



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ ТАТАРСТАН
МУНИЦИПАЛЬНОЕ АВТОНОМНОЕ УЧРЕЖДЕНИЕ ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
«ЦЕНТР ТЕХНИЧЕСКОГО ТВОРЧЕСТВА И ПРОФОРИЕНТАЦИИ»
НИЖНЕКАМСКОГО МУНИЦИПАЛЬНОГО РАЙОНА РЕСПУБЛИКИ ТАТАРСТАН

СОГЛАСОВАННО

Заместитель директора по УР
МАУ ДО «ЦТТиП» НМР РТ
 М. В. Киселева
от «31» 08 2022г.

«УТВЕРЖДАЮ»

Директор МАУ ДО «ЦТТиП» НМР РТ
 М. А. Кирпичонок
Приказ № 196
от «31» 08 2022г.



Принята на заседании
педагогического совета
Протокол № 1
от «31» 08 2022 года

Дополнительная общеобразовательная
общеразвивающая программа
«ОСНОВЫ ПРОГРАММИРОВАНИЯ»

(Базовый уровень)

Направленность: техническая
Возраст обучающихся: 11-13 лет
Срок реализации: 1 год (144 часа)

Автор-составитель:
Маркелова Юлия Сергеевна,
Гурина Анастасия Сергеевна,
педагоги дополнительного образования

г. Нижнекамск 2022

ОГЛАВЛЕНИЕ

1. Пояснительная записка	3
2. Структура программы.....	18
2.1. Объем программы.....	18
2.2. Учебный план	18
2.3. Содержание учебного плана	21
3. Условия реализации программы.....	27
3.1. Материально-техническое оснащение.....	27
3.2. Методическое обеспечение реализации программы.....	27
4. Список литературы.....	30
4.1. Список литературы, используемой педагогом.....	30
4.2. Список рекомендуемой литературы для обучающихся.....	30
Приложение 1. Контрольно-измерительные материалы	31

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Направленность программы – техническая.

Нормативно-правовое обеспечение программы.

1. Федеральный закон от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации»;

2. Федеральный закон от 31 июля 2020 г. № 304-ФЗ «О внесении изменений в Федеральный закон «Об образовании в Российской Федерации» по вопросам воспитания обучающихся»;

3. Концепция развития дополнительного образования детей, утвержденная Распоряжением Правительства Российской Федерации от 31 марта 2022 г. № 678-р;

4. Приказ Министерства просвещения Российской Федерации от 03.09.2019 г. №467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей»;

5. Приказ Министерства просвещения Российской Федерации от 09 ноября 2018 N 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;

6. Стратегия развития воспитания в Российской Федерации на период до 2025 года, утвержденная Распоряжением Правительства Российской Федерации от 29 мая 2015 г. № 996-р;

7. СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи» от 28 сентября 2020 года N 28;

8. Устав МАУ ДО «Центр технического творчества и профориентации» Нижнекамского муниципального района Республики Татарстан.

Актуальность программы.

Сегодня мы живем в огромном потоке молниеносно меняющейся информации. Беспрецедентным примером высоких темпов развития IT- сферы являются веб-технологии. Прогрессивность этой отрасли проявляется в том, что новые технологии и идеологии разрабатываются не только специализированными компаниями и корпорациями, но и «рядовыми» программистами, студентами и школьниками, имеющими потребность в создании новой технологии для дальнейшего использования, как в своих целях, так и для общества. Эффективное применение современных технологий и идеологий веб-программирования способствует не только повышению качества программного продукта, но и экономии трудозатрат, в том числе,

финансовых ресурсов, производительности и безопасности приложения, легкой расширяемости проектов и многому другому.

В условиях реализации программы «Цифровая экономика Российской Федерации», на первое место вышли «сквозные» технологии, включающие в себя развитие искусственного интеллекта, AR и VR, промышленное и спортивное программирование, робототехнику, аддитивные технологии (3D-моделирование, проектирование и конструирование). Это определяется, прежде всего, требованиями современного общества, которые диктуют необходимость владения технологиями работы в самых передовых технологиях XXI века. Новой тенденцией в сфере веб-программирования на сегодняшний день является разработка прогрессивных веб-приложений (PWA), которая основывается на технологиях искусственного интеллекта, что является релевантным сквозных цифровых технологий программы «Цифровая экономика Российской Федерации».

Программа реализуется в рамках объединения по углубленному изучению математики и информатики «Математическая мастерская «Цифра+», созданной для популяризации в молодежном сообществе и углубления уровня компетенций школьников в базовых дисциплинах цифровой среды.

Отличительная особенность и новизна дополнительной общеобразовательной программы «Основы программирования» заключается в том, что она является практико-ориентированной и позволяет в динамичном формате познакомить школьников с основными IT-направлениями современного мира.

Во-первых, общеобразовательная программа имеет практическую ценность, что мотивирует обучающихся к профессиональной интерпретации полученных результатов, во-вторых, позволяет избавиться от всего лишнего в образовательной концепции (удалить "образовательный шум") и, в-третьих, позволяет выстроить траекторию, в которой предыдущий этап был бы частью последующего, тем самым предоставляет возможность рассчитывать на эволюцию в мыслительной деятельности учащихся, а также осознание важности и необходимости полученных навыков.

Кейсы, выполняемые обучающимися в процессе обучения ориентированы на точки роста города Нижнекамск и Республики Татарстан, что позволяет обучающимся оценить практическую значимость своих разработок.

Педагогическая целесообразность.

Программирование учит детей самостоятельно мыслить, находить и решать проблемы, привлекая для этого знания из разных областей, уметь прогнозировать результаты и возможные последствия различных вариантов решения.

Применение детьми на практике теоретических знаний, полученных на уроках математики (выполнение расчетов), физики (физические законы при настройке анимации) и информатики ведет к более глубокому пониманию основ программирования, закрепляет полученные знания.

Занятия по программе способствуют развитию у детей технического мышления, повышению интереса к инженерным специальностям, стимулируют к продолжению образования в научно-технической сфере.

Возможность прикоснуться к неизведанному миру программирования для современного ребенка является очень мощным стимулом к познанию нового, преодолению инстинкта потребителя и формированию стремления к самостоятельному созиданию. Сфера IT воспитывает в ребенке характер исследователя, внимательного и ответственного человека. Он получает собственный интеллектуальный продукт, который можно показать друзьям, родителям.

Основные теоретическая идеи программы. В основу программы заложены следующие педагогические идеи:

- работы А.А. Кузнецова, А.Г. Кушниренко, М.И. Лапчика;
- теория развития мотивации ребенка к познанию и творчеству (А.К.Бруднов), возможности выбора индивидуального образовательного пути (Е.Б. Евладова, Л.Н. Николаева);
- разноплановая творческая деятельность, позволяющая развивать частные, индивидуальные интересы личности (О.Е. Лебедев, А.Е. Асмолов).

Практическая работа реализуется через:

- научно-исследовательскую деятельность, в ходе которой обучающиеся получают возможность ознакомиться с различными цифровыми технологиями, такими как искусственный интеллект, большие данные и машинное обучение, виртуальная и дополненная реальность.

- проектную деятельность, развивающую технические способности и инженерное мышление, техническую смекалку и высокое профессиональное мастерство при выполнении практических работ, связанных с проектированием, разработкой и отладкой программного кода.

Образовательный процесс предусматривает овладение теоретическими знаниями одновременно с формированием деятельностно-практического опыта, в основу которого положен творческий потенциал каждого учащегося:

создание авторских инженерных решений и участие в конкурсах, конференциях, соревнованиях и хакатонах.

Немаловажным является приобретение опыта работы в команде, а также индивидуальное техническое творчество.

Цель программы

Целью программы является создание оптимальных условий для развития ИКТ-компетентности обучающихся, их профессиональной ориентации на успешную деятельность в современном информационном обществе; формирование интереса к техническим видам творчества, развитие логического, алгоритмического мышления, создание условий для творческой самореализации личности ребёнка посредством получения навыков работы с современными IT-технологиями.

Задачи программы

Достижение поставленной цели складывается из выполнения следующих задач:

- сформировать понятийный аппарат, связанный с созданием, получением, обработкой, интерпретацией и хранением информационного контента;
- обеспечить базовое изучение языка разметки гипертекста HTML, каскадных стилей CSS;
- познакомить с различными способами создания графической информации, особенностями использования графических элементов при построении веб-сайтов;
- обеспечить базовое изучение основ программирования на Python;
- рассмотреть и практически освоить создание компьютерных игр;
- развивать логическое и алгоритмическое мышление, пространственное воображение;
- формировать интерес к цифровой трансформации современной экономики в стране и мире;
- формировать 4К компетенций (критическое мышление, креативное мышление, коммуникация, кооперация);
- воспитывать ответственное отношение к создаваемому информационному контенту, его содержанию и культуре оформления.

Адресат программы. Программа ориентирована на дополнительное образование подростков в возрасте 11-13 лет, мотивированных к обучению и обладающих системным мышлением.

Режим занятий - 2 раза по 2 часа в неделю.

Продолжительность 1 занятия: 2 академических часа.

Структура двухчасового занятия:

- 40 минут – рабочая часть;
- 10 минут – перерыв (отдых);
- 40 минут – рабочая часть.

Формы организации образовательного процесса.

Формы занятий групповые, количество обучающихся в группе – 15 человек. Состав групп постоянный.

Образовательный процесс по данной программе может строиться как в традиционной очной форме, так и с использованием дистанционных технологий обучения с помощью Интернет-ресурсов дистанционного обучения, блогов, сообществ, рассылки обучающих материалов по электронной почте. Программа предусматривает предоставление учащимся возможности очной защиты подготовленных заочно проектов.

Методы: проблемный, поисковый, исследовательский, кейс-метод, проектная деятельность.

Формы работы:

- практическое занятие;
- занятие – соревнование;
- деловая игра;
- самостоятельная работа.

Виды учебной деятельности:

- анализ проблемных учебных ситуаций;
- систематизация данных;
- программирование;
- построение математических моделей физических процессов;
- построение алгоритмических конструкций для программной реализации математических моделей;
- определение свойств приборов по чертежам и моделям;
- поиск необходимой информации;
- выполнение практических работ;
- конструирование и моделирование;
- подготовка выступлений и докладов с использованием разнообразных источников информации;
- публичное выступление.

Требования к результатам освоения программы:

В результате освоения общеобразовательной программы «Разработчик веб и мультимедийных приложений» обучающийся должен:

уметь:

- самостоятельно разрабатывать корректную структуру веб-

приложения;

- создавать динамические веб-сайты;
- пользоваться наиболее распространенными HTML-редакторами;
- реализовывать дизайн веб-сайта, используя графические редакторы

Adobe Photoshop или Figma;

- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;
- настраивать рабочую среду Python;
- активно участвовать в коллективной работе, самостоятельно и творчески решать поставленные задачи в среде программирования Scratch.

знать:

- определения ключевых понятий и терминов веб-программирования;
- синтаксис каскадных таблиц стилей (CSS);
- основные возможности языка гипертекстовой разметки HTML;
- синтаксис функционального программирования на Python;
- способы создания и добавления на веб-страницу основных ее

элементов;

- основные функции и возможности веб-редактора Sublime Text;
- методы разработки приложений средствами Python.

обладать навыками:

- разработки полноценных приложений средствами HTML, CSS, Python;
- создания прототипа и макета веб-сайта или приложения согласно принципам дизайна.

Результат (освоенные компетенции)	Основные показатели оценки результата	Формы и методы контроля и оценки
Личностные компетенции	Умение работать в команде: работа в общем ритме, эффективное распределение задач и др.	Проектная деятельность Деловые игры Кейс-задачи
	Социальная ответственность и способность выполнять роль лидера в данной ответственности.	Творческие задания Ролевая игра Коллоквиум
	Критическое мышление	Портфолио
	Эмоциональный интеллект, способность воспринимать чувства и эмоции других людей (сенситивность), понимать их намерения, мотивацию и желания.	Практические задачи Участие в конференциях, выставках, хакатонах, конкурсов

	Способность резюмировать, осмысливать, анализировать и объединять различные исходные данные в последовательную систему.	
	Обладание чувством самоорганизации и чувством времени.	
Метапредметные компетенции	умение ориентироваться в информационном пространстве, продуктивно использовать техническую литературу для поиска сложных решений	Проектная деятельность Деловые игры Кейс-задачи Творческие задания Ролевая игра Коллоквиум Портфолио Практические задачи Участие в конференциях, выставках, хакатонах, конкурсов
	способность творчески решать технические задачи	
	готовность и способность применения теоретических знаний по физике, информатике для решения задач в реальном мире	
	способность правильно организовывать рабочее место и время для достижения поставленных целей	
	Знание основ ТРИЗ, навыки публичного выступления и презентации результатов, навык генерации идей	
Предметные компетенции	система команд исполнителя Scratch; линейный алгоритм, цикл, ветвления, их реализация в среде Scratch;	Проектная деятельность Кейс-задачи Творческие задания Практические задачи Участие в конференциях, выставках, хакатонах, конкурсов
	понятие проекта, его структура и реализация в среде Scratch.	
	понимание принципов составления математических и алгоритмических моделей для решения прикладных задач веб-программирования	
	овладение практическими базисными знаниями веб-программирования на HTML и CSS	
	знание и понимание основных алгоритмических конструкций	
	знание основ и овладение практическими базисными знаниями программирования в Python	

Формы подведения итогов реализации программы

Основной формой подведения итогов дополнительной общеразвивающей программы «Основы программирования» является

проектная деятельность (разработка проекта и полное сопровождения ресурса отраслевой тематики).

Критерии оценки защиты проекта.

Критерии оценки содержания проекта

№	Критерий оценивания	Аспект оценивания	Максимальный балл
ОЦЕНКА ПРОЕКТА			
1	Целеполагание	<p>0 баллов:</p> <ul style="list-style-type: none"> - отсутствует описание цели проекта. - не определён круг потенциальных заказчиков / потребителей / пользователей. - не определены показатели назначения. <p>1 балл:</p> <ul style="list-style-type: none"> -обозначенная цель проекта не обоснована (не сформулирована проблема, которая решается в проекте) или не является актуальной в современной ситуации. -круг потенциальных заказчиков / потребителей / пользователей не конкретен. -заявленные показатели назначения не измеримы, либо отсутствуют. <p>3 балла:</p> <ul style="list-style-type: none"> -цель проекта обоснована (сформулирована проблема, которая решается в проекте) и является актуальной в современной ситуации. -представлено только одно из следующего: 1) чётко обозначен круг потенциальных заказчиков / потребителей / пользователей. 2) заявленные показатели назначения измеримы. <p>5 баллов:</p> <p>Есть:</p> <ul style="list-style-type: none"> -конкретная формулировка цели проекта и проблемы, которую проект решает; -актуальность проекта обоснована; -чётко обозначен круг потенциальных заказчиков / потребителей / пользователей. -заявленные показатели назначения измеримы. 	5

2	Анализ существующих решений и методов	<p>0 баллов: -нет анализа существующих решений.</p> <p>1 балл: -есть неполный анализ существующих решений проблемы и их сравнение.</p> <p>3 балла: -дана сравнительная таблица аналогов с указанием показателей назначения. Выявленные в результате сравнительного анализа преимущества предлагаемого решения не обоснованы, либо отсутствуют.</p> <p>5 баллов: -есть подробный анализ существующих в практике решений, сравнительная таблица аналогов с указанием преимуществ предлагаемого решения.</p>	5
3	Планирование работ, ресурсное обеспечение проекта	<p>0 баллов: -отсутствует план работы. Ресурсное обеспечение проекта не определено. Способы привлечения ресурсов в проект не проработаны.</p> <p>5 баллов: Есть только одно из следующего: 1) план работы, с описанием ключевых этапов и промежуточных результатов, отражающий реальный ход работ; 2) описание использованных ресурсов; 3) способы привлечения ресурсов в проект.</p> <p>7 баллов: Есть только два из следующего: 1) план работы, с описанием ключевых этапов и промежуточных результатов, отражающий реальный ход работ; 2) описание использованных ресурсов; 3) способы привлечения ресурсов в проект.</p> <p>10 баллов: - есть подробный план, описание использованных ресурсов и способов их привлечения для реализации проекта.</p>	10
4	Качество результата	<p>0 баллов: -нет подробного описания достигнутого результата. -нет подтверждений (фото, видео, скриншотов) полученного результата. -отсутствует программа и методика испытаний/тестового запуска. -не приведены полученные в ходе испытаний показатели назначения.</p> <p>5 баллов:</p>	10

		<p>- дано подробное описание достигнутого результата.</p> <p>- есть видео и фото-подтверждения работающего образца/макета/прототипа.</p> <p>- отсутствует программа и методика испытаний/тестового запуска.</p> <p>- тестовые запуски не проводились.</p> <p>7 баллов:</p> <p>- дано подробное описание достигнутого результата.</p> <p>- есть видео и фото-подтверждения работающего образца/макета/прототипа.</p> <p>- приведена программа и методика испытаний/тестового запуска.</p> <p>- полученные в ходе испытаний показатели назначения не в полной мере соответствуют заявленным.</p> <p>10 баллов:</p> <p>- дано подробное описание достигнутого результата.</p> <p>- есть видео и фото-подтверждения работающего образца/макета/модели.</p> <p>- приведена программа и методика испытаний/тестового запуска.</p> <p>- полученные в ходе испытаний показатели назначения в полной мере соответствуют заявленным.</p>	
		ОЦЕНКА ПРЕЗЕНТАЦИИ ПРОЕКТА	
1	Качество устного выступления	<p>0 баллов:</p> <p>- устное выступление участника не логично, присутствуют грамматические и лексические ошибки, которые затрудняют понимание</p> <p>3 балла:</p> <p>- устное выступление участника не всегда логично, присутствуют незначительные грамматические и лексические ошибки</p> <p>5 баллов:</p> <p>- устное выступление участника не всегда логично, но отсутствуют грамматические и лексические ошибки.</p> <p>7 баллов:</p> <p>- устное выступление участника логично, отсутствуют грамматические и лексические ошибки</p>	7
2	Самостоятельность работы над проектом и	<p>0 баллов:</p> <p>- участник не может точно описать ход работы над проектом, нет понимания</p>	6

	уровень командной работы	<p>личного вклада и вклада других членов команды.</p> <p>-низкий уровень осведомлённости в профессиональной области.</p> <p>2 балла:</p> <p>-участник может описать ход работы над проектом, выделяет личный вклад в проект, но не может определить вклад каждого члена команды.</p> <p>-уровень осведомлённости в профессиональной области, к которой относится проект не достаточен для дискуссии.</p> <p>4 балла:</p> <p>-участник может описать ход работы над проектом, выделяет личный вклад в проект, но не может определить вклад каждого члена команды.</p> <p>-уровень осведомлённости в профессиональной области, к которой относится проект достаточен для дискуссии.</p> <p>6 баллов:</p> <p>-участник может описать ход работы над проектом, выделяет личный вклад в проект и вклад каждого члена команды.</p> <p>-уровень осведомлённости в профессиональной области, к которой относится проект, достаточен для дискуссии.</p>	
3	Качество ответов на вопросы экспертов	<p>0 баллов:</p> <p>- ответы на вопросы отсутствовали в полном объеме.</p> <p>1 балл:</p> <p>- участник затруднялся давать правильные ответы на вопросы.</p> <p>4 балла:</p> <p>- в ходе устного выступления даны ответы на некоторые вопросы.</p> <p>6 баллов:</p> <p>- в ходе устного выступления даны ответы на все вопросы.</p>	6
4	Качество оформления презентации	<p>0 баллов:</p> <p>- презентация отсутствует.</p> <p>1 балл:</p> <p>- оформление презентации на низком уровне: нечитабельный шрифт, несоразмерные таблицы/количество текста на слайде.</p> <p>2 балла:</p> <p>- оформление презентации на среднем уровне: визуальная информация</p>	3

		воспринимается хорошо, но есть мелкие недочеты. -3 балла: - оформление презентации на высоком уровне: информация визуально приятная, понятная и ориентирована на целевую аудиторию	
5	Соответствие текста доклада и презентации	0 баллов: - текст выступления в полном объеме дублирует презентацию. 1 балл: - выступление частично повторяет текст презентации или публикации. 3 балла: - выступление не повторяет текст презентации или публикации, но логично дополняет его	3
		ИТОГО	55

Критерии оценки технической части IT-проекта

№	Критерий оценивания	Аспект оценивания	Максимальный балл
ОЦЕНКА ПРОГРАММНОГО ПРОДУКТА			
1	Соответствие реализации функций программного обеспечения задачам пользователя	0 баллов: функционал программного обеспечения (прототипа) не решает поставленные пользователем задачи. 1 балл: функционал программного обеспечения (прототипа) решает поставленные пользователем задачи на 30% 2 балла: функционал программного обеспечения (прототипа) решает поставленные пользователем задачи на 50% и более. 3 балла: функционал программного обеспечения (прототипа) в полной мере решает задачи, поставленные заказчиком	3
2	Соответствие применяемых цифровых технологий поставленной задаче	0 баллов: все выбранные цифровые технологии не предназначены для решения задач создаваемого программного обеспечения. 1 балл: выбранные цифровые технологии частично предназначены для решения задач создаваемого программного обеспечения. 2 балла: выбранные цифровые технологии в полной мере предназначены для решения задач создаваемого программного	3

		<p>обеспечения, но не рассмотрены альтернативные цифровые технологии.</p> <p>3 балла: все выбранные цифровые технологии в полной мере предназначены для решения задач создаваемого программного обеспечения.</p>	
3	<p>Полнота автоматизации задач пользователя</p>	<p>0 баллов: функционал программного обеспечения (прототипа) не автоматизирует операции пользователя</p> <p>1 балл: функционал программного обеспечения (прототипа) частично автоматизирует операции пользователя</p> <p>2 балла: функционал программного обеспечения (прототипа) на 50 и более процентов автоматизирует операции пользователя</p> <p>3 балла: функционал программного обеспечения (прототипа) в полной мере автоматизирует операции пользователя</p>	3
	<p>Выходные результаты работы приложения (данные) представлены в удобном виде для пользователя</p>	<p>0 баллов: нет</p> <p>1 балл: да</p>	1
	<p>Возможность адаптации программного обеспечения к конкретным условиям эксплуатации предназначенными для этого средствами</p>	<p>0 баллов: адаптация программного продукта не возможна, или требует несоизмеримых ресурсов.</p> <p>1 балл: адаптация программного продукта возможна, но требует несоизмеримых ресурсов</p> <p>2 балла: адаптация программного продукта возможна в полной мере со значительными ресурсными затратами.</p> <p>3 балла: адаптация программного продукта возможна в полной мере с минимальными ресурсными затратами.</p>	3
	<p>Обеспечение безопасности данных, вводимых пользователем программного продукта</p>	<p>0 баллов: не соблюдены требования по защите данных.</p> <p>1 балл: предпринята попытка организации безопасности данных, но практически не реализована.</p>	3

		<p>2 балла: предпринята попытка организации безопасности данных, практически реализована на 30-50%.</p> <p>3 балла: организованы мероприятия по защите данных пользователя на более чем 50%.</p>	
	Соответствие программного обеспечения деловой практике (терминологии, стандартным формам документов, логике решения задач)	<p>0 баллов: не соответствует</p> <p>1 балл: соответствие минимальное</p> <p>2 балла: частичное соответствие</p> <p>3 балла: полное соответствие</p>	3
НАЛИЧИЕ ОШИБОК			
1	Противоречие в выполнении алгоритмов	<p>3 балла: отсутствие ошибок</p> <p>- 0,5 балла за каждый вид ошибки</p>	3
2	Ошибка в вычислениях	<p>3 балла: отсутствие ошибок</p> <p>- 0,5 балла за каждый вид ошибки</p>	3
3	Несовместимость форматов файлов и данных	<p>3 балла: отсутствие ошибок</p> <p>- 0,5 балла за каждый вид ошибки</p>	3
4	Отсутствие диагностического сообщения в случае сбоя или отказа	<p>1 балл: отсутствие ошибок</p> <p>- 0,5 балла за каждый вид ошибки</p>	1
5	Неполнота контроля корректности, полноты и непротиворечивости входных, выходных данных и баз данных	<p>2 балла: отсутствие ошибок</p> <p>- 0,5 балла за каждый вид ошибки</p>	2
6	Затруднения при загрузке и запуске программного обеспечения	<p>2 балла: отсутствие ошибок</p> <p>- 0,5 балла за каждый вид ошибки</p>	2
7	Неудобство ввода данных	<p>1 балл: отсутствие ошибок</p> <p>- 0,5 балла за каждый вид ошибки</p>	1
8	Отсутствие значений по умолчанию	<p>3 балла: отсутствие ошибок</p> <p>- 0,5 балла за каждый вид ошибки</p>	3
9	Затруднения восприятия выходных данных	<p>2 балла: отсутствие ошибок</p> <p>- 0,5 балла за каждый вид ошибки</p>	2
10	Наличие непонятных сообщений	<p>2 балла: отсутствие ошибок</p> <p>- 0,5 балла за каждый вид ошибки</p>	2

11	Низкая читабельность кода	2 балла: отсутствие ошибок - 0,5 балла за каждый вид ошибки	2
12	Недостаточно комментариев	2 балла: отсутствие ошибок - 0,5 балла за каждый вид ошибки	2
		ИТОГО	45

2. СТРУКТУРА ПРОГРАММЫ

2.1. Объем программы

Год обучения	Уровень	Кол-во часов
1 год	углубленный уровень	144

2.2. Учебный план

№ п/п	Наименование раздела, темы	Количество часов			Формы организации занятий	Формы контроля
		Всего	Теория	Практика		
	Вводный урок. Знакомство с профессиями IT-сектора. Техника безопасности и организация рабочего места.	2	1	1	Лекция Деловая игра Практическая работа	Заполнение карточки профессий
1.	Раздел 1. Основы разработки сайтов на HTML 5 и CSS	46	8	38		
1.1	Язык HTML 5. Структура HTML документа. Понятие тег, атрибут тега.	2	1	1	Лекция, практическая работа на платформе HTML-академия	Упражнение Head Body
1.2	HTML-теги. Одиночные и парные. Знакомство и работа в Brackets. Служебные теги.	2	0	2	Учебная игра, практическая работа на платформе HTML-академия	Опрос в Learning Apps
1.3	Форматы графических изображений, используемых в Интернет их особенности. Подготовка графических изображений к публикации. Оптимизация графики. Вставка изображений на страницу и их свойства	4	1	3	Пресс-конференция, практическая работа	Заполнение сводной таблицы
1.4	Абсолютные и локальные ссылки. Создание	6	1	5	Учебная игра	Тематический диктант

	многостраничного документа					
1.5	Параметры таблицы в HTML. Строки и ячейки. Объединение ячеек	6	1	5	Практическая работа	Самостоятельная индивидуальная работа (Задание 1-2)
1.6	Атрибут, селектор, подключение CSS.	4	1	4	Лекция, практическая работа в HTML Academy	Тест в Learning Apps. Соревнование по исправлению базовой разметки с ограничением времени
1.7	Флекбоксы	6	1	5	Практическая работа в HTML Academy	Игра на скорость во Flexbox Froggy
1.8	Div, span и display. Блочная модель верстки сайта	6	1	5	Проблемная лекция, практическая работа в HTML Academy	Настольная игра на запоминание тегов Отчет по валидации кода
1.9	Практика блочной верстки макетов веб-приложения	6	1	5	Практическая работа	Чек-лист «Блочная верстка»
1.10	Разработка макетов сайта средствами графического редактора Figma	4	0	4	Творческая мастерская	Заполнение чек-листа Смотр творческих работ
2	Раздел 2. Разработка приложений на Python	48	8	40		
2.1	История языков программирования. Введение в язык программирования Python. Среда программирования Python. Установка программы.	2	1	1	Практическая работа	Тестирование
2.2	Типы данных и функции вывода.	4	1	3	Практическая работа	Опрос

	Определение переменной. Переменные и арифметические выражения.					
2.3	Чтение данных. Операции над строками. Примеры решения задач.	4	1	3	Практическая работа	Самостоятельная работа
2.4	Отработка навыков решения простейших задач.	4	0	4	Практическая работа	Заполнение чек-листа по итогам тестирования ПО
2.5	Логический тип данных и операции. Примеры использования логических выражений.	4	1	3	Учебная игра	Написание отчета
2.6	Логические выражения и операторы. Условный оператор IF. Множественное ветвление	4	1	3	Практическая работа	Решение задач
2.7	Отработка навыков решения простейших задач.	4	0	4	Практическая работа	Заполнение чек-листа по итогам тестирования ПО
2.8	Цикл WHILE. Примеры решения задач.	4	1	3	Практическая работа	Мини-проект
2.9	Оператор цикла с параметром FOR	2	1	1	Практическая работа	Написание отчета
2.10	Отработка навыков решения простейших задач.	4	0	4	Практическая работа	Решение задач
2.11	Создание функций. Локальные и глобальные переменные	2	1	1	Практическая работа	Тестирование
2.12	Решение задач с использованием функций	2	0	4	Практическая работа	Заполнение чек-листа по итогам тестирования ПО
2.13	Разработка собственных консольных приложений	8	0	8	Творческая мастерская	Заполнение чек-листа Смотр творческих работ
3	Раздел 3. Разработка компьютерных игр на Scratch	46	16	30		

3.1	Знакомство со средой. Написание первой программы	2	1	1	Практическая работа	Опрос
3.2	Координаты. Система координат. Новые объекты	2	1	1	Практическая работа	Выполнение задания
3.3	Одновременное выполнение скриптов (программ)	2	1	1	Практическая работа	Решение задач
3.4	Последовательное выполнение скриптов (программ)	2	1	1	Практическая работа	Решение задач
3.5	Рассмотрение циклов. Создание простых мультфильмов	2	1	1	Практическая работа	Мини-проект
3.6	Обмен сообщениями	2	1	1	Практическая работа	Решение задач
3.7	Создание программы с обменом сообщениями	2	1	1	Практическая работа	Разработка сценария игры
3.8	Знакомство с музыкальными возможностями Scratch.	2	1	1	Практическая работа	Решение задач
3.9	Редактирование фона. Создание нового фона. Программирование фона	2	1	1	Практическая работа	Разработка сценария игры
3.10	Сценарий путешествия объекта по нескольким сценам	2	1	1	Практическая работа	Разработка сценария игры
3.11	Интерактивность	2	1	1	Практическая работа	Решение задач
3.12	Переменные и условный оператор	2	1	1	Практическая работа	Решение задач
3.13	Случайное число. Сценарий со случайными числами	2	1	1	Практическая работа	Мини-проект
3.14	Использование генератора случайных чисел при создании программ в среде Scratch. Камень, ножницы, бумага	2	1	1	Практическая работа	Решение задач
3.15	Рисование мышью. Работа с костюмами	2	1	1	Практическая работа	Мини-проект
3.16	Использование библиотек и объектов	2	1	1	Практическая работа	Решение задач
3.17	Программирование игр интерактивных мультфильмов	12	0	12	Творческая мастерская	Заполнение чек-листа

						Смотр творческих работ
Защита итоговых проектов	2	0	2	Семинар		Протокол оценки проекта
Итого	144	33	111			

2.3. Содержание учебного плана

Вводный урок. Знакомство с профессиями IT-сектора. Техника безопасности и организация рабочего места.

Теория. Инструктаж учащихся по правилам поведения, технике безопасности работы в компьютерной лаборатории и правилам дорожного движения. Обзор профессий IT-отрасли с применением Атласа профессий будущего.

Практика. Создание собственных папок на Рабочем столе. Настройка рабочего стола. Создание специального почтового ящика для получения заданий на урок.

РАЗДЕЛ 1. ОСНОВЫ РАЗРАБОТКИ САЙТОВ НА HTML 5

Тема 1.1. Язык HTML 5. Структура HTML документа. Понятие тег, атрибут тега.

Теория. Синтаксис языка. Тэги. Тэг-комментарий. Структура Веб - страницы Заголовок документа. Тело документа. Атрибуты тэгов.

Практика. Разработка проекта первой страницы. Создать страницу с приветствием и сведениями о себе. Создать страницу с различными списками. Создать страницы, содержащие всевозможные тэги текста. Увеличение, уменьшение, стили, шрифты, индексы и т.д.

Тема 1.2. HTML-теги. Одиночные и парные. Знакомство и работа в Brackets. Служебные теги.

Практика. Цвет фона. Изображение как фон. Цвет текста. Цвета. Размер и форма шрифта. Тэги форматирования текста. Взаимодействие тэгов Текстовые блоки. Заголовки. Абзацы. Перевод строки. Разделительная линия. Списки. Дополнить созданную ранее страницу объектами.

Тема 1.3. Форматы графических изображений, используемых в Интернет их особенности. Подготовка графических изображений к публикации. Оптимизация графики. Вставка изображений на страницу и их свойства

Теория. Растровые форматы JPEG, SVG, GIF, PNG. Прозрачная графика.

Анимация. Тэг `img` и его параметры: `src` – URL файла источника изображения, `width` – ширина изображения, `height` – высота изображения, `alt` – альтернативный текст

Практика. Разработка проекта страницы с эксклюзивной графикой (фон, логотип, иконка). Создать страницу, содержащую графику, как фон, как содержимое ячейки, как отдельный абзац.

Тема 1.4. Абсолютные и локальные ссылки. Создание многостраничного документа

Практика. Виды ссылок в HTML. Общий вид кода якоря. Код классического якоря (тег `<a>`). Якорь в заголовке, в списке, в таблице. Ссылка на якорь. Локальные ссылки на якорь, расположенный в пределах одной страницы. Абсолютная ссылка на якорь. Работа с коллекцией гипертекстовых ссылок и программирование гипертекстовых переходов в зависимости от условий просмотра HTML-страниц и действий пользователя. Разработка программного кода тренировочных заданий.

Тема 1.5. Параметры таблицы в HTML. Строки и ячейки. Объединение ячеек

Практика. Тэг `table` и его атрибуты: Ширина таблицы. Границы таблицы. Выравнивание таблицы в родительском окне. Цвет фона таблицы. Отступ от содержимого до границы ячейки. Расстояние между ячейками. Запрет пустых ячеек - код пробела. Строка-заголовок. Объединение двух ячеек по горизонтали: `<td colspan="2">1`. Объединение двух ячеек по вертикали: `<td rowspan="2">3`. Создать страницу, содержащую таблицу-расписание событий в текущем месяце. Разработка проекта сайта с помощью таблиц. Создать страницу, содержащую почтовую ссылку, предложить написать отзыв на работу. Организовать скачивание книги с Вашей страницы.

Тема 1.6. Атрибут, селектор, подключение CSS.

Теория. Стил. Таблицы стилей. Язык CSS. Расширение CSS. Стил, имеющий уникальное имя - стилевой класс. Синтаксис языка CSS: `selector {attribute1:value1[; attribute2:value2]}`. Комментарии в CSS. Три способа задания стиля.

Практика. Написать примеры стилевых описаний для различных тэгов. Задать на странице всеми способами стилевые описания и определить приоритет стилевых описаний. Создать страницу, где одни и те же тэги относятся к различным классам и поэтому имеют различные стили.

Тема 1.7. Флекбоксы

Практика. Стилиевое описание тэга `A`. Главное свойство `display: flex` и `display`: Разработка проекта сайта с динамическими панелями навигации. Создать страницу с горизонтальными и вертикальными панелями навигации на тему «Мои Интересы и Мои Умения». Создать страницу с динамической панелью навигации на тему «кружки, которые я посещаю». Создать страницу, соответствующую шаблону, состоящую из блоков с заданными ID.

Тема 1.8. Div, span и display. Блочная модель верстки сайта

Теория. Блочный шаблон (шапка, навигация, контент, подвал). Достоинства и недостатки. Содержимое блока. Геометрия. Расположение. Вложенность. Блочные элементы. Тэг div. Ширина и высота. Единицы измерения размеров. Спецификации CSS для блока div. Свойства блоков. Расположение друг относительно друга.

Практика. Разработка проекта сайта с блочной версткой. Создать страницу «Космические полеты», в соответствии с заданным шаблоном используя блочную верстку. Построить и оформить блок «Заголовок», состоящий из логотипа, названия и сведений об авторе.

Тема 1.9. Практика блочной верстки макетов веб-приложения

Практика. Разработка проекта многостраничного сайта средствами HTML и CSS. Тематика проектов и необходимый текстовый и графический контент выбирается обучающимися самостоятельно.

Тема 1.10. Разработка макетов сайта средствами графического редактора Figma

Практика. Обзор интерфейса, манипуляция объектами, слои и маски, работа с текстом, цветовое решение, сетки и направляющие. Монтажные области. Размер модулей и модульной сетки. Разработка тематического макета веб-приложения средствами Figma с учетом базовых принципов дизайна.

РАЗДЕЛ 2. РАЗРАБОТКА ПРИЛОЖЕНИЙ НА PYTHON

Тема 2.1. История языков программирования. Введение в язык программирования Python. Среда программирования Python. Установка программы.

Теория. Установка Python. Доступ к документации. Основы ввода и вывода данных. Первая программа на языке Python. Основы Алгоритмизации.

Практика. Решение задач для развития логики и понимания работы алгоритма. Установка программы Python. Написание первой программы вывод на экран.

Тема 2.2. Типы данных и функции вывода. Определение переменной. Переменные и арифметические выражения.

Теория. Типы данных. Преобразование типов. Различия типов данных. Базовые функции (abs, round, int, math)

Практика. Решение математических задач с использованием функции import math.

Тема 2.3. Чтение данных. Операции над строками. Примеры решения задач.

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.

Практика. Решение задач со строками.

Тема 2.4. Отработка навыков решения простейших задач.

Практика. Решение задач

Тема 2.5. Логический тип данных и операции. Примеры использования логических выражений.

Теория. Математические 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.

Практика. Решение задач на логический тип данных.

Тема 2.6. Логические выражения и операторы. Условный оператор IF. Множественное ветвление

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.

Практика. Практическое закрепление знаний по условным операторам. Создание программ.

Тема 2.7. Отработка навыков решения простейших задач.

Практика. Решение задач

Тема 2.8. Цикл WHILE. Примеры решения задач.

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.

Практика. Числа Фибоначчи. Создание игры угадай число. Повторение пройденного.

Тема 2.9. Оператор цикла с параметром FOR

Теория. 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.

Практика. Решение задачи с циклом for.

Тема 2.10. Отработка навыков решения простейших задач.

Практика. Решение задач

Тема 2.11. Создание функций. Локальные и глобальные

переменные.

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.

Практика. Создание игры русская рулетка. Создание всех ранее созданных программ с использованием функций.

Тема 2.12. Решение задач с использованием функций.

Практика. Решение задач

Тема 2.13. Разработка собственных консольных приложений

Практика. Разработка консольных приложений согласно техническому заданию.

РАЗДЕЛ 3. РАЗРАБОТКА КОМПЬЮТЕРНЫХ ИГР НА SCRATCH

Тема 3.1. Знакомство со средой. Написание первой программы

Теория. Знакомство с детьми. О программе и прогнозируемых результатах. Правила ТБ и внутреннего распорядка. Знакомство со средой Скретч. Понятие спрайта и объекта.

Практика. Создание и редактирование спрайтов и фонов для сцены.

Тема 3.2. Координаты. Система координат. Новые объекты

Теория. Создание и редактирование спрайтов и фонов для сцены. Пользуемся помощью Интернета. Поиск, импорт и редакция спрайтов и фонов из Интернета.

Практика. Создание и редактирование спрайтов и фонов для сцены, работа с координатной плоскостью.

Тема 3.3. Одновременное выполнение скриптов (программ)

Теория. Датчики в Скретче и их значение. Понятие переменной и константы. Создание переменных. Предоставление переменным значений, пересмотр значений переменных. Команды предоставления переменных значений. Использование переменных.

Практика. Создание и редактирование готовых сценариев.

Тема 3.4. Последовательное выполнение скриптов (программ)

Теория. Управление спрайтами: команды идти, повернуться на угол, опустить перо, поднять перо, очистить. Координатная плоскость. Точка отсчёта, оси координат, единица измерения расстояния, абсцисса и ордината. Навигация в среде Scratch. Определение координат спрайта. Команда идти в точку с заданными координатами.

Практика. Создание проекта «Кругосветное путешествие Магеллана». Команда плыть в точку с заданными координатами. Режим презентации.

Тема 3.5. Рассмотрение циклов. Создание простых мультфильмов

Теория. Понятие цикла. Команда повторить. Рисование узоров и орнаментов. Конструкция всегда. Создание проектов «Берегись автомобиля!» и «Гонки по вертикали». Команда если край, оттолкнуться. Ориентация по компасу. Управление курсом движения. Команда повернуть в направление.

Практика. Проект «Полёт самолёта». Спрайты меняют костюмы. Анимация. Создание проектов «Осьминог», «Девочка, прыгающая на скакалке» и «Бегущий человек». Создание мультипликационного сюжета «Кот и птичка».

Тема 3.6. Обмен сообщениями

Теория. Понятие сообщения. Передача сообщения, запуск скриптов при условии получения сообщения вызова. Обмен данными между скриптами.

Практика. Разработка игр с передачей сообщений.

Тема 3.7. Создание программы с обменом сообщениями

Практика. Разработка игр с передачей сообщений с проработкой сценария.

Тема 3.8. Знакомство с музыкальными возможностями Scratch.

Теория. Команды музыкального сопровождения. Использование библиотеки звуков, запись собственной озвучки.

Практика. Создание проектов с программируемым построением музыкальных фрагментов на сцене путем перемещения спрайтов.

Тема 3.9. Редактирование фона. Создание нового фона.

Программирование фона

Теория. Команды рисования. Создание проектов с программируемым построением изображений на сцене путем перемещения спрайтов. Использование команды Штамп.

Практика. Разработка игр с применением команд рисования.

Тема 3.10. Сценарий путешествия объекта по нескольким сценам.

Теория. Изучение правил взаимодействия двух героев. Введение нового персонажа. Задание определенных действий. Построение взаимодействия между ними.

Практика. Создание нескольких игровых сцен, организация их взаимодействия.

Тема 3.11. Интерактивность

Практика. Разработка и создание небольшой программы с использованием заранее подготовленных материалов. Тестирование и отладка проекта. Защита проекта

Тема 3.12. Переменные и условный оператор

Теория. Соблюдение условий. Сенсоры. Блок если. Управляемый стрелками спрайт. Переменные. Их создание. Использование счётчиков. Проект «Голодный кот». Ввод переменных.

Практика. Создание коллекции игр: «Лабиринт», «Кружащийся котёнок». Пополнение коллекции игр: «Опасный лабиринт». Составные

условия. Проекты «Хождение по коридору», «Слепой кот», «Тренажёр памяти». Датчик случайных чисел. Проекты «Разноцветный экран», «Хаотичное движение», «Кошки-мышки», «Вырастим цветник». Циклы с условием. Проект «Будильник». Запуск спрайтов с помощью мыши и клавиатуры. Проекты «Переодевалки» и «Дюймовочка». Самоуправление спрайтов. Обмен сигналами. Блоки передать сообщение и Когда я получу сообщение. Проекты «Лампа» и «Диалог». Датчики. Проекты «Котёнок-обжора», «Презентация».

Тема 3.13. Случайное число. Сценарий со случайными числами

Теория. Понятие операции и выражения. Арифметические операции. Основные правила построения, вычисления и использования выражений. Присвоение значений выражений переменным. Понятие локальной и глобальной переменной. Генератор псевдослучайных чисел.

Практика. Разработка игр по сценарию.

Тема 3.14. Использование генератора случайных чисел при создании программ в среде Scratch. Камень, ножницы, бумага.

Практика. Разработка мультфильма с написанием сценария.

Тема 3.15. Рисование мышью. Работа с костюмами

Теория. Создание собственноручно нарисованных персонажей. Добавление их в программную среду и в приложение. Действия с ними.

Практика. Создание собственных спрайтов.

Тема 3.16. Использование библиотек и объектов

Теория. Списки Понятие списка. Создание списков. Понятие индекса, как номера элемента списка. Предоставление значений элементам списка и отображения его содержания. Поиск необходимых данных в списке. Вычисление итоговых показателей для списка. Вычисление итоговых показателей для элементов списка, которые отвечают определенным критериям. Алгоритмы сортировки списков.

Практика. Разработка сценариев с алгоритмами сортировки.

Тема 3.17. Программирование игр интерактивных мультфильмов

Практика. Создание тестов – с выбором ответа и без. Создание проектов по собственному замыслу. Регистрация в Scratch-сообществе. Публикация проектов в Сети.

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

3.1. Материально-техническое оснащение

Компьютерное оборудование:

- персональный компьютер – 15 шт.
- моноблок/персональный компьютер – 1 шт.
- маршрутизатор – 1 шт.
- коммутатор – 1 шт.

Программное обеспечение:

- ОС Windows
- Figma (версия Desktop)
- Sublime Text 3
- GIMP
- Microsoft Office
- Интерактивная платформа HTML-академия
- Scratch 3.0
- IDLE — Python 3.10.6

Презентационное оборудование:

- проектор – 1 шт.

Дополнительное оборудование:

- учительский стол – 1 шт.
- учительский стул – 1 шт.
- парты двухместные – 8 шт.
- стулья ученические – 16 шт.

3.2. Методическое обеспечение реализации программы

Основными принципами в освоении дополнительной общеобразовательной общеразвивающей программы «Основы программирования» являются:

- наглядность,
- систематичность и последовательность обучения, доступность,
- связь теории с практикой.

В процессе обучения педагог исходит из индивидуальных особенностей детей, опираясь на сильные стороны ребенка, доводит его подготовленность до уровня общих требований.

Принцип *наглядности* вытекает из сущности процесса восприятия, осмысления и обобщения учащимися изучаемого материала. На отдельных этапах изучения учебного материала наглядность выполняет различные функции. Когда учащиеся изучают внешние свойства предмета, то, рассматривая предмет или его изображение, они могут сами извлекать необходимые знания.

Обучение должно быть систематичным и последовательным.

Необходимо руководствоваться правилами дидактики:

- от близкого к далекому,
- от простого к сложному,
- от более легкого к более трудному,
- от известного к неизвестному.

Систематичность обучения предполагает такое построение учебного процесса, в ходе которого происходит связывание ранее усвоенного материала с новым. В процессе обучения происходит знакомство с основной терминологией веб-программирования, математики, информатики, принципами построения различных видов алгоритмов.

Учёт возрастных различий и особенностей учащихся находит выражение в принципе *доступности* обучения, которое должно проводиться так, чтобы изучаемый материал по содержанию и объёму был посилен учащимся. Применяемые методы обучения должны соответствовать возрасту учащихся, развивать их силы и способности.

Для реализации программы используются разнообразные формы и методы проведения занятий.

Занятия сопровождаются использованием наглядного материала. Программно-методическое и информационное обеспечение помогают проводить занятия интересно и грамотно. Разнообразные занятия дают возможность учащимся проявить свою индивидуальность, самостоятельность, способствуют гармоничному развитию личности. Игровые приемы, соревнования в рамках объединения, тематические вопросы также помогают при творческой работе.

На занятиях используются различные формы организации образовательного процесса:

- фронтальные (беседа, рассказ, демонстрация, показ, проверочная работа);
- групповые (олимпиады, фестивали, соревнования, хакатоны);
- индивидуальные (инструктаж, разбор ошибок и т.д.).

Эффективность обучения по данной программе зависит от организации занятий, проводимых с применением следующих методов по способу получения знаний:

- *объяснительно-иллюстративный* - представление информации различными способами (объяснение, рассказ, беседа, инструктаж, демонстрация, работа с техническими приспособлениями для проведения опытов, и др.);
- *эвристический* - метод творческой деятельности (создание творческих проектов и т.д.);
- *проблемный* - постановка проблемы и поиск её решения учащимися;
- *программированный* - набор операций, которые необходимо выполнить в ходе выполнения практических работ (формы: компьютерный практикум, проектная деятельность);
- *репродуктивный* - воспроизводство знаний и способов деятельности (формы: верстка страниц по образцу, беседа, упражнения по аналогу);
- *частично-поисковый* - решение проблемных задач с помощью педагога;
- *поисковый* - самостоятельное решение проблем;

В реализации программы используются педагогические технологии, ориентированные на формирование компетенций учащихся:

- информационные технологии;
- компьютерные технологии;
- личностно- ориентированная технология;
- технология компетентного и деятельностного подхода;
- педагогика сотрудничества;
- здоровьесберегающие технологии;
- технология ТРИЗ.

4. СПИСОК ЛИТЕРАТУРЫ

4.1. Список литературы, используемой педагогом

Основная:

1. Уитни Д. Программирование для детей. Учимся создавать сайты, приложения и игры. HTML, CSS и JavaScript, Питер; Санкт-Петербург; 2020/ Уитни Дэвид
2. Дронов В.А. JavaScript. 20 уроков для начинающих, BHV, ISBN: 978-5-9775-6589-9, год издания: 2021/ В.А. Дронов
3. Брайсон Пэйн: Python для детей и родителей. 2-е издание, 2021
4. Бриггс, Джейсон Python для детей. Самоучитель по программированию / Джейсон Бриггс ; пер. с англ. Станислава Ломакина ; [науч. ред. Д. Абрамова]. — М. : Манн, Иванов и Фербер, 2017. — 320 с.

Интернет источники(дополнительная):

1. <https://htmlacademy.ru/>
2. <http://uroki-html.ru/>
3. https://www.youtube.com/playlist?list=PL0lO_mlqDDFUpe6yMyXAlcrfT6AO0KW1a

4.2. Список рекомендуемой литературы для обучающихся

Основная:

1. Уитни Д. Учимся создавать игры на HTML, CSS и JavaScript, Питер; Санкт-Петербург; 2020/ Уитни Дэвид
2. JavaScript для детей. Самоучитель по программированию / Ник Морган ; пер. с англ. Станислава Ломакина ; [науч. ред. Д. Абрамова]. — М. : Манн, Иванов и Фербер, 2016. — 288 с.
3. Брайсон Пэйн: Python для детей и родителей. 2-е издание, 2021
4. Бриггс, Джейсон Python для детей. Самоучитель по программированию / Джейсон Бриггс ; пер. с англ. Станислава Ломакина ; [науч. ред. Д. Абрамова]. — М. : Манн, Иванов и Фербер, 2017. — 320 с.

Интернет-источники (дополнительная):

5. <http://old.code.mu/books/css>
6. <https://html5css.ru/html/default.php>