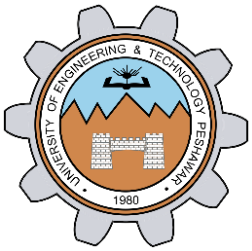
Unity API (Input System, Collision and Trigger Methods)

LAB # 5



Fall 2024

CSE-411L Intro to Game Development Lab

Submitted by: **Ali Asghar**

Registration No.: **21PWCSE2059**

Class Section: **A**

"On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Submitted to:

Engr. Abdullah Hamid

Date:

21st December 2024

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

Objective:

In this lab we further explored the Unity API.

Tasks:

- Open/create a Unity scene.

- Create a player cube that moves forward, backward, left, and right.

- The camera in the scene should be set to a top-down view.

- The scene should have a plane with a maze on it (see the following picture for reference).

- The walls of the maze should be made from cubes of different sizes.

- At the start of the scene, there should be a sphere (ball) and a player cube. When the player moves the ball to the goal in the middle of the maze, the goal should turn green.

- If the ball touches the walls of the maze, the walls should turn red and return to normal when the ball moves away from them.

- Additionally, when the player touches the ball, the ball should turn yellow. When the player releases the ball, it should return to white.

Code:

Player class

Assets > Labs > Lab5 > Scripts > Player.cs

```
6   public class Player : MonoBehaviour{
7       public float speed = 1f;
8
9       // Update is called once per frame
10      void Update(){
11          if (Input.GetKey(KeyCode.W))
12              transform.Translate(Vector3.forward * Time.deltaTime * speed
13
14          if (Input.GetKey(KeyCode.S))
15              transform.Translate(Vector3.back * Time.deltaTime * speed);
16
17          if (Input.GetKey(KeyCode.A))
18              transform.Translate(Vector3.left * Time.deltaTime * speed);
19
20          if (Input.GetKey(KeyCode.D))
21              transform.Translate(Vector3.right * Time.deltaTime * speed);
22
23          if (Input.GetKey(KeyCode.Q))
24              transform.Rotate(Vector3.up * Time.deltaTime * speed * 100f)
25
26          if (Input.GetKey(KeyCode.E))
27              transform.Rotate(Vector3.down * Time.deltaTime * speed * 100
28      }
```

```
30      void LateUpdate(){
31          if (transform.rotation.x != 0 || transform.rotation.z != 0)
32              SetPosRot();
33
34      }
35
36      void OnCollisionEnter(Collision col){
37          if (col.gameObject.CompareTag("ball"))
38              col.gameObject.GetComponent<MeshRenderer>().material.color =
39
40      }
41      void OnCollisionExit(Collision col){
42          if (col.gameObject.CompareTag("ball"))
43              col.gameObject.GetComponent<MeshRenderer>().material.color =
44
45      }
```

```
47      public void SetPosRot(){
48          // Preserve the y-axis rotation, reset x and z rotation
49          float yRotation = transform.rotation.eulerAngles.y;
50          Quaternion newRotation = Quaternion.Euler(0f, yRotation, 0f);
51
52          // Apply position and rotation
53          transform.SetPositionAndRotation(transform.position, newRotation
54
55      }
56  }
57
```

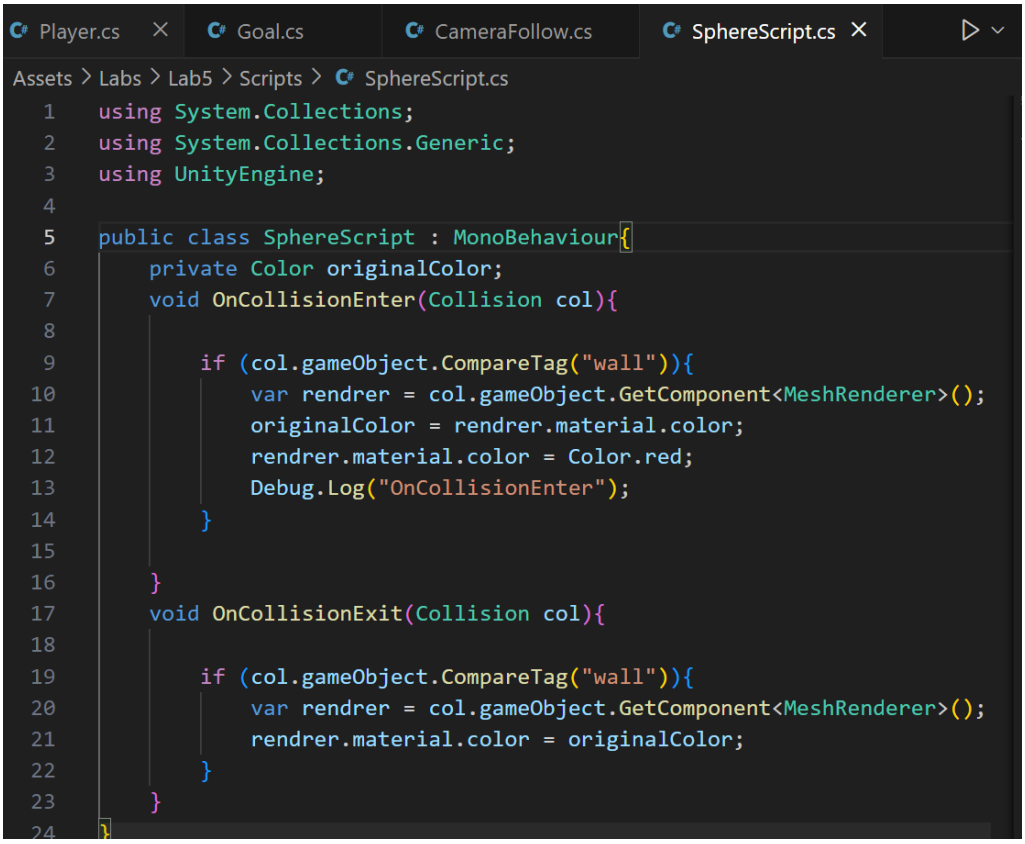
Goal class

Player.cs | Goal.cs | CameraFollow.cs | SphereScript.cs

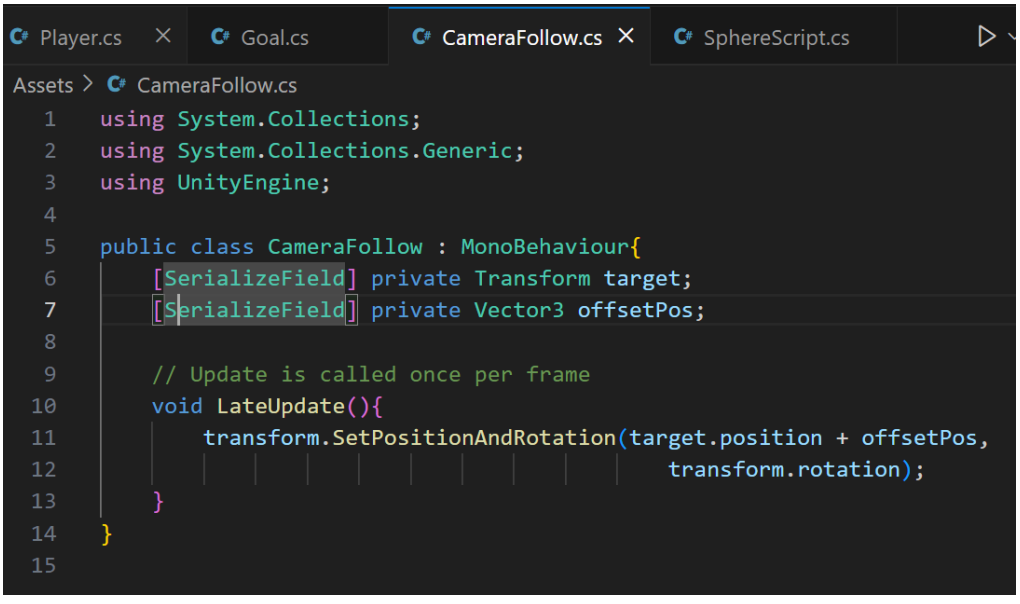
Assets > Labs > Lab5 > Scripts > Goal.cs

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   public class Goal : MonoBehaviour{
6
7       void OnTriggerEnter(Collider other){
8           if (other.gameObject.CompareTag("ball")){
9               var rendrer = gameObject.GetComponent<MeshRenderer>();
10              rendrer.material.color = Color.green;
11          }
12      }
13
14      void OnTriggerExit(Collider other){
15          if (other.gameObject.CompareTag("ball")){
16              var rendrer = gameObject.GetComponent<MeshRenderer>();
17              rendrer.material.color = Color.white;
18          }
19      }
20  }
```

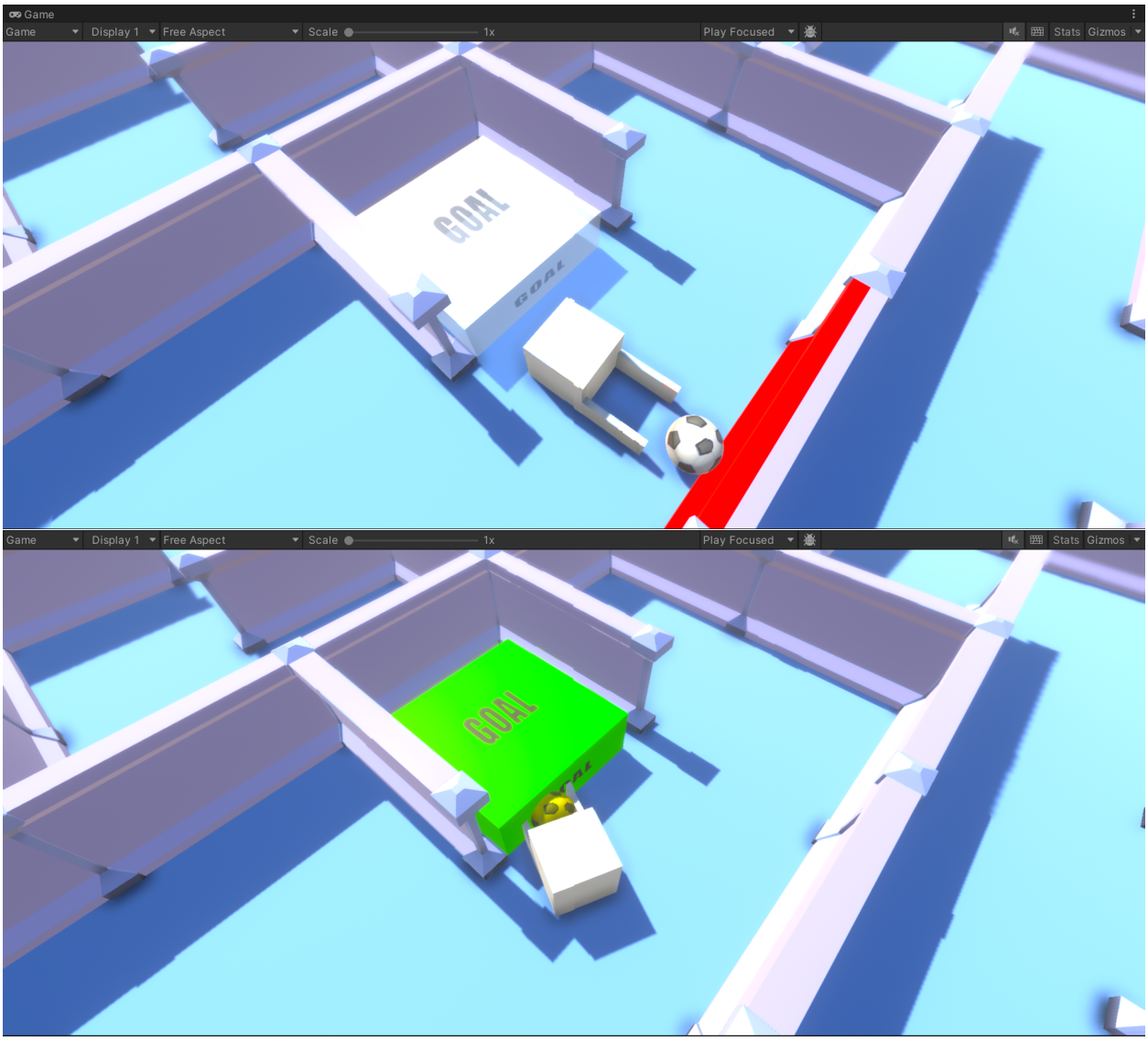
Sphere class



CameraFollow class(additional)



Output:


Game
Game
Display 1
Free Aspect
Scale
1x
Play Focused
Stats
Gizmos
GOAL
GOAL
Game
Display 1
Free Aspect
Scale
1x
Play Focused
Stats
Gizmos
GOAL