# Unity 3D Practicals

**System Requirements for Unity version 2020.3.12**

OS- Windows 7 SP1+, 8, 10, macOS 10.11+

GPU- Graphics Card with DX9 (shader model 3.0) or DX11 with feature level 9.3 capabilities

Install Unity Hub. Create your account on Unity Hub

Go to Unity Hub Click on Installs then click on Add and select the Unity version under Recommended release

After Installation is complete Close Unity Hub

**Starting Unity Practicals**

Start Menu-> Unity-> Unity **2020.3.12f1 (64 bit) ->Unity**

Unity Hub opens

Go to Projects and click on New

# 2D UFO Tutorial

Project Name: 2D UFO Game

Template : 2D

Click on Create Project

Window -> Asset Store or press Ctrl + 9

Search for 2D UFO Tutorial

Click on Download, Click Accept, Click Import, Select all assets, click Import

File-> Save Scene As

Go To 2D UFO Game->Assets->Scenes folder

Give the name Main and Save

**Step 1:- Setting Up the Play Field**

1. In the Project View Assets->Sprites-> select Background and drag into the hierarchy of the left
2. Select the background Game object in the hierarchy and in the Inspector window set the name as Background
3. Set Sorting layer as Background
4. Move the cursor to the scene in the Scene View and press 'F' to see entire game object
5. Drag Assets->Sprites->UFO in the hierarchy
6. Select UFO and in the Inspector window set the name as Player
7. Set Sorting layer as Player
8. Click on Game View to view what the player will see
9. Click on Main Camera in the Inspector window set Size as 16.5
10. Set its Background color of your choice

**Step2:- Controlling the Player**

11. Select Player Game object and in inspector window Click Add Component->Physics 2D-> Rigidbody 2D
12. Again Click Add Component->New Script, Give name as PlayerController and Click on Create and Add
13. In Project View below, Maximize Completed-Assets folder, Now drag the PlayerController script from Assets folder to the Completed-Assets-> Scripts folder
14. Double click on the PlayerController script file to edit it

**PlayerController.cs**

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

public class PlayerController : MonoBehaviour {
   public float speed;
```

```
    private Rigidbody2D rb2d;
      // Use this for initialization
      void Start () {
       rb2d = GetComponent<Rigidbody2D>();
      }

      // Update is called once per frame
      void FixedUpdate () {
       float moveHorizontal = Input.GetAxis("Horizontal");
       float moveVertical = Input.GetAxis("Vertical");
       Vector2 movement = new Vector2(moveHorizontal, moveVertical);
       rb2d.AddForce(movement * speed);
      }
}
```

Save and Close

15. Select Player Game object and in inspector set Gravity scale=0 and speed=10

**Step 3:- Adding Collision**

16. Select Player Game object and in inspector window Click Add Component->Physics 2D-> Circle Collider 2D
17. In the inspector window under Circle Collider 2D set radius to 2.15
18. Select Background in the hierarchy and in inspector window Click Add Component->Physics 2D-> Box Collider 2D. Set Offset X=14.3 and Y=0, Size X=3.3 and Y=31.64
19. Click on its gear control on top right and select Copy Component, Select it again and select Paste Component As New. Set Offset X=-14.3 and Y=0, Size X=3.3 and Y=31.64
20. Click on its gear control on top right and select Copy Component, Select it again and select Paste Component As New. Set Offset X=0 and Y=14.3, Size X=31.64 and Y=3.3
21. Click on its gear control on top right and select Copy Component, Select it again and select Paste Component As New. Set Offset X=0 and Y=-14.3, Size X=31.64 and Y=3.3

## Step 4:- Following the Player with the Camera

22. Select Main Camera in the hierarchy and in inspector window Click Add Component-> New Script, Give name as CameraController and click on Create and Add
23. Drag this script in the Scripts folder
24. Double click on the CameraController script file to edit it

**CameraController.cs**

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

public class CameraController : MonoBehaviour {
    public GameObject player;
    private Vector3 offset;
      // Use this for initialization
      void Start () {
       offset = transform.position - player.transform.position;
      }

      // Update is called once per frame
      void LateUpdate () {
       transform.position = player.transform.position + offset;
      }
}
```

Save and Close

25. Select Main Camera in the hierarchy and in inspector window, under CameraController, click on Player box and drag Player object from the hierarchy over this box

## Step 5:- Creating Collectible Objects

26. Drag Assets->Sprites->Pickup in the hierarchy
27. Select Pickup object in the hierarchy and set sorting layer as Pickups

28. Select Player object in the hierarchy and in the Inspector window deselect the checkbox next to its name to deactivate it temporarily
29. Select Pickup object in the hierarchy and in inspector window Click Add Component-> Physics 2D-> Circle Collider 2D
30. In the inspector window, Under Circle Collider 2D, Set the Radius to 1
31. Select Pickup object in the hierarchy and Click on Add Component->New Script, name it as Rotator, click on Create and Add
32. Drag this script in the Scripts folder

**Rotator.cs**

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

public class Rotator : MonoBehaviour {

    // Use this for initialization
    void Start () {
    }
    // Update is called once per frame
    void Update () {
     transform.Rotate(new Vector3(0, 0, 45) * Time.deltaTime);
    }
}
```

Save and Close

## Step 6:- Picking Up Collectibles

33. Drag Pickup from hierarchy to Completed-Assets->Prefabs folder in Project View
34. Select Pickup from Prefabs folder, click on Tag and Select Pickup. If not added Click on Add Tag and Click on + and give name as Pickup and select it
35. Select Pickup from Prefabs folder, under Circle Collider 2D, check Is Trigger
36. Select from menu, Game Object-> Create Empty, Name it as Pickups
37. Drag Pickup over Pickups to make it child of Pickups
38. Select the Pickup and from menu Edit->Duplicate or press Ctrl+D , repeat this 9 times

39. Place the Pickups at proper places in the scene
40. Select the Player and reactivate it, by selecting checkbox next to its name
41. Select Pickup from Prefabs folder, Add Component->2D Physics-> Rigidbody 2D and set gravity to 0

## Step 7:- Counting Collectibles

42. In Project Hierarchy, select Create->UI->Text
43. Select Text Element and give name as CountText, Set the Text to CountText and color to any color of your choice
44. Place this Text Element at a proper place in the scene, press F or use mouse wheel to zoom out to see where it is. From the gear menu select reset
45. Now again open PlayerController script and make the following changes in the code after using the namespace for UI

**PlayerController.cs**

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

public class PlayerController : MonoBehaviour {
    public float speed;
    private Rigidbody2D rb2d;
    private int count;
    public Text countText;

    // Use this for initialization
    void Start () {
      rb2d = GetComponent<Rigidbody2D>();
      count = 0;
      SetCountText();
    }

    // Update is called once per frame
    void FixedUpdate () {
      float moveHorizontal = Input.GetAxis("Horizontal");
      float moveVertical = Input.GetAxis("Vertical");
```

```
        Vector2 movement = new Vector2(moveHorizontal, moveVertical);
        rb2d.AddForce(movement * speed);
      }

    void OnTriggerEnter2D(Collider2D other)
    {
      if(other.gameObject.CompareTag("PickUp"))
      {
        other.gameObject.SetActive(false);
        count = count + 1;
        SetCountText();
      }
    }
    void SetCountText()
    {
      countText.text = "Count: " + count.ToString();
     }
}
```

46. Now select Player object in the hierarchy and click on the box for Count Text, Now drag CountText from the hierarchy to this box
47. Now create a new text box. Select from the hierarchy Create-> UI-> Text
48. Give the name WinText, text as “Win Text”, give an appropriate color and size ,PosX and PosY so that it doesn’t cover the UFO
49. Make the changes in the code as below

**PlayerController.cs**

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

public class PlayerController : MonoBehaviour {
    public float speed;
    private Rigidbody2D rb2d;
    private int count;

    public Text countText;
    public Text winText;
```

```
// Use this for initialization
void Start () {
 rb2d = GetComponent<Rigidbody2D>();
 count = 0;
 winText.text = "";
 SetCountText();
}

// Update is called once per frame
void FixedUpdate () {
 float moveHorizontal = Input.GetAxis("Horizontal");
 float moveVertical = Input.GetAxis("Vertical");
 Vector2 movement = new Vector2(moveHorizontal, moveVertical);
 rb2d.AddForce(movement * speed);
}

void OnTriggerEnter2D(Collider2D other)
{
   if(other.gameObject.CompareTag("PickUp"))
   {
      other.gameObject.SetActive(false);
      count = count + 1;
      SetCountText();
   }
}
void SetCountText()
{
   countText.text = "Count: " + count.ToString();
   if (count >= 10)
      winText.text = "You win!";
}
}
```

Save and Close

50. Now select Player object in the hierarchy and click on the box for Win Text, Now drag WinText from the hierarchy to this box

## Step 8:- Building our 2D UFO Game

51. To Build the game, File menu-> Build Settings or press Shift+Ctrl+B, Current platform displayed is PC, MAC & Linux Software and on the right Target Platform is Windows. Keep as it is.
52. Click on Add Open Scenes or drag the scenes from the Scenes folder inh the area on the top
53. Create a folder by the name Builds in your game folder. Select that folder and click on Select Folder. An exe file is created