

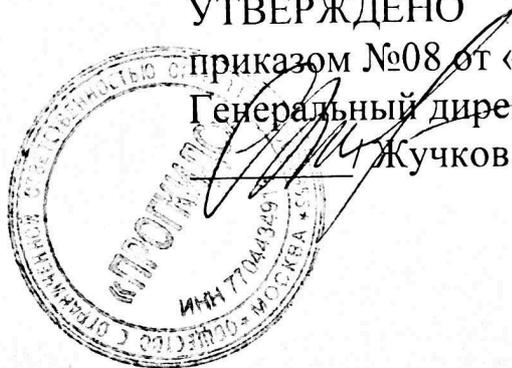
ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ "ПРОГКИДС"

УТВЕРЖДЕНО

приказом №08 от «19» мая 2025 г.

Генеральный директор

Жучков С.В.



**Дополнительная общеобразовательная программа
- дополнительная общеразвивающая программа**

«ROBLOX - ОБУЧЕНИЕ ПО СОЗДАНИЮ ИГР В ROBLOX STUDIO»

Срок освоения: 72 часа

Возраст обучающихся: обучающиеся старшего
школьного возраста, старше 18 лет

Форма обучения: заочная с исключительным применением
электронного обучения и дистанционных образовательных технологий

Направленность программы: техническая

г. Москва, 2025 г.

Содержание программы

№пп	Наименование разделов программы	Страница
1.	Пояснительная записка	3
1.1.	Нормативно-правовые основания разработки программы	3
1.2.	Актуальность образовательной программы	3
1.3.	Цель и задачи реализации программы	4
1.4.	Планируемые результаты обучения	5
1.5.	Адресат программы	6
1.6.	Трудоемкость программы	6
1.7.	Форма обучения	6
1.8.	Язык обучения	6
1.9.	Форма организации занятий	6
1.10.	Документ об обучении	6
2.	Учебный план	8
3.	Календарный учебный график	9
4.	Содержание рабочей программы	10
5.	Организационно-педагогические условия реализации программы	
5.1.	Организация образовательного процесса	29
5.2.	Материально-технические условия реализации программы	29
5.3.	Требования к кадровым условиям реализации программы	29
5.4.	Учебно-методическое обеспечение	30
6.	Оценка качества освоения образовательной программы	32
7.	Методические материалы	34

1. Пояснительная записка

Дополнительная общеобразовательная программа – дополнительная общеразвивающая программа «Roblox - обучение по созданию игр в Roblox Studio» (далее – программа) направлена на удовлетворение индивидуальных потребностей обучающихся в интеллектуальном развитии, создание и обеспечение необходимых условий для личностного развития, формирование общей культуры обучающихся, удовлетворение иных образовательных потребностей и интересов обучающихся, не противоречащих законодательству Российской Федерации, осуществляемых за пределами федеральных государственных образовательных стандартов.

Направленность программы – техническая.

Настоящая программа разработана в соответствии с п.9. ст. 2 Федерального закона от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации» и представляет собой комплекс основных характеристик образования (объем, содержание, планируемые результаты) и организационно-педагогических условий, которые представлены в виде учебного плана, календарного учебного графика, рабочих программ учебных предметов, курсов, дисциплин (модулей), иных компонентов, оценочных и методических материалов, а также форм аттестации.

1.1. Нормативно-правовые основания разработки программы

Настоящая программа разработана в соответствии с:

п.9 ст. 2, ст. 12, ст. 75 Федерального закона от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации» (далее – Федеральный закон об образовании);

Постановлением Правительства РФ от 11.10.2023 № 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;

приказом Министерства просвещения РФ от 27.07.2022 № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;

приказом Министерства труда и социальной защиты РФ от 22.09.2021 № 652н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых».

1.2. Актуальность образовательной программы

Roblox стал глобальной платформой, объединяющей миллионы пользователей по всему миру. Особенностью Roblox является уникальная возможность для игроков не только играть, но и самим создавать уникальные

игры и мировую среду, превращая хобби в профессию.

Курсы по созданию игр в Roblox Studio становятся все более популярными и востребованными по нескольким причинам:

1. Высокая популярность Roblox: платформа насчитывает сотни миллионов активных пользователей ежемесячно, многие из которых заинтересованы в изучении геймдизайна и программировании.

2. Широкие карьерные перспективы: владея навыками создания игр в Roblox Studio, подростки и молодые специалисты получают реальный шанс зарабатывать деньги, продавая созданные игры и внутриигровые активы.

3. Легкость входа и обучения: Roblox Studio доступна бесплатно, обладает интуитивным интерфейсом и возможностью изучать кодирование на Lua, что упрощает освоение программирования даже новичками.

4. Разнообразие жанров и направлений: Roblox позволяет создавать игры любого жанра, от симуляторов и приключений до шутеров и головоломок, что привлекает широкую аудиторию создателей и геймеров.

5. Образование будущего: Roblox Studio способствует развитию важных soft skills, таких как креативность, пространственное мышление, логическое мышление, проектирование и дизайн, что делает этот курс актуальным и полезным для будущих профессий в ИТ-сфере и геймдизайне.

Наш курс призван восполнить дефицит профессиональных кадров в игровой индустрии, поддержать инновационные инициативы молодежи и развить полезные навыки, востребованные рынком труда XXI века.

1.3. Цель образовательной программы – формирование у обучающихся навыков разработки компьютерных игр в среде Roblox Studio, развитие алгоритмического мышления, навыков программирования и дизайнерского творчества для успешной самореализации в цифровой среде и будущей профессии.

Задачи образовательной программы:

Обучающие:

- освоить основные инструменты и функции редактора Roblox Studio;
изучить основы программирования на языке Lua для создания сценариев и механик игр;

научиться проектировать и создавать игровые объекты, персонажи и окружения;

овладеть базовыми приемами игрового дизайна и тестирования;

разработать и опубликовать собственную игру на платформе Roblox.

Развивающие:

- развитие алгоритмического и логического мышления через решение игровых задач и программирование;
- формирование навыков креативного мышления и художественного вкуса при проектировании игровых миров;
- развитие навыков анализа и самоанализа при поиске путей улучшения созданных игр;
- совершенствование коммуникативных навыков через обмен опытом и совместное создание проектов;
- повышение информационной грамотности и культуры безопасного пользования интернет-ресурсами.

Воспитательные задачи:

- способствовать социально-профессиональному самоопределению;
- способствовать саморазвитию.

1.4. Планируемые результаты обучения:

Знать:

- стандартные объекты Roblox Studio и их параметры;
- основы программирования на языке Lua (переменные, операторы, условные конструкции, циклы);
- основные сервисы Roblox Studio (Instance, Players, ClickDetector, ProximityPrompt) и их назначение;
- механизмы работы событий объектов и методы их обработки;
- методы объектов и их применение для изменения поведения объектов в игре;
- принципы работы с аксессуарами, одеждой и магазинами предметов;
- основы создания анимации персонажей и объектов;
- технические аспекты создания NPC (неигровых персонажей), включая дружественных и враждебных;
- назначение и принципы работы ограничителей для ограничения передвижений;
- структуры данных Lua (таблицы) и обработка коллекций с помощью цикла foreach.

Уметь:

- создавать и настраивать объекты в редакторе Roblox Studio;
- работать с переменными, условиями и циклами на языке Lua для написания скриптов;
- проектировать и создавать игровые уровни с использованием механики препятствий (obby);

- настраивать магазины и создавать виртуальные предметы и аксессуары;
- создавать анимацию персонажей и объектов для оживления игры;
- интегрировать NPC и интерактивные элементы (ClickDetector, ProximityPrompt, Tool) в игру;
- тестировать и устранять баги в созданной игре, публиковать ее на платформе Roblox;
- применять таблицы и обработку коллекций для оптимизации производительности приложения;
- устанавливать ограничения на передвижение игрока и объектов с помощью ограничителей;
- объединять полученные знания и навыки для комплексного обновления и улучшения игры.

1.5. Адресат программы

Адресат программы: обучающиеся старшего школьного возраста, лица старше 18 лет. К освоению программы допускаются любые лица без предъявления требований к уровню образования.

Зачисление обучающихся на программу осуществляется в соответствии с локальным нормативным актом.

1.6. Трудоемкость (срок освоения образовательной программы) составляет 72 часа.

Трудоёмкость настоящей образовательной программы включает все виды учебной работы обучающегося (теоретические занятия, самостоятельная работа), а также время, отводимое на текущий контроль успеваемости, промежуточную и итоговую аттестацию.

1.7. Форма обучения

Обучение по образовательной программе осуществляется в заочной форме с исключительным применением электронного обучения и дистанционных образовательных технологий.

1.8. Язык обучения. Обучение по образовательной программе ведется на русском языке.

1.9. Форма организации занятий – групповая.

Количество обучающихся в группе: не более 15 человек в группе.

1.10. Документ об обучении

Лицу, освоившему образовательную программу, выдается документ об обучении по образцу, установленному самостоятельно организацией, осуществляющей образовательную деятельность – сертификат.

Лицам, освоившим часть программы и (или) отчисленным из организации, осуществляющей образовательную деятельность, выдается справка об обучении или о периоде обучения по образцу, самостоятельно устанавливаемому организацией, осуществляющей образовательную деятельность.

2. Учебный план

№ п.п.	Наименование модулей	Объем дополнительной общеобразовательной программы в академических часах					Форма промежуточной и итоговой аттестации
		Всего	Теоретическое занятие	Самостоятельная работа	Промежуточная аттестация	Итоговая аттестация	
1	Модуль 1 - Введение в программирование и создание игр в Roblox Studio	12	6	6			
2	Модуль 2 - Методы и события в Roblox Studio	22	11	11			
3	Модуль 3 - Персонализация и анимация: создание магазина и персонажей в Roblox	12	6	6			
4	Промежуточная аттестация	2			2		Тестирование
5	Модуль 4 - NPC и физика объектов: расширение игрового мира	10	5	5			
6	Модуль 5 - Продвинутая логика и структуры в Roblox Studio	12	6	6			
7	Итоговая аттестация	2				2	Тестирование
	Итого	72	34	34	2	2	

3. Календарный учебный график

Начало обучения по мере комплектования учебных групп.

Учебные группы формируются в течение всего календарного года.

№ п/п	Наименование модулей	Общая трудоемкость, час.	Период обучения
1	Модуль 1 - Введение в программирование и создание игр в Roblox Studio	12	1 месяц обучения
2	Модуль 2 - Методы и события в Roblox Studio	22	1-2 месяц обучения
3	Модуль 3 - Персонализация и анимация: создание магазина и персонажей в Roblox	12	2 месяц обучения
4	Промежуточная аттестация	2	2 месяц обучения
5	Модуль 4 - NPC и физика объектов: расширение игрового мира	10	3месяц обучения
6	Модуль 5 - Продвинутая логика и структуры в Roblox Studio	12	3 месяц обучения
7	Итоговая аттестация	2	3 месяц обучения
	Итого	72	3 месяца обучения

4. Содержание рабочей программы

Модуль 1 - Введение в программирование и создание игр в Roblox Studio

Учебно-тематический план

№ п.п.	Наименование урока	Всего	Теоретическое занятие	Самостоятельная работа
1	Урок 1. Объекты и параметры	2	1	1
2	Урок 2. Первая переменная	2	1	1
3	Урок 3. Цикл for и Vector3	2	1	1
4	Урок 4. Цикл while и условие if	2	1	1
5	Урок 5. Функции и сервис Instance	2	1	1
6	Урок 6. Создание простого obby с помощью полученных знаний. Настройка и публикация игры	2	1	1
	Итого	12	6	6

Урок 1. Объекты и параметры

Теоретическое занятие

Как создавать и редактировать игровые объекты. Как использовать основные сервисы (Workspace, Lighting). Как применять простые эффекты (огонь, дым, свет). Как менять гравитацию и время суток.

Самостоятельная работа

Задание:

Самостоятельно постройте любые объекты из деталей с простыми эффектами (например: дом, мост, факел, костёр, лампу). Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 2. Первая переменная

Теоретическое занятие

Компонент Script. Переменная. Оператор обращения (.). Иерархия и поиск объектов с помощью кода. Изменение параметров объекта через код.

Самостоятельная работа

Задание:

Самостоятельно добавь компонент Script, создай переменную и запомни в неё простое значение. Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 3. Цикл for и Vector3

Теоретическое занятие

Функция wait(). Цикл for. Тип данных Vector3.

Самостоятельная работа

Задание:

Самостоятельно добавьте компонент Script. Создайте переменную и запомните в неё объект Part, Seat или VehicleSeat.

С помощью цикла for измените параметры объекта.

Дополнительно:

Запрограммируйте полёт Seat по квадрату (вверх, вправо, вниз, влево).

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 4. Цикл while и условие if

Теоретическое занятие

Как работает условие if. Как работает цикл while. В чём разница между if и while. Как использовать цикл для повторения действий. Как применить цикл для смены времени суток.

Самостоятельная работа

Задание:

Самостоятельно создайте объект Part и компонент Script.

С помощью цикла while измените любые параметры блока (позицию, размер, цвет и т.д.).

Добавьте компонент Seat и запрограммируйте его на бесконечный полёт с помощью цикла while.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 5. Функции и сервис Instance

Теоретическое занятие

Как создавать функции и вызывать их в нужный момент. Как использовать Instance для динамического создания объектов. Как работать с математической библиотекой для создания случайных значений. Как улучшить игровой процесс, создавая различные объекты и эффекты в Roblox Studio.

Самостоятельная работа

Задание:

С помощью функции опиши любое действие (например, смену цвета, передвижение объекта, изменение размера) и вызови эту функцию;

С помощью класса Instance создай объект (например, блок или модель).

Создай лестницу с помощью цикла for и Instance.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 6. Создание простого obbu с помощью полученных знаний. Настройка и публикация игры

Теоретическое занятие

Как создавать и редактировать игровые объекты с помощью скриптов. Как использовать основные сервисы (Workspace, Lighting) для настройки игры. Как программировать и изменять поведение объектов с помощью скриптов (движение, вращение, исчезновение). Как публиковать свою игру и делиться ею с другими пользователями Roblox.

Самостоятельная работа

Задание:

Создайте свой собственный obbu, используя примитивные объекты и скрипты для создания интересных механик (например, вращающиеся блоки, движущиеся платформы, исчезающие блоки).

Примените хотя бы два различных эффекта для улучшения игрового процесса (например, свет, огонь, дым).

Используя цикл for, создайте динамические изменения для блоков (например, изменение размеров, движение или смена цвета).

Опубликуйте вашу игру на Roblox и поделитесь ссылкой с друзьями.

Попробуйте создать уникальные препятствия для своего obbu, используя различные механизмы (толкающие блоки, перемещающиеся платформы, поднимающиеся лестницы).

Прикрепите скриншот постройки к ответу на образовательной платформе.

Модуль 2 - Методы и события в Roblox Studio

Учебно-тематический план

№ п.п.	Наименование урока	Всего	Теоретическое занятие	Самостоятельная работа
1	Урок 1. Методы объектов. Часть 1	2	1	1
2	Урок 2. Методы объектов. Часть 2	2	1	1
3	Урок 3. Методы объектов. Часть 3	2	1	1
4	Урок 4. Методы объектов. Часть 4	2	1	1
5	Урок 5. События компонента Part	2	1	1
6	Урок 6. Обновление obbu с помощью полученных знаний. Часть 1	2	1	1
7	Урок 7. Сервис Players	2	1	1

8	Урок 8. События объекта ClickDetector	2	1	1
9	Урок 9. События объекта ProximityPrompt	2	1	1
10	Урок 10. Компонент Tool	2	1	1
11	Урок 11. Обновление obby с помощью полученных знаний. Часть 2	2	1	1
	Итого	22	11	11

Урок 1. Методы объектов. Часть 1

Теоретическое занятие

Как вызывать методы поиска и ожидания объекта. Как работать с компонентами Humanoid и Atmosphere. Как изменять параметры персонажа, такие как скорость, сила прыжка и размер головы. Как работать с атмосферой, управляя туманностью, цветом и бликами.

Самостоятельная работа

Задание:

Повторите работу с методом WaitForChild.

Повторите работу с методом FindFirstChild.

Повторите работу с методом FindFirstChildOfClass.

Измените параметры персонажа, например, скорость бега или высоту прыжка. С помощью цикла while постепенно повышайте скорость персонажа (не забудьте использовать функцию wait()).

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 2. Методы объектов. Часть 2

Теоретическое занятие

Как вызывать методы поиска и ожидания объекта. Как использовать методы ApplyImpulse для задания скорости объектам. Как клонировать объекты с помощью метода Clone. Как удалять объекты с помощью метода Destroy для оптимизации игры. Как создать бесконечный фонтан из объектов, используя эти методы.

Самостоятельная работа

Задание:

Повторите работу с методом ApplyImpulse.

Повторите работу с методом Clone.

Повторите работу с методом Destroy.

Попробуйте создать объект с применением методов ApplyImpulse и Clone, а затем удалите его с помощью метода Destroy.

Прикрепите скриншот постройки к ответу на образовательной платформе.

платформе.

Урок 3. Методы объектов. Часть 3

Теоретическое занятие

Как создать модель и назначить ей основной элемент (PrimaryPart). Как использовать метод MoveTo для телепортации модели с учетом препятствий. Как применять метод PivotTo для перемещения и поворота модели без учета препятствий. Как использовать метод TranslateBy для смещения модели на заданное расстояние.

Самостоятельная работа

Задание:

Создайте собственную модель, состоящую из нескольких объектов.

Назначьте один из объектов как PrimaryPart.

Повторите работу с методом MoveTo — переместите модель в новые координаты.

Используйте метод PivotTo для перемещения и поворота модели.

Попробуйте плавно сместить модель с помощью метода TranslateBy.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 4. Методы объектов. Часть 4

Теоретическое занятие

Как использовать метод MoveTo для перемещения персонажей. Как применять метод TakeDamage для нанесения урона персонажам. Как заставить персонажа преследовать игрока и прыгать при приближении. Как рассчитывать расстояние между объектами с помощью Magnitude.

Самостоятельная работа

Задание:

Повторите работу с методами MoveTo и TakeDamage у персонажей.

Попробуйте изменить условие срабатывания урона — например, увеличьте дистанцию или добавьте проверку на высоту.

Создайте второго персонажа и заставьте его тоже преследовать игрока.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 5. События компонента Part

Теоретическое занятие

Что такое событие и как оно работает. Как использовать метод Connect для обработки событий. Как срабатывают события Touched и TouchEnded. Как взаимодействовать с персонажами при касании. Как создать интерактивный блок, реагирующий на прикосновение.

Самостоятельная работа

Задание:

Создайте объект Part и добавьте к нему компонент Script.

Запрограммируйте событие Touched на любое действие: изменение цвета, размера, добавление эффекта и т.д.

Попробуйте использовать событие TouchEnded, чтобы отменить или изменить действие после завершения касания.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 6. Обновление obby с помощью полученных знаний. Часть 1

Теоретическое занятие

Как применять события и методы в своей игре. Как создавать игровые механики, основанные на касании объектов. Как опубликовать игру и настроить её параметры в Roblox Studio.

Самостоятельная работа

Задание:

Дополните свой obby новыми препятствиями, используя изученные события и методы.

Придумайте и реализуйте как минимум два разных типа блоков с уникальными эффектами (например, телепорт, исчезновение, урон).

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 7. Сервис Players

Теоретическое занятие

Как использовать событие PlayerAdded для отслеживания входа игроков. Как работает событие CharacterAdded и чем оно полезно. Как создавать и настраивать папку leaderstats. Как сохранить скрипт в Toolbox для повторного использования.

Самостоятельная работа

Задание:

Самостоятельно создать дополнительное значение (например, очки, опыт или уровень) и добавить его в папку leaderstats. Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 8. События объекта ClickDetector

Теоретическое занятие

Как использовать ClickDetector для обработки щелчков по объекту. Какие события поддерживает ClickDetector. Как создать функциональные объекты в

игре. Как связать ClickDetector с игровыми значениями. Как сделать игру, в которой за клики начисляются очки.

Самостоятельная работа

Задание:

Создайте собственный интерактивный объект с ClickDetector, который при нажатии меняет своё поведение: например, исчезает, меняет цвет, увеличивается или вращается.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 9. События объекта ProximityPrompt

Теоретическое занятие

Как работать с компонентом ProximityPrompt. Как настраивать внешний вид подсказки. Порядок обработки события Triggered и взаимодействия с персонажем. Как клонировать и заменять персонажей в игре. Как создавать интересные игровые механики на основе близости.

Самостоятельная работа

Задание:

Создайте собственный объект с компонентом ProximityPrompt, который при активации будет выполнять одно из следующих действий:

Делать часть персонажа невидимой;

Телепортировать персонажа;

Превращать персонажа во что-то другое (например, в животное или предмет).

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 10. Компонент Tool

Теоретическое занятие

Как работает компонент Tool. Какие события существуют у инструмента. Как инструмент взаимодействует с объектами. Как создавать простые игровые механики с помощью инструмента. Как изменить свойства персонажа с помощью инструмента.

Самостоятельная работа

Задание:

Повторить работу с компонентом Tool, создать интерактивный инструмент, который изменяет свойства объектов при активации.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 11. Обновление obby с помощью полученных знаний. Часть 2

Теоретическое занятие

Как использовать ClickDetector для реагирования на клики. Как работать с ProximityPrompt для создания интерактивных элементов. Как использовать Tool для создания объектов, которые можно взять. Как создавать препятствия и интерактивные элементы в Obby.

Самостоятельная работа

Задание:

Использовать компоненты ClickDetector, ProximityPrompt и Tool для создания новых препятствий и интерактивных элементов в вашем Obby.

Добавьте новые объекты, которые игроки смогут активировать при клике или вблизи с помощью ProximityPrompt.

Создайте Tool, который будет выдавать предметы, открывать проходы или изменять окружающую среду при взаимодействии. Прикрепите скриншот постройки к ответу на образовательной платформе.

Модуль 3 - Персонализация и анимация: создание магазина и персонажей в Roblox

Учебно-тематический план

№ п.п.	Наименование урока	Всего	Теоретическое занятие	Самостоятельная работа
1	Урок 1. Аксессуары	2	1	1
2	Урок 2. Одежда	2	1	1
3	Урок 3. Создание анимации	2	1	1
4	Урок 4. Создание магазина с предметами. Часть 1	2	1	1
5	Урок 5. Создание магазина с предметами. Часть 2	2	1	1
6	Урок 6. Создание магазина с предметами. Часть 3	2	1	1
	Итого	12	6	6

Урок 1. Аксессуары

Теоретическое занятие

Как создать аксессуар с помощью компонента Accessory. Что такое Handle и для чего он нужен. Как прикреплять аксессуары к частям тела персонажа. Как изменить положение и размер аксессуара. Как сделать так, чтобы аксессуар надевался на персонажа игрока автоматически.

Самостоятельная работа

Задание:

Попробуйте создать свой собственный аксессуар (или несколько) для персонажа.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 2. Одежда

Теоретическое занятие

Что такое компоненты `Pants` и `Shirt`. Как использовать шаблоны для создания одежды. Как загрузить изображение одежды в `Roblox Studio`. Как прикрепить одежду к персонажу вручную. Как написать код, чтобы переодеть персонажа при касании или с помощью кнопки.

Самостоятельная работа

Задание:

Нарисуйте свою одежду по шаблону `Shirt` или `Pants`;

Загрузите её в `Roblox Studio`;

Попробуйте с помощью кода надеть её на персонажа — через касание или интерактивную кнопку.

Можете использовать любой из двух способов — или оба!

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 3. Создание анимации

Теоретическое занятие

Что такое анимация в `Roblox Studio` и как она устроена. Как пользоваться инструментом `Animation Editor`. Как настроить параметры и приоритет анимации. Как загрузить и воспроизвести анимацию через скрипт.

Самостоятельная работа

Задание:

1. Создайте собственную анимацию движения, действия или эмоции.
2. Загрузите её в `Roblox Studio`.
3. Напишите код, чтобы воспроизвести её в игре.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 4. Создание магазина с предметами. Часть 1

Теоретическое занятие

Как использовать компоненты `Part`, `Light`, `Attachment` и `ProximityPrompt`. Как собрать элементы сцены в модели и задать `PrimaryPart`. Как применять `CFrame` и `PivotTo` для поворота объектов. Как подключать функции к событиям и создавать простую анимацию в коде.

Самостоятельная работа

Задание:

- 1) Украсьте интерьер своего магазина: добавьте полки, прилавок, кассу и витрину.
- 2) Создайте собственные аксессуары и предметы (Tool, Accessory, Pants и Shirt) и разместите их в магазине.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 5. Создание магазина с предметами. Часть 2

Теоретическое занятие

Как использовать AnimationEditor для создания анимаций. Как публиковать и применять анимации через компонент Animation. Как запускать анимации с помощью скриптов и циклов. Как задать приоритеты анимациям и управлять поведением персонажа в игре.

Самостоятельная работа

Задание:

Украсьте интерьер магазина используя знания из предыдущих уроков, создай:

- 1) Инструменты (компонент Tool)
- 2) Аксессуары (компонент Accessory)
- 3) Одежду (компоненты Pants и Shirt)

Эти вещи будут продаваться в твоём магазине в будущем!

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 6. Создание магазина с предметами. Часть 3

Теоретическое занятие

Как превращать 3D-модели в игровые предметы. Как организовывать покупку предметов через скрипты. Как работать с одеждой и аксессуарами. Как создавать монеты, которые дают очки при сборе.

Самостоятельная работа

Задание:

Добавьте собственные товары в ваш магазин!

Используйте полученные знания, чтобы создать:

- 1) Инструменты (Tool);
- 2) Аксессуары (Accessory);
- 3) Одежду (Pants и Shirt).

Продумайте, как они будут размещены в вашем магазине, и настройте для них покупку. Прикрепите скриншот постройки к ответу на образовательной платформе.

Промежуточная аттестация

Тестирование:

1. Что происходит с объектом, если отключить параметр `CanCollide`?

Объект становится невидимым

Объект начинает падать

Объект проходит сквозь другие объекты

2. Какой из следующих циклов существует в программировании?

`do-while`

`for` и `while`

`repeat-until`

3. Какой из примеров показывает перемещение объекта вверх на единицу в цикле?

`part.Position = part.Position + Vector3.new(0, 1, 0)`

`part.Position = part.Position + Vector3.new(1, 0, 0)`

`part.Position = part.Position + Vector3.new(0, 0, 1)`

4. Какой результат будет после выполнения следующего кода? `if 10 > 5 then print("True") end`

"False"

"True"

Ошибка

5. Какой из следующих операторов используется для проверки на равенство?

`~=`

`==`

`=`

6. Что должно быть добавлено в бесконечный цикл `while`, чтобы избежать зависания игры?

Основные условия

Функция `wait()`

Логические операторы

7. Что делает ключевое слово `return` в функции?

Завершает выполнение программы

Возвращает значение из функции

Вызов другая функция

8. Какая из следующих строк создаёт новый объект в Lua?

`local part = new("Part")`

```
local part = Instance.new("Part")  
local part = create("Part")
```

9. Какой метод используется для ожидания загрузки дочернего объекта в Roblox?

```
WaitForChild  
FindFirstChildGet  
Children
```

10. Какое свойство объекта отвечает за его положение в игровом мире?

```
Position  
Location  
Coordinates
```

11. Какое событие срабатывает, когда один объект касается другого в Roblox?

```
TouchEnd  
Touched  
OnContact
```

12. Какой метод используется для подключения функции к событию?

```
Add  
Connect  
Bind
```

13. Какую функцию можно вызвать для нанесения урона персонажу при касании с объектом Part?

```
TakeDamage  
InflictDamage  
HitCharacter
```

14. Какое событие срабатывает при нажатии левой клавиши мыши на объект с компонентом ClickDetector?

```
MouseHoverEnter  
MouseClicked  
LeftMouseClicked
```

15. Какое событие компонента Tool срабатывает, когда игрок использует (нажимает) инструмент?

```
Unequipped  
Equipped  
Activated
```

Модуль 4 - NPC и физика объектов: расширение игрового мира

Учебно-тематический план

№ п.п.	Наименование урока	Всего	Теоретическое занятие	Самостоятельная работа
1	Урок 1. Создание неигровых персонажей (NPC)	2	1	1
2	Урок 2. Создание враждебного NPC	2	1	1
3	Урок 3. Созданием питомца - защитника	2	1	1
4	Урок 4. Изучение работы ограничителей. Часть 1	2	1	1
5	Урок 5. Изучение работы ограничителей. Часть 2	2	1	1
	Итого	10	5	5

Урок 1. Создание неигровых персонажей (NPC)

Теоретическое занятие

Как запрограммировать персонажа на смену анимации в зависимости от действий. Как использовать компоненты Dialog и DialogChoice для создания диалогов в игре. Как управлять взаимодействием игрока с NPC через текстовые подсказки.

Самостоятельная работа

Задание:

Самостоятельно создайте других общающихся персонажей.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 2. Создание враждебного NPC

Теоретическое занятие

Как создать враждебного NPC, который ищет игрока, приближается к нему и наносит урон. Как добавлять компонент Configuration и использовать его для хранения параметров поведения. Как назначать цель NPC через компонент ObjectValue. Как запускать и останавливать анимации в зависимости от условий в коде. Как проверять расстояние до игрока и задавать действия NPC в зависимости от этого.

Самостоятельная работа

Задание:

Повторить реализацию всех событий (Running, Jumping, FreeFalling, MoveToFinished).

Добавить новые эффекты (например, дым при прыжке).
Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 3. Созданием питомца - защитника

Теоретическое занятие

Как использовать компонент VectorForce для управления движением объектов. Как создавать и настраивать питомцев-защитников. Как реализовать атаку с помощью снарядов, использующих физику VectorForce. Как отличать дружественные и враждебные цели с помощью ObjectValue и Humanoid.

Самостоятельная работа

Задание:

Самостоятельно создать других питомцев-защитников.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 4. Изучение работы ограничителей. Часть 1

Теоретическое занятие

Как использовать компоненты ограничителей для связывания объектов. Как отключать столкновения между двумя объектами. Как создавать эффекты верёвки, пружины и стержня. Как применять ограничители для взаимодействия объектов в физическом движке Roblox.

Самостоятельная работа

Задание:

Повторите работу с компонентами NoCollisionConstraint, RodConstraint, SpringConstraint, RopeConstraint и WeldConstraint.

Создайте сцены, в которых соедините объекты разными типами ограничителей и поэкспериментируйте с их параметрами.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 5. Изучение работы ограничителей. Часть 2

Теоретическое занятие

Как использовать компонент HingeConstraint для создания вращающихся объектов. Как соединять части конструкции с помощью различных физических ограничителей. Как создавать объекты, которые взаимодействуют с персонажем и другими объектами с учётом физики Roblox. Как программно управлять скоростью вращения объектов с помощью HingeConstraint..

Самостоятельная работа

Задание:

Повторите создание физической карусели с использованием компонента HingeConstraint.

Поэкспериментируйте с различными типами ограничителей (RodConstraint, RopeConstraint, SpringConstraint, WeldConstraint), соединяя их с каруселью.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Модуль 5 - Продвинутая логика и структуры в Roblox Studio

Учебно-тематический план

№ п.п.	Наименование урока	Всего	Теоретическое занятие	Самостоятельная работа
1	Урок 1. Общие события объектов	2	1	1
2	Урок 2. Создание ездовых питомцев	2	1	1
3	Урок 3. Таблицы	2	1	1
4	Урок 4. Методы объектов	2	1	1
5	Урок 5. Цикл foreach	2	1	1
6	Урок 6. Обновление obby с помощью полученных знаний. Часть 3	2	1	1
	Итого	12	6	6

Урок 1. Общие события объектов

Теоретическое занятие

Как использовать событие Changed для отслеживания любых изменений свойств объекта. Как с помощью событий ChildAdded и ChildRemoved реагировать на добавление или удаление дочерних объектов. Как программировать объекты, которые автоматически изменяют своё поведение в ответ на взаимодействия игроков.

Самостоятельная работа

Задание:

Повторить реализацию всех событий (Changed, ChildAdded, ChildRemoved).

Создать объект, меняющий цвет при изменении позиции.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 2. Создание ездовых питомцев

Теоретическое занятие

Как запрограммировать персонажа на смену анимации в зависимости от действий. Как использовать компоненты Dialog и DialogChoice для создания диалогов в игре. Как управлять взаимодействием игрока с NPC через текстовые подсказки.

Самостоятельная работа

Задание:

Создайте нового ездового питомца с уникальной механикой (например, летающего питомца).

Добавьте визуальные эффекты при движении питомца (следы, частицы или свечение).

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 3. Таблицы

Теоретическое занятие

Как создавать и заполнять таблицы в Lua. Как получать элементы из таблицы и изменять их. Как применять функции insert, remove, sort для работы с таблицами. Как использовать оператор # для получения длины таблицы. Как использовать таблицы для хранения и обработки данных в Roblox Studio.

Самостоятельная работа

Задание:

Создайте таблицу с координатами точек для перемещения NPC.

Реализуйте систему случайного выбора элемента из таблицы.

Попробуйте добавить и удалить элементы из таблицы с помощью функций insert и remove.

Отсортируйте элементы таблицы (если таблица подходит для сортировки).

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 4. Методы объектов

Теоретическое занятие

Как получать потомков объекта с помощью методов GetChildren и GetDescendants. Как работать с сервисами Roblox с помощью метода GetService. Как получать список всех игроков с помощью метода GetPlayers. Как использовать таблицы и циклы для массовой обработки объектов.

Самостоятельная работа

Задание:

Создайте систему спавна NPC в точках из таблицы.

Реализуйте поиск всех объектов с определённым тегом.

Попробуйте изменить параметры (например, цвет или размер) сразу у нескольких объектов с помощью цикла.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 5. Цикл foreach

Теоретическое занятие

Как использовать `ipairs` для упорядоченных таблиц. Как применять `pairs` для нелинейных структур. Как создавать таблицы объектов и управлять группами элементов.

Самостоятельная работа

Задание:

Создать таблицу NPC с рандомным выбором через `math.random()`.

Реализовать систему поиска объектов по тегу через `pairs()`.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Урок 6. Обновление obby с помощью полученных знаний. Часть 3

Теоретическое занятие

Как использовать таблицы и циклы для создания динамических элементов в игре. Как создавать и управлять такими элементами, как динамические лестницы, летающие блоки и стены с пушками. Как добавлять случайные элементы и ловушки, активируемые при приближении игрока.

Самостоятельная работа

Задание:

Самостоятельно добавь систему случайного спавна препятствий через `math.random()`. Реализуй «умные» ловушки, активирующиеся при приближении игрока.

Прикрепите скриншот постройки к ответу на образовательной платформе.

Итоговая аттестация

Тестирование:

1. Какой компонент используется для применения постоянной силы к объекту?

BodyVelocity

LinearVelocity

BodyForce

VectorForce

2. Какой ограничитель предотвращает столкновения между двумя частями?

SpringConstraint
WeldConstraint
RodConstraint
NoCollisionConstraint

3. Какое событие срабатывает при добавлении дочернего объекта?

DescendantAdded
Changed
ChildRemoved
ChildAdded

4. Какой ограничитель соединяет две части с эффектом пружины?

SpringConstraint
RopeConstraint
WeldConstraint
RodConstraint

5. Какой компонент позволяет контролировать вращение объекта?

AngularVelocity
VectorForce
LinearVelocity
BodyGyro

6. Какой метод возвращает всех потомков объекта, включая вложенные?

GetService
GetDescendants
GetChildren
FindFirstChild

7. Какой метод задерживает выполнение скрипта?

wait()
delay()
pause()

8. Какой из этих проектов лучше всего подойдёт для использования HingeConstraint?

Карусель
Телепорт
Питомец
Диалог

9. Что произойдет, если условие в операторе if будет ложным?

Код внутри if не выполнится

Код внутри if выполнится
Код всегда выполнится

10. Какой компонент используется для хранения настроек NPC, таких как здоровье и урон?

Folder
Configuration
Script
ModuleScript

11. Какой компонент используется для создания диалогов у NPC?

Dialog
Hint
DialogChoice
Message

12. Какой синтаксис используется для создания таблицы в Lua?

table = ()
table = {}
table = []
table =

13. Какой компонент позволяет добавить варианты ответов в диалоге NPC?

DialogOption
Choice
DialogChoice
Hint

14. Какой цикл чаще всего используют с таблицами, если важен порядок элементов?

for
while
for...in pairs
for...in ipairs

15. Что произойдет, если не указать время в функции wait()?

Будет использовано минимальное время - 29 миллисекунд
Будет использовано максимальное время
Программа завершится

5. Организационно-педагогические условия реализации программы

5.1. Организация образовательного процесса

Режим занятий устанавливается в соответствии с учебным расписанием.

Для всех видов занятий академический час устанавливается продолжительностью 40 минут.

Выбор методов обучения для каждого занятия определяется педагогом дополнительного образования в соответствии с составом и уровнем подготовленности обучающихся, степенью сложности излагаемого материала, наличием и состоянием учебного оборудования, технических средств обучения, местом и продолжительностью проведения занятий.

Теоретические занятия проводятся с целью изучения нового учебного материала. Изложение материала необходимо вести в форме, доступной для понимания обучающихся, соблюдать единство терминологии, условных обозначений. В ходе занятий педагог дополнительного образования обязан соотносить новый материал с ранее изученным, дополнять основные положения примерами из практики, соблюдать логическую последовательность изложения.

Самостоятельная работа проводится с целью закрепления теоретических знаний и выработки у обучающихся основных умений и навыков работы в ситуациях, максимально имитирующих реальные процессы.

5.2. Материально-технические условия реализации программы

Обучение осуществляется через систему дистанционного обучения – на образовательной платформе «Геткурс», режим доступа к курсу:

<https://progkids.getcourse.ru/teach/control/stream/view/id/934424302>

Обеспечение функционирования информационно-коммуникационной системы: 1) Web-камера Logitech HD Webcam C310 -1 шт.;

2) микрофон HIPER H-M008 - 1шт.;

3) ноутбук Asus Vivobook 16 X1605ZA-MB321 16" – 1 шт.;

4) наушники JBL T510 -1 шт.;

5) стол -1 шт.; 6) стул -1 шт.

Обеспечение доступа к информационно-телекоммуникационным сетям: сеть Интернет.

5.3. Требования к кадровым условиям реализации программы

К педагогу дополнительного образования предъявляются следующие требования к образованию и обучению:

Высшее образование или среднее профессиональное образование в рамках укрупненных групп специальностей и направлений подготовки высшего образования и специальностей среднего профессионального образования «Образование и педагогические науки»

или

Высшее образование либо среднее профессиональное образование в рамках иных укрупненных групп специальностей и направлений подготовки высшего образования и специальностей среднего профессионального образования при условии его соответствия дополнительным общеразвивающим программам, дополнительным предпрофессиональным программам, реализуемым организацией, осуществляющей образовательную деятельность, и получение при необходимости после трудоустройства дополнительного профессионального образования педагогической направленности

или

Успешное прохождение обучающимися промежуточной аттестации не менее чем за два года обучения по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующим направленности дополнительных общеобразовательных программ (приказ Министерства труда и социальной защиты РФ от 22.09.2021 № 652н).

К занятию педагогической деятельностью по дополнительным общеобразовательным программам допускаются лица, обучающиеся по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующим направленности дополнительных общеобразовательных программ, и успешно прошедшие промежуточную аттестацию не менее чем за два года обучения. Соответствие образовательной программы высшего образования направленности дополнительной общеобразовательной программы определяется работодателем (ч. 4 ст. 46 Федерального закона от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»).

5.4. Учебно-методическое обеспечение:

1. Официальная документация Roblox Creator Hub

<https://create.roblox.com/docs>

Полный справочник по Roblox Studio, скриптингу, интерфейсам, физике и анимации.

2. Введение в скриптинг в Roblox

<https://create.roblox.com/docs/tutorials/use-case-tutorials/scripting/basic-scripting/intro-to-scrip...>

Пошаговое руководство по созданию скриптов в Roblox Studio, включая переменные, функции и события.

3. Учебные ресурсы для преподавателей от Roblox Education

<https://create.roblox.com/docs/education/resources/resources-landing>

Учебные планы, карточки с проектами и PDF-файлы для преподавателей и учеников.

4. Полный учебник по скриптингу для начинающих

<https://devforum.roblox.com/t/full-tutorial-how-to-script-on-roblox-beginners/1991385>

Подробный гайд по основам скриптинга в Roblox, включая переменные, функции, условия и события.

5. Справочник по скриптингу в Roblox

<https://cheatography.com/ozzytig/cheat-sheets/roblox-general-scripting/>

Краткий справочник по скриптингу в Roblox, включая основные команды и структуры.

6. Руководство по программированию Roblox Lua на GitHub

<https://github.com/PacktPublishing/Coding-Roblox-Games-Made-Easy>

Репозиторий с примерами кода и проектами для обучения программированию в Roblox.

7. Руководство по программированию Roblox для детей

<https://github.com/murrekatt/programming-roblox-with-lua>

Учебные материалы и лекции по программированию в Roblox, ориентированные на детей и начинающих.

6. Оценка качества освоения образовательной программы

При реализации образовательной программы оценка результатов освоения программы проводится в рамках текущего контроля успеваемости, промежуточной аттестации.

Итоговая аттестация по программе не осуществляется.

Текущий контроль успеваемости

Текущий контроль успеваемости осуществляется в ходе изучения тем образовательной программы.

Текущий контроль успеваемости проводится с целью получения оперативной информации о качестве усвоения обучающимися учебного материала, управления учебным процессом и совершенствования методики проведения занятий.

Форма текущего контроля – контроль активности на образовательной платформе, анализ выполненных заданий самостоятельной работы.

Критерии оценивания текущей аттестации: правильность ответа по содержанию занятия (учитывается количество и характер ошибок при ответе); рациональность использованных приемов и способов решения поставленной учебной задачи.

Промежуточная аттестация

Освоение образовательной программы сопровождается промежуточной аттестацией обучающихся, проводимой в форме, определенной учебным планом, и в порядке, установленной организацией, осуществляющей образовательную деятельность.

Промежуточная аттестация обучающихся предназначена для определения степени достижения учебных целей при изучении тем программы в целом.

Форма промежуточной аттестации – тестирование.

Промежуточная аттестация проводится после освоения отдельных модулей образовательной программы согласно календарному учебному графику и расписанию учебных занятий.

При проведении промежуточной аттестации применяется зачетная система оценки: «зачтено»/ «не зачтено».

При проведении промежуточной аттестации обучающихся используются **оценочные материалы**, установленные разделом 4 настоящей образовательной программы: Содержание рабочей программы.

Критерием оценки служит следующая шкала количества верных ответов (в %):

0–70 % – не зачтено;

71%–100% – зачтено.

Итоговая аттестация обучающихся

Освоение программы завершается итоговой аттестацией обучающихся.

Форма итоговой аттестации: тестирование.

К итоговой аттестации допускается обучающийся, не имеющий академической задолженности и в полном объеме выполнивший учебный план (индивидуальный учебный план) по программе.

Объем времени аттестационных испытаний, входящих в итоговую аттестацию обучающихся, устанавливается учебным планом.

Лицам, не прошедшим итоговой аттестации или получившим на итоговой аттестации неудовлетворительные результаты, а также лицам, освоившим часть образовательной программы и (или) отчисленным из организации, осуществляющей образовательную деятельность, выдается справка об обучении или о периоде обучения по образцу, самостоятельно устанавливаемой организацией, осуществляющей образовательную деятельность.

Оценочные материалы итоговой аттестации устанавливаются разделом 4 настоящей образовательной программы: Содержание рабочей программы.

При проведении итоговой аттестации применяется зачетная система оценки: «зачтено»/ «не зачтено».

Критерием оценки служит следующая шкала количества верных ответов (в %):

0–70 % – не зачтено;

71%–100% – зачтено.

7. Методические материалы

Методические указания по освоению программы

Вид учебных занятий	Методические указания для обучающихся по освоению модуля по видам учебных занятий
Теоретическое занятие	<p>Конспектирование видеолекций, работа с конспектом. Изучение тем программы требует систематического и последовательного накопления знаний, поэтому пропуски отдельных тем не позволяют глубоко освоить программу. Желательно запись осуществлять на одной странице листа или оставляя поля, на которых позднее, при самостоятельной работе с конспектом, можно сделать дополнительные записи, отметить непонятные места.</p> <p>Обучающимся необходимо:</p> <ul style="list-style-type: none">- перед каждым теоретическим занятием просматривать рабочую программу программы – содержание темы, что позволит сэкономить время на записывание темы, ее основных вопросов;- перед очередным теоретическим занятием необходимо просмотреть по конспекту материал предыдущего теоретического занятия. При затруднениях в восприятии материала следует обратиться к педагогу. <p>Обратная связь представляется через электронную почту или чат в мессенджере.</p>
Самостоятельная работа	<p>К выполнению задания самостоятельной работы необходимо приступить после освоения теоретического занятия.</p> <p>В случае возникновения вопросов при выполнении обучающимся заданий практических занятий педагогом предоставляется обратная связь.</p> <p>Обратная связь представляется через электронную почту или чат в мессенджере либо на онлайн-консультациях.</p> <p>Онлайн-консультации проводятся в режиме видеоконференцсвязи через сервис «МТС Линк» https://events.webinar.ru</p>
Подготовка к промежуточной и итоговой аттестации	<p>Работа с конспектами теоретических занятий. При подготовке к промежуточной и итоговой аттестации необходимо обратиться к оценочным материалам, установленным настоящей образовательной программой.</p>