

Индивидуальный предприниматель

 М.А. Иванова

«12» января 2026 г.

**Дополнительная общеобразовательная общеразвивающая
программа «Основы программирования в среде Scratch»**

Программа реализуется исключительно с применением электронного обучения и дистанционных образовательных технологий.

Направленность: техническая. Возраст обучающихся: от 8 лет.

Срок реализации: 28 академических часов в течение 14 недель

Автор программы: Иванова Марина Александровна

Содержание

1. Основные характеристики образовательной программы.....	3
2. Учебный план.....	5
3. Календарный учебный график	6
4. Рабочая программа.....	7
5. Оценка качества освоения программы.....	32
6. Условия реализации образовательной программы	32
7. Фонд оценочных материалов	33
8. Электронная библиотека	35

1. Основные характеристики образовательной программы

1.1. Актуальность программы

В условиях цифровизации общества и развития информационных технологий программирование становится базовым навыком XXI века. Раннее знакомство детей с основами алгоритмического мышления и логики через визуальное программирование создаёт прочный фундамент для дальнейшего технического образования и профессионального развития.

Scratch — визуальная среда программирования, разработанная Массачусетским технологическим институтом специально для обучения детей. Благодаря интуитивному блочному интерфейсу, яркой графике и игровому формату работы Scratch делает программирование доступным и увлекательным для детей от 8 лет, позволяя создавать анимации, игры и интерактивные проекты без необходимости печатать код.

Данная программа отвечает на запрос современного образования в развитии у детей вычислительного мышления, творческих способностей, навыков решения задач и проектной деятельности. Освоение Scratch формирует базовые программистские компетенции и готовит обучающихся к изучению более сложных языков программирования.

1.2. Направленность и уровень освоения программы

Программа имеет техническую направленность и ориентирована на развитие алгоритмического и логического мышления, формирование навыков программирования и цифрового творчества.

Уровень освоения — базовый.

1.3. Новизна программы

Новизна программы заключается в системном подходе к обучению основам программирования через проектную деятельность и творческое самовыражение. Программа интегрирует:

- поэтапное освоение ключевых концепций программирования от простого к сложному;
- практико-ориентированный подход с созданием законченных проектов на каждом этапе;
- развитие алгоритмического мышления через решение практических задач;
- освоение работы с координатами, переменными, событиями и объектно-ориентированными концепциями в доступной форме;
- развитие творческих способностей через создание собственных игр и анимаций;
- подготовку к публичной презентации результатов своей работы.

Реализация программы в электронном формате с применением дистанционных технологий позволяет обучающимся осваивать материал в комфортном темпе,

неоднократно возвращаться к записям занятий и получать индивидуальную поддержку преподавателя.

1.4. Обучающиеся

Программа рассчитана на обучающихся в возрасте от 8 лет без требований к предварительной подготовке в области программирования.

1.5. Объем и срок реализации программы

- Срок реализации: 14 недель
- Общий объем программы: 28 академических часов (академический час равен 45 астрономическим минутам)
- Режим занятий: 1 онлайн-занятие в неделю продолжительностью 2 академических часа (90 минут, с 10-минутным перерывом после первого часа) в формате прямого эфира с преподавателем. На занятии преподаватель демонстрирует каждое действие, обучающиеся сразу повторяют за ним на своих компьютерах, создавая проект синхронно с преподавателем.

1.6. Цели и задачи программы

Цель программы: развитие алгоритмического мышления и творческих способностей обучающихся путём освоения основ программирования через создание анимированных историй, игр и интерактивных проектов в визуальной среде Scratch.

Задачи программы:

Обучающие:

- сформировать представление об основных понятиях программирования (алгоритм, последовательность, цикл, условие, переменная, событие, функция);
- обучить работе в среде Scratch: навигация по интерфейсу, работа со спрайтами, костюмами, фонами и звуками;
- сформировать навыки работы с системой координат и управления движением объектов;
- обучить использованию основных алгоритмических конструкций: линейный алгоритм, ветвление, циклы различных типов;
- сформировать понимание работы с переменными, списками и сообщениями;
- обучить принципам создания и использования клонов объектов;
- сформировать навыки создания пользовательских блоков (функций) для структурирования кода;
- обучить основам проектной деятельности: от замысла до реализации и презентации.

Развивающие:

- развить алгоритмическое и логическое мышление;
- развить пространственное мышление через работу с координатной плоскостью;
- развить творческое мышление и способность к генерации идей;

- развить навыки отладки программ и поиска ошибок;
- развить навыки самостоятельного обучения и работы с технической документацией.

Воспитательные:

- воспитать интерес к техническому творчеству и программированию;
- сформировать ответственное отношение к результатам своей работы;
- сформировать культуру цифрового творчества и уважение к интеллектуальной собственности.

1.7. Планируемые результаты обучения

По завершении обучения по программе обучающиеся будут:

Знать:

- основные понятия программирования: алгоритм, последовательность команд, цикл, условие, переменная, событие, клон, функция;
- интерфейс и основные возможности среды Scratch;
- принципы работы с системой координат на плоскости;
- типы циклов и условных операторов;
- назначение переменных и способы работы с ними;
- принципы работы с сообщениями для взаимодействия объектов;
- основы создания и управления клонами;
- правила создания пользовательских блоков (функций);
- этапы разработки проекта от идеи до презентации.

Уметь:

- создавать, сохранять и открывать проекты в Scratch;
- добавлять, удалять и редактировать спрайты, костюмы и фоны;
- программировать движение спрайтов с использованием координат;
- использовать различные типы циклов для повторяющихся действий;
- применять условные операторы для создания интерактивности;
- создавать и использовать переменные для хранения данных;
- работать с сообщениями для синхронизации действий разных объектов;
- создавать клоны и управлять ими;
- создавать собственные блоки (функции) с параметрами;
- находить и исправлять ошибки в программах;
- создавать законченные интерактивные проекты (игры, анимации, открытки).

Владеть:

- навыками алгоритмического мышления;
- навыками работы в визуальной среде программирования Scratch;
- базовыми приёмами программирования: последовательность, циклы, условия, переменные, события.

1.8. Итоговый документ

Обучающемуся, успешно прошедшему итоговую аттестацию (освоившему программу), выдаётся сертификат о прохождении обучения по дополнительной общеобразовательной общеразвивающей программе. Обучающемуся, не прошедшему итоговую аттестацию или освоившему часть программы, по заявлению выдаётся справка об обучении.

2. Учебный план

№ п/п	Наименование разделов и тем занятий	Количество ак. часов		Форма контроля
		Всего	Практика	
1.	Знакомство со средой Scratch. Создание первого проекта	2	2	практическое задание
2.	Работа со звуком	2	2	практическое задание
3.	Движение спрайтов и внешний вид	2	2	практическое задание
4.	Координаты и движение в Scratch	2	2	практическое задание
5.	Циклы в Scratch	2	2	практическое задание
6.	Циклы и ожидание. Создание анимаций	2	2	практическое задание
7.	Условия и сенсоры	2	2	практическое задание
8.	Ожидание и цикл с условием	2	2	практическое задание
9.	Переменные. Создание таймера для игры	2	2	практическое задание
10.	Клоны и события	2	2	практическое задание
11.	Создание собственного блока. Повторяющиеся коды	2	2	практическое задание
12.	Создание собственных блоков с параметрами	2	2	практическое задание
13.	Подготовка проекта. Создание интерактивной открытки	2	2	практическое задание
14.	Итоговый контроль	2	2	защита проекта
	Итого часов	28	28	

3. Календарный учебный график

№	Наименование разделов и тем	Неделя и день обучения
1.	Знакомство со средой Scratch. Создание первого проекта	1 неделя, суббота
2.	Работа со звуком	2 неделя, суббота
3.	Движение спрайтов и внешний вид	3 неделя, суббота
4.	Координаты и движение в Scratch	4 неделя, суббота
5.	Циклы в Scratch	5 неделя, суббота
6.	Циклы и ожидание. Создание анимаций	6 неделя, суббота
7.	Условия и сенсоры	7 неделя, суббота
8.	Ожидание и цикл с условием	8 неделя, суббота
9.	Переменные. Создание таймера для игры	9 неделя, суббота
10.	Клоны и события	10 неделя, суббота

№	Наименование разделов и тем	Неделя и день обучения
11.	Создание собственного блока. Повторяющиеся коды	11 неделя, суббота
12.	Создание собственных блоков с параметрами	12 неделя, суббота
13.	Подготовка проекта. Создание интерактивной открытки	13 неделя, суббота
14.	Итоговый контроль	14 неделя, суббота

Конкретное расписание составляется для каждой учебной группы по мере набора учебных групп. Численность учебных групп — 5–8 человек, осваивающих образовательную программу одновременно.

Занятия проводятся по субботам в 12:00 МСК.

4. Рабочая программа

УРОК 1. ЗНАКОМСТВО СО СРЕДОЙ SCRATCH. СОЗДАНИЕ ПЕРВОГО ПРОЕКТА

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Введение в программирование и Scratch (5 минут)

Преподаватель рассказывает:

- Что такое программирование и зачем оно нужно детям
- Что такое Scratch и какие проекты можно создавать

2. Запуск Scratch и настройка интерфейса (5 минут)

Преподаватель показывает → ученики повторяют:

- Открыть браузер
- Перейти на сайт scratch.mit.edu
- Нажать кнопку «Создавать»
- Переключить язык интерфейса (глобус → русский)

3. Знакомство с интерфейсом Scratch (10 минут)

Преподаватель показывает каждый элемент → ученики находят его на своём экране:

- Палитра блоков команд (слева)
- Область скриптов (центр)
- Сцена (справа вверху)
- Список спрайтов и фонов (справа внизу)
- Спрайт кота (по умолчанию)

4. Создание первой программы «Привет, мир!» (15 минут)

Преподаватель делает каждый шаг → ученики повторяют:

- Открыть вкладку «События»
- Перетащить блок «когда флажок нажат» в область скриптов
- Открыть вкладку «Внешний вид»
- Присоединить блок «говорить ... 2 секунды» к блоку «когда флажок нажат»
- Изменить текст на «Привет!»
- Нажать зелёный флажок — проверить работу программы
- Все вместе проверяют результат

Перерыв 10 минут

5. Система координат в Scratch (10 минут)

Преподаватель объясняет и показывает:

- Центр сцены (0, 0)

- Направление осей X и Y
- Где находятся положительные и отрицательные значения

Преподаватель делает → ученики повторяют:

- Открыть вкладку «Движение»
- Перетащить блок «перейти в x: 0 y: 0»
- Изменить координаты на x: 100 y: 100
- Нажать на блок — посмотреть, куда переместился кот
- Попробовать разные координаты

6. Управление движением с клавиатуры (30 минут)

Преподаватель делает пошагово → ученики повторяют каждый шаг:

Шаг 1: Движение влево

- Открыть вкладку «События»
- Перетащить блок «когда клавиша пробел нажата»
- В выпадающем списке выбрать «стрелка влево»
- Открыть вкладку «Движение»
- Присоединить блок «изменить x на 10»
- Изменить значение на -10
- Проверить — нажать стрелку влево

Шаг 2: Движение вправо

- Создать новый скрипт «когда клавиша стрелка вправо нажата»
- Присоединить блок «изменить x на 10»
- Проверить – нажать стрелку вправо

Шаг 3: Добавление реплики

- Создать скрипт «когда клавиша пробел нажата»
- Присоединить блок «говорить ... 1 секунду»
- Изменить текст на «Мяу!»
- Проверить

7. Свободная практика под наблюдением (5 минут)

- Ученики экспериментируют со своими программами
- Преподаватель отвечает на вопросы
- Помогает тем, кто отстал или столкнулся с проблемами

8. Подведение итогов (5 минут)

- Преподаватель спрашивает, что узнали на уроке
- Напоминает основные понятия
- Анонсирует тему следующего урока

Результат урока:

- Все обучающиеся создали свою первую программу в Scratch
- Освоили базовый интерфейс

- Научились управлять спрайтом с клавиатуры

УРОК 2. РАБОТА СО ЗВУКОМ

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Введение в работу со звуками (5 минут)

Преподаватель рассказывает:

- Зачем нужны звуки в проектах Scratch
- Какие возможности есть для работы со звуком

2. Знакомство со вкладкой «Звуки» (10 минут)

Преподаватель показывает → ученики повторяют:

- Нажать на вкладку «Звуки» (вверху в центре)
- Увидеть звук «Meow», который есть у кота по умолчанию
- Нажать на кнопку воспроизведения – послушать звук

3. Добавление звука из библиотеки (10 минут)

Преподаватель делает → ученики повторяют:

- Нажать на кнопку «Выбрать звук» (значок динамика внизу слева)
- Найти звук «Dog1» в категории «Животные»
- Нажать на него – звук добавлен
- Вернуться на вкладку «Код»

4. Блоки воспроизведения звука (15 минут)

Преподаватель объясняет и показывает → ученики повторяют:

Шаг 1: Звук при запуске программы

- Открыть вкладку «События»
- Перетащить блок «когда флажок нажат»
- Открыть вкладку «Звук»
- Присоединить блок «играть звук ... до конца»
- В выпадающем списке выбрать «Meow»
- Нажать флажок – проверить

Шаг 2: Звук по нажатию клавиши

- Создать новый скрипт «когда клавиша M нажата»
- Присоединить блок «играть звук Dog1 до конца»
- Проверить – нажать клавишу M

Перерыв 10 минут

5. Запись собственного звука (15 минут)

Преподаватель показывает → ученики повторяют:

- Перейти на вкладку «Звуки»

- Нажать кнопку «Записать» (значок микрофона)
- Разрешить доступ к микрофону (если спросит браузер)
- Нажать красную кнопку записи
- Сказать в микрофон своё имя или «Привет!»
- Нажать стоп
- Прослушать запись
- Если не понравилось – удалить и записать заново
- Вернуться на вкладку «Код»

6. Использование записанного звука (10 минут)

Преподаватель делает → ученики повторяют:

- Создать скрипт «когда клавиша пробел нажата»
- Присоединить блок «играть звук ... до конца»
- Выбрать свой записанный звук (обычно называется «recording1»)
- Проверить – нажать пробел

7. Остановка звуков (5 минут)

Преподаватель показывает → ученики повторяют:

- Создать скрипт «когда клавиша S нажата»
- Присоединить блок «остановить все звуки»
- Проверить: нажать M (начнёт играть), быстро нажать S (остановится)

8. Управление громкостью (опционально, если осталось время) (10 минут)

Преподаватель показывает → ученики повторяют:

- В любом скрипте со звуком перед блоком воспроизведения
- Добавить блок «установить громкость на 50%»
- Попробовать разные значения: 100%, 30%, 10%
- Послушать разницу

9. Подведение итогов (5 минут)

- Преподаватель спрашивает, что узнали на уроке
- Обсуждает, в каких проектах пригодятся звуки

Результат урока:

- Все обучающиеся научились добавлять звуки из библиотеки
- Записали собственный звук
- Научились управлять воспроизведением звуков

УРОК 3. ДВИЖЕНИЕ СПРАЙТОВ И ВНЕШНИЙ ВИД

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Введение: что такое спрайт (5 минут)

Преподаватель рассказывает:

- Что такое спрайт – программируемый объект в Scratch
- Чем спрайт отличается от фона

2. Добавление спрайтов (10 минут)

Преподаватель показывает → ученики повторяют:

- Нажать на значок «Выбрать спрайт» (кот с плюсом внизу справа)
- Найти спрайт «Ballerina» в категории «Люди»
- Нажать на него – спрайт добавлен
- Повторить – добавить спрайт «Dinosaur1»
- Увидеть оба новых спрайта на сцене

3. Знакомство с костюмами (15 минут)

Преподаватель показывает → ученики повторяют:

- Выбрать спрайт балерины (кликнуть на неё внизу)
- Перейти на вкладку «Костюмы»
- Увидеть 4 костюма балерины
- Нажимать на разные костюмы – смотреть, как меняется внешний вид
- Вернуться на вкладку «Код»

Преподаватель делает → ученики повторяют:

- Для балерины: «когда клавиша пробел нажата»
- Присоединить блок «следующий костюм» (вкладка «Внешний вид»)
- Проверить – нажимать пробел, балерина меняет позу

4. Блоки управления внешним видом (15 минут)

Преподаватель показывает → ученики повторяют каждый блок:

Шаг 1: Скрыть и показать

- Для динозавра: «когда клавиша H нажата»
- Присоединить блок «скрыть»
- Проверить – динозавр исчезает
- Создать скрипт: «когда клавиша S нажата» → «показать»
- Проверить

Шаг 2: Размер

- Для балерины: «когда клавиша + нажата»
- Присоединить «изменить размер на 10»
- Создать: «когда клавиша - нажата» → «изменить размер на -10»
- Проверить

Перерыв 10 минут

5. Работа с фонами (10 минут)

Преподаватель показывает → ученики повторяют:

- Навести на значок «Выбрать фон» (в правом нижнем углу)
- Нажать на него
- Выбрать фон «Party»
- Выбрать ещё фон «Bedroom 1»
- Перейти на вкладку «Код» для Сцены (внизу справа)
- Создать скрипт: «когда клавиша стрелка вправо нажата» → «следующий фон»
- Проверить – фоны переключаются

6. Создание комплексного проекта (20 минут)

Преподаватель делает пошагово → ученики повторяют:

Для балерины:

- «Когда флажок нажат»
- «Перейти в x: -100 y: -50»
- «Показать»
- Проверить

Для динозавра:

- «Когда флажок нажат»
- «Перейти в x: 100 y: -50»
- «Показать»
- Проверить

Для Сцены:

- «Когда флажок нажат»
- «Переключить фон на Party»
- Проверить

7. Свободная практика (5 минут)

- Ученики пробуют добавить графические эффекты
- Преподаватель подсказывает: вкладка «Внешний вид» → блок «изменить эффект цвет на 25»

8. Подведение итогов (5 минут)

- Преподаватель спрашивает: что такое костюм, как его переключить?
- Напоминает основные блоки внешнего вида

Результат урока:

- Все обучающиеся создали проект с несколькими спрайтами и фонами
- Научились переключать костюмы и фоны
- Освоили основные блоки внешнего вида

УРОК 4. КООРДИНАТЫ И ДВИЖЕНИЕ В SCRATCH

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Теория системы координат (10 минут)

Преподаватель объясняет с демонстрацией на экране:

- Сцена – это координатная плоскость
- Центр сцены – точка (0, 0)
- Ось X: влево -240, вправо +240
- Ось Y: вниз -180, вверх +180
- Показывает на экране, где находятся точки (100, 100), (-100, -100)

2. Эксперименты с координатами (15 минут)

Преподаватель показывает → ученики повторяют:

- Для кота: перетащить блок «перейти в x: 0 y: 0» из вкладки «Движение»
- Не присоединять к скрипту – просто кликнуть на блок
- Кот переместился в центр
- Изменить на x: 200 y: 150 → кликнуть → кот в правом верхнем углу
- Попробовать x: -200 y: -150 → левый нижний угол
- Попробовать x: 0 y: 150 → вверх по центру
- Каждый раз кликать на блок и смотреть, куда перемещается кот

3. Блоки точного перемещения (20 минут)

Преподаватель делает → ученики повторяют:

Программа 1: Мгновенное перемещение

- «Когда флажок нажат»
- «Перейти в x: -200 y: 100»
- «Ждать 1 секунду»
- «Перейти в x: 200 y: 100»
- «Ждать 1 секунду»
- «Перейти в x: 200 y: -100»
- «Ждать 1 секунду»
- «Перейти в x: -200 y: -100»
- Проверить – кот прыгает по углам
- Программа 2: Плавное перемещение*
- Создать новый скрипт: «когда клавиша пробел нажата»
- «Перейти в x: -150 y: 100»
- «Плыть 2 секунды в x: 150 y: 100»
- «Плыть 2 секунды в x: 150 y: -100»
- «Плыть 2 секунды в x: -150 y: -100»
- «Плыть 2 секунды в x: -150 y: 100»

- Проверить – кот плавно движется по квадрату

Перерыв 10 минут

4. Создание бесконечного движения (15 минут)

Преподаватель показывает → ученики повторяют:

- Удалить предыдущие скрипты (перетащить в палитру блоков)
- «Когда флажок нажат»
- «Перейти в x: -150 y: 100»
- Добавить блок «повторять всегда»
- Внутри цикла:
 - «Плыть 2 секунды в x: 150 y: 100»
 - «Плыть 2 секунды в x: 150 y: -100»
 - «Плыть 2 секунды в x: -150 y: -100»
 - «Плыть 2 секунды в x: -150 y: 100»
- Проверить – кот бесконечно движется по квадрату

5. Блок «если касается края, оттолкнуться» (15 минут)

Преподаватель делает → ученики повторяют:

- Создать новый скрипт: «когда флажок нажат»
- «Перейти в x: 0 y: 0»
- «Повернуться в направление 90» (смотрит вправо)
- Добавить «повторять всегда»
- Внутри цикла:
 - «Идти 10 шагов»
 - «Если касается края, оттолкнуться»
- Проверить – кот ходит от края до края
- Заметить: кот переворачивается вверх ногами
- Исправить: добавить перед циклом блок «установить способ вращения влево-вправо»
- Проверить снова

6. Свободная практика (5 минут)

- Ученики экспериментируют с координатами
- Пробуют изменить траекторию движения

7. Подведение итогов (5 минут)

- Преподаватель спрашивает: где находится центр сцены? Какой блок для плавного движения?

Результат урока:

- Все обучающиеся понимают систему координат
- Умеют использовать блоки точного перемещения
- Создали программы с различными типами движения

УРОК 5. ЦИКЛЫ В SCRATCH

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Введение в циклы (10 минут)

Преподаватель объясняет:

- Цикл – это повторение действий
- Зачем нужны циклы: чтобы не копировать одинаковые блоки много раз
- Показывает пример: 10 раз написать блок «идти 10 шагов» или один раз цикл «повторить 10 раз»

2. Бесконечный цикл «Повторять всегда» (20 минут)

Преподаватель делает → ученики повторяют:

Шаг 1: Простое движение

- Удалить все предыдущие скрипты у кота
- «Когда флажок нажат»
- Добавить блок «повторять всегда» (вкладка «Управление»)
- Внутри цикла: «идти 10 шагов»
- Проверить – кот уходит за край

Шаг 2: Отталкивание от края

- Внутри цикла добавить: «если касается края, оттолкнуться»
- Проверить – кот ходит туда-сюда
- Если переворачивается – добавить перед циклом «установить способ вращения влево-вправо»

Шаг 3: Изменение скорости

- Изменить «идти 10 шагов» на «идти 5 шагов» → медленнее
- Попробовать «идти 15 шагов» → быстрее
- Вернуть 10

3. Добавление анимации в цикл (15 минут)

Преподаватель показывает → ученики повторяют:

- Убедиться, что у кота есть 2 костюма (вкладка «Костюмы»)
- Вернуться на «Код»
- Внутри цикла (после блока отталкивания):
 - Добавить «следующий костюм»
 - Добавить «ждать 0,2 секунды»
- Проверить – кот идёт с анимацией
- Попробовать изменить время ожидания: 0,1 с – быстрее, 0,3 с – медленнее

Перерыв 10 минут

4. Ограниченный цикл «Повторить N раз» (15 минут)

Преподаватель объясняет:

- Этот цикл выполняется не всегда, а определённое количество раз
- Полезен, когда нужно что-то сделать несколько раз и остановиться

Преподаватель делает → ученики повторяют:

- Создать новый скрипт: «когда клавиша пробел нажата»
- «Перейти в x: 0 y: 0»
- Добавить блок «повторить 10 раз»
- Внутри цикла: «идти 20 шагов»
- Проверить – кот делает 10 шагов и останавливается

5. Практический пример: полный оборот (15 минут)

Преподаватель делает → ученики повторяют:

- Создать скрипт: «когда клавиша стрелка вверх нажата»
- «Повторить 36 раз»
- Внутри цикла:
 - «Повернуть направо на 10 градусов»
 - «Изменить эффект цвет на 10»
 - «Ждать 0,05 секунды»
- Проверить – кот делает полный оборот (360°), меняя цвет
- Объяснить математику: $36 \times 10 = 360$ градусов

6. Создание итогового проекта (10 минут)

Преподаватель помогает объединить всё:

- Бесконечный цикл при флаге – ходьба с анимацией
- Пробел – 10 шагов вперёд
- Стрелка вверх – полный оборот

7. Подведение итогов (5 минут)

- Чем отличается «повторять всегда» от «повторить N раз»?
- Зачем нужен блок «ждать» в цикле?

Результат урока:

- Все обучающиеся понимают концепцию циклов
- Умеют использовать оба типа циклов
- Создали проект с анимацией через циклы

УРОК 6. ЦИКЛЫ И ОЖИДАНИЕ. СОЗДАНИЕ АНИМАЦИЙ

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Важность блока «Ждать» (8 минут)

Преподаватель объясняет:

- Блок «ждать» создаёт паузу в программе
- Без него анимация работает слишком быстро и мы её не видим
- Показывает пример: смена 10 костюмов без ожидания — мгновенно

2. Эксперименты с ожиданием (15 минут)

Преподаватель делает → ученики повторяют:

Эксперимент 1: Слишком быстро

- Удалить предыдущие скрипты у кота
 - «Когда флажок нажат»
 - «Повторить 10 раз»
 - Внутрь: «следующий костюм»
 - Проверить – почти ничего не видно
- Эксперимент 2: Оптимальная скорость
- Добавить внутрь цикла после «следующий костюм»: «ждать 0.2 секунды»
 - Проверить – видна анимация
- Эксперимент 3: Разные скорости
- Попробовать 0,1 с – быстро
 - Попробовать 0,5 с – медленно
 - Попробовать 1 с – очень медленно
 - Вернуться к 0,2 с

3. Выбор подходящего спрайта для анимации (12 минут)

Преподаватель показывает → ученики повторяют:

- Удалить кота (правой кнопкой → удалить)
- Добавить спрайт «Parrot» (категория «Животные»)
- Кликнуть на попугая → вкладка «Костюмы»
- Посмотреть – у него 2 костюма
- Вернуться на «Код»

4. Создание летящего попугая (20 минут)

Преподаватель делает пошагово → ученики повторяют каждый шаг:

- «Когда флажок нажат»
- «Перейти в x: -200 y: 0»
- «Повернуться в направление 90» (вправо)
- «Повторять всегда»
- Внутрь цикла:
 - «Идти 5 шагов»
 - «Следующий костюм»
 - «Ждать 0,1 секунды»
 - «Если касается края, оттолкнуться»

- Добавить перед циклом: «установить способ вращения влево-вправо»
- Проверить – попугай летит

Перерыв 10 минут

5. Улучшение анимации – волнообразное движение (20 минут)

Преподаватель показывает → ученики повторяют:

- Создать переменную «направление_Y» (вкладка «Переменные» → «Создать переменную»)
- В скрипте после «когда флажок нажат», перед циклом:
 - «Установить направление_Y в 5»
- Внутри цикла (после «идти 5 шагов»):
 - «Изменить у на направление_Y»
- После цикла добавить ещё один цикл:
 - «Повторять всегда»
 - Внутри: «если < у > 100 >», то:
 - «Установить направление_Y в -5»
 - После блока «если» добавить ещё «если < у < -100 >», то:
 - «Установить направление_Y в 5»
- Проверить – попугай летит волнами

6. Упрощённый вариант (если предыдущий сложный) (10 минут)

Преподаватель предлагает более простой проект:

- Оставить только основной цикл с движением и анимацией
- Добавить смену цвета: внутри цикла «изменить эффект цвет на 5»
- Каждый выбирает, какой вариант ему больше нравится

7. Подведение итогов (5 минут)

- Какое время ожидания подходит для плавной анимации?
- Что происходит без блока «ждать»?

Результат урока:

- Все обучающиеся понимают роль блока «ждать»
- Создали плавную анимацию движения
- Научились подбирать оптимальное время ожидания

УРОК 7. УСЛОВИЯ И СЕНСОРЫ

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Введение в условия (10 минут)

Преподаватель объясняет:

- Условие – это проверка: правда или ложь

- «Если условие правда, то выполнить действие»
- Пример из жизни: «Если идёт дождь, то взять зонт»
- В Scratch: «Если кот касается мыши, то изменить цвет»

2. Блок «Если, то» (15 минут)

Преподаватель показывает → ученики повторяют:

- Удалить предыдущие скрипты
- Добавить спрайт кота (если его нет)
- «Когда флажок нажат»
- «Повторять всегда»
- Внутри цикла: блок «если ... то» (вкладка «Управление»)
- В шестиугольное отверстие: из вкладки «Сенсоры» блок «касается указателя мыши?»
- Внутри блока «если»: «изменить эффект цвет на 25»
- Проверить – при наведении мыши кот меняет цвет

3. Блок «Если, то, иначе» (15 минут)

Преподаватель делает → ученики повторяют:

- Создать новый скрипт: «когда флажок нажат»
- «Повторять всегда»
- Внутри: блок «если ... то, иначе»
- В условие: «касается указателя мыши?»
- В часть «то»: «установить размер 150%»
- В часть «иначе»: «установить размер 100%»
- Проверить – кот увеличивается при наведении и уменьшается обратно

Перерыв 10 минут

4. Сенсоры клавиатуры (10 минут)

Преподаватель показывает → ученики повторяют:

- Создать скрипт: «когда флажок нажат»
- «Повторять всегда»
- «Если ... то»
- В условие: «клавиша пробел нажата?» (вкладка «Сенсоры»)
- Внутри «если»: «играть звук Meow до конца»
- Проверить

5. Цветовые сенсоры – подготовка (15 минут)

Преподаватель делает → ученики повторяют:

Шаг 1: Рисуем цветную линию

- Выбрать Сцену (внизу справа)
- Вкладка «Фоны»
- Выбрать инструмент «Линия»

- Выбрать красный цвет
- Увеличить толщину линии
- Нарисовать горизонтальную линию внизу сцены

Шаг 2: Добавляем управление

- Вернуться к коту → вкладка «Код»
- Создать скрипт: «когда флажок нажат»
- «Перейти в х: 0 у: 50»
- «Повторять всегда»
- Внутри цикла:
 - «Если клавиша стрелка вправо нажата», то: «изменить х на 10»
 - «Если клавиша стрелка влево нажата», то: «изменить х на -10»
 - «Если клавиша стрелка вниз нажата», то: «изменить у на -10»
 - «Если клавиша стрелка вверх нажата», то: «изменить у на 10»
- Проверить – кот управляется стрелками

6. Использование цветового сенсора (20 минут)

Преподаватель показывает → ученики повторяют:

- Внутри основного цикла добавить блок «если ... то»
- В условие: «касается цвета ...?» (вкладка «Сенсоры»)
- Кликнуть на цветной квадратик в блоке
- Появится пипетка – кликнуть на красную линию на сцене
- Цвет в блоке стал красным
- Внутри «если»:
 - «Говорить Стоп! 1 секунду»
 - «Перейти в х: 0 у: 50»
- Проверить – при касании красной линии кот говорит «Стоп!» и возвращается

7. Подведение итогов (5 минут)

- Что делает блок «если»?
- Как работает цветовой сенсор?

Результат урока:

- Все обучающиеся понимают условные операторы
- Умеют использовать различные сенсоры
- Создали интерактивную программу с условиями

УРОК 8. ОЖИДАНИЕ И ЦИКЛ С УСЛОВИЕМ

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Блок «Ждать до» (10 минут)

Преподаватель объясняет:

- Блок «ждать» – пауза на определённое время
- Блок «ждать до» – пауза, пока условие не станет правдой
- Пример: «ждать до <касается указателя мыши?>»

Преподаватель показывает → ученики повторяют:

- «Когда флажок нажат»
- «Перейти в x: 0 y: 0»
- «Ждать до <касается указателя мыши?>» (вкладка «Управление»)
- «Говорить Ты коснулся меня! 2 секунды»
- Проверить — программа ждёт касания

2. Цикл «Повторять, пока не» (15 минут)

Преподаватель объясняет:

- Это цикл, который выполняется, пока условие НЕ стало правдой
- Как только условие стало правдой – цикл останавливается

Преподаватель делает → ученики повторяют:

- Создать новый скрипт: «когда клавиша пробел нажата»
- «Перейти в x: 0 y: 0»
- «Повторять, пока не <касается указателя мыши?>»
- Внутри цикла:
 - «Повернуть направо на 15 градусов»
 - «Ждать 0.1 секунды»
- После цикла: «говорить Остановлено! 2 секунды»
- Проверить – кот вращается, пока не коснется мышью

3. Логические операторы (15 минут)

Преподаватель объясняет:

- Оператор «И» – оба условия должны быть правдой
- Оператор «ИЛИ» – хотя бы одно условие правда
- Оператор «НЕ» – инвертирует условие

Преподаватель показывает → ученики повторяют:

- Открыть вкладку «Операторы»
- Найти зелёные блоки: «... и ...», «... или ...», «не ...»
- Создать условие: «< <касается указателя мыши?> или <клавиша пробел нажата?> >»
- Вставить это в предыдущий скрипт вместо простого условия
- Проверить – останавливается при касании ИЛИ при пробеле

Перерыв 10 минут

4. Создание фона с цветными областями (10 минут)

Преподаватель делает → ученики повторяют:

- Выбрать Сцену → вкладка «Фоны»
- Создать новый фон (кнопка рисования)
- Инструмент «Круг» (с заливкой)
- Нарисовать большой синий круг справа
- Нарисовать большой красный круг слева
- Вернуться к коту

5. Комплексная программа с условиями (25 минут)

Преподаватель делает пошагово → ученики повторяют каждый шаг:

Программа 1: Вращение до касания цвета

- «Когда флажок нажат»
- «Перейти в x: 0 y: 0»
- «Ждать до <касается указателя мыши?>»
- «Повторять, пока не <касается цвета [синий]?> или <касается цвета [красный]?>»
- Внутри цикла:
 - «Повернуть направо на 10 градусов»
 - «Ждать 0.05 секунды»
- «Говорить Коснулся цвета! 2 секунды»
- Использовать пипетку для выбора синего и красного цветов
- Проверить Программа 2: Движение до касания
- «Когда клавиша стрелка вверх нажата»
- «Повернуться в направление 0» (вверх)
- «Повторять, пока не <касается цвета [синий]?>»
- Внутри: «идти 5 шагов»
- «Говорить Достиг синего! 2 секунды»
- Проверить

6. Свободная практика (5 минут)

- Ученики экспериментируют с разными комбинациями условий

7. Подведение итогов (5 минут)

- Чем отличается «ждать» от «ждать до»?
- Когда использовать «и», а когда «или»?

Результат урока:

- Понимание блока «ждать до»
- Умение использовать цикл «повторять, пока не»
- Навык комбинирования условий

УРОК 9. ПЕРЕМЕННЫЕ. СОЗДАНИЕ ТАЙМЕРА ДЛЯ ИГРЫ

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Что такое переменная (10 минут)

Преподаватель объясняет:

- Переменная – это «коробочка» для хранения информации
- Примеры: счёт очков, количество жизней, время
- Переменная может изменяться во время программы
- Показывает аналогию: переменная «возраст» — сейчас 8, через год будет 9

2. Создание первой переменной (10 минут)

Преподаватель показывает → ученики повторяют:

- Открыть вкладку «Переменные»
- Нажать «Создать переменную»
- Написать имя: «счёт»
- Выбрать «Для всех спрайтов»
- Нажать «ОК»
- Переменная появилась на сцене (показывает значение)
- Увидеть новые блоки во вкладке «Переменные»

3. Блоки работы с переменными (15 минут)

Преподаватель делает → ученики повторяют каждый блок:

Блок «Установить»:

- Перетащить блок «установить счёт в 0»
- Кликнуть на него – на сцене счёт стал 0
- Изменить на «установить счёт в 100»
- Кликнуть – счёт стал 100

Блок «Изменить»:

- Перетащить «изменить счёт на 1»
- Кликать несколько раз – счёт увеличивается на 1
- Попробовать «изменить счёт на -1» – счёт уменьшается

Блок «Показать/Скрыть»:

- «Скрыть переменную счёт» – исчезла с экрана
- «Показать переменную счёт» – появилась обратно

4. Создание игры со счётом (10 минут)

Преподаватель делает → ученики повторяют:

- «Когда флажок нажат»
- «Установить счёт в 0»
- «Показать переменную счёт»

- «Повторять всегда»
- Внутрь: «если <касается указателя мыши?>», то:
 - «Изменить счёт на 1»
 - «Ждать 0,5 секунды»
- Проверить – счёт увеличивается при касании

Перерыв 10 минут

5. Создание переменной «таймер» (5 минут)

Преподаватель показывает → ученики повторяют:

- Создать новую переменную с именем «таймер»
- Теперь на сцене две переменные

6. Программирование таймера обратного отсчёта (20 минут)

Преподаватель делает пошагово → ученики повторяют:

Скрипт 1: Таймер (отдельный скрипт)

- «Когда флажок нажат»
- «Установить таймер в 30»
- «Показать переменную таймер»
- «Повторять, пока не <таймер = 0>» (вкладка «Операторы» для знака «=»)
- Внутрь цикла:
 - «Изменить таймер на -1»
 - «Ждать 1 секунду»
- После цикла:
 - «Говорить Время вышло! 2 секунды»
 - «Остановить все»
- Проверить – таймер отсчитывает от 30 до 0

7. Создание фона с финишем (10 минут)

Преподаватель делает → ученики повторяют:

- Сцена → вкладка «Фоны»
- Создать новый фон
- Нарисовать большой зелёный круг справа вверху (финиш)
- Вернуться к коту

8. Программа управления с проверкой финиша (15 минут)

Преподаватель делает → ученики повторяют:

Скрипт 2: Управление (параллельный скрипт)

- «Когда флажок нажат»
- «Перейти в x: -200 y: -130»
- «Повторять всегда»
- Внутрь цикла:
 - «Если <клавиша стрелка вправо нажата?>», то: «изменить x на 10»

- «Если <клавиша стрелка влево нажата?>», то: «изменить x на -10»
- «Если <клавиша стрелка вверх нажата?>», то: «изменить y на 10»
- «Если <клавиша стрелка вниз нажата?>», то: «изменить y на -10»
- «Если <касается цвета [зелёный]?>», то:
- «Говорить Победа! 2 секунды»
- «Остановить все»

- Проверить – нужно дойти до зелёного за 30 секунд

9. Подведение итогов (5 минут)

- Что такое переменная?
- Как создать таймер?

Результат урока:

- Понимание концепции переменных
- Умение создавать и использовать переменные
- Создали игру с таймером

УРОК 10. КЛОНЫ И СОБЫТИЯ

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Что такое клоны (10 минут)

Преподаватель объясняет:

- Клон – это копия спрайта
- Зачем нужны: чтобы создать много одинаковых объектов
- Примеры: падающие снежинки, звёзды, враги в игре
- Показывает картинку снегопада – сотни снежинок, но это клоны одного спрайта

2. Блоки для работы с клонами (10 минут)

Преподаватель показывает блоки:

- Вкладка «Управление»
- «Создать клон себя» – создаёт копию
- «Когда я начинаю как клон» – что делает каждый клон
- «Удалить этот клон» – клон исчезает

3. Подготовка спрайта для клонирования (10 минут)

Преподаватель делает → ученики повторяют:

- Удалить кота
- Добавить спрайт «Snowflake» (снежинка)
- Изменить размер снежинки: блок «установить размер 30%»
- Кликнуть на него – снежинка стала меньше

4. Создание клонов в цикле (15 минут)

Преподаватель делает пошагово → ученики повторяют:

Скрипт для оригинального спрайта:

- «Когда флажок нажат»
- «Скрыть» (оригинал не должен быть виден)
- «Повторить 10 раз»
- Внутри цикла:
 - «Создать клон себя»
 - «Ждать 0,5 секунды»
- Пока не проверяем — клонов не видно, потому что нет скрипта для них

Перерыв 10 минут

5. Программирование поведения клонов (20 минут)

Преподаватель делает пошагово → ученики повторяют:

Скрипт для клонов:

- «Когда я начинаю как клон»
- «Перейти в х: [случайное число от -200 до 200] у: 180»
 - Для «случайное число»: вкладка «Операторы» → зелёный овал «выдать случайное от ... до ...»
 - Вложить его в х: изменить значения на -200 и 200
- «Показать»
- «Повторять, пока не $y < -180$ »
 - Для условия: вкладка «Операторы» → «... < ...»
 - В левую часть: вкладка «Движение» → синий овал «у»
 - В правую часть: -180
- Внутри цикла:
 - «Изменить у на -5»
 - «Ждать 0,05 секунды»
- После цикла: «Удалить этот клон»
- Проверить – снежинки падают!

6. Улучшение снегопада (10 минут)

Преподаватель показывает → ученики повторяют:

- В основном скрипте изменить «повторить 10 раз» на «повторять всегда»
- Изменить ожидание с 0,5 на 0,3 секунды
- Проверить - снежинки падают непрерывно

7. Знакомство с сообщениями (опционально, если осталось время) (10 минут)

Преподаватель объясняет:

- Сообщения – способ «связи» между спрайтами
- Один спрайт передаёт сообщение, другие его получают

Преподаватель показывает пример:

- Добавить кнопку (спрайт «Button1»)
- Для кнопки: «когда этот спрайт нажат» → «передать старт снегопада»
- Для снежинки: изменить «когда флажок нажат» на «когда я получу старт снегопада»
- Проверить – снегопад начинается по клику на кнопку

8. Подведение итогов (5 минут)

- Зачем нужны клоны?
- Какие блоки используются для клонов?

Результат урока:

- Понимание концепции клонов
- Умение создавать и удалять клоны
- Создали проект «Снегопад»

УРОК 11. СОЗДАНИЕ СОБСТВЕННОГО БЛОКА. ПОВТОРЯЮЩИЕСЯ КОДЫ

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Проблема повторяющегося кода (10 минут)

Преподаватель показывает пример плохого кода:

- Создаёт скрипт с управлением стрелками (4 блока «если» для каждой стрелки)
- Показывает, что если нужно добавить звук к каждому движению – придётся копировать 4 раза
- Объясняет: код сложно читать и изменять

2. Введение в собственные блоки (10 минут)

Преподаватель объясняет:

- Собственный блок – это группа команд под одним именем
- Как рецепт: вместо перечисления всех шагов говорим «испечь торт»
- Преимущества: код короче, понятнее, легче менять

3. Создание первого собственного блока (15 минут)

Преподаватель делает → ученики повторяют:

Шаг 1: Создать блок

- Открыть вкладку «Мои блоки»
- Нажать «Создать блок»
- Написать имя: «Управление»
- Нажать «ОК»
- Появился новый блок и блок «Определить Управление»

Шаг 2: Определить, что делает блок

- Под блок «Определить Управление» присоединить:
 - «Если <клавиша стрелка вправо нажата?>», то: «изменить x на 10»
 - «Если <клавиша стрелка влево нажата?>», то: «изменить x на -10»
 - «Если <клавиша стрелка вверх нажата?>», то: «изменить y на 10»
 - «Если <клавиша стрелка вниз нажата?>», то: «изменить y на -10»

Шаг 3: Использовать блок

- «Когда флажок нажат»
- «Перейти в x: 0 y: 0»
- «Повторять всегда»
- Внутри цикла: перетащить блок «Управление» (из вкладки «Мои блоки»)
- Проверить – работает!

4. Создание блока для анимации (15 минут)

Преподаватель делает → ученики повторяют:

- Создать новый блок с именем «Анимация»
- Под «Определить Анимация»:
 - «Следующий костюм»
 - «Ждать 0,2 секунды»
- В основной цикл добавить после блока «Управление»: блок «Анимация»
- Проверить – кот двигается с анимацией

Перерыв 10 минут

5. Сравнение кода до и после (5 минут)

Преподаватель показывает:

- Было: большой цикл с множеством блоков
- Стало: цикл с двумя простыми блоками – понятно и коротко

6. Создание блока с более сложной логикой (20 минут)

Преподаватель делает → ученики повторяют:

Создать блок «Проверить границы»:

- Создать блок с именем «Проверить границы»
- Под «Определить Проверить границы»:
 - «Если < x > 230 >», то: «установить x в 230»
 - «Если < x < -230 >», то: «установить x в -230»
 - «Если < y > 170 >», то: «установить y в 170»
 - «Если < y < -170 >», то: «установить y в -170»
- Добавить этот блок в основной цикл после «Управление»
- Проверить – кот не выходит за границы

7. Итоговая программа (10 минут)

Преподаватель помогает собрать всё вместе:

- «Когда флажок нажат»
- «Перейти в x: 0 y: 0»
- «Повторять всегда»
- Внутрь:
 - «Управление»
 - «Анимация»
 - «Проверить границы»
- Проверить – всё работает, код короткий и понятный

8. Подведение итогов (5 минут)

- Зачем создавать собственные блоки?
- Как создать блок?

Результат урока:

- Понимание необходимости структурирования кода
- Умение создавать собственные блоки
- Код стал короче и понятнее

УРОК 12. СОЗДАНИЕ СОБСТВЕННЫХ БЛОКОВ С ПАРАМЕТРАМИ

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Зачем нужны параметры (10 минут)

Преподаватель объясняет:

- Параметр – это «настройка» блока
- Пример: блок «идти ... шагов» – число шагов это параметр
- Без параметров: блок «идти быстро», блок «идти медленно» – нужно много блоков
- С параметром: один блок «идти [скорость]» – можем задать любую скорость

2. Создание блока с числовым параметром (20 минут)

Преподаватель делает пошагово → ученики повторяют:

Шаг 1: Создать блок

- Вкладка «Мои блоки» → «Создать блок»
- Написать имя: «Движение вправо»
- Нажать «Добавить параметр» → выбрать «число или текст»
- Появилось поле – написать: «скорость»
- Нажать «ОК»

- Появился блок «Движение вправо скорость» с овальным отверстием

Шаг 2: Определить блок

- Под «Определить Движение вправо скорость»:
 - «Если <клавиша стрелка вправо нажата?>», то:
 - «Изменить x на ...»
 - В отверстие перетащить овал «скорость» (он рядом с блоком «Определить»)
 - «Если <клавиша стрелка влево нажата?>», то:
 - «Изменить x на ...»
 - Вкладка «Операторы» → «... - ...»
 - Слева 0, справа овал «скорость»
 - Получится: изменить x на (0 - скорость)

Шаг 3: Использовать блок

- «Когда флажок нажат»
- «Перейти в x: 0 y: 0»
- «Повторять всегда»
- Внутрь: «Движение вправо [10]»
- Проверить – работает со скоростью 10
- Изменить на 5 – медленнее, на 20 – быстрее

Перерыв 10 минут

3. Создание блока «Повернуть на угол» (15 минут)

Преподаватель делает → ученики повторяют:

- Создать блок «Повернуть на угол»
- Добавить параметр «угол» (число или текст)
- Под «Определить Повернуть на угол»:
 - «Повернуть направо на ...» (вставить овал «угол»)
- Использовать:
 - «Когда клавиша R нажата» → «Повернуть на угол [90]»
 - «Когда клавиша T нажата» → «Повернуть на угол [180]»
- Проверить

4. Логический параметр (15 минут)

Преподаватель объясняет:

- Логический параметр – это «да» или «нет»
- Пример: включить или выключить звук

Преподаватель делает → ученики повторяют:

- Создать блок «Движение со звуком»
- Добавить параметр → выбрать «логический»
- Написать имя: «звук включён»
- Под «Определить Движение со звуком звук включён»:

- «Идти 10 шагов»
- «Если <звук включён>», то:
- «Играть звук Рор до конца»
- Использовать:
 - «Когда стрелка вверх нажата» → «Движение со звуком <истина>» (галочка)
 - «Когда стрелка вниз нажата» → «Движение со звуком <ложь>» (пусто)
- Проверить

5. Создание универсального блока движения (20 минут)

Преподаватель делает → ученики повторяют:

- Создать блок «Двигаться»
- Добавить параметры:
 - «скорость» (число)
 - «анимация» (логический)
- Под «Определить Двигаться»:
 - «Если <клавиша стрелка вправо нажата?>», то: «изменить x на скорость»
 - «Если <клавиша стрелка влево нажата?>», то: «изменить x на (0 - скорость)»
 - «Если <анимация>», то:
 - «Следующий костюм»
 - «Ждать 0.1 секунды»
- Использовать в цикле: «Двигаться [5] <истина>»
- Попробовать разные варианты: [10] <ложь>, [3] <истина>

6. Подведение итогов (5 минут)

- Что такое параметр?
- Какие бывают типы параметров?

Результат урока:

- Понимание параметров функций
- Умение создавать блоки с разными типами параметров
- Создали гибкие многоразовые блоки

УРОК 13. ПОДГОТОВКА ПРОЕКТА. СОЗДАНИЕ ИНТЕРАКТИВНОЙ ОТКРЫТКИ

Длительность: 2 академических часа (90 минут)

Формат: синхронная работа – преподаватель показывает, ученики сразу повторяют

ХОД ЗАНЯТИЯ:

1. Планирование проекта (10 минут)

Преподаватель объясняет концепцию:

- Интерактивная открытка – это открытка, которая «открывается» на экране
- Показывает пример готовой открытки
- Обсуждает элементы: анимация открытия, персонажи, музыка, кнопка запуска

2. Создание костюмов фона для анимации открытия (20 минут)

Преподаватель делает пошагово → ученики повторяют каждый шаг:

- Сцена → вкладка «Фоны»
- Создать новый фон (кнопка рисования)
- Выбрать инструмент «Прямоугольник» (с заливкой)
- Выбрать красивый цвет (например, синий)
- Нарисовать два прямоугольника – левая и правая створка закрытой открытки
- Инструментом «Текст» написать «С Днём Рождения!» посередине
- Это костюм 1 – закрытая открытка
- Кликнуть правой кнопкой на костюм → «Дублировать»
- Для костюма 2: выбрать инструмент «Выбрать»
- Выделить левый прямоугольник → немного сдвинуть влево
- Выделить правый прямоугольник → немного сдвинуть вправо
- Открытие началось
- Дублировать костюм 2 → получится костюм 3
- Сдвинуть створки ещё дальше
- Повторить ещё 2-3 раза → получится 5-6 костюмов
- Последний костюм – створки полностью разъехались
- Вернуться на вкладку «Код»

3. Программирование анимации открытия (15 минут)

Преподаватель делает → ученики повторяют:

Для Сцены:

- «Когда я получу открыть» (вкладка «События» → создать новое сообщение «открыть»)
- «Переключить фон на backdrop1» (первый костюм)
- «Повторить 5 раз» (или сколько костюмов создали)
- Внутри цикла:
 - «Следующий фон»
 - «Ждать 0,3 секунды»
- Пока не проверим – сообщение никто не отправляет

Перерыв 10 минут

4. Создание персонажа с репликами (15 минут)

Преподаватель делает → ученики повторяют:

- Добавить спрайт «Avery» (или любой другой персонаж)
- Уменьшить размер: «установить размер 50%» → кликнуть
- Перетащить персонажа в правую часть сцены

Для персонажа:

- «Когда я получу открыть»
- «Скрыть»
- «Перейти в х: 150 у: -50»
- «Ждать 2 секунды» (пока открывается)
- «Показать»
- «Говорить Поздравляю! 2 секунды»
- «Говорить Желаю счастья! 2 секунды»
- «Говорить И веселья! 2 секунды»
- Пока не проверяем

5. Создание интерактивной кнопки (10 минут)

Преподаватель делает → ученики повторяют:

- Добавить спрайт «Button1»
- Перетащить кнопку в центр экрана внизу

Для кнопки:

- «Когда этот спрайт нажат»
- «Передать открыть»
- «Скрыть»
- Проверить – нажать на кнопку – открытка открывается, персонаж говорит!

6. Добавление музыки (10 минут)

Преподаватель показывает → ученики повторяют:

- Выбрать Сцену
- Вкладка «Звуки» → «Выбрать звук»
- Найти музыку (например, «Birthday» или любая весёлая)
- Вернуться на «Код»
- В скрипте Сцены после «Когда я получу открыть»:
 - Добавить перед циклом: «Играть звук Birthday до конца» (или «Включить звук Birthday»)
- Проверить

7. Добавление финального эффекта (5 минут)

Преподаватель показывает → ученики повторяют:

- Добавить спрайт «Star» (звезда)
- «Когда я получу открыть»
- «Скрыть»

- «Ждать 4 секунды»
- «Повторить 10 раз»
- Внутрь:
 - «Показать»
 - «Изменить эффект цвет на 25»
 - «Ждать 0.1 секунды»
 - «Скрыть»
 - «Ждать 0.1 секунды»
- Проверить – звезда мигает в конце

8. Подведение итогов (5 минут)

- Преподаватель напоминает ключевые элементы
- Анонсирует следующий урок – защиту проекта

Результат урока:

- Все создали свою интерактивную открытку
- Освоили работу с сообщениями для синхронизации
- Применили все изученные навыки

УРОК 14. ИТОГОВЫЙ КОНТРОЛЬ

Длительность: 2 академических часа (90 минут)

Формат: презентация и обсуждение проектов

ХОД ЗАНЯТИЯ:

1. Объяснение процедуры защиты

Преподаватель рассказывает:

- Каждый покажет свою открытку
- Расскажет, что сделал
- Ответит на 1-2 вопроса
- Время на одного – до 10 минут

2. Защита проектов

Каждый обучающийся по очереди:

- Запускает свою открытку
- Показывает все элементы: кнопку, анимацию, персонажей, эффекты

Рассказ:

- Какую тему выбрал
- Что было сложно
- Чем гордится в проекте

Вопросы преподавателя:

- Сколько костюмов создал для анимации?
- Как использовал сообщения?

- Какие блоки были самыми полезными?

Обратная связь от преподавателя:

- Что получилось хорошо
- Одна рекомендация для улучшения

3. Подведение итогов курса

Преподаватель:

- Поздравляет всех с завершением курса
- Перечисляет, чему научились:
 - Интерфейс Scratch
 - Движение и координаты
 - Циклы и условия
 - Переменные
 - Клоны
 - Собственные блоки
 - Создание проектов
- Рассказывает о возможности публиковать проекты на scratch.mit.edu
- Предлагает продолжать практиковаться самостоятельно
- Отвечает на финальные вопросы

Результат урока:

- Все презентовали свои проекты
- Получили обратную связь
- Подвели итоги обучения

5. Оценка качества освоения программы

5.1. Формы аттестации

Оценка качества освоения программы включает текущий контроль и итоговую аттестацию и проводится в соответствии с Положением о формах, периодичности и порядке текущего контроля успеваемости, промежуточной и итоговой аттестации обучающихся.

Текущий контроль проводится на каждом занятии в формате онлайн-практики с преподавателем без выставления отметок (безотметочная система) в форме:

- наблюдения преподавателя за работой обучающихся в режиме реального времени;
- проверки правильности выполнения практических заданий непосредственно на занятии;
- оценки работоспособности созданных программ;
- оценки активности и самостоятельности обучающихся.

По результатам текущего контроля преподаватель даёт обучающемуся устную обратную связь: что получилось, над чем стоит поработать.

Итоговая аттестация проводится по завершении обучения в форме защиты итогового проекта на онлайн-занятии.

5.2. Форма итоговой аттестации

Защита итогового проекта — создание и презентация интерактивной открытки в Scratch с демонстрацией работы программы в прямом эфире, объяснением функционала и особенностей реализации, ответами на вопросы преподавателя.

Результат итоговой аттестации определяется по уровню освоения программы (высокий, средний, начальный) в соответствии с критериями, приведёнными в разделе 7.2. Обучающийся, показавший высокий, средний или начальный уровень, считается освоившим программу («освоил»). Если итоговый проект не выполнен или не работает, результат — «не освоил».

6. Условия реализации образовательной программы

6.1. Требования к минимальному материально-техническому обеспечению

Технические требования:

- компьютер или ноутбук;
- стабильное интернет-соединение (скорость не менее 5 Мбит/с);
- веб-камера и микрофон для участия в онлайн-занятиях.

Программное обеспечение:

- онлайн-среда программирования Scratch 3.0 (<https://scratch.mit.edu>);
- программа для видеосвязи «Яндекс Телемост».

6.2. Организационно-педагогические условия реализации программы

Кадровое обеспечение:

Для реализации образовательной программы привлекаются педагоги дополнительного образования, соответствующие требованиям профессионального стандарта «Педагог дополнительного образования детей и взрослых» и имеющие:

- высшее образование или среднее профессиональное образование в рамках укрупнённых групп специальностей и направлений подготовки по профилю «информатика», «программирование», «компьютерные науки» либо педагогическое образование с дополнительной подготовкой в области программирования;
- или высшее образование либо среднее профессиональное образование в рамках иных укрупнённых групп специальностей и направлений подготовки при условии его соответствия дополнительным общеобразовательным общеразвивающим программам, реализуемым организацией, и получение при необходимости после трудоустройства дополнительного профессионального образования педагогической направленности;

- или успешное прохождение промежуточной аттестации не менее чем за два года обучения по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующим направленности дополнительных общеобразовательных программ.

Требования к опыту практической работы педагогов: не менее одного года в должности педагога дополнительного образования, иной должности педагогического работника — для старшего педагога дополнительного образования.

Информационное обеспечение:

Обучающиеся получают доступ к:

- онлайн-занятиям в режиме реального времени;
- записям всех занятий для повторного просмотра;

7. Фонд оценочных материалов

7.1. Задания для текущего контроля

Текущий контроль осуществляется на каждом занятии в формате онлайн-практики без выставления отметок. Преподаватель наблюдает за работой обучающихся в режиме реального времени, оценивает правильность выполнения практических заданий, помогает с возникающими трудностями и даёт обратную связь.

Критерии текущего контроля:

- правильность повторения действий за преподавателем;
- темп работы (успевает ли обучающийся следовать за преподавателем);
- работоспособность создаваемого проекта;
- умение исправлять ошибки с подсказками преподавателя.

Ниже приведены описания проектов, которые создаются на каждом занятии в формате «преподаватель показывает → ученики повторяют».

Урок 1. Знакомство со средой Scratch

Проект: «Управляемый кот»

Обучающиеся создают программу управления спрайтом, повторяя каждое действие за преподавателем.

Урок 2. Работа со звуком

Проект: «Кот со звуками»

Обучающиеся добавляют звуки из библиотеки, записывают собственный звук, программируют воспроизведение.

Урок 3. Движение спрайтов и внешний вид

Проект: «Анимированные персонажи»

Обучающиеся работают с несколькими спрайтами, костюмами, фонами, создают анимацию.

Урок 4. Координаты и движение

Проект: «Движение по квадрату»

Обучающиеся программируют точное перемещение спрайта по координатам, используют плавное движение.

Урок 5. Циклы

Проект: «Анимированное движение»

Обучающиеся создают программу с бесконечным и ограниченным циклами, добавляют анимацию.

Урок 6. Циклы и ожидание

Проект: «Летающая птица»

Обучающиеся создают плавную анимацию движения с подбором оптимального времени ожидания.

Урок 7. Условия и сенсоры

Проект: «Интерактивный спрайт»

Обучающиеся программируют условия с различными сенсорами: мышь, клавиатура, цвет.

Урок 8. Ожидание и цикл с условием

Проект: «Вращение до условия»

Обучающиеся используют блоки «ждать до», «повторять, пока не», логические операторы.

Урок 9. Переменные

Проект: «Игра с таймером»

Обучающиеся создают переменные, программируют таймер обратного отсчёта, создают игру с целью.

Урок 10. Клоны и события

Проект: «Снегопад»

Обучающиеся создают и удаляют клоны, программируют поведение каждого клона, используют сообщения.

Урок 11. Создание собственного блока

Проект: «Структурированная программа»

Обучающиеся создают собственные блоки для группировки повторяющегося кода.

Урок 12. Создание блоков с параметрами

Проект: «Гибкие блоки»

Обучающиеся создают блоки с числовыми и логическими параметрами.

Урок 13. Создание интерактивной открытки

Проект: «Интерактивная открытка»

Обучающиеся создают комплексный проект с анимацией открытия, персонажами, музыкой, синхронизацией через сообщения.

7.2. Задание для итоговой аттестации

Урок 14. Итоговый контроль. Защита итогового проекта: интерактивная открытка

Описание задания:

Обучающийся презентует созданную на уроке 13 интерактивную открытку в Scratch, демонстрируя все элементы проекта и объясняя принципы реализации.

Обязательные элементы проекта:

1. Анимация открытия (не менее 5 костюмов фона).
2. Интерактивная кнопка запуска.
3. Не менее одного персонажа с репликами (2–3 фразы).
4. Использование сообщений для синхронизации.
5. Фоновая музыка или звуковые эффекты.

Критерии оценивания по уровням освоения программы:

Уровень	Критерии
Высокий	Проект полностью работает; выполнены все обязательные элементы, большая часть — самостоятельно; добавлены собственные элементы (эффекты, дополнительная анимация); обучающийся уверенно объясняет, как устроена программа, и отвечает на вопросы преподавателя.
Средний	Проект работает; выполнены все или почти все обязательные элементы (не менее 4 из 5), допускаются небольшие недочёты; часть работы выполнена с помощью преподавателя; обучающийся объясняет основные блоки программы с помощью наводящих вопросов.
Начальный	Проект запускается; выполнено не менее 3 из 5 обязательных элементов, в основном по образцу или с помощью преподавателя; обучающийся затрудняется в объяснении, но может показать, что делает программа.
Не освоил	Проект не выполнен, не запускается или выполнено менее 3 обязательных элементов.

Итог аттестации: высокий, средний или начальный уровень — «освоил» (выдаётся сертификат о прохождении обучения); проект не выполнен или не работает — «не освоил» (по заявлению выдаётся справка об обучении).

8. Электронная библиотека

1. Рейна Бердитт. Программирование на Scratch с нуля. Создаём весёлые игры, охотимся за багами и пишем первые программы! — М.: Эксмо, 2023. <https://www.litres.ru/book/reyna-berditt/programmirovanie-na-scratch-s-nulya-sozdaem-veselye-igry-oho-69190843/>

2. Scratch для юных программистов / пер. с англ. Банкрашкова А. — М.: Издательство АСТ, 2019. <https://www.litres.ru/book/aleksandr-bankrashkov/scratch-dlya-unyh-programmistov-24312493/>

Читать книги можно по кнопке «Читать по подписке». Логин и пароль доступны ученикам в личном кабинете на образовательной платформе.