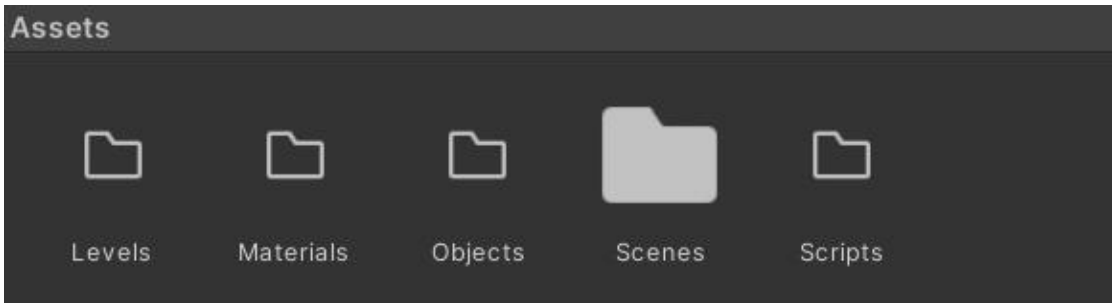
GAME PROGRAMMING LAB - 7

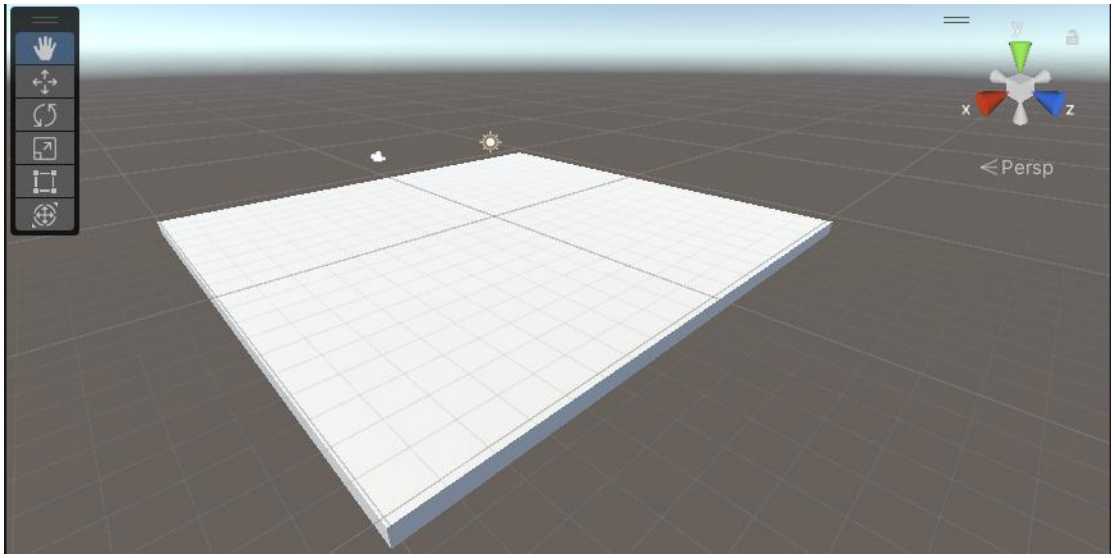
- by Meghna Sinha, 20BAI1133

Create the folders for the following -

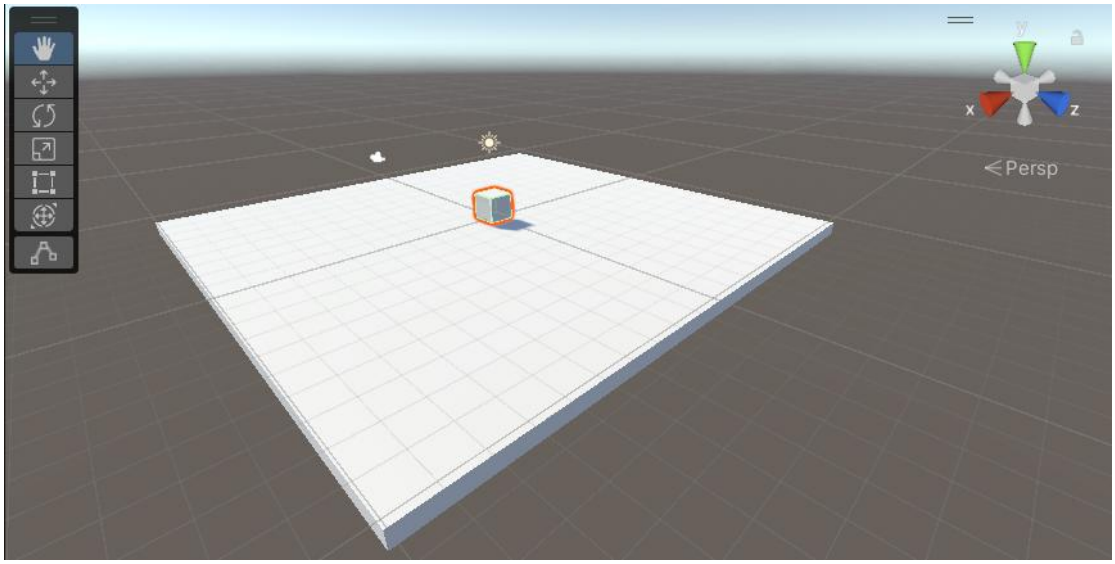
- Scripts
- Materials
- Objects
- Levels



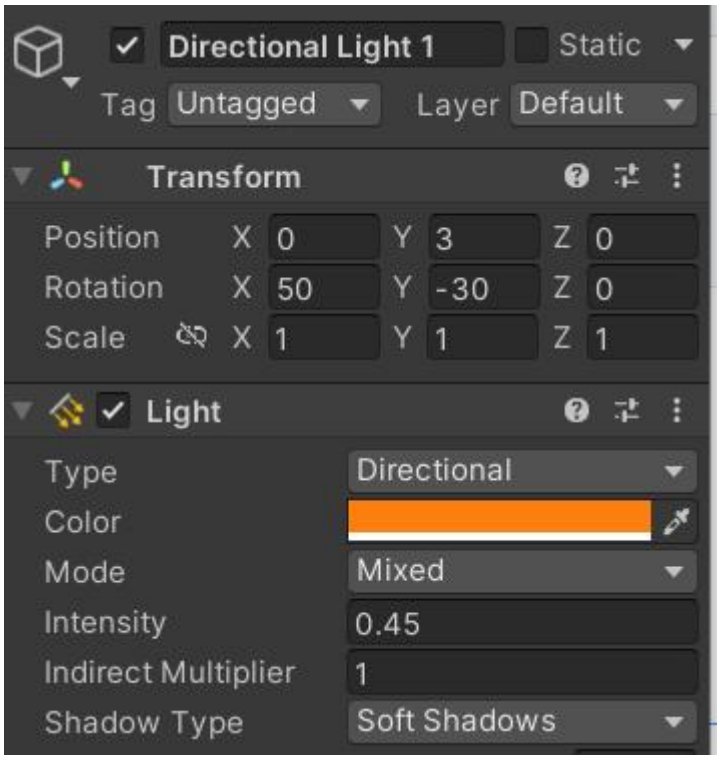
Create gameobject floor (scale:- x:20, y:0.5, z:20; position:- y:-0.5)

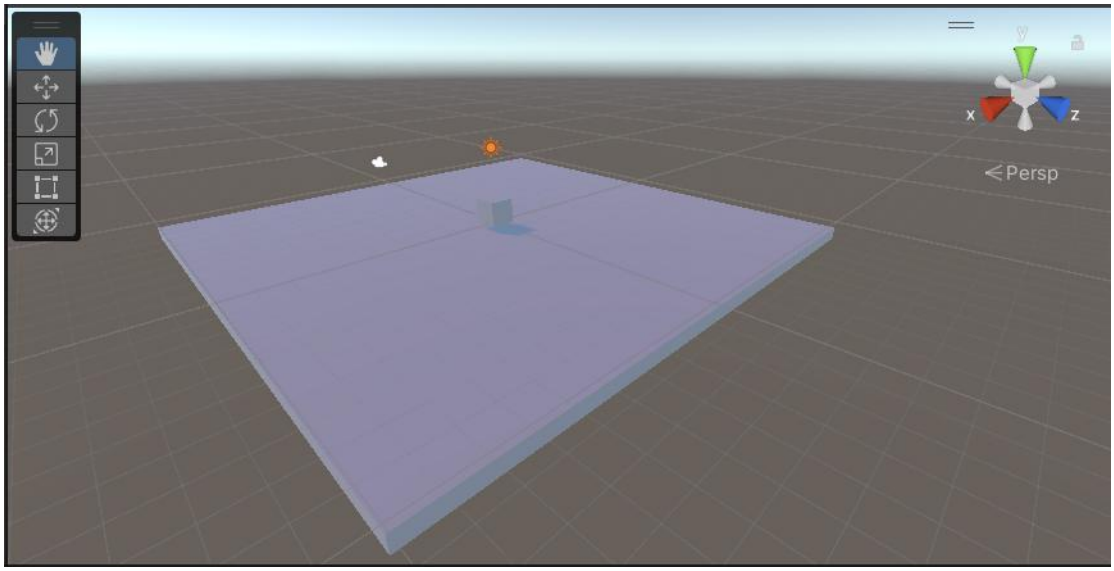


Create gameobject player (position:- y:0.25)

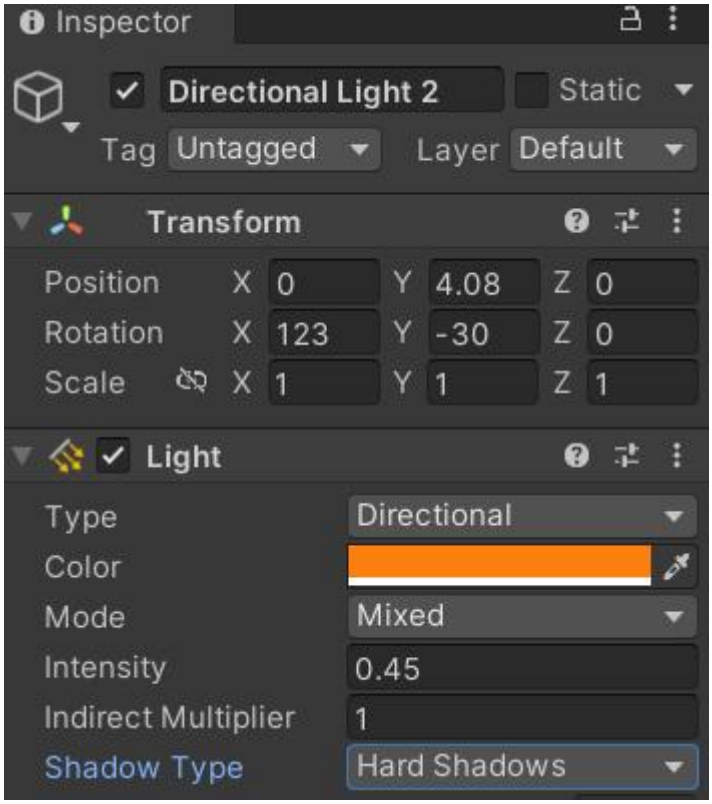


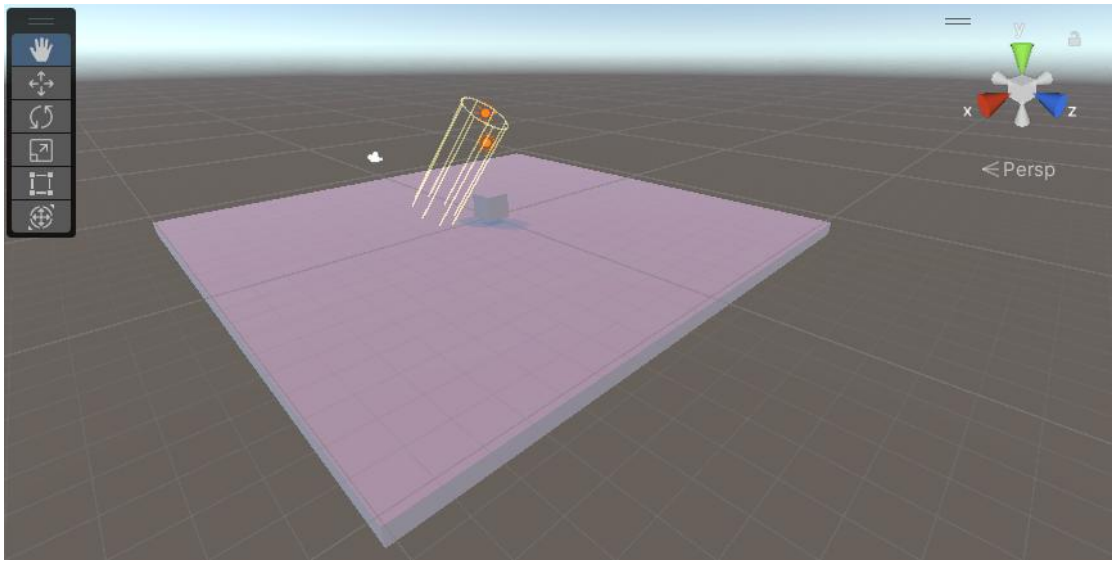
Make changes to the directional light attribute. Changing color to orange and intensity to 0.45



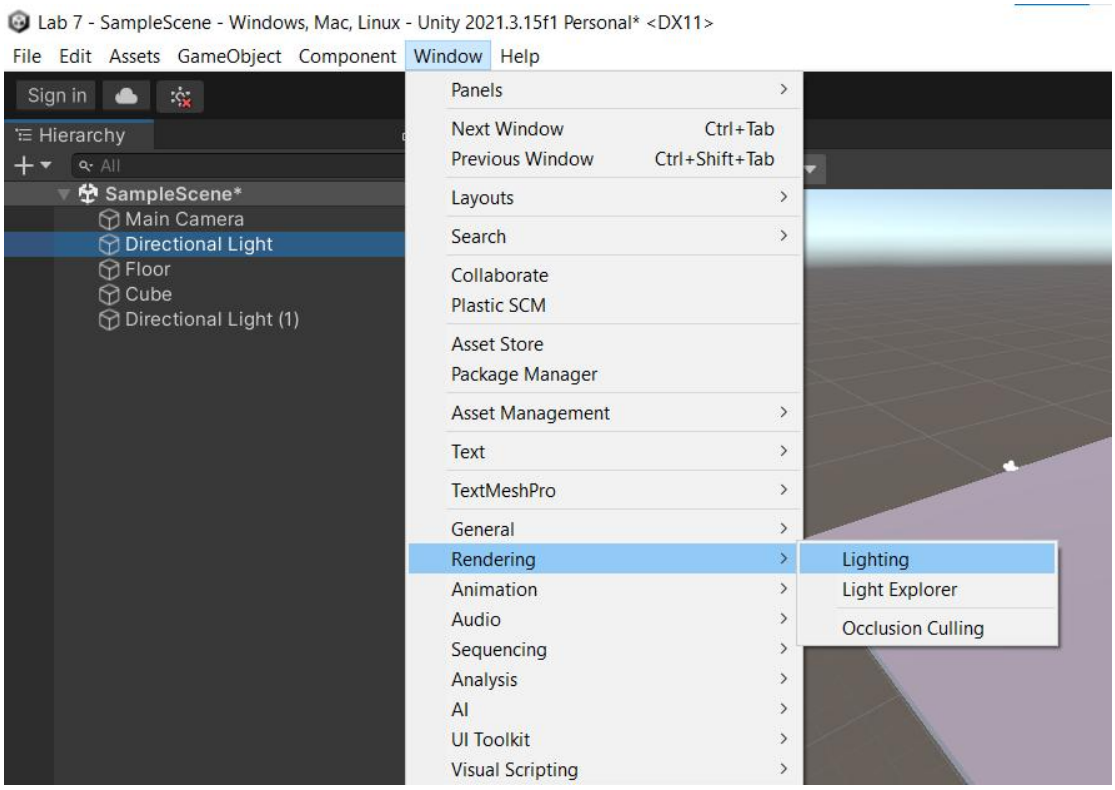


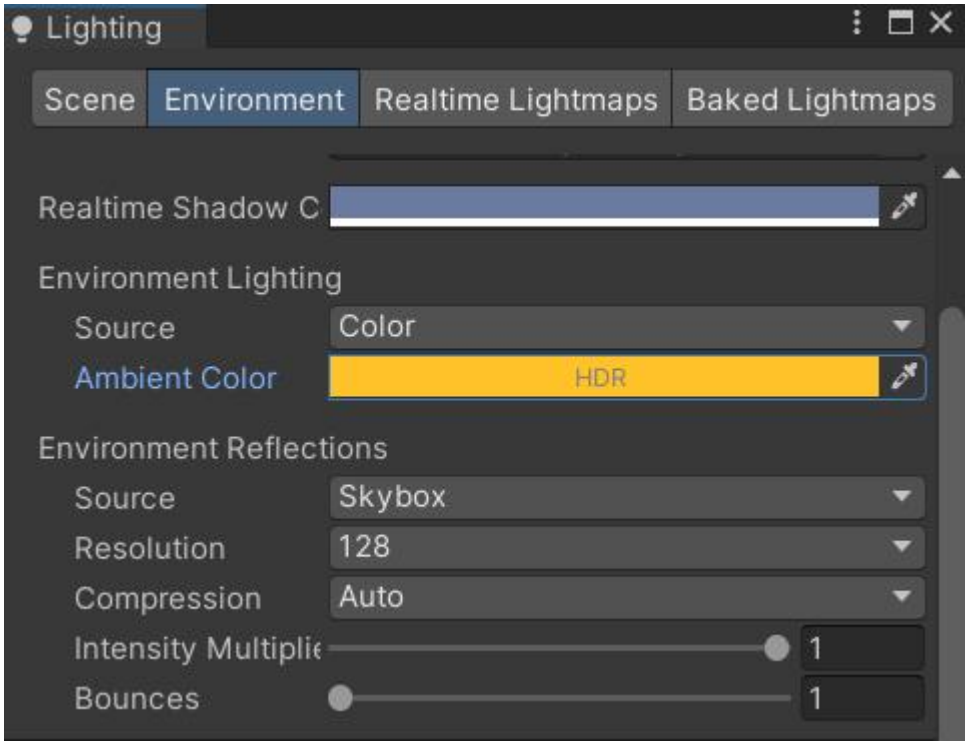
Create another directional light by duplicating the above. This will have shadow type as hard shadows

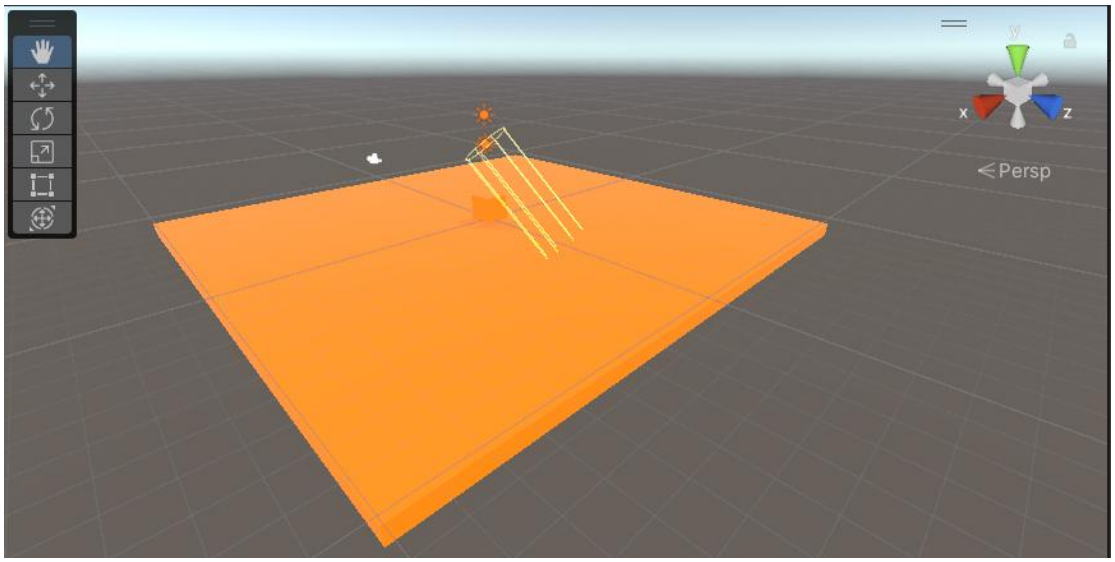




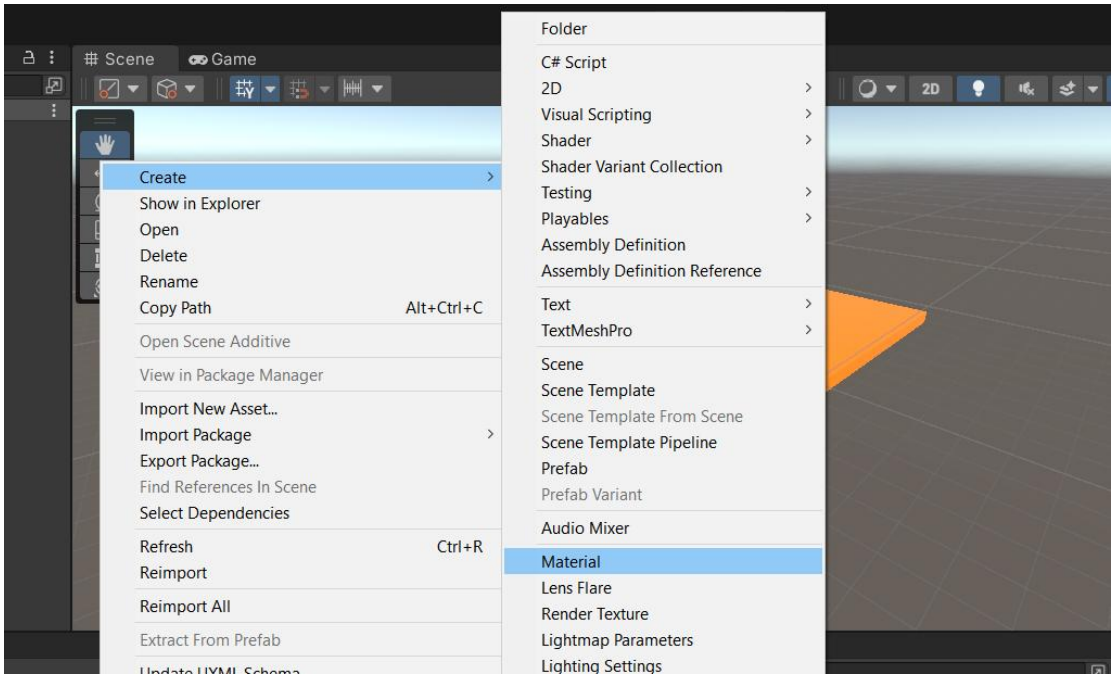
Applying ambient lighting to the environment







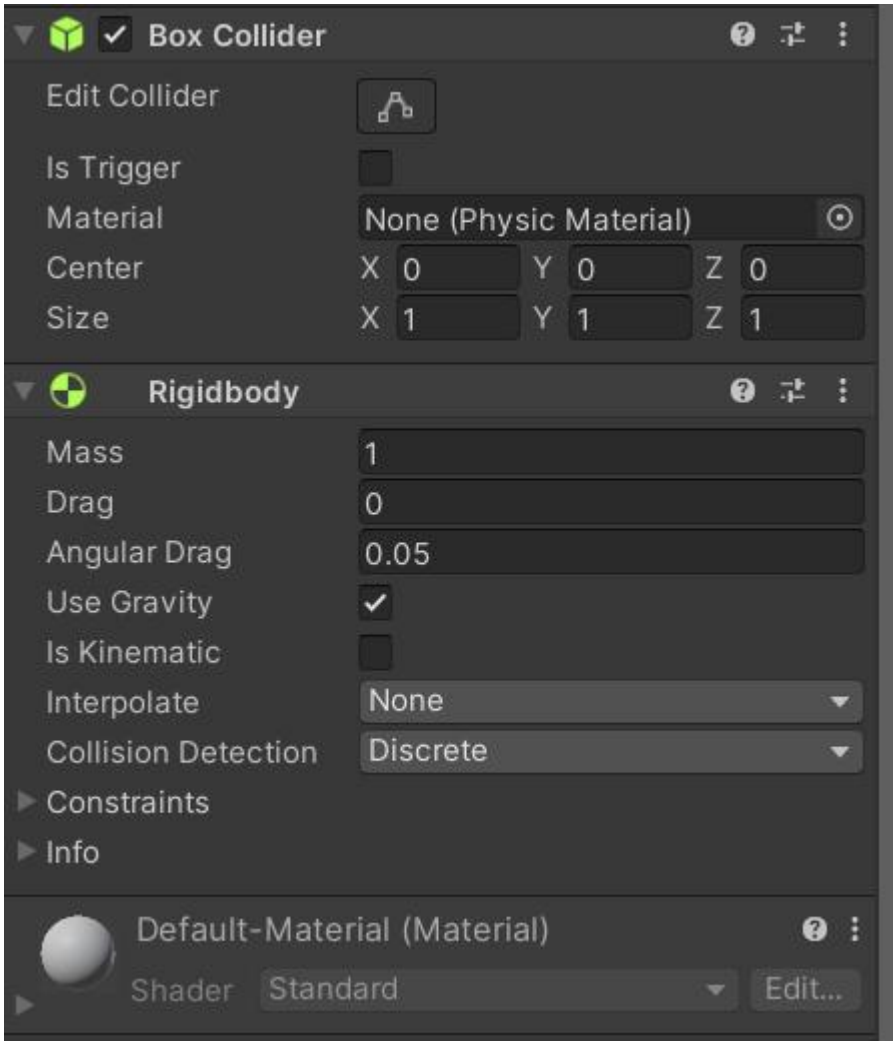
Creating different materials for player, enemy



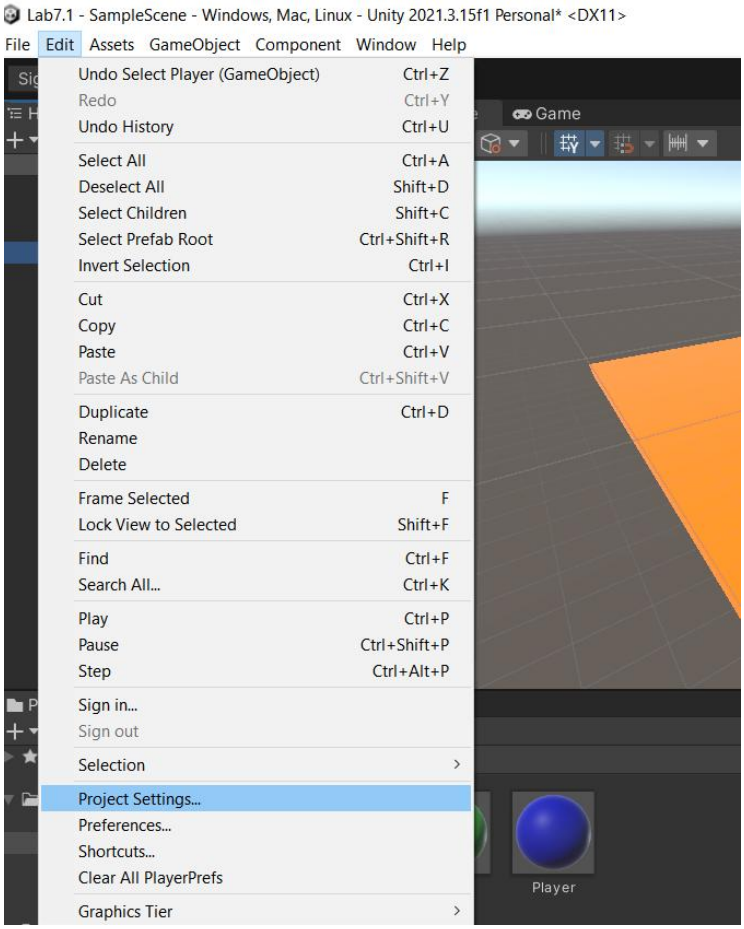
Shader --> Diffuse --> Main color --> name it as player

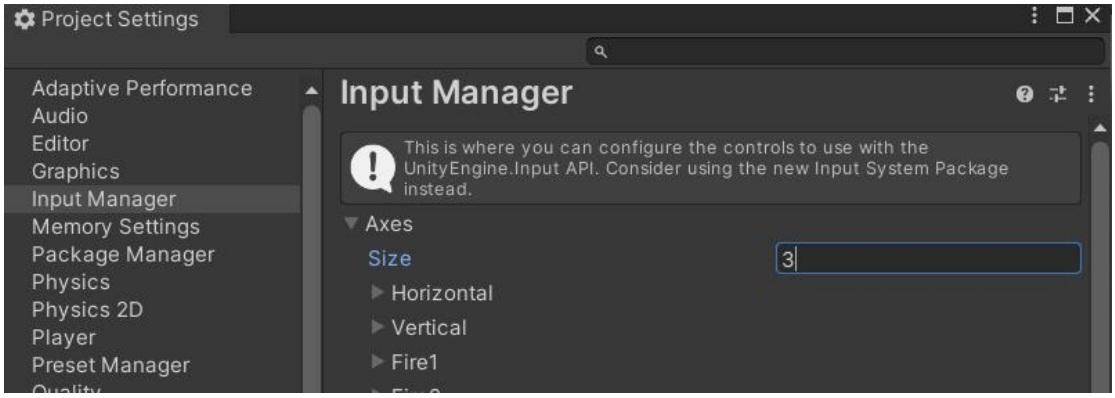
Similarily create material for enemy and ground/grass

For the cube player check for the Box Collider component and add the RigidBody component

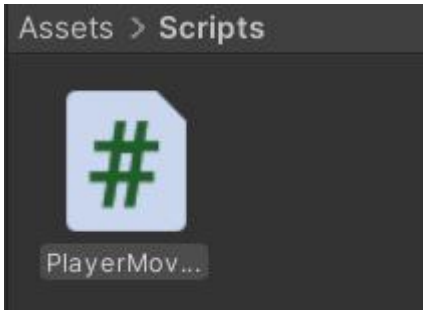


Click player object (cube)
Edit --> Project Settings --> Input --> Size = 3





Create script for Player Movement



```
Almost out of storage … If you run out, you can't create or edit files, send or receive email
on Gmail, or back up to Google Photos.
PlayerMovement.cs
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PlayerMovement : MonoBehaviour {
    public float moveSpeed; //part1
    public GameObject deathParticles;
    private float maxSpeed = 5f;//part1
    private Vector3 input; //xyz coordinate -> one variable to hold three values//part1
    private Vector3 spawn;
```

```
    // Use this for initialization
    void Start () {
        spawn = transform.position;
    }

    // Update is called once per frame
    void Update () {
```

```
        input = new Vector3(Input.GetAxis("Horizontal"), 0, Input.GetAxis("Vertical"));//x-
horizontal                                              //y-no jump; z- vertical
                                        //part1
```

```
        if (GetComponent<Rigidbody>().velocity.magnitude < maxSpeed)         //part1
        {
            GetComponent<Rigidbody>().AddForce(input * moveSpeed);
        }
```

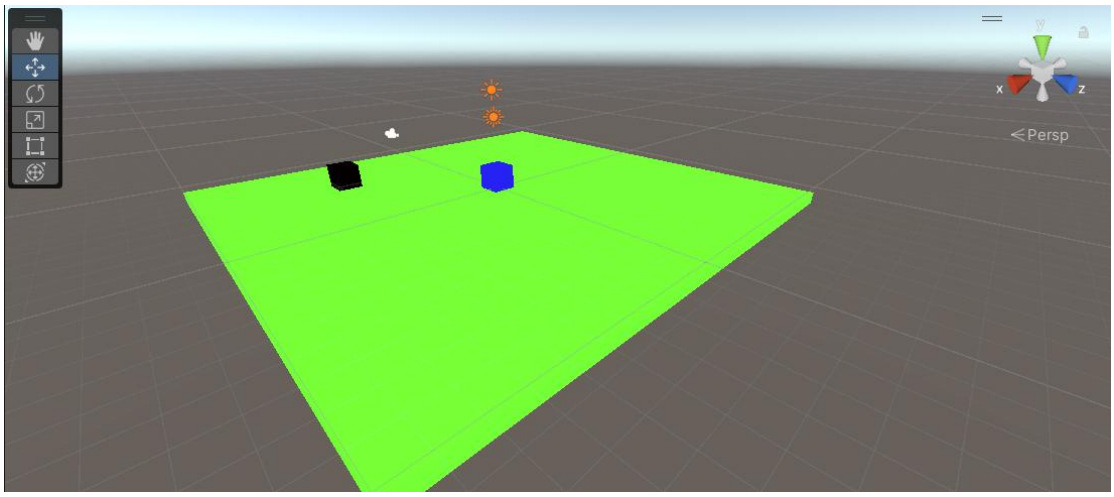
```
        if (transform.position.y < -2)
        {
            Die();
        }

    }
```

```
    private void OnCollisionEnter(Collision other) //part3
    {
        if(other.transform.tag == "Enemy")
        {
            //print("I hit enemy");
            Die();
        }
    }
```

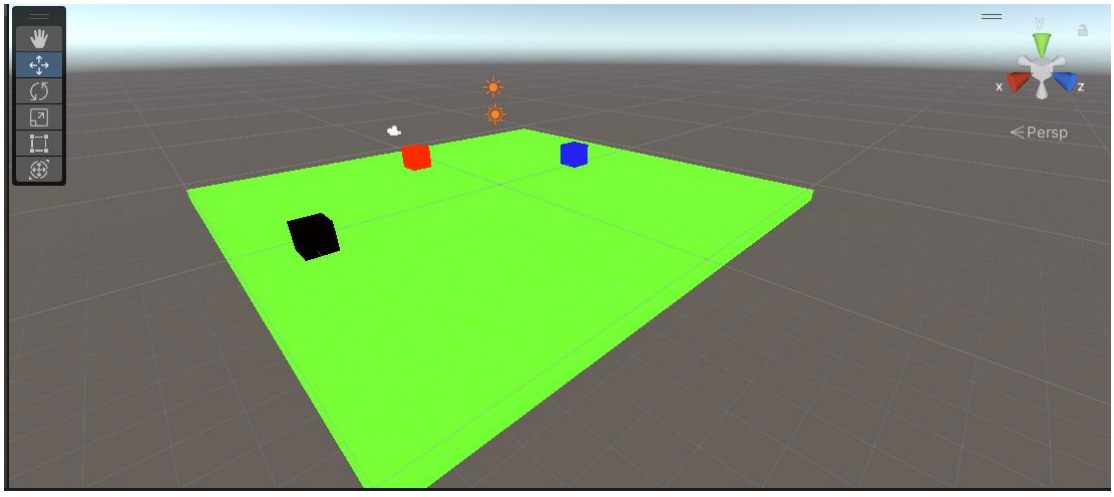
```
    private void OnTriggerEnter(Collider other)
    {
        if(other.transform.tag=="Goal")
        {
            GameManager.CompleteLevel();
        }
    }
```

```
    void Die()
    {
        Instantiate(deathParticles, transform.position, Quaternion.Euler(270,0,0));
            transform.position = spawn;
    }
}
```

Create another cube for goal



Now create the enemy cubes



Add the enemy cube in the prefabs folder

Create two empty game objects as patrol points for enemy to move

Create script for patrol and add it to the enemy

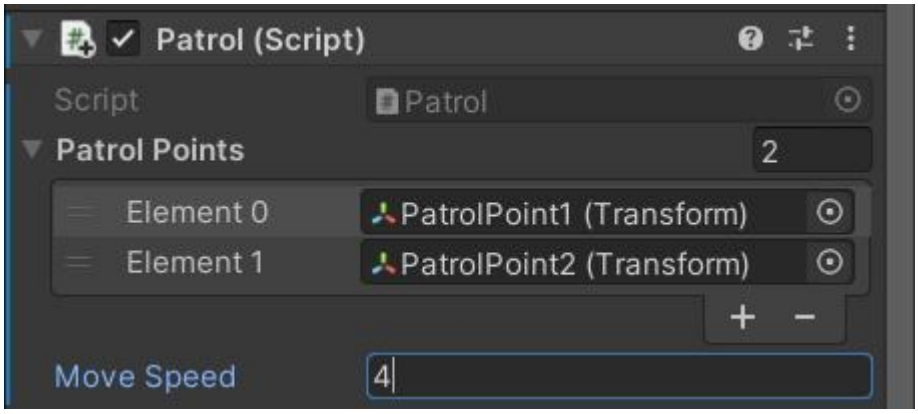
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Patrol : MonoBehaviour {
```

```
    public Transform[] patrolPoints;
    public float moveSpeed;
    private int currentPoint;
```

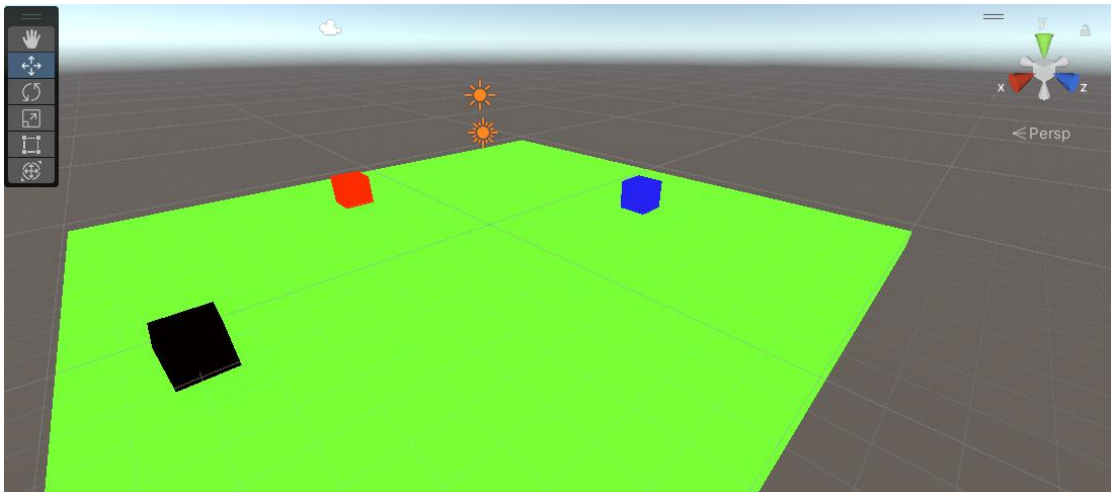
```
    // Use this for initialization
    void Start () {
        transform.position = patrolPoints[0].position;
        currentPoint = 0;
    }

    // Update is called once per frame
    void Update () {
```

```


        if(transform.position == patrolPoints[currentPoint].position)
        {
            currentPoint++;
        }
        if (currentPoint >=  patrolPoints.Length)
        {
            currentPoint = 0;
        }
        transform.position = Vector3.MoveTowards(transform.position,
patrolPoints[currentPoint].position, moveSpeed * Time.deltaTime);
    }
}
```

Adding patrol points to the enemy



Create scene 2 for level 2


x
z
Persp