

Технологическая карта урока

ФИО педагогического работника: Фостаковская Екатерина Викторовна

Тип занятия: Комбинированный урок

Дисциплина: МДК 12.02 Программирование и разработка на платформе Unity

Тема: Добавление объектов (клонирование) на сцену программно.

Нормативные документы: 1) ФГОС СПО по специальности 09.02.07 Информационные системы и программирование

Квалификация: программист

2) Учебная программа ПМ.12. Разработка игр

С целью овладения профессиональными компетенциями обучающихся в ходе освоения дисциплины должен уметь:

- разрабатывать программные модули для пользовательского интерфейса, игровых уровней и объектов;
- правильно составлять программный код в соответствии с общепринятыми парадигмами;
- исправлять, возникающие в процессе написания, и в процессе сборки, ошибки;
- выполнять поставленные в соответствии с ТЗ задачи;

знать:

- принципы разработки программного кода;
- методы работы с вспомогательным специализированным программным обеспечением, сопровождающим разработку игры;
- особенности всесторонней работы с Unity;

- основные принципы и методы написания компьютерных программ на языке программирования высокого уровня;
- основные принципы систематизации информации к решению практических задач по программированию.

Цель:

Обучающая: с целью формирования у обучающихся умений выполнять работу в Unity, работать с компонентами, сценами, пользовательским интерфейсом студент должен знать принципы создания ;

уметь

Развивающая: развивать умение обучающихся самостоятельно работать с печатными материалами для получения навыков работы в программе Unity. Развивать трудолюбие, ответственность, внимательность, коммуникативность.

Воспитательная: воспитание профессиональных качеств личности обучающегося, таких как аккуратность и самостоятельность при выполнении заданий. Формирование интереса к профессиональной деятельности по специальности.

Форма занятия: практическое занятие.

Межпредметные связи:

1. ОУДД.16 Введение в креативные индустрии
2. ОП.03. Информационные технологии
3. ОП.04. Основы алгоритмизации и программирования
4. МДК 12.01 Гейм-дизайн

Структура занятия:

1. Организация занятия – 3 мин.
2. Проверка знаний и умений обучающихся - 10 мин.

3. Постановка задания для обучающихся – 5 мин.
4. Изложение нового материала – 20 мин.
5. Инструктаж – 5 мин.
6. Самостоятельная работа обучающихся и текущий инструктаж - 40 мин.
7. Итоговый контроль выполнения работы – 5 мин.
8. Домашнее задание – 2 мин.

Используемая литература:

1. Корнилов, А. В. UNITY. Полное руководство. (+виртуальный DVD 10 Гб с Unity-проектами, примерами из книги и ассетами) / А. В. Корнилов. — 2-е изд. — Санкт-Петербург : Наука и Техника, 2021. — 496 с. — ISBN 978-5-94387-721-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/191482> (дата обращения: 28.09.2023). — Режим доступа: для авториз. пользователей.
2. Курбанисмаилов, З. М. Современные подходы в программировании при создании интерактивной анимации на C# и Unity : учебно-методическое пособие / З. М. Курбанисмаилов. — Москва : РТУ МИРЭА, 2021. — 142 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/176569> (дата обращения: 28.09.2023). — Режим доступа: для авториз. пользователей.
3. Ларкович, С. Н. UNITY на практике. Создаем 3D-игры и 3D-миры : учебное пособие / С. Н. Ларкович. — 2-е изд. перераб. и доп. — Санкт-Петербург : Наука и Техника, 2022. — 384 с. — ISBN 978-5-907592-02-5. — Текст :

электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/297155> (дата обращения: 28.09.2023). — Режим доступа: для авториз. пользователей.

4. Руководство пользователя Unity 2022: официальная документация кроссплатформенной среды разработки компьютерных игр - docs.unity3d.com
5. Unity в действии. Мультиплатформенная разработка на C# / Джозеф Хокинг/ 2-е международное издание — СПб.: Питер, 2019. — 352 с.

Ход урока

Этапы	Цель	Деятельность преподавателя	Деятельность студента	Результат
1. Организация занятия. Сообщение темы и цели занятия	Организация группы на урок Мотивация учебной деятельности	Приветствие обучающихся, проверка присутствующих	Приветствие преподавателя, подготовка к уроку	Готовность группы к уроку
2. Проверка знаний и умений обучающихся	Актуализация знаний	Фронтальный опрос, вопросы преподаватель выводит на экран проектора, для того чтобы слабослышащие студенты могли участвовать в опросе. Вопросы для опроса: - Зачем нужно окно Project? - Зачем нужно окно Hierarchy?	Воспроизведение знаний при ответах на вопросы преподавателя. Слабослышащие студенты ответы набирают в текстовом редакторе на ПК.	Актуализация знаний по работе в кроссплатформенной среде разработки компьютерных игр Unity. Развитие умения грамотно, четко и точно

		<ul style="list-style-type: none"> - Что такое компонент, в каком окне можно добавить компонент объекту? - Как изменить свойства компонента Transform для объекта? - Как добавить объекту свойства физического тела? - Как создать скрипт в программе Unity? - В какой среде разработки редактируются скрипты? - Как прикрепить скрипт к объекту? 		<p>выражать свои мысли письменно или устно.</p> <p>Воспитание вежливости при общении с преподавателем и одноклассниками.</p>
<p>Метод- устный комбинированный опрос с демонстрацией вопросов на экране проектора</p> <p>Методический приём - ответы на вопросы устные либо письменные</p> <p>Форма обучения- групповая</p> <p>Средства обучения- ПК, проектор и экран, программа Unity.</p>				
3. Постановка задания для обучающихся	<p>Организовать группу на выполнение работы</p> <p>- прослушать новый материал по созданию</p>	<p>Преподаватель озвучивает тему и цель занятия: Добавление объектов (клонирование) на сцену программно.</p>	<p>Слушают задание. При необходимости задают вопросы для уточнения.</p>	<p>Группа готова к восприятию нового материала и выполнению практической работы.</p>

	префабов с целью добавления объектов (клонирование) на сцену программно.	Рассказывает ход занятия, демонстрируя его на экране проектора		
<p>Метод - информационно-сообщающий</p> <p>Методический приём - рассказ, сопровождаемый демонстрацией слайдов с темой и целью занятия и ходом работы.</p> <p>Форма обучения - индивидуально-групповая</p> <p>Средства обучения - ПК, проектор и экран</p>				
4. Изложение нового материала	Обеспечение восприятия, первичного запоминания знаний и способов деятельности, связей и отношений в объекте изучения: добавление объектов (клонирование) на сцену программно с помощью префабов.	Преподаватель рассказывает новый материал с применением проектора, используя презентацию. Пример создания префаба из объекта и клонирования его на сцену программно преподаватель показывает в программе Unity и интегрированной среде разработки (IDE) программного обеспечения Visual Studio. Новый материал: Префаб (Prefab) в Unity - это шаблон объекта, который может быть использован для создания	Восприятие и осознание знаний, их запоминание. Краткое конспектирование основных целей применения эффектов и приемов их создания.	Первичное осмысление новых знаний по теме: добавление объектов (клонирование) на сцену программно с помощью префабов. Развитие внимательности, умения сосредотачиваться на изучаемом материале. Воспитание положительного интереса к изучаемому предмету.

		<p>игровых объектов в сцене или в процессе игры. Префабы обычно используются для многократного использования ассетов, таких как 3D модели, материалы, анимации и скрипты.</p> <p>Система Prefab Unity позволяет создавать, настраивать и хранить Игровой объект (GameObject) в комплекте со всеми компонентами, значениями свойств и дочерними игровыми объектами как многоразовый объект. Prefab действует как шаблон, на основе которого вы можете создавать новые экземпляры Prefab на Сцене.</p> <p>Если нужно повторно использовать игровой объект, настроенный определенным образом (например, неигровой персонаж (NPC), реквизит или часть декорации), в</p>		
--	--	--	--	--

		<p>нескольких местах сцены или в нескольких сценах проекта, следует преобразовать его в Prefab. Это лучше, чем простое копирование и вставка GameObject, поскольку система Prefab позволяет автоматически синхронизировать все копии.</p> <p>Любые изменения, которые вносятся в префаб, автоматически отражаются в экземплярах этого префаба, что позволяет легко вносить значительные изменения во весь проект без необходимости повторно вносить одно и то же редактирование в каждую копию актива.</p> <p>Unity поддерживает иерархию префабов, что позволяет создавать сложные сцены с использованием иерархии родительских и дочерних объектов. Это обеспечивает</p>		
--	--	--	--	--

		<p>большую гибкость и контроль над структурой сцены и поведением объектов в ней.</p> <p>Можно вкладывать префабы в другие префабы, чтобы создавать сложные иерархии объектов, которые легко редактировать на нескольких уровнях.</p> <p>Однако это не означает, что все экземпляры Prefab должны быть идентичными. Можно переопределить настройки отдельных экземпляров префаба, если хотите, чтобы некоторые экземпляры префаба отличались от других. Также можно создавать варианты префабов, которые позволяют группировать набор переопределений в значимый вариант префаба.</p> <p>Можно использовать префабы, если хотите во время выполнения</p>		
--	--	---	--	--

		<p>создавать экземпляры игровых объектов, которые не существовали в сцене в начале - например, чтобы бонусы, специальные эффекты, снаряды или неигровые персонажи появлялись в нужные моменты во время игры.</p> <p>Некоторые распространенные примеры использования Prefab^</p> <ul style="list-style-type: none">• Экологические активы — например, определенный тип дерева, используемый несколько раз на уровне• Неигровые персонажи (NPC) — например, робот определенного типа может появляться в вашей игре несколько раз на разных уровнях. Они могут отличаться (с использованием <i>переопределений</i>) скоростью		
--	--	--	--	--

		<p>движения или звуком, который издают.</p> <ul style="list-style-type: none">• Снаряды — например, пиратская пушка может создавать экземпляр префаба пушечного ядра каждый раз при выстреле.• Главный герой игрока - префаб игрока может быть размещен в начальной точке каждого уровня (отдельных сцен) вашей игры. <p>Создание префаба:</p> <p>Создание префаба состоит из создания экземпляра объекта в сцене, настройки его свойств и сохранения его как префаба.</p> <p>Пример:</p> <ul style="list-style-type: none">• Загрузите новый ассет изображение цветка (Assets – Import New Asset) в папку Sprites. Назовите его Cvetok.		
--	--	---	--	--

		<ul style="list-style-type: none">• Создайте папку Prefabs в папке Assets• Теперь из цветка нужно сделать префаб. Для этого из окна иерархии перетащите объект Svetok в папку Prefabs.• Удалите со сцены цветок <p>Создание экземпляра префаба программно:</p> <p>Чтобы создать экземпляр префаба во время выполнения скрипта, коду нужна ссылка на этот префаб. Можно сделать эту ссылку, создав в своем коде общедоступную (public) переменную для хранения ссылки на Prefab. Открытая (public) переменная в коде отображается как назначаемое поле в Инспекторе. Затем нужно назначить в Инспекторе фактический префаб, который должен использоваться.</p>		
--	--	---	--	--

		<p>Пример:</p> <p>Создадим экземпляр префаба цветка на сцене программно. Для этого создадим скрипт, назовем его Svetki. Откроем скрипт и в нем объявим переменную Svetok</p> <pre>using UnityEngine; public class InstantiationExample : MonoBehaviour { public GameObject Svetok;</pre> <p>Сохраним скрипт и вернемся опять в Unity</p> <p>В окно переменной Svetok, там где отражается добавленный скрипт, перетащим пиктограмму префаба Svetok из папки префабов.</p> <p>Для создания экземпляра (клона) префаба в языке C# используется метод</p>		
--	--	---	--	--

		<p>Instantiate (исходный объект, позиция Vector3, вращение кватерниона);</p> <p>аргументами которого являются название префаба, его координаты на сцене и угол вращения на сцене. В приведенном примере сценария есть одна общедоступная переменная «Cvetok» , которая является ссылкой на Prefab. Он создает экземпляр этого префаба в методе Start().</p> <pre>using UnityEngine; public class InstantiationExample : MonoBehaviour { public GameObject Cvetok; void Start() { Instantiate (Cvetok, new Vector3(0, 0, 0), Quaternion.identity); } }</pre>		
--	--	---	--	--

		<p>Редактирование префаба из его экземпляров</p> <p>Инспектор экземпляра префаба содержит три кнопки, которые у обычных объектов отсутствуют: <i>Select</i> (Выделить), <i>Open</i> (Открыть) и <i>Overrides</i> (Переопределить).</p> <p>Кнопка <i>Select</i> выделяет файл префаба, из которого был получен данный экземпляр. Это позволяет быстро найти префаб.</p> <p>Кнопка <i>Open</i> позволяет открыть и отредактировать оригинальный префаб, применяя изменения ко всем его экземплярам.</p> <p>Также вы можете сохранить переопределённые свойства из экземпляра в сам оригинальный префаб с помощью кнопки <i>Overrides</i> . Это позволяет эффективно редактировать все экземпляры через любой из них, и</p>		
--	--	--	--	--

		это быстрый и правильный способ вносить глобальные изменения.		
<p>Метод - Объяснительно-иллюстративный</p> <p>Методический приём - рассказ преподавателя с демонстрацией изучаемого материала на экране проектора.</p> <p>Форма обучения - групповая</p> <p>Средства обучения - ПК, проектор и экран, слайды презентации, программа Unity и IDE Visual Studio</p>				
5. Инструктаж	Обеспечить правильное выполнение практической работы	<p>Объясняет порядок выполнения практической работы. На экране проектора демонстрирует текстовые инструкции практической работы и видеоматериалы, которыми надо будет воспользоваться при выполнении практического задания.</p> <p>При работе разрешается задавать вопросы преподавателю и товарищам.</p>	Слушают преподавателя, наблюдают демонстрируемое преподавателем методическое обеспечение.	<p>Группа готова к восприятию инструктажа по выполнению практической работы.</p> <p>Создаются условия для развития умений и навыков работы по инструкциям, использования видеоматериалов для получения информации.</p> <p>Обеспечиваются условия для развития творческого отношения к избранной профессии.</p>

<p>Метод - информационно-сообщающий</p> <p>Методический приём - рассказ, сопровождаемый демонстрацией размещения методических материалов и видеоматериалов.</p> <p>Форма обучения – групповая.</p> <p>Средства обучения - ПК, проектор и экран, файл практической работы.</p>				
<p>6. Самостоятельная работа обучающихся и текущий инструктаж</p>	<p>Формирование умений по добавлению объектов (клонированию) на сцену программно с помощью префабов.</p>	<p>Спрашивает, все ли получается при выполнении заданий, отвечает на вопросы, помогает в выполнении заданий.</p> <p>Для диалога со слабослышащими студентами преподаватель может использовать любой текстовый редактор.</p>	<p>Выполняют задания в программе Unity и IDE Visual Studio по инструкции.</p>	<p>Формирование умений по добавлению объектов (клонированию) на сцену программно с помощью префабов, с целью применения полученных навыков в профессиональной деятельности при разработке компьютерных игр.</p> <p>Продолжить формирование умения студентов пользоваться интерфейсом программы Unity на иностранном (английском) языке.</p>

				<p>Развитие у обучающихся аналитического и алгоритмического мышления.</p> <p>Формирование навыка самостоятельной учебной деятельности.</p> <p>Формирование навыка работы в коллективе.</p>
<p>Метод – репродуктивный</p> <p>Методический приём - выполнение задания по инструкции</p> <p>Форма обучения - индивидуальная</p> <p>Средства обучения – ПК, программа Unity и IDE Visual Studio, текстовый редактор.</p>				
7. Итоговый контроль выполнения работы	Проверка правильности выполнения заданий по созданию клонов префаба на сцене.	Подходит к студентам, смотрит работы. При необходимости задает вопросы по ходу выполнения работы. Для диалога со слабослышащими студентами преподаватель может использовать любой текстовый редактор на ПК.	Студенты отвечают на вопросы преподавателя. Слабослышащие студенты ответы набирают в текстовом редакторе на ПК.	Осмысление полученных знаний и навыков по созданию клонов префаба на сцене.
<p>Метод- практическая проверка</p> <p>Форма обучения- индивидуальная</p>				

<p>8. Домашнее задание</p>	<p>Повторение изученного материала по теме «Добавление объектов (клонирование) на сцену программно».</p>	<p>Повторить материал по конспекту и подготовиться к фронтальному опросу. Вопросы выводятся на экран проектора. Примерные вопросы для опроса:</p> <ul style="list-style-type: none"> - Что такое Prefab? - Расскажите алгоритм создания префаба из объекта? - Как создать сценарий в программе Unity? - Какой метод применяется для клонирования объектов из префаба на сцене? - Как объявить общедоступную переменную в скрипте? - Как прикрепить фактический префаб к объявленной в скрипте общедоступной переменной в среде разработке Unity? 	<p>Воспринимают информацию, заносят задание в тетрадь.</p>	<p>Обучающиеся ознакомлены с домашним заданием. Формирование ответственности и дисциплинированности при подготовке к следующему занятию.</p>
----------------------------	--	--	--	--