Учебный план:

Unity для начинающих

1. Введение в курс, установка среды, обьяснение условий пользования движком Unity3D. Регистрация в Unity
   * 1. -Unity Hub, Unity3D
     2. -Visual Studio, Visual Studio Code, MonoDevelop(На выбор)(примерно 15 мин)
2. Знакомство с интерфейсом редактора Unity3D. Настройка редактора. Создание первого проекта. Создание первого обьекта из примитивов.(30 мин теория 10 мин практика)
3. Понятие «Префаб», создание нескольких сложных обьектов из примитивов. (30 мин теория, 1-4 часа практика)
4. VS Code и первый скрипт(25 мин теория, 10 мин практика)
5. Подготовка проекта «Терминал»(15-20 мин практика)
6. Скрипт управления терминалом: Termninal.Writeline()(15 мин теория)
7. Функции(методы).(10 мин теория, 10 мин практика)
8. Переменные (10 мин теория, 10 мин практика)
9. Параметры функций(10 мин теория, 20-30 мин практика)
10. Input данные, метод OnUserInput()(10 мин теория, 10 мин практика)
11. Операторы сравнения +=, ==, <, <=, >, >=, !=. Логические операторы !, !=, &&, ||, ^(30 мин теория, 1 час практика)
12. Условные операторы If, else, тернар(15 мин теория, 40 мин практика)
13. Переменные и их использование(INT)(10 мин теория, 10 мин практика)
14. ENUM(10 мин теория, 10-20 мин практика)
15. Структурирование кода(10 мин теория, 40 мин практика)
16. Самостоятельная работа(40-60 мин практика)
17. Массивы(10 мин теория, 40 мин практика)
18. Switch(25 мин теория, 1 час практика)
19. Random.Range()(15 мин теория,10-20 мин практика)
20. Циклы for, while, do while, foreach. Операторы continue, break(40 мин теория, 20-30 мин практика)
21. Экран победы, ASCII(20 мин теория, 30-40 мин практика)
22. Подсказка для пароля, функция Anagram(30 мин теория, 40-60 мин практика)
23. Сборка игры(10 мин теория, 10 мин практика)
24. Контрольная работа и итоги()(1-4 часа практика)

Unity Стандартный

1. Создание нового проекта, ракеты и местности из примитивов, материалы(30 мин теория, 30-40 мин практика)
2. Управление клавиатурой, Input.GetKey()(15 мин теория, 10-20 мин практика)
3. RigidBody, AddForce, transform.Rotate управление ракетой(30 мин теория, 1-час практика)
4. AudioSource(15 мин теория, 15-20 мин практика)
5. Ограничение перемещения ракеты по плоскости(10 мин теория, 20-30 мин практика)
6. Time.deltaTime фикс скорости ракеты(15 мин теория, 5 мин практика)
7. Tags, Collision(15 мин теория, 10-20 мин практика)
8. Создание первого уровня(15 мин теория, 30 мин практика)
9. Создание второго уровня(10 мин теория, 30 мин практика)
10. Подробнее о префабах(20 мин теория)
11. SceneManager(10 мин теория, 10 -20 мин практика)
12. Invoke(), задержка в скрипте(20 мин теория, 20 мин практика)
13. Больше звуков для игры(15 мин теория, 40-60 мин практика)
14. ParticleSystem – эффекты(20 мин теория, 40-60 мин практика)
15. Движение обьектов через скрипт(20 мин теория, 40-60 мин практика)
16. Mathf.PinPong()(15 мин теория, 30-60 мин практика)
17. Освещение в Unity3d(30 мин теория, 30 мин практика)
18. Разнообразие в геймплее(15 теория,???? – пересмотреть время для практики, возможно не нужно)
19. Debug методы – читы(15 мин теория, 20-30 мин практика)
20. Переходы между уровнями(20 мин теория, 10-20 мин практика)
21. Главное меню(20 мин теория, 30-40 мин практика)
22. Показатель энергии ракеты и батарейки на уровнях(40 мин теория,40-60 мин практика)
23. Контрольная работа и итоги(1-4 часа практика)

**Unity Продвинутый**

1. Создание нового проекта, подробнее о системе версий Unity(10 мин теория)
2. Создание ландшафта (Terrain)(10 мин теория, 20 мин практика)
3. Добавление текстур к ландшафту(15 мин теория,20-40 мин практика)
4. Skybox настройка(10-15 мин теория, 10-20 мин практика)
5. Оптимизация Terrain. Модель космического корабля(15 мин теория, 20 – 40 мин практика)
6. Splash Screen, фоновая музыка(10 мин теория, 20 – 40 мин практика)
7. Метод DontDestroyOnLoad(10 мин теория, 5-10 мин практика)
8. Waypoints, Standart Assets(15 мин теория, 40-60 мин практика)
9. CrossPlatformInput(20 мин теория, 10-20 мин практика)
10. Движение по Оси X(15 мин теория)
11. Перемещение корабля по горизонтали и вертикали, ограничение движения(20 мин теория, 40-60 мин практика)
12. Вращение localRotation(15 мин теория)
13. Вращение корабля относительно перемещения(25 мин теория, 40-60 мин практика)
14. Улучшение геймплея(10 мин теория, 20-30 мин практика)
15. Выстрел, взрыв – Particle System(35 мин теория, 30-60 мин практика)
16. Добавляем препятствия(15 мин теория, 15-40 мин практика)
17. Приводим проект в порядок(10 мин теория, 15-30 мин практика)
18. Триггеры, коллайдеры и из взаимодействие(20 мин теория, 20-40 мин практика)
19. SendMessage(), вызов метода в скрипте(20 мин теория, 10-20 мин практика)
20. GameOver – взрыв корабля(15 мин теория, 10-30 мин практика)
21. Удаляем дубликаты объекта Music(10 мин теория)
22. Взаимодействие частиц с коллайдерами(15 мин теория,10-25 мин практика)
23. Addcomponent() – добавление компонента через скрипт(10 мин теория, 20-60 мин практика)
24. Уничтожение врагов(20 мин теория,20-40 мин практика)
25. UI – показатель количества очков(15 мин теория, 10-30 мин практика)
26. Задаем UI текст через скрипт(15 мин теория, 10-20 мин практика)
27. Добавляем очки за уничтожение врага(15 мин теория, 10 -20 мин практика)
28. Выстрел корабля по нажатию на кнопку(15 мин теория, 10 -20 мин практика)
29. Очки жизни для врагов(10 мин теория, 10 - 30 мин практика)
30. Timeline (1 час теория,1-2 часа практика)
31. Визуальное улучшение игры(20 мин теория,10-20 мин практика)
32. Контрольная работа и итоги. (1-4 часа практика, 40 мин теория)
33. Подробнее про Asset Store, FPS, TPS, Racing(30 мин теория, 1-1.30 час практика)

Домашние задания:

Unity для начинающих

1. Установить необходимую среду разработки

2. Расставить вкладки Hierarchy, Inspector, Scene, Game, Console, Project как будет удобно

3. Создайте 5 объектов из примитивов одной тематики и поместите их в префабы.

Примеры тематик: Объекты на улице, Оружие, Транспорт, Мебель, Компьютер и другие тех. устройства (принтер, телевизор и тп.)

Создать сцену, например: городское шоссе.

4. Создайте скрипт, который будет выводить в консоль анекдот или полезную информацию при каждом запуске игры.

6. Создайте скрипт с названием, "terminal", сохраните его. Переименуйте его в "terminalHacker".Исправьте ошибку в консоле путём редактирования кода внутри скрипта.

7. Создайте функцию, которая будет показывать в консоле имя автора, при запуске игры.

8. Создайте скрипт, который будет выводить сообщение в консоль для вас, с использованием переменной типа string.

9. Как в уроке создайте метод, который будет принимать ДВА параметр, вместо одного. Используйте оба параметра в коде на свое усмотрение.

Подсказка, параметры перечисляйте через запятую

Пример, void NewFunc(string a, string b){}

10. Выведите на экран и в консоль разработчика, введенную строку.

11. Выведите в консоль, результат выражения: 118+8920-(8952/888)+896-7893; Сравните два любых выражения, получите значения true(1), false(0); (добавить задания с остальными логическими операторами)

12.С помощью операторов if, else сравните выражение и в зависимости от результата выведите в терминал «Выражения совпадают» или «Выражения не совпадают». Добавьте пасхалку.

13. Создайте переменную с любым значением типа INT, и выведите ее значение при помощи if и Terminal.WriteLine;

15. Сделайте свой код читабельным и менее громоздким, обязательно использование комментариев

17. Создайте массив для паролей с большим диапазоном, и используйте его.

18. Замените, где это возможно If, else на switch, case.

19. Создайте переменную, которой будет рандомно генерироваться значение от 0 до 10, при каждом новом запуске игры, если число окажется 7 или 9 вывести «Да ты сегодня в ударе».

20. При помощи цикла выведите каждое значение из любого массива паролей.

22. Создайте еще один экран игры, который будет называться "секретное меню",

Если игрок в экране меню вводит слово "valve", покажите эму секретное меню.

Секретное меню должно содержать предложение зашифрованное предложение "The Cake Is a Lie", причем зашифрованными должны быть только слова в предложении, пробелы НЕ шифровать.

Если игрок вводит пароль правильно, покажите эму тортик:

\_!\_!\_!\_!\_

| || ||

| || |||

}{{{{}}}{{{

\_\_||\_\_

Игрок в любой момент может выйти из секретного меню, написав "меню"

Также можете придумать свою задачку для секретного меню. Это просто пример

24. Создать на базе проекта «Терминал» свою игру, это может быть своя угадайка, или же текстовый квест. В итоговой работе должны быть: 2 секретных уровня, 3 пасхалки.

Unity Стандартный

1. Создайте новый проект, из примитивов постройте посадочную платформу (используйте больше одного примитива)
2. Создайте ракету из примитивов
3. Подберите нужные параметры для RelativeForce, Mass
4. С помощью компонента Rigidbody, сделайте так, чтобы ракета вела себя как в невесомости
5. Создайте куб, создайте для него скрипт, который по нажатию E – будет вращать куб по оси X, по нажатию на R – будет вращать куб по оси Y, по нажатию на T – будет вращать куб по оси Z.
6. Найдите звук, который будет проигрываться сразу после запуска игры
7. Задайте обьектам уникальные теги
8. Создайте первый уровень
9. Создайте второй уровень, поместите на уровне несколько дружественных посадочных платформ, которые не будут уничтожать ракету. Игрок сможет приземлиться на них, по пути к финишу
10. --------------------------------------------------------------------------------------------------------------------------------
11. Подумайте над тем, как можно переключаться между несколькими уровнями без костылей.
12. С помощью метода Invoke(), выведите в консоль обратный отсчет, добавьте звук обратного отсчета
13. Добавьте звук проигрыша, который будет проигрываться после взрыва ракеты. Задайте задержку между сменой уровня, чтобы звук смог проиграться до конца
14. Найдите другие бесплатные эффекты в Asset Store, и используйте их для вашей игры.
15. --------------------------------------------------------------------------------------------------------------------------------
16. Создайте уровень, где будут двигающиеся препятствия
17. Поэкспериментируйте с освещением, создайте уникальный уровень с необычным освещением, или освещением как геймплейного элемента
18. Создайте несколько уровней с разными геймплейными особенностями
19. Добавьте звук который будет проигрываться при включении Debug – метода, а также вывод в консоль состояния читов(вкл/выкл)
20. --------------------------------------------------------------------------------------------------------------------------------
21. Создайте свое главное меню с интересным дизайном
22. Добавьте показатель энергии и батарейки для остальных уровней. Исправьте код таким образом, чтобы использовать батарейки в сцене было необязательно. Сейчас при запуске уровня на котором нет батареек и UI в консоли вызывается ошибка – Null references
23. Закончите игру с ракетой, как минимум в игре должно быть 7 уровней, главное меню, экран проигрыша, Debug – методы должны быть отключены (НЕЛЬЗЯ УДАЛЯТЬ КОД ИЗ СКРИПТА ИЛИ ЕГО КОММЕНТИРОВАТЬ)

Unity Продвинутый

1. Создайте новый проект, обновите UNITY
2. Создайте свою горную, равнинную местности, на выбор
3. Найдите подходящую текстуру для местности, и раскрасьте ее в разные цвета
4. Найдите в Asset store подходящий скайбокс и примените его
5. Оптимизируйте местность, найдите для себя модель космического корабля
6. Найдите музыку и сделайте свой экран приветствия
7. --------------------------------------------------------------------------------------------------------------------------------
8. Задайте камере маршрут по местности
9. Задайте управление в игре
10. --------------------------------------------------------------------------------------------------------------------------------
11. Сделайте управление кораблем через скрипт
12. Добавьте кораблю вращение
13. --------------------------------------------------------------------------------------------------------------------------------
14. --------------------------------------------------------------------------------------------------------------------------------
15. Создайте свои эффекты взрывов и выстрелов
16. Добавьте много препятствий, проверьте игру на сложность
17. --------------------------------------------------------------------------------------------------------------------------------
18. Добавьте триггеры на уровень
19. --------------------------------------------------------------------------------------------------------------------------------
20. Добавьте экран проигрыша в игру
21. --------------------------------------------------------------------------------------------------------------------------------
22. Добавьте взаимодействия частиц с обьектами
23. Добавьте несколько препятствий через Addcomponent
24. Сделайте врагов убиваемыми
25. Добавьте свой показатель очков в игру
26. --------------------------------------------------------------------------------------------------------------------------------
27. --------------------------------------------------------------------------------------------------------------------------------
28. Сделайте корабль стреляющим
29. Добавьте очки жизни для врагов, сделайте врагов разной сложности по уровню здоровья
30. Если возможно добавьте анимацию врагу, сделайте анимацию взлета корабля игрока
31. --------------------------------------------------------------------------------------------------------------------------------
32. Сделайте второй уровень и экран выигрыша

Журнал