CREATE GAMES !

Welcome to issue 9.

Welcome to a new issue of Create Games. From this issue on I will not go into much details of explaining all scripts. You should be able to use them with the knowledge you got of scripting by going through the previous issues. When you download the assets of this issue you will get the full open resource project that you can open in Unity. This way you can start right away and change/add things to your own likings. If you're new to Unity and C-Script I advise you to read and experiment with all previous issues.

In this Issue we will create a basic setup game of Plants Vs Zombies. You can see this all in action right here : https://youtu.be/87zvaK92yJ4

I will tell you what each script does so you will understands this example game mechanics. I am sure things could be scripted better but hey it works so go nuts and make it better when needed.

GameManager Script.

This script uses UI elements so make sure to insert the UnityEngine.UI line. The script holds the spawn function that spawns the fallen sun objects at random places every 10 seconds. The score is shown in the UI text element SunscoreT.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class GameManager : MonoBehaviour
{
/// Plants Vs Zombies Create Games Issue December 2021////
/// René Pol 01-12-2021///

public float SunScore;
public Text SunscoreT;
public GameObject Suns;
public float SpawnTime = 5f;
public int WhatToBuilt;
public int Build;

public void SpawnSuns()
{
Instantiate(Suns, new Vector3(-7.5f + Random.Range(1, 14), 9.15f, 0),
Quaternion.identity);
}
```

```
void Update()
{
SpawnTime -= Time.deltaTime;
if (SpawnTime <= 0)
{
SpawnSuns();
SpawnTime = 10f;
}
SunscoreT.text = SunScore.ToString("0");
}
}
```

Falling Sun Script.

This object has a circle 2D Collider so it can be clicked on with the mouse. It is spawned from the GameManager script and depending on the random picked Droprow number it lands on the same row of grass. When not picked up it will remove itself. If clicked on it will move to the Vector2 given in the inspector’s view and adds 25 points to the score that is made in the GameManager script. After reaching the Vector2 it will be removed.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class FallingSun : MonoBehaviour
{

/// Plants Vs Zombies Create Games Issue December 2021////
/// René Pol 01-12-2021///

public AudioClip Sunscoring;
public int DropRow;
public float Speed = 1;
public float Speed2 = 5;
public bool MoveToScore;
public Vector2 TargetLocation;
private GameManager GM;
public bool PlaySound;

void Start()
{
GM = FindObjectOfType<GameManager>();
MoveToScore = false;
DropRow = Random.Range(0, 5);
PlaySound = true;
}

public void OnMouseDown()
{
MoveToScore = true;
}
```

```
IEnumerator Holding()
{
yield return new WaitForSeconds(5f);
if (MoveToScore == false)
{
Destroy(this.gameObject);
}
}

void Update()
{
if (DropRow == 0 && transform.position.y > 3.5f)
{
transform.Translate(Vector2.down * Speed * Time.deltaTime);
}
if (DropRow == 0 && transform.position.y < 3.5f)
{
StartCoroutine(Holding());
}
if (DropRow == 1 && transform.position.y > 1.4f)
{
transform.Translate(Vector2.down * Speed * Time.deltaTime);
}
if (DropRow == 1 && transform.position.y < 1.4f)
{
StartCoroutine(Holding());
}
if (DropRow == 2 && transform.position.y > -0.6f)
{
transform.Translate(Vector2.down * Speed * Time.deltaTime);
}
if (DropRow == 2 && transform.position.y < -0.6f)
{
StartCoroutine(Holding());
}
if (DropRow == 3 && transform.position.y > -2.8)
{
transform.Translate(Vector2.down * Speed * Time.deltaTime);
}
if (DropRow == 3 && transform.position.y < -2.8f)
{
StartCoroutine(Holding());
}
if (DropRow == 4 && transform.position.y > -4.9)
{
transform.Translate(Vector2.down * Speed * Time.deltaTime);
}
if (DropRow == 4 && transform.position.y < -4.9f)
{
StartCoroutine(Holding());
}
if (MoveToScore == true)
{
if (PlaySound == true)
{
AudioSource.PlayClipAtPoint(Sunscoring, new Vector3(0, 0, -10));
PlaySound = false;
}
```

```
transform.position = Vector2.MoveTowards(transform.position, TargetLocation,
Speed2 * Time.deltaTime);
DropRow = 10;
StopCoroutine(Holding());
if (transform.position.x == TargetLocation.x && transform.position.y ==
TargetLocation.y)
{
GM.SunScore += 25;
Destroy(this.gameObject);
}
}
}
}
```

Rotate Script.

This object is a child object of the falling sun. It rotates and stays with the fallen sun game object.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Rotate : MonoBehaviour
{
/// Plants Vs Zombies Create Games Issue December 2021////
/// René Pol 01-12-2021///

public float Speed = 2;

void Update()
{
transform.Rotate(0, 0, 20 * Speed * Time.deltaTime);
}
}
```

GrassTile Script.

The tile object has a box 2d collider so it can be clicked on with the mouse. When the Spot is free bool variable is true it will create the plant number set in the game manager WhatToBuilt. The cost of the plant to build (100) Is subtracted of the score. A rebuild function is called from the script BP. (BuiltPeaplant) Every kind of plant could get a script similar to this one.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Grasstile : MonoBehaviour
{
/// Plants Vs Zombies Create Games Issue December 2021////
/// René Pol 01-12-2021///

public GameObject Peaplant;
```

```
public bool SpotIsFree;
public AudioClip planted;


private GameManager GM;
private BuiltPeaplant BP;

void Start()
{
GM = FindObjectOfType<GameManager>();
BP = FindObjectOfType<BuiltPeaplant>();
SpotIsFree = true;
}

public void OnMouseDown()
{
if (SpotIsFree == true && GM.SunScore >= 100 && GM.WhatToBuilt == 1)
{
BP.IsSelected = false;
AudioSource.PlayClipAtPoint(planted, new Vector3(0, 0, -10));
Instantiate(Peaplant, transform.position, Quaternion.identity);
SpotIsFree = false;
GM.SunScore -= 100;
GM.WhatToBuilt = 0;
BP.Rebuildplant();
}
}
}
```

BuildPeaplant script

The building image is an UI element so don’t forget the UnityEngine.UI line. When the rebuildplant function is called it will use the fill amount to make the picture disappear vertical. If it’s gone and the score is enough the plant can be selected. Once selected it sets the number in the GameManager (Whattobuild) and now each empty tile will build this plant once clicked on. After placement the rebuildplant function is called again.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;


public class BuiltPeaplant : MonoBehaviour
{
/// Plants Vs Zombies Create Games Issue December 2021////
/// René Pol 01-12-2021///

public Image BuildingBar;
public float CurrentBuild;
public float Maxbuild = 100f;
public bool firstReveal;

public GameObject Selector;
private GameManager GM;
public bool IsSelected;
public AudioClip Selecting;
```

```
public bool rebuilding;


void Start()
{
rebuilding = false;
firstReveal = true;
IsSelected = false;
GM = FindObjectOfType<GameManager>();
Selector.SetActive(false);
}

public void Rebuildplant()
{
rebuilding = true;
CurrentBuild = 100;
}

public void Selected()
{
if (IsSelected == false && GM.SunScore >= 100 && rebuilding == false)
{
GM.WhatToBuilt = 1;
AudioSource.PlayClipAtPoint(Selecting, new Vector3(0, 0, -10));
Selector.SetActive(true);
IsSelected = true;
return;
}
if (IsSelected == true && GM.SunScore >= 100 && rebuilding == false)
{
GM.WhatToBuilt = 0;
Selector.SetActive(false);
IsSelected = false;
return;
}
}


void Update()
{
if (rebuilding == true)
{
CurrentBuild -= 20 * Time.deltaTime;
if (CurrentBuild <= 0)
{
rebuilding = false;
}
}
if (firstReveal == true && GM.SunScore >= 100)
{
CurrentBuild = 0;
firstReveal = false;
}
BuildingBar.fillAmount = CurrentBuild / Maxbuild;
if (IsSelected == false)
{
Selector.SetActive(false);
}
```

```
}
}
```

Peabullet Script.

This is a simple script that makes the pea bullet object move to the right once it is created. When the position on the x-axis is bigger than 10 it will be removed. For collision it has a circle 2D collider. The game object has the tag PeaBullet.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PeaBullet : MonoBehaviour
{

/// Plants Vs Zombies Create Games Issue December 2021////
/// René Pol 01-12-2021///

public float Speed = 8f;
public AudioClip peashot;

void Start()
{
AudioSource.PlayClipAtPoint(peashot, new Vector3(0, 0, -10));
}

void Update()
{
transform.Translate(Vector2.right * Speed * Time.deltaTime);
if (transform.position.x > 10)
{
Destroy(this.gameObject);
}
}
}
```

Peaplant Script.

Once this plant is placed it will spawn peas every 2 seconds. There created from the new Vector3 position given. The pea and it’s script is above this one.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PeaPlant : MonoBehaviour
{
/// Plants Vs Zombies Create Games Issue December 2021////
/// René Pol 01-12-2021///

public GameObject pea;
public float spawntime = 2f;
```

```
public void spawnPea()
{
Instantiate(pea, new Vector3(transform.position.x + 0.6f, transform.position.y +
0.5f, 0), Quaternion.identity);
}

void Update()
{
spawntime -= Time.deltaTime;
if (spawntime <= 0)
{
spawnPea();
spawntime = 2f;
}
}
}
```

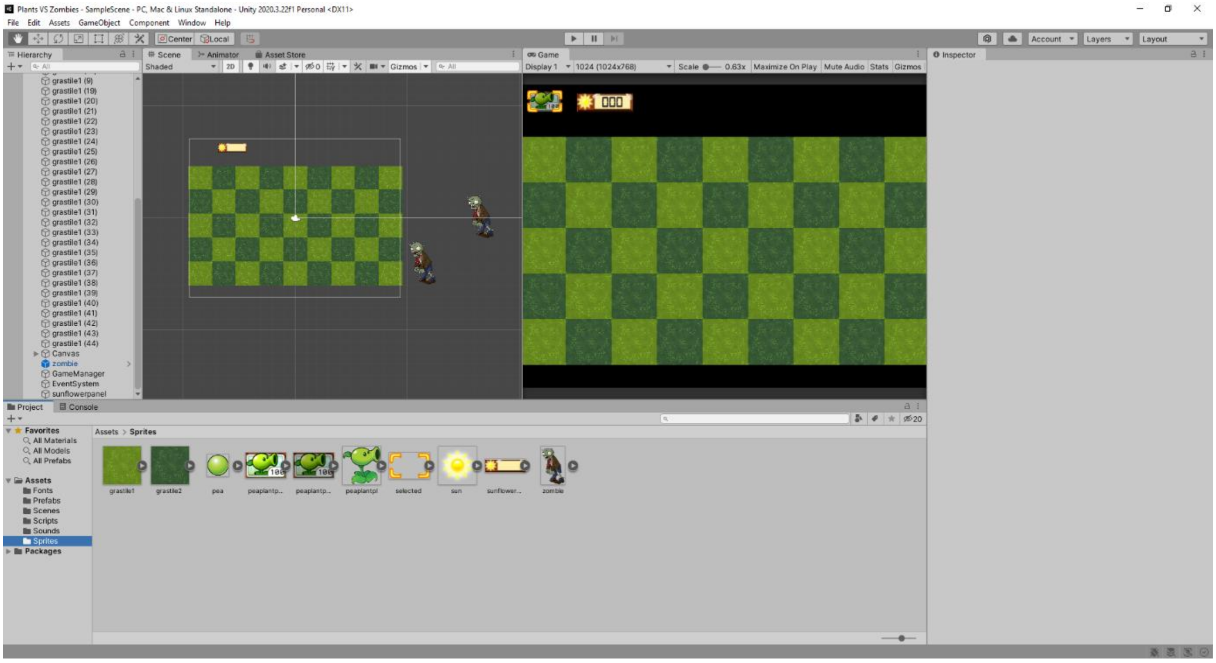
ZombieWalk Script.

For collision to work the zombie object has a box 2D collider and a Rigidbody2D collider with the gravity set to 0. It moves slowly to the left. When hit by an object with the tag name PeaBullet 1 point is subtracted of the given health. When the health reaches zero the object removes itself.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Zombiewalk : MonoBehaviour
{
/// Plants Vs Zombies Create Games Issue December 2021////
/// René Pol 01-12-2021///

public AudioClip hit;
public float walkingspeed = 0.2f;
public int Health = 10;

private void OnTriggerEnter2D(Collider2D collision)
{
if (collision.gameObject.tag == ("PeaBullet"))
{
AudioSource.PlayClipAtPoint(hit, new Vector3(0, 0, -10));
Destroy(collision.gameObject);
Health -= 1;
}
}

void Update()
{
transform.Translate(Vector2.left * walkingspeed * Time.deltaTime);
if (Health <= 0)
{
Destroy(this.gameObject);
}
```

```
    }
}
```

All of this is a basic Plant vs Zombie game. You can add new plants or change the complete theme of the game to your likings. These scripts are given a good start so happy game making !



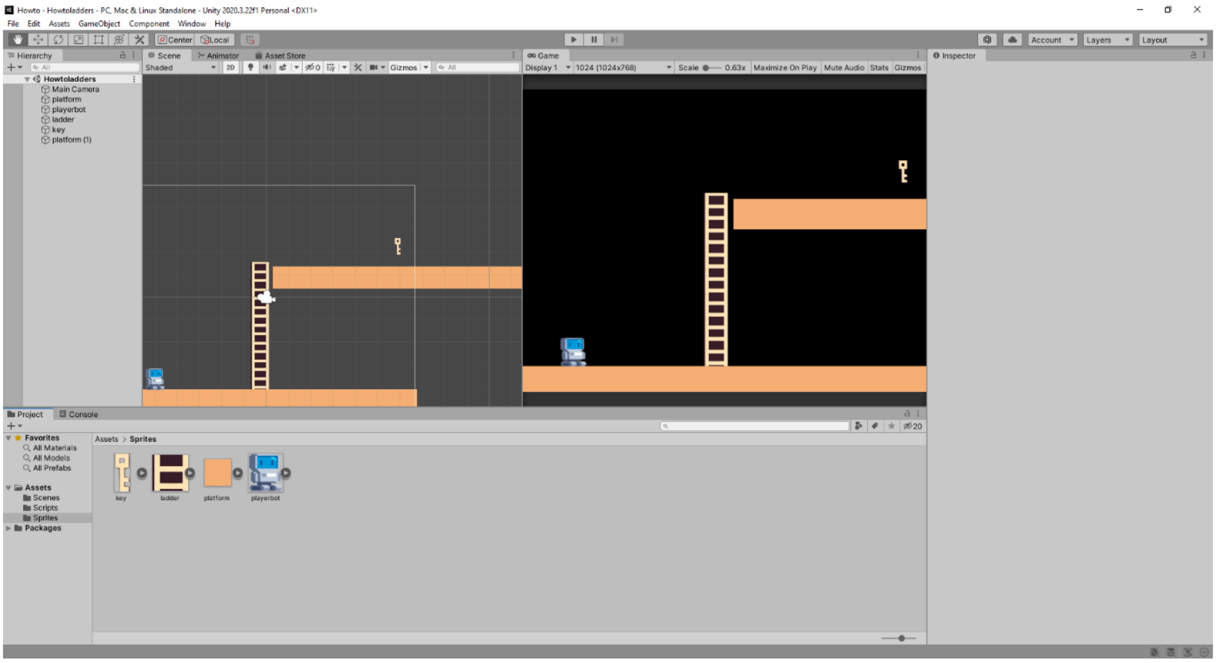
HOW TO

The next script is a player movement script that holds also the script to use in game 2D Ladders to move up and down on without any problems. It's a basic script that should be easy to adapt into your own likings.

The ladder game object should have a matching Box Collider 2D. It's tag name should be Ladder so the player can recognize it when coming close to it. The box is set to be a Trigger. When the player is in range and only then it can use the vertical movement and the gravity is set to 0. When out of range it will no longer use the vertical movement and the gravity is set in this example to 20 So it falls down till it hits the ground or platform. The player checks to see if the ladder holds the tag name Ladder so it knows how to respond. You can see this all working at : https://youtu.be/hkF2cp-a_Gs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PlayerMovement : MonoBehaviour
{
/// How to do Ladders Issue December 2021////
/// René Pol 01-12-2021///


Rigidbody2D body;
float vertical;
float horizontal;

public bool IsLadder;
public bool IsClimbing;
public bool IsFacingRight;
public float runSpeed = 5.0f;
private Rigidbody2D rb;

void Start()
{
rb = GetComponent<Rigidbody2D>();
}

public void Flip()
{
IsFacingRight = !IsFacingRight;
transform.Rotate(0f, 180f, 0f);
}

void Update()
{
if (IsFacingRight && horizontal > 0)
{
Flip();
}
else if (horizontal < 0 && !IsFacingRight)
{
Flip();
}

rb.velocity = new Vector2(horizontal * runSpeed, 0 * runSpeed);
horizontal = Input.GetAxisRaw("Horizontal");
vertical = Input.GetAxis("Vertical");
if (IsLadder == true && Mathf.Abs(vertical) > 0f)
{
IsClimbing = true;
}
}
```

```
public void OnTriggerEnter2D(Collider2D collision)
{
if (collision.CompareTag("Key"))
{
Destroy(collision.gameObject);
}
if (collision.CompareTag("Ladder"))
{
IsLadder = true;
}
}

public void OnTriggerExit2D(Collider2D collision)
{
if (collision.CompareTag("Ladder"))
{
IsLadder = false;
IsClimbing = false;
}
}

private void FixedUpdate()
{
if (IsClimbing)
{

rb.gravityScale = 0f;
rb.velocity = new Vector2(rb.velocity.x, vertical * runSpeed);
}
else
{
rb.gravityScale = 20f;
}
}
}
```


Howto - Howtoladders - PC, Mac & Linux Standalone - Unity 2020.3.22f1 Personal <DX11>
File Edit Assets GameObject Component Window Help
Center Local
Account Layers Layout
Hierarchy
Howtoladders
Main Camera
platform
playerbot
ladder
key
platform (1)
Scene Animator Asset Store
Shaded 2D Gizmos
Game
Display 1 1024 (1024x768) Scale 0.63x Maximize On Play Mute Audio Stats Gizmos
Inspector
Project Console
Favorites
All Materials
All Models
All Prefabs
Assets
Scenes
Scripts
Sprites
Packages
Assets > Sprites
key
ladder
platform
playerbot