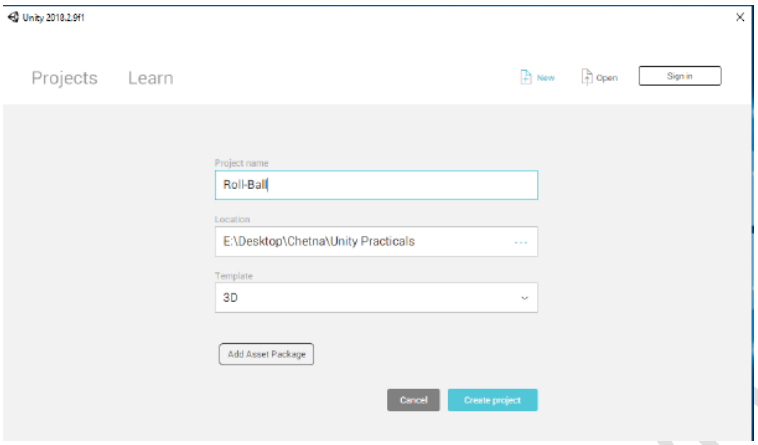
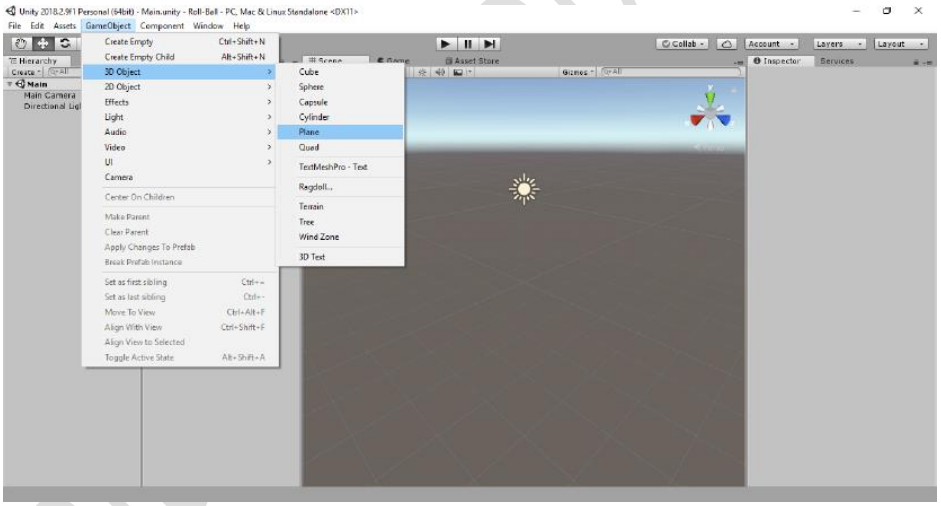
Unity Practical 2- Roll A Ball Game

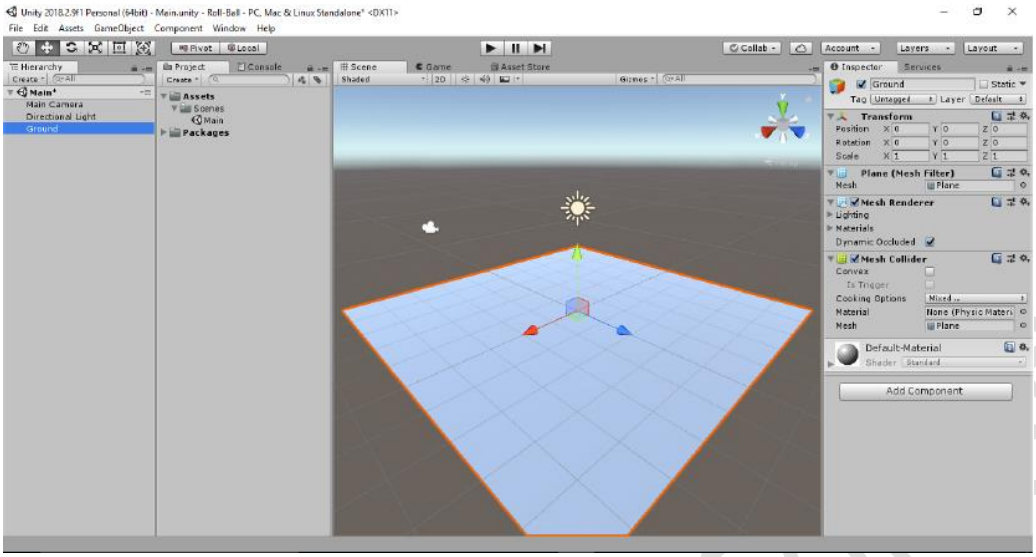
Setting Up the Game

1. From menu, File-> New Project, Name: Roll A Ball, Set appropriate path, 3D, Click on Create Project

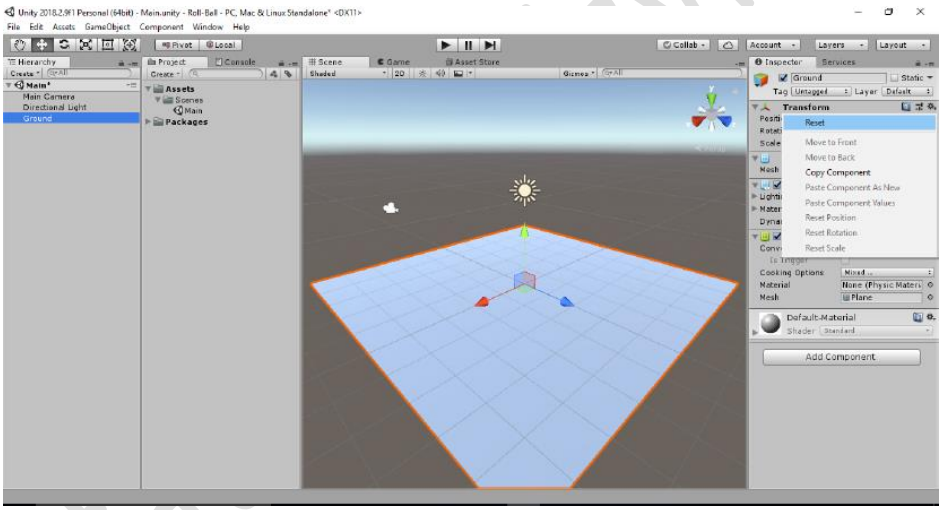


2. From menu, File->Save Scene As, Assets/Scenes, Name: Main
3. In hierarchy, Create -> 3D Object-> Plane, name: Ground

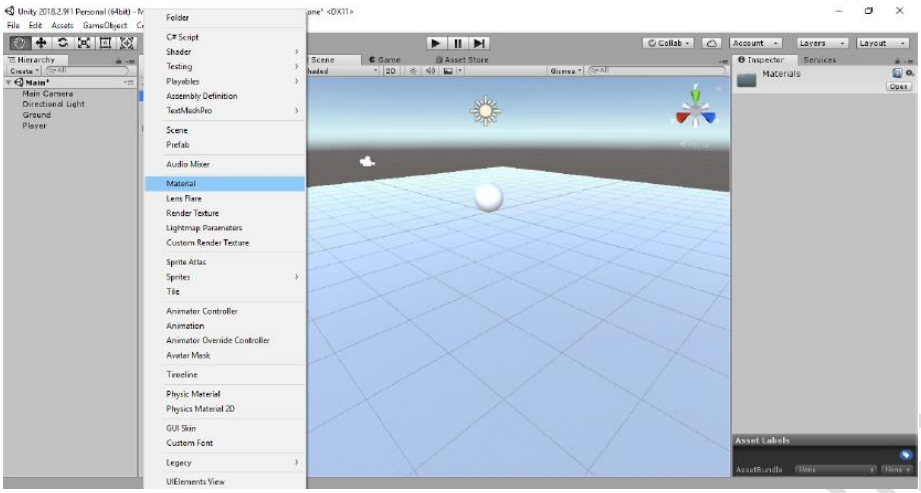


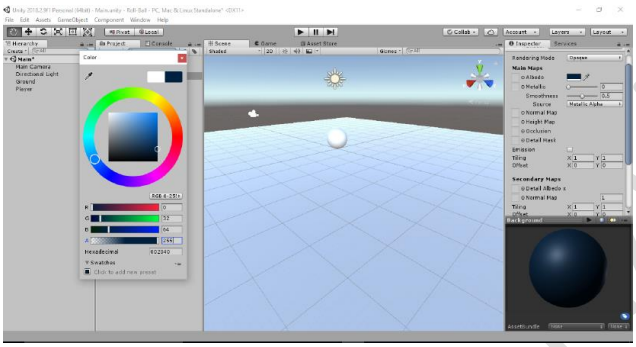


4. In hierarchy, Select ground, Press F to focus, Click on Scale tool and resize
5. In hierarchy, Select Main Camera, Position: X=0, y=10, z=-10, Rotation: X=45
6. In hierarchy, Create->3D Object-> Sphere, name: Player, Reset position using gear menu, Position: Y=0.5



7. In Project View, Create-> Folder, Name: Ground , open it , Create->Material, Name: Material , Albedo: select blue colour, drag the Material on Ground in the hierarchy





Moving the Player

1. In hierarchy, Select Player, Add Component->Physics -> Rigidbody
2. In hierarchy, Select Player, Add Component-> New Script, Name: PlayerController
3. In Project View, under Assets, Create-> Folder, Name: Scripts, Drag PlayerController in this folder, open it and add the following code

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

public class PlayerController : MonoBehaviour {
    private Rigidbody rb;
    public float speed;
     // Use this for initialization
     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);
    }
}
```

4. In hierarchy, Select Player, set speed to 20 and run and test the game

Moving the Camera

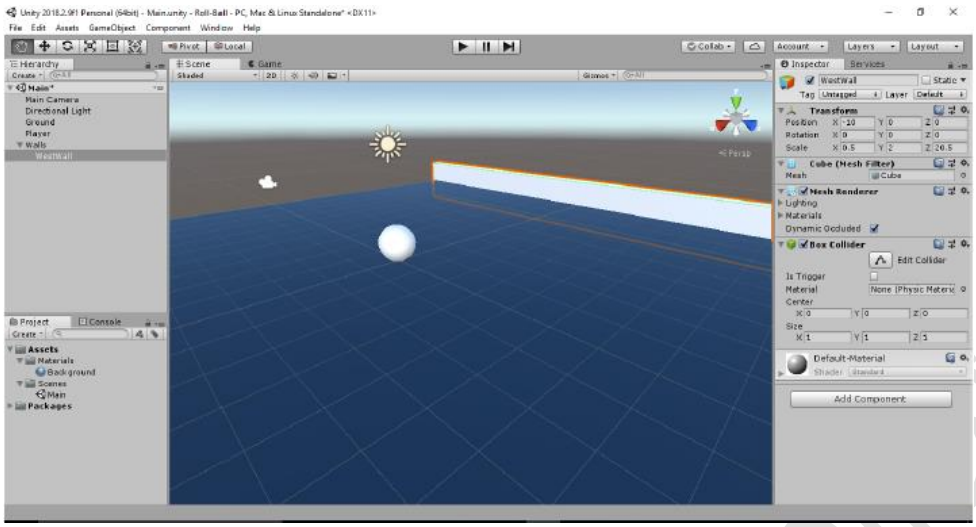
1. In hierarchy, Select Main Camera, Add Component-> New Script, Name: CameraController, drag it to the scripts folder
2. Open it and add the following code

```
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;
        }
}
```

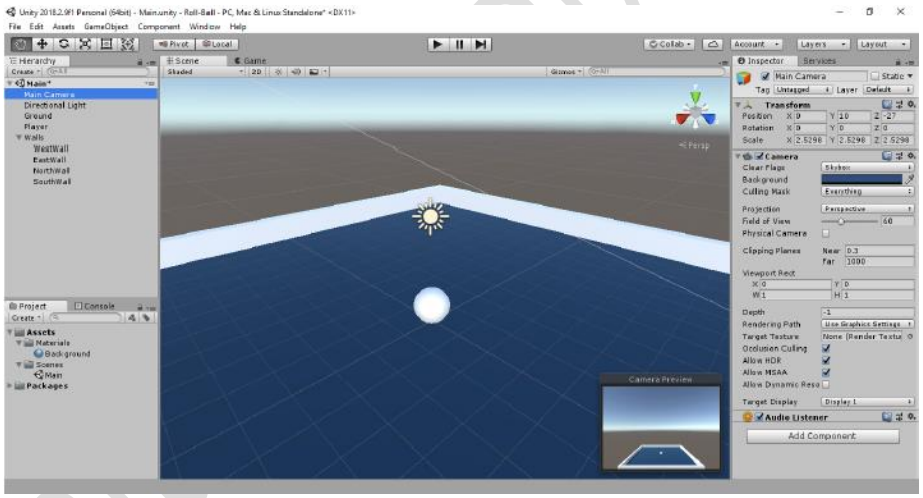
3. In hierarchy, select Main Camera, click on Player attribute under CameraController (it turns blue), drag Player from hierarchy on this slot

Setting up the play Area

1. From menu, GameObject-> Create Empty, Name: Walls, Reset its position from gear menu
2. From menu, GameObject->3D Object-> Cube, Name: WestWall, Reset its position from gear menu, Scale: X=0.5, Y=2, Z=10 (give Z value according to length of the ground), Position X=-5

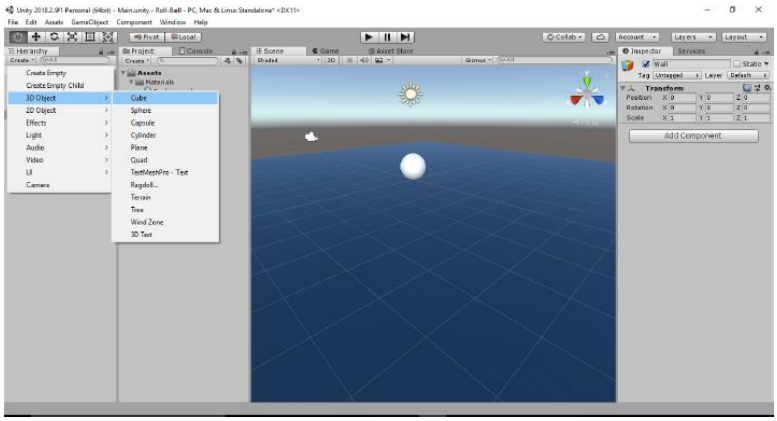


3. In hierarchy, select WestWall, from menu, Edit->Duplicate, Name: East Wall, Position X=5
4. In hierarchy, select EastWall, From menu, Edit->Duplicate, Name: NorthWall, Scale: X=10, Y=2, Z=0.5(give X value according to length of the ground) ,Position Z=5
5. In hierarchy, select NorthWall, From menu, Edit->Duplicate, Name: SouthWall, Position Z=-5

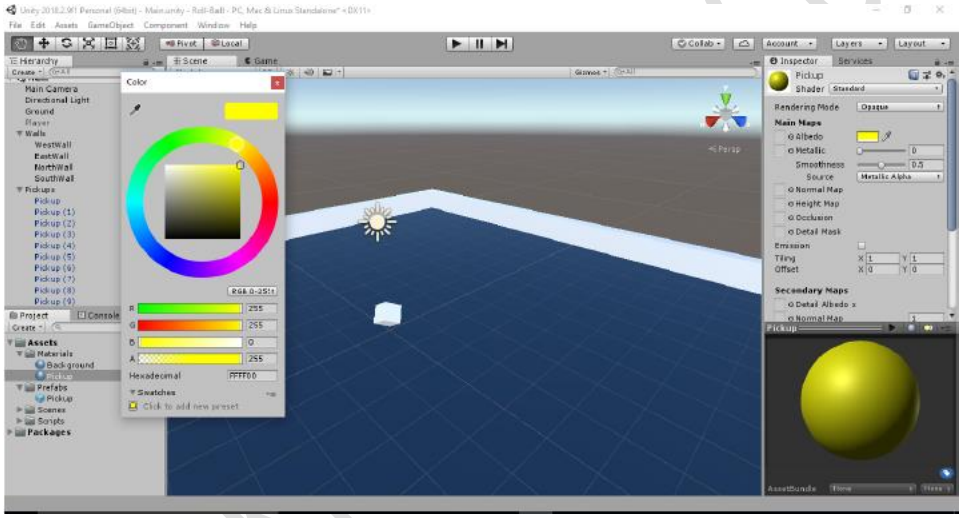


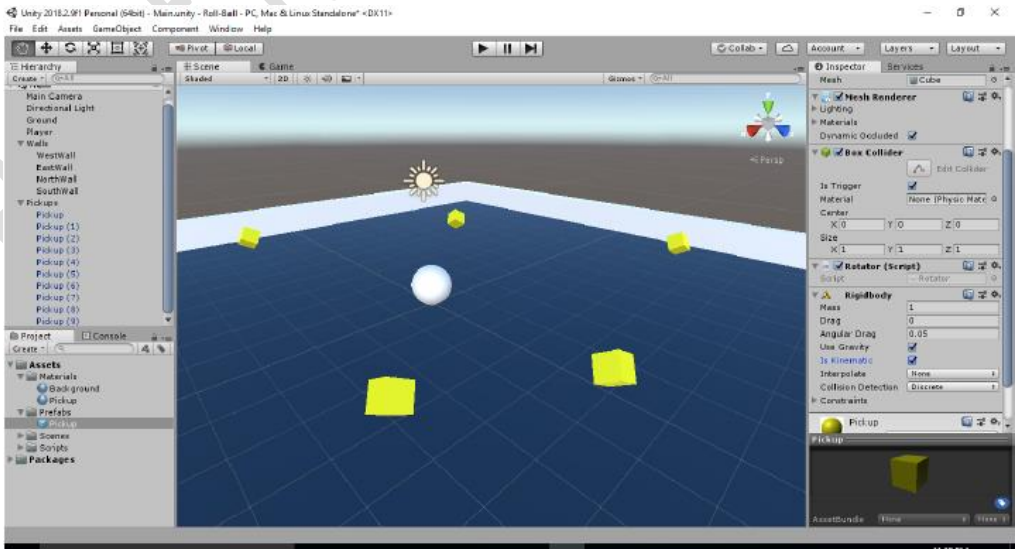
Creating Collectible Objects

1. From menu, GameObject->3D Object-> Cube, Name: PickUp, Reset Position

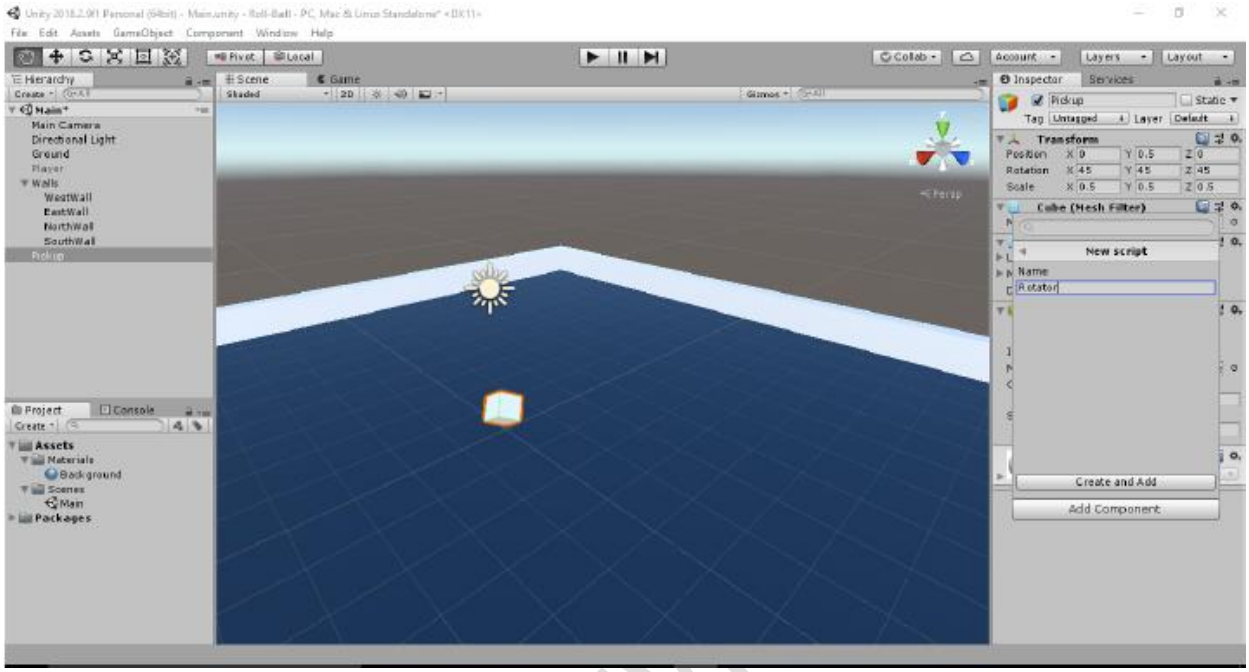


2. Deactivate Player object
3. In hierarchy, select PickUp, Position : Y=0.5, Rotation: X=45, Y=45, Z=45
4. Under Materials folder, select Material, Edit menu-> Duplicate, Name: PickUp, change Albedo to yellow colour, drag it and drop over PickUp

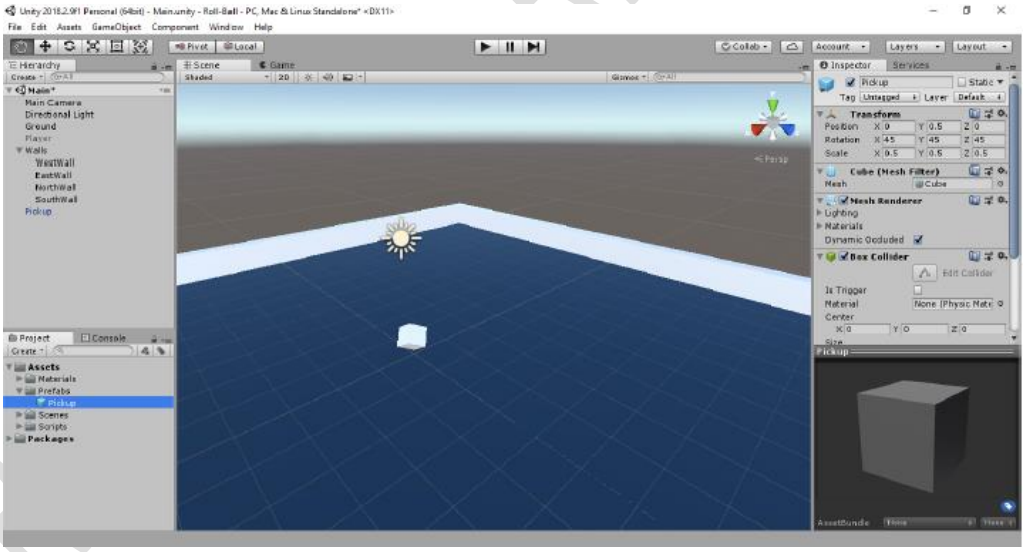




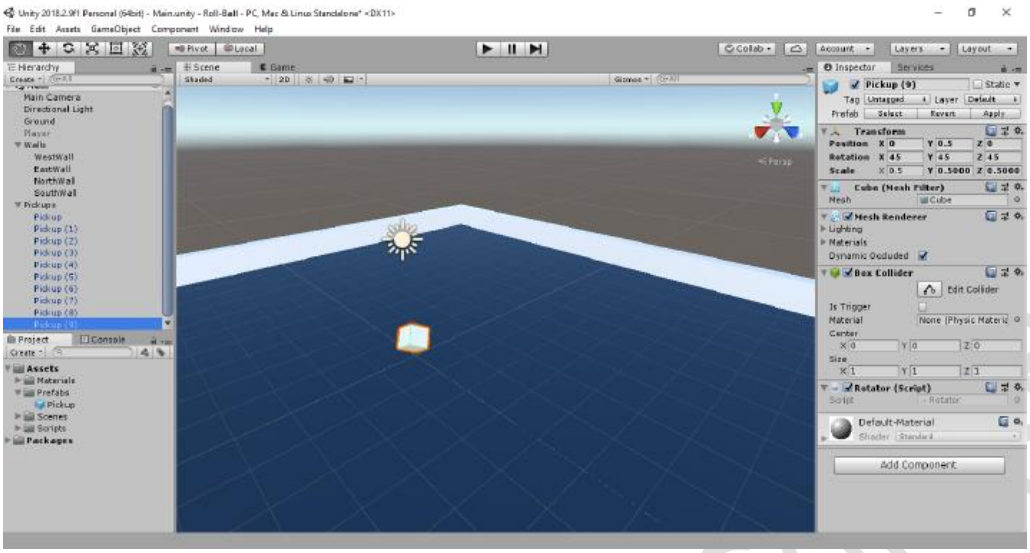
5. Add Component-> Script, Name: Rotator, move to Scripts folder



6. Open Rotator script and add the following code:
7. In Project View, Create->Folder, Name: Prefabs, drag the PickUp collectible in this folder

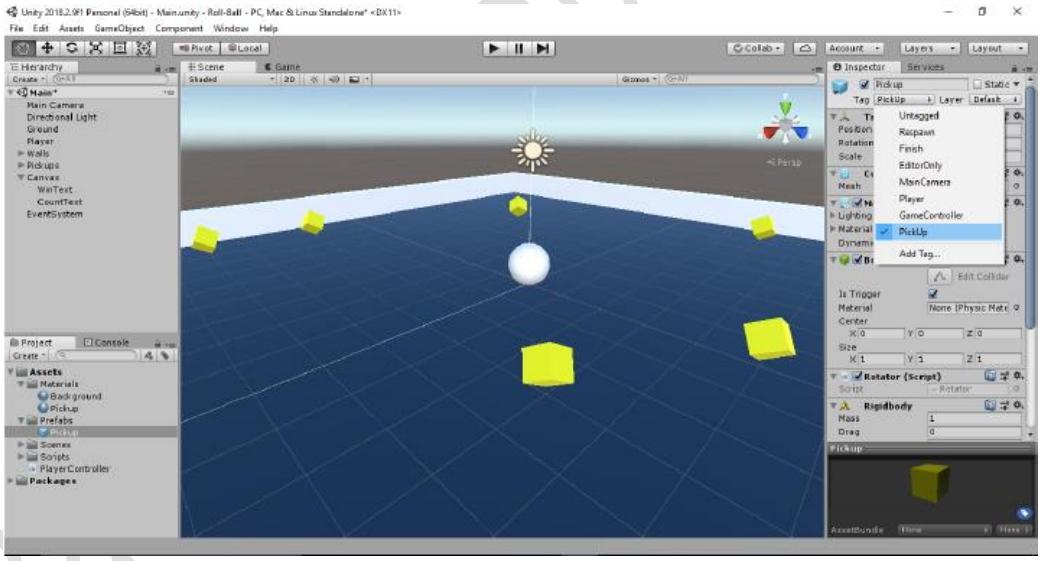


8. In hierarchy, Create-> Create Empty, Name: PickUps, drag PickUp into PickUps
9. Go to Scene view, Change Gizmo to see top down view
10. In hierarchy, Select PickUp , Edit menu-> Duplicate, place it at an appropriate place, repeat for other pickups



Collecting PickUp Objects

1. Under Prefabs folder, select PickUp prefab, Tag: click on Add Tag, Click on +, Name: PickUp, Again select PickUp prefab, click on Tag-> select PickUp
2. Check Is Trigger checkbox



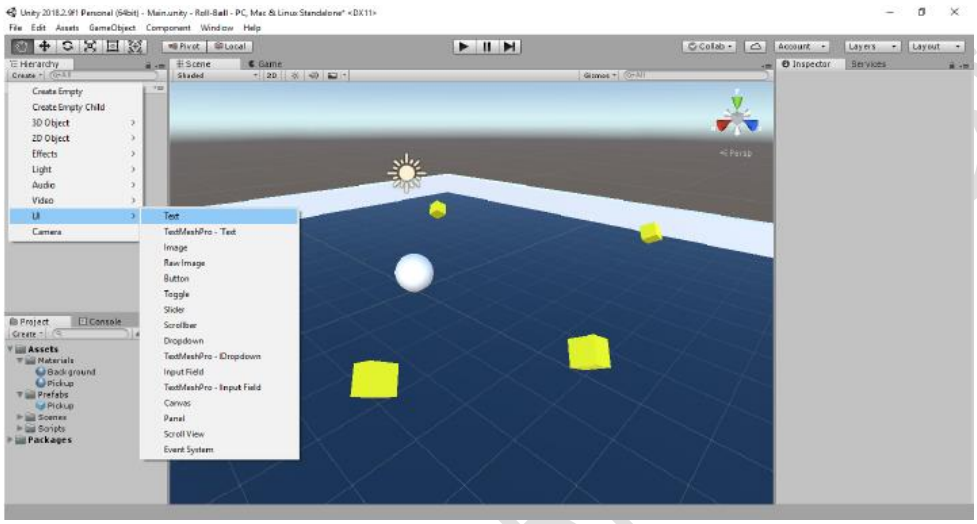
3. Add Component-> Physics-> Rigidbody, Under Rigidbody, Select Is Kinematic checkbox
4. Open PlayerController Scripts from Scripts folder and add the following function in it:

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

}

Displaying the Score and Win Text

1. In Hierarchy, Create->UI-> Text, Name: countText, Click Anchors box, press Shift+Alt, and select top left, select appropriate and Font size for the text
2. In Hierarchy, Create->UI-> Text, Name: winText, Click Anchors box, press Shift+Alt, and select top left, select appropriate and Font size for the text

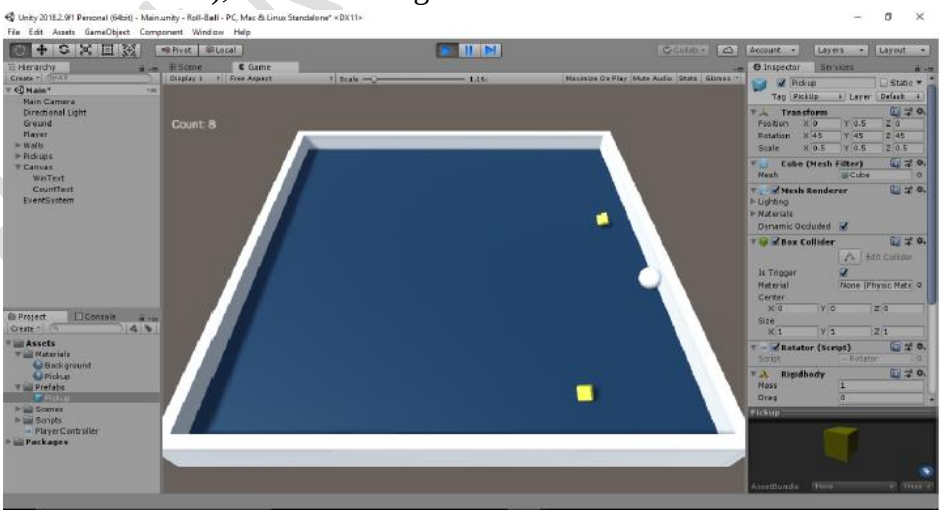


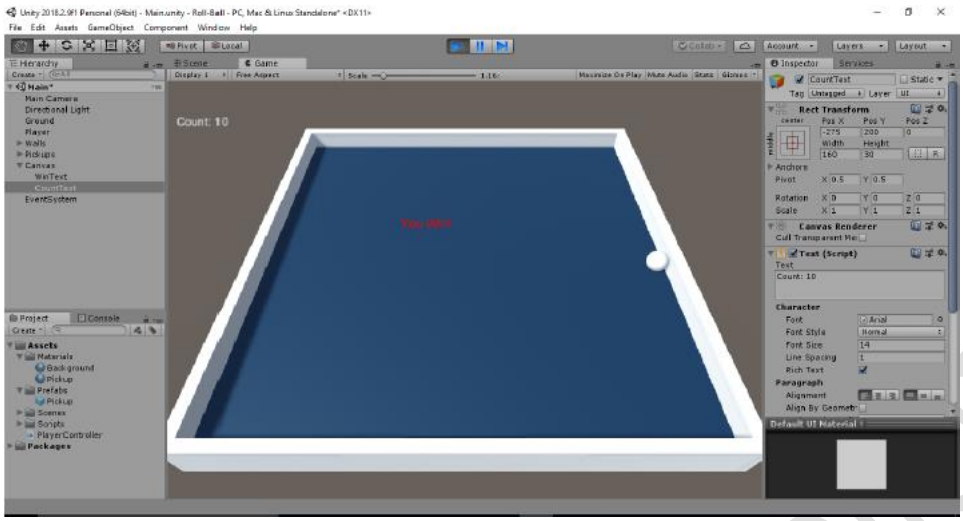
3. Open PlayerController Scripts from Scripts folder and make changes to the code as following : First add namespace UnityEngine.UI

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

public class PlayerController : MonoBehaviour {
    private Rigidbody rb;
    public float speed;
    private int count;
    public Text countText;
    public Text winText;
        // Use this for initialization
        void Start () {
      rb = GetComponent<Rigidbody>();
      count = 0;
         SetCountText();
         winText.text = "";
        }

 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("PickUp"))
        {
            other.gameObject.SetActive(false);
            count = count + 1;
            SetCountText();
        }
    }

    void SetCountText()
    {
        countText.text = " Count : " + count.ToString();
        if(count>=4)
        {
            winText.text = "You Win";
        }
    }
}
```

4. In Hierarchy, select Player object, under PlayerController, click on slot for countText(it should turn blue), select and drag countText in this slot
5. In Hierarchy, select Player object, under PlayerController, click on slot for winText(it should turn blue), select and drag winText in this slot





Building the Game

1. File menu-> Build Settings, click on add open scenes or drag the scene from Scenes folder in the top area, don't include SampleScene
2. Click on Build, Create a folder under the root folder Roll A Ball, Name it Builds, select it, Click on select Folder