем составом группы. Группа формируется из обучающихся 13-18летнего возраста. Состав группы постоянный.

Формы реализации образовательной программы: традиционная.

Режим занятий.

Продолжительность одного академического часа - 40 мин. Общее количество часов в неделю – 2 часа. Занятие проводится 1 раз в неделю по 2 часа.

1.2. Цели и задачи дополнительной общеразвивающей программы «Программирование на языке Python»

Цель программы: формирование творческой личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python..

Задачи программы:

Образовательные (предметные):

- Познакомить с основными алгоритмическими конструкциями и правилами их записи, с основными способами организации данных.
- Научить составлять и записывать алгоритмы с использованием соответствующих алгоритмических конструкций.
- Научить распознавать необходимость применения той или иной алгоритмической конструкции при решении задач.
- Научить организовывать данные для эффективной алгоритмической обработки.

- Научить разрабатывать алгоритмы и реализовывать их на языке программирования Python.
- Научить осуществлять отладку и тестирование программы.

Личностные:

- повышение общекультурного уровня;
- вооружение правильным методологическим подходом к познавательной и практической деятельности;
- выделение и раскрытие роли информационных технологий и компьютеров в развитии современного общества;
- привитие навыков сознательного и рационального использования компьютера в своей учебной, а затем и профессиональной деятельности;
- воспитание стремления к овладению техникой исследования;
- воспитание трудолюбия, инициативности и настойчивости в преодолении трудностей.

Метапредметные:

- владение умениями организации собственной учебной деятельности, включающими: целеполагание как постановку учебной задачи на основе соотнесения того, что уже известно, и того, что требуется установить;
- планирование – определение последовательности промежуточных целей с учетом конечного результата, разбиение задачи на подзадачи, разработка последовательности и структуры действий, необходимых для достижения цели при помощи фиксированного набора средств;
- прогнозирование – предвосхищение результата.

1.3. Содержание дополнительной общеобразовательной общеразвивающей программы

Учебный план

№ п/п	Тема занятий	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
	Инструктаж по технике безопасности в компьютерном классе	2	1	1	Опрос
1.	Раздел 1. Знакомство с языком Python	4	2	2	Тестирование, решение практических задач
2.	Раздел 2. Переменные и выражения	8	4	4	Тестирование, решение практических задач
3	Раздел 3. Условные предложения	6	3	3	Тестирование, Решение практических задач
4	Раздел 4. Циклы	10	4	6	Тестирование, решение практических задач, творческая работа

5	Раздел 5. Функции	8	4	4	Тестирование, решение практических задач
6	Раздел 6. Строки - последовательности символов	8	3	5	Решение практических задач
7	Раздел 7. Сложные типы данных	18	7	11	Тестирование, Решение практических задач
8	Раздел 8. Стил программирования и отладка программ	8	3	5	Решение практических задач, презентация проекта, рефлексия.
	ВСЕГО	72	31	41	

Содержание учебного плана

Вводное занятие.

Теория: Знакомство с планом работы объединения, инструктаж по ТБ. Практика: Опрос. Тренинг на командообразование.

Раздел 1. Знакомство с языком Python.

Теория: Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программы на языке Python. Комментарии.

Практическая работа 1.1. Установка программы Python Практическая работа 1.2. Режимы работы с Python Тест № 1. Знакомство с языком Python

Учащиеся должны знать / понимать:

- понятие программы;
- структура программы на Python;
- режимы работы с Python.

Учащиеся должны уметь:

- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе.

Раздел 2. Переменные и выражения.

Теория: Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова.

Выражения. Операции. Порядок выполнения операций. Магические функции. Композиция.

Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.

Практическая работа 2.1. Работа со справочной системой Практическая работа 2.2. Переменные

Практическая работа 2.3. Выражения

Практическая работа 2.5. Задачи на элементарные действия с числами Тест № 2.

Выражения и операции.

Учащиеся должны знать / понимать:

- общую структуру программы;
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания;

Раздел 3. Условные предложения.

Теория: Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.

Практическая работа 3.1. Логические выражения Практическая работа 3.2. "Условный оператор" Практическая работа 3.3. Множественное ветвление Практическая работа 3.4. "Условные операторы"

Самостоятельная работа № 1. Решение задач по теме "Условные операторы". Зачетная работа № 1. "Составление программ с ветвлением".

Тест № 3. "Условные операторы".

Учащиеся должны знать / понимать:

- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;
- Учащиеся должны уметь:
- использовать условный оператор;
- создавать сложные условия с помощью логических операторов.

Раздел 4. Циклы.

Теория: Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов.

Оператор цикла с параметром for. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Случайные числа. Функция randrange. Функция random. Примеры решения задач с циклом.

Практическая работа 4.1. "Числа Фибоначчи" Практическая работа 4.2. Решение задачи с циклом for.

Практическая работа 4.3. Реализация циклических алгоритмов Практическая работа 4.4. Случайные числа

Практическая работа 4.5. Решение задач с циклом. Самостоятельная работа № 2 "Составление программ с циклом"

Тест № 4. Циклы

Творческая работа № 1. "Циклы"

Учащиеся должны знать / понимать:

- циклы с условием и их виды;
- правила записи циклов условием;
- назначение и особенности использования цикла с параметром;
- формат записи цикла с параметром;
- примеры использования циклов различных типов.

Учащиеся должны уметь:

- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием;
- определять целесообразность применения и использовать цикл с параметром для решения поставленной задачи.

Раздел 5. Функции.

Теория: Создание функций. Параметры и аргументы. Локальные и глобальные переменные.

Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda. Примеры решения задач с использованием функций.

Рекурсивные функции. Вычисление факториала. Числа Фибоначчи. Практическая работа 5.1. Создание функций

Практическая работа 5.2. Локальные переменные

Практическая работа 5.3. Решение задач с использованием функций Практическая работа 5.4. Рекурсивные функции

Самостоятельная работа № 3 по теме "Функции" Тест № 5. Функции

Учащиеся должны знать / понимать:

- понятие функции;
- способы описания функции;
- принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- понятие формальных и фактических параметров подпрограмм;
- способ передачи параметров.

Учащиеся должны уметь:

- создавать и использовать функции;
- использовать механизм параметров для передачи значений.

Раздел 6. Строки - последовательности символов.

Теория: Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки.

Срезы строк. Строки нельзя изменить. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки, списки, кортежи). Примеры решения задач со строками.

Практическая работа 6.1. Строки

Практическая работа 6.2. Решение задач со строками.

Учащиеся должны знать / понимать:

- назначение строкового типа данных;
- операторы для работы со строками;
- процедуры и функции для работы со строками;
- операции со строками.

Учащиеся должны уметь:

- описывать строки;
- соединять строки;
- находить длину строки;
- вырезать часть строки;
- находить подстроку в строке;
- находить количество слов в строке.

Раздел 7. Сложные типы данных.

Теория: Списки. Тип список (list). Индексы. Обход списка. Проверка вхождения в список. Добавление в список. Суммирование или изменение списка. Операторы для списков. Срезы списков. Удаление списка. Клонирование списков. Списочные параметры. Функция range. Списки: примеры решения задач.

Матрицы. Вложенные списки. Матрицы. Строки и списки. Генераторы списков в Python. Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения

Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы.

Множества в языке Python. Множества. Множественный тип данных. Описание множеств. Операции, допустимые над множествами: объединение, пересечение, разность, включение. Оператор определения принадлежности элемента множеству.

Практическая работа 7.1. Списки.

Практическая работа 7.2. Решение задач со списками. Тест № 7. Списки

Учащиеся должны знать / понимать:

- сложные типы данных;
- способ описания списка;
- способ доступа к элементам списка;
- способ описания кортежа;
- способ описания словаря;

- операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.

Учащиеся должны уметь:

- описывать списки;
- вводить элементы списка;
- выводить элементы списка;
- выполнять поиск элемента в списке, поиск минимума и максимума, нахождение суммы элементов списка;
- использовать вложенные списки;
- приводить примеры использования вложенных списков (матриц);
- описывать множества;
- определять принадлежность элемента множеству;
- вводить элементы множества;
- выводить элементы множества.

Раздел 8. Стил ь программирования и отладка программ.

Стил ь программирования. Отладка программ.

Зачет по курсу «Программирование на языке Python»

Учащиеся должны знать / понимать:

- что такое стил ь программирования;
- правила именования объектов;
- основные рекомендации при написании программ.

Учащиеся должны уметь:

- определять вид ошибок и находить ошибки в программе.
- выполнять тестирование и отладку программ.

1.4. Планируемые результаты освоения дополнительной общеразвивающей программы «Программирование на языке Python»

Реализация программы предполагает достижение обучающимися следующих личностных, метапредметных и предметных результатов:

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;
- формирование представлений об основных предметных понятиях («информация», «алгоритм», «исполнитель», «модель») и их свойствах;
- 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) несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных;
- навыки и опыт 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;
- формирование умения соблюдать нормы информационной этики и права.

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; понимание, что в программировании длинная программа не всегда лучшая;
- умение критически оценивать правильность решения учебно- исследовательской задачи;
- умение корректировать свои действия, вносить изменения в программу и отлаживать её 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 учителем и сверстниками в процессе проектной и учебно- исследовательской деятельности.

Личностные:

- формирование ответственного отношения к учению, способности довести до конца начатое дело аналогично завершённым творческим учебным проектам;
- формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой 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.

II Комплекс организационно-педагогических условий реализации дополнительной общеразвивающей программы «Программирование на языке Python»

2.1. Календарный учебный график на 2023-2024 учебный год «Программирование на языке Python» техническая направленность

№ п/п	Сроки проведения урока, занятия.		Тема урока, тема раздела.	Количество часов	примечание
	Плановая дата	Скорректированная дата			
			<i>Инструктаж по технике безопасности в компьютерном классе (2 ч)</i>		
1			Вводное занятие. Правила техники безопасности.	2	
<i>Раздел 1. Знакомство с языком Python (4ч)</i>					
2			Общие сведения о языке Практическая работа: Установка программы Python	2	
3			Практическая работа «Режимы работы с Python.	2	
<i>Переменные и выражения (8 ч)</i>					
4			Переменные. Практическая работа со справочной системой.	2	
5			Выражения. Практическая работа «Выражения»	2	
6			Ввод и вывод.	2	
7			Задачи на элементарные действия с числами.	2	
<i>Условные предложения (6ч)</i>					
8			Логические выражения и операторы. Практическая работа «Логические выражения»	2	
9			Условный оператор. Практическая работа «Условный оператор»	2	
10			Множественное ветвление.	2	
<i>Циклы. (10 ч)</i>					
11			Оператор цикла с условием. Практическая работа «Числа Фибоначчи»	2	
12			Оператор цикла for. Практическая работа «Решение задач с циклом for»	2	
13			Вложенные циклы. Практическая работа «Реализация циклических алгоритмов»	2	
14			Случайные числа.	2	
15			Циклы.	2	
<i>Функции. (8 ч)</i>					

16			Создание функций. Практическая работа «Создание функций»	2	
17			Локальные переменные.	2	
18			Решение задач с использованием функций.	2	
19			Рекурсивные функции. Практическая работа «Рекурсивные функции»	2	
Строки – последовательности символов. (8 ч)					
20			Строки. Практическая работа «Строки»	2	
21			Срезы строк.	2	
22			Примеры решения задач со строками Практическая работа: Решение задач со строками	2	
23			Примеры решения задач со строками Практическая работа: Решение задач со строками.	2	
Сложные типы данных (18ч)					
24			Списки Практическая работа: Списки.	2	
25			Срезы списков.	2	
26			Списки: примеры решения задач.	2	
27			Решение задач со списками.	2	
28			Матрицы.	2	
29			Кортежи.	2	
30			Введение в словари.	2	
31			Множества в языке Python.	2	
32			Множества в языке Python.	2	
Стиль программирования и отладка программ (8ч)					
33			Стиль программирования.	2	
34			Отладка программ.	2	
35			Зачет по курсу «Программирование на языке Python»	2	
36			Зачет по курсу «Программирование на языке Python»	2	
Итого				72 ч	

2.2. Условия реализации дополнительной общеразвивающей программы «Программирование на языке Python»

Материально-техническое обеспечение:

Помещение: кабинет для занятий, расположенный по адресу Владимирская область Гусь-Хрустальный район, п.Добрятино , МБОУ»Добрятинская СОШ». Кабинет, соответствующий санитарно-гигиеническим требованиям.

Для реализации образовательной программы используются:

1. столы для компьютера;
2. компьютерные стулья;
3. шкафы для дидактических материалов, пособий;
4. специальная и научно-популярная литература для педагога и учащихся;
5. канцтовары;

Информационно-методическое обеспечение:

1. персональный компьютер (10 шт.);
2. мультимедийный проектор;
3. видеоматериалы разной тематики по программе;
4. оргтехника;
5. выход в сеть Internet;

Аппаратное обеспечение:

1. Процессор не ниже Core2 Duo;
2. Объем оперативной памяти не ниже 4 ГбDDR3;
3. Дисковое пространство на менее 128 Гб;
4. Монитор диагональю на мене 19’;

Программное обеспечение:

1. Операционная система Windows 7 Профессиональная или выше;
2. Интерпретатор Python версии 3.7 и выше;
3. IDE JatBrains PyCharm;
4. Foxit Reader или другой просмотрщик PDF файлов;
5. WinRAR;
6. Пакет офисных программ;
7. Adobe Photoshop или другой растровый графический редактор;
8. Любой браузер для интернет серфинга.

Кадровое обеспечение программы: (Педагог, работающий по данной программе должен знать основы программирования.) Кубылина Наталья Николаевна, учитель информатики первой квалификационной категорией. Педагогический стаж работы 25 лет.

2.3. Формы аттестации/контроля

Входящий контроль осуществляется при комплектовании группы в начале учебного года. *Цель* – определить исходный уровень знаний учащихся, определить формы и методы работы с учащимися.

Форма контроля: тестирование.

Текущий контроль осуществляется после изучения отдельных тем, раздела программы. В практической деятельности результативность оценивается качеством выполнения практических работ, поиску и отбору необходимого материала, умению работать с различными источниками информации. Анализируются положительные и отрицательные стороны работы, корректируются недостатки. Контроль знаний осуществляется с помощью заданий педагога (решение практических задач средствами языка программирования); взаимоконтроля, самоконтроля и др. Они активизируют, стимулируют работу учащихся, позволяют более полно проявлять полученные знания, умения, навыки.

Промежуточный контроль осуществляется в конце I полугодия учебного года.

Форма контроля: тест, решение практических задач средствами языка программирования.

Итоговый контроль осуществляется в конце учебного года.

Форма контроля: защита творческого проекта.

Отслеживание личностного развития учащихся осуществляется методом наблюдения, анкетирования. По итогам первого полугодия и по итогам года заполняется «Диагностическая карта», в которой проставляется уровень усвоения программы каждым учащимся объединения.

2.4. Оценочные материалы

Диагностируются результаты различным образом и на различных этапах деятельности. Для определения уровня знаний и умений обучающихся проводятся: в начале года - входной контроль, в середине года - промежуточный контроль, в конце года — итоговый контроль. Основной способ оценки – самооценка выполнения программ на языке программирования. Эта оценка присутствует на всех этапах. Одной из форм диагностики результатов является зачет или контрольная работа.

Входящий контроль: *Тестирование (Приложение 2)*

Промежуточный контроль: *Тестирование, решение задач (Приложение 3)*

Итоговый контроль: *учащимся предлагается самостоятельно выбрать тему творческого проекта и на основании темы разработать программу, пояснительную записку, презентацию.*

2.5. Методические материалы

Учебно-воспитательный процесс направлен на развитие природных задатков обучающихся, на реализацию их интересов и способностей. Каждое занятие обеспечивает развитие личности. При планировании и проведении занятий применяется личностно-ориентированная технология обучения, в центре внимания которой неповторимая личность, стремящаяся к реализации своих возможностей, а также системно-деятельностный метод обучения. Данная программа допускает творческий, импровизированный подход со стороны детей и педагога того, что касается возможной замены порядка раздела, введения дополнительного материала, методики проведения занятий. Руководствуясь данной программой, педагог имеет возможность увеличить или уменьшить объем и степень технической сложности материала в зависимости от состава группы и конкретных условий работы.

Методы обучения:

Выбор метода обучения зависит от содержания занятий, уровня подготовки и опыта обучающихся.

1. Традиционные методы развития интереса обучающихся к техническому творчеству можно объединить в группы, каждая из которых наиболее продуктивна на том или ином этапе работы. 1. Словесный: объяснение нового материала; рассказ обзорный для раскрытия новой темы; беседы с учащимися в процессе изучения темы.

2. Наглядный: применение демонстрационного материала, наглядных пособий, презентаций по теме.

3. Практический: индивидуальная и совместная продуктивная деятельность, выполнение учащимися определенных заданий, решение задач.

4. Интерактивный: создание специальных заданий, моделирующих реальную жизненную ситуацию, из которой учащимся предлагается найти выход.

Педагогические технологии, применяемые в реализации программы:

1. Технология проблемного диалога. Учащимся не только сообщаются готовые знания, но и организуется такая их деятельность, в процессе которой они сами делают «открытия», узнают что-то новое и используют полученные знания и умения для решения жизненных задач.

2. Технология коллективного взаимобучения («организованный диалог», «сочетательный диалог», «коллективный способ обучения (КСО)», «работа учащихся в парах сменного

состава») позволяет плодотворно развивать у обучаемых самостоятельность и коммуникативные умения.

3. Игровая технология. Игровая форма в образовательном процессе создаётся при помощи игровых приёмов и ситуаций, выступающих как средство побуждения к деятельности. Способствует развитию творческих способностей, продуктивному сотрудничеству с другими учащимися. Приучает к коллективным действиям, принятию решений, учит руководить и подчиняться, стимулирует практические навыки, развивает воображение.

4. Элементы здоровьесберегающих технологий являются необходимым условием снижения утомляемости и перегрузки учащихся.

5. Проектная технология предлагает практические творческие задания, требующие от учащихся их применение для решения проблемных заданий, знания материала на данный исторический этап. Овладевая культурой проектирования, школьник приучается творчески мыслить, прогнозировать возможные варианты решения стоящих перед ним задач.

6. Информационно-коммуникационные технологии активизируют творческий потенциал учащихся; способствует развитию логики, внимания, речи, повышению качества знаний; формированию умения пользоваться информацией, выбирать из нее необходимое для принятия решения, работать со всеми видами информации, программным обеспечением, специальными программами и т.д.

Формы организации учебного занятия: комбинированное занятие; беседа; игра; викторина, творческая встреча, соревнование, экскурсия и др. **Алгоритм учебного занятия.**

I этап - организационный.

Задача: подготовка детей к работе на занятии.

Содержание этапа: организация начала занятия, создание психологического настроя на учебную деятельность и активизация внимания.

II этап - подготовительный (подготовка к восприятию нового содержания).

Задача: Организовать занимающихся, сообщение задач занятия, подготовка к основной части занятия, к предстоящей работе. Содержание этапа: сообщение темы, цели учебного занятия и мотивация учебной деятельности детей (пример, познавательная задача, проблемное задание детям).

III этап - основной.

В качестве основного этапа могут выступать следующие:

1. Усвоение новых знаний и способов действий.

Задача: обеспечение восприятия, осмысления и первичного запоминания связей и отношений в объекте изучения. Целесообразно при усвоении новых знаний использовать задания и вопросы, которые активизируют познавательную деятельность детей.

2. Первичная проверка понимания.

Задача: установление правильности и осознанности усвоения нового учебного материала, выявление неверных представлений, их коррекция. Применяют пробные практические задания, которые сочетаются с объяснением соответствующих правил или обоснованием.

3. Закрепление знаний и способов действий, применяют творческие задания, выполняемые детьми самостоятельно.

4. Обобщение и систематизация знаний.

Задача: формирование целостного представления знаний по теме. Распространенными способами работы являются беседа и практические задания.

IV этап – контрольный.

Задача: выявление качества и уровня овладения знаниями, их коррекция.

Используются виды устного и письменного опроса, вопросы и задания различного уровня сложности (репродуктивного, творческого, поисково-исследовательского).

V этап - итоговый.

Задача: дать анализ и оценку успешности достижения цели и наметить перспективу последующей работы.

Содержание этапа: педагог сообщает, как работали обучающиеся на занятии, что нового узнали, какими умениями и навыками овладели.

VI этап - рефлексивный.

Задача: мобилизация детей на самооценку. Может оцениваться работоспособность, психологическое состояние, результативность работы, содержание и полезность учебной работы.

Изложенные этапы могут по-разному комбинироваться, какие-либо из них могут не иметь места в зависимости от педагогических целей и учитывая формы занятия.

Дидактические материалы:

- таблицы;
- схемы;
- памятки;
- видеозаписи, мультимедийные материалы;
- упражнения.

экскурсия и др.

Воспитательное значение игры, ее влияние на развитие ребенка трудно переоценить. Игра органически присуща детскому возрасту и при умелом руководстве со стороны взрослых способна творить чудеса. Ленивого она может сделать трудолюбивым, незнайку - знающим, неумелого - умельцем. Игра помогает сплотить детский коллектив, включить в активную деятельность детей замкнутых и застенчивых. В играх воспитывается сознательная дисциплина, дети приучаются к соблюдению правил, справедливости, умению контролировать свои поступки, правильно и объективно оценивать поступки других.

Основными формами работы с детьми в объединениях являются фронтальная, групповая, индивидуальная.

Список литературы

Литература для обучающихся:

1. Сайт/справочные материалы [Электронный ресурс] – Режим доступа: <https://metanit.com/python/>, свободный.
2. Сайт / интерактивный сборник задач для практики программирования [Электронный ресурс] – Режим доступа: <http://pythontutor.ru/>, свободный.
3. Сайт/Адаптивный тренажер Python [Электронный ресурс] – Режим доступа: <https://stepik.org/course/431> , свободный.
4. Сайт / среда разработки для языка Python [Электронный ресурс] – Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu> , свободный.
1. Сайт проекта Open Book Project. Практические примеры на Python Криса Мейерса [Электронный ресурс] – Режим доступа: openbookproject.net , свободный.

Литература для педагога

1. Бизли, Дэвид М. Python. Подробный справочник. – М.–СПб.: Символ-Плюс, 2010.
2. Лутц, Марк Python. Справочник. – М.: Вильямс, 2015.
3. Официальный сайт программы [Электронный ресурс] – Режим доступа: <https://docs.python.org/>, свободный.

4. Сайт, среда разработки для языка Python. [Электронный ресурс] – Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu>, свободный.
