1277

PRACTICAL NO:3

AIM: Write a program to implement unity 3D Roll A Ball Game.

STEPS:

1.ENVIRONMENT AND PLAYER:

I. SETTING UP THE GAME:

1. Click on New Project -> Name it as Roll a Ball -> Set the path or destination -> Select 3D Template -> Click on Create.
2. Save scene as MiniGame under the Scenes folder.(File -> Save scene as -> create a folder in Assets folder named _Scenes).
3. Click on Create in the hierarchy view -> 3D Object ->Plane -> Rename as Ground and reset the transform.
4. Select Frame selected to Ground.
5. Select Gizmos and deselect Show grid.
6. Put scale as x->2, y->1, z->2.
7. Create -> 3D object ->Sphere -> Rename as Player and put Frame selected.
8. Put transform POSITION of player as x=0, y=0.5, z=0.
9. Go to Projects -> create a folder as Materials and with Materials folder selected click on Create -> Material -> New Material -> Rename it as Background. In the inspector window, Click on Albedo and change color to 0, 32, 64, 255.
10. Drag the Background in Scene view of plane.
11. Click on Directional light -> Change the transform rotation -> x=50, y=60, z=0. Position : x=0, y=3, z=0.

II. MOVING THE PLAYER:

12. Select Player -> Add component -> Physics -> Rigid body.
13. In the Project View -> Click on Create -> click on Folder -> name it as Scripts.
14. Select Player -> Add Component -. New Script -> name it as PlayerController and select C# script from the drop down menu. Drop the PlayerController script in it. ADD THE CODE.
15. Change speed as 10.

2. CAMERA AND PLAY AREA:

I. MOIVNG THE CAMERA:

1. Main Camera -> Adjust the Transform: POSITION: x=0, y=10, z=-10. ROTATION: x=45, y=0, z=0.
2. Add component -> New script -> Add script for main camera as CameraController.
3. Drag the CameraController in the Scripts folder.
4. Drag the Player object in the Camera's Player slot in the CameraController's component.

CAMERACONTROLLER.cs

```
using UnityEngine;
using System.Collections;

public class CameraController : MonoBehaviour {

    public GameObject player;

    private Vector3 offset;

    void Start ()
    {
        offset = transform.position - player.transform.position;
    }

    void LateUpdate ()
    {
        transform.position = player.transform.position + offset;
    }
}
```

II. SETTING UP THE PLAY AREA:

1. In the Hierarchy view -> Click on Create -> Game Object -> name it as Walls.
2. Reset the transform to origin.
3. Game Object -> 3D Object -> Cube -> rename this as West Wall and reset it to origin.
4. Drag and drop it in the Walls object.
5. Select the West Wall object and in the transform, change the Scale of x, y and z-axis to 0.5, 2 and 20.5 respectively.
6. Set the transform's position value of x to -10.
7. Click on Edit -> Duplicate -> rename it as East Wall. and remove the negative sign of x-axis in position transform.
8. Duplicate the East Wall and rename it as North Wall.
9. Reset the transform, set Scale -> x-axis: 20.5, y: 2, z: 0.5 an position of z-axis to 10.
10. Duplicate North Wall and rename it as South Wall, change position value of z-axis from 10 to -10.
11. Highlight the Player object, and set the editor to Local Mode.

Creating collectable objects.

1.create a new cube and rename it as pickup

2.reset the pickup transform to origin.

3.select the player object and deselect the check box of the name field.

Change the transform rotation of pickup to 45,45,45

Keeping the pickup selected create a new script by clicking on the addcomponent on the inspector and give a name as Rotator.

Place the script into the scripts folder .open and write the code

Rotator.cs

```
usingUnityEngine;
usingSystem.Collections;

publicclass Rotator :MonoBehaviour {

void Update ()
    {
transform.Rotate (new Vector3 (15, 30, 45) *Time.deltaTime);
    }
}
```

Select the project view and create a folder name it as Prefabs.Drag the pickup object from hierarchy and place it inside the prefabs folder.

Create a new gameobject and give it a name Pickups.make this gameobject as parent and place the other cube(pickup) inside it.

place the pickup on the game area and make a duplicate of it and arrange it on the palyer area.

In the project view make the duplicate of material(Background) and rename it as pickup.keeping this selected change the albidocolor property to yellow.

Change the color of the pickup by selecting the material(pickup).

Collecting the pickup objects.

Change the code in PlayerContoller.cs and save the script.

```
using UnityEngine;
using System.Collections;

public class PlayerController : MonoBehaviour {

    public float speed;

    private Rigidbody rb;

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

    void FixedUpdate ()
    {
        float moveHorizontal = Input.GetAxis ("Horizontal");
        float moveVertical = Input.GetAxis ("Vertical");

        Vector3 movement = new Vector3 (moveHorizontal, 0.0f, moveVertical);

        rb.AddForce (movement * speed);
    }

    void OnTriggerEnter(Collider other)
    {
        if (other.gameObject.CompareTag("Pick up"))
        {
            other.gameObject.SetActive (false);
        }
    }
}
```

Select the prefab asset for the pickup object.select add tag and create a new custom tag by using '+' button and type Pick Up(which will be in row 0 and it is case sensitive).

Now select the prefab asset tag manager and select the pick up tag.

Select the player object(sphere) and disable the mesh render in the inspector.

Select the prefab asset and under the box collider component enable the is trigger checkbox.

Add a rigid body to pickups object and enable the " is kinematic " property .

Displaying score and text

Update the code in PlayerController.cs and save the script.

```
using UnityEngine;
using System.Collections;

public class PlayerController : MonoBehaviour {

    public float speed;

    private Rigidbody rb;
    private int count;

    void Start ()
    {
        rb = GetComponent<Rigidbody>();
        count = 0;
    }

    void FixedUpdate ()
    {
        float moveHorizontal = Input.GetAxis ("Horizontal");
        float moveVertical = Input.GetAxis ("Vertical");

        Vector3 movement = new Vector3 (moveHorizontal, 0.0f, moveVertical);

        rb.AddForce (movement * speed);
    }

    void OnTriggerEnter(Collider other)
    {
        if (other.gameObject.CompareTag("Pick Up"))
        {
            other.gameObject.SetActive (false);
            count = count + 1;
        }
    }
```

Create a new UI element by clicking on a create menu of hierarchy and name it as text.

Make sure that the UI element (text) is inside Canvas.

Rename the text element to Count Text.

Keeping count text selected we can reset the rect transform by using context sensitive game menu.

Change the color by using colorswatch.

Change the position of a count text by changing the position in transform are by dragging.

To display pickup message add another UI element.

Adjust react transform y=75,paragraph alignment in text component.

Make text layer by setting font size and adjust color.

4/10/18

Building game

Same steps which are followed in 2D UFO .refer it for this step.