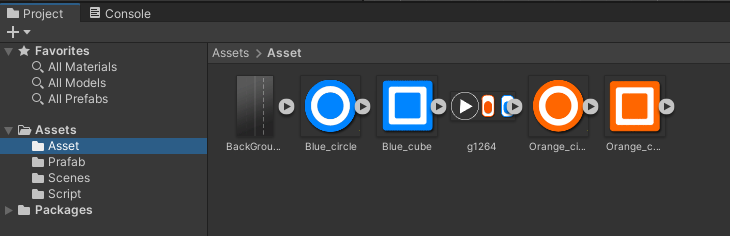
**Лабораторное занятие № 21.**

**Тема: разработка игры “ 2 CARS”**

**Цель: разработать игру “ 2 CARS”**

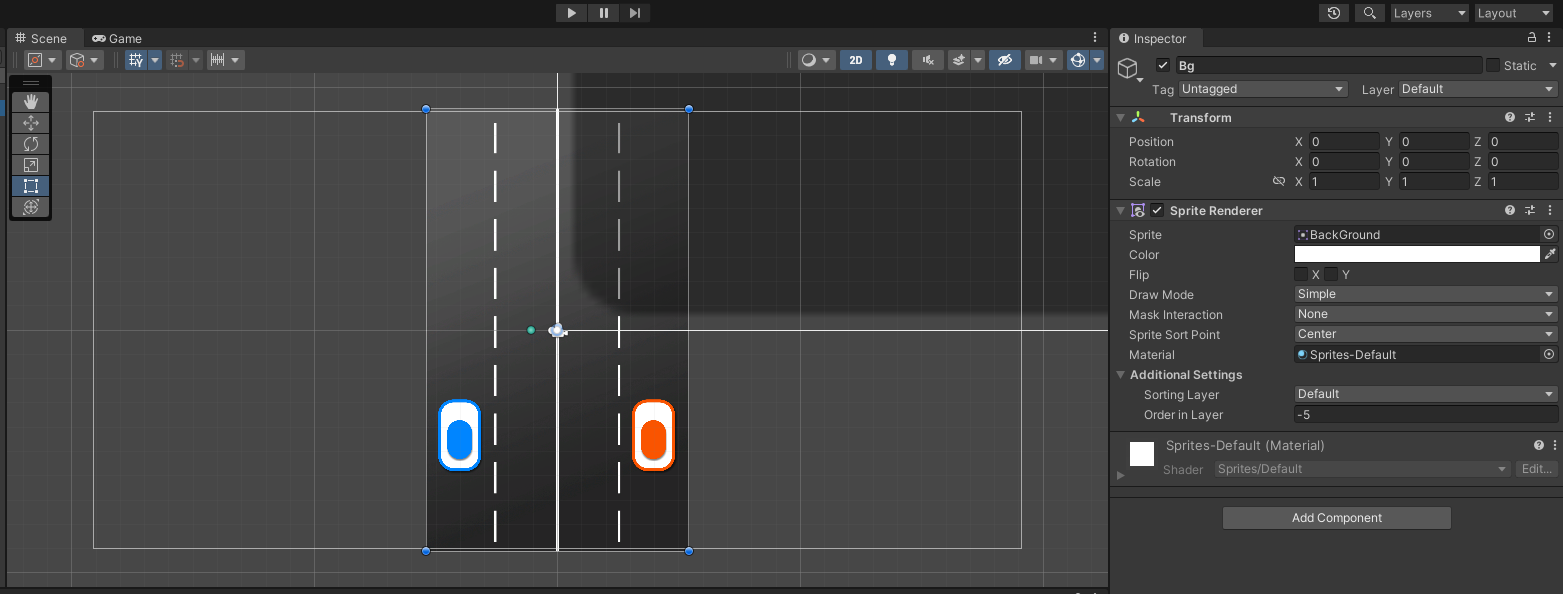
**Ход работы:**

1.Перекидываем папку с ресурсами.



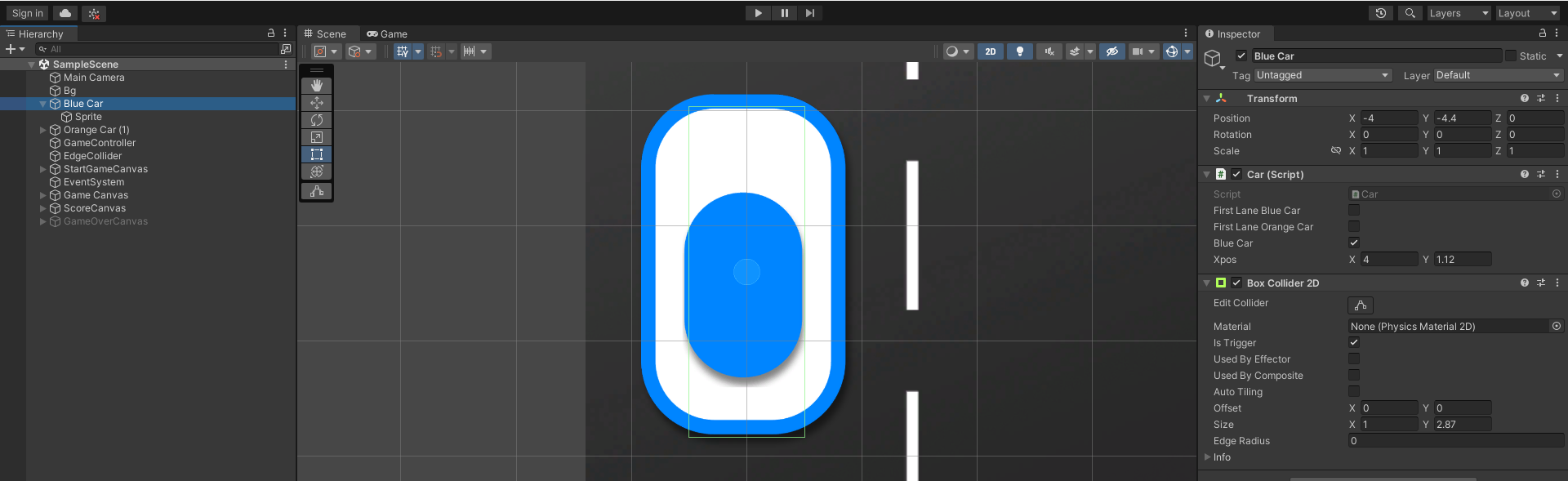
**Рис.21.1** – Папка Sprites.

2.Создаём поле(фон).



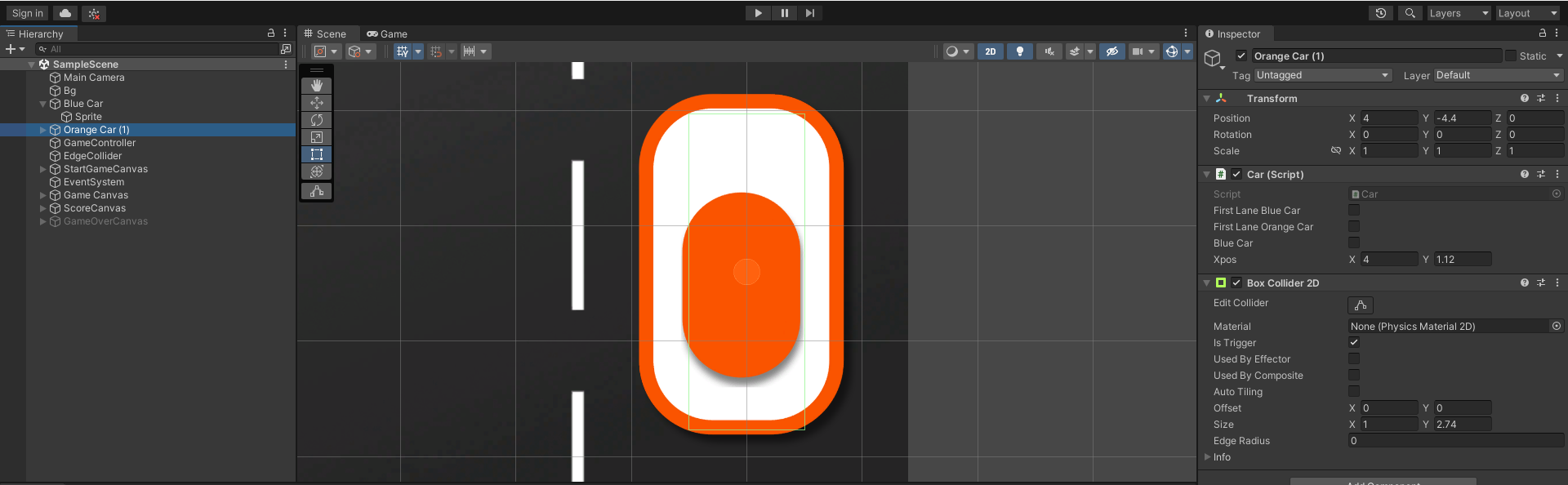
**Рис.21.2** – Создание фона.

3.Создание машинки (Blue Car)



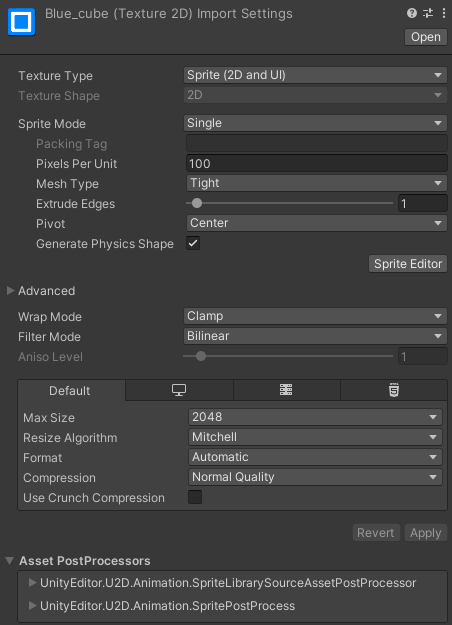
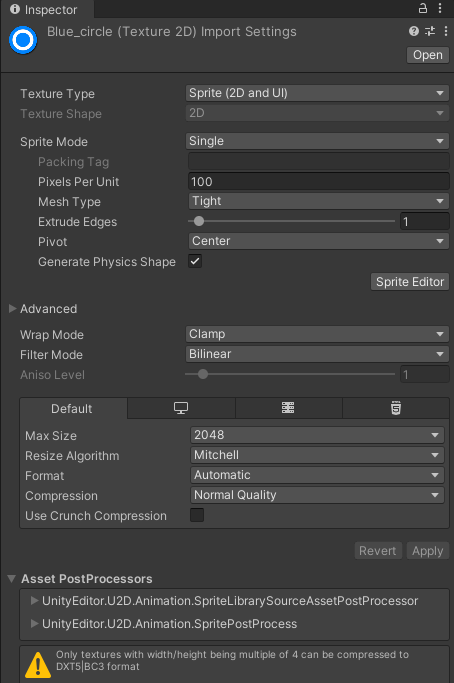
**Рис.21.3** – Blue Car.

4.Создание машинки (Orange Car)



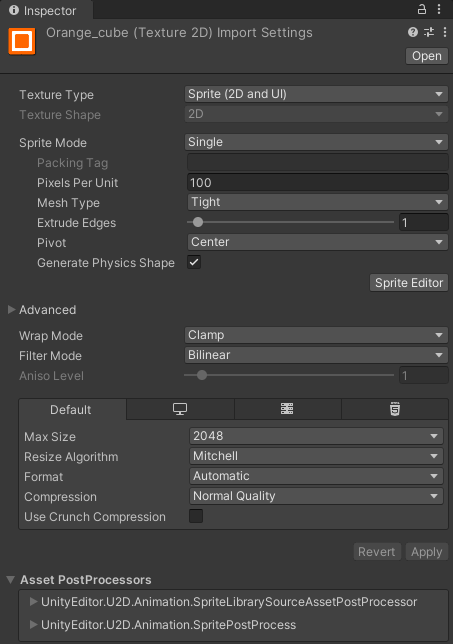
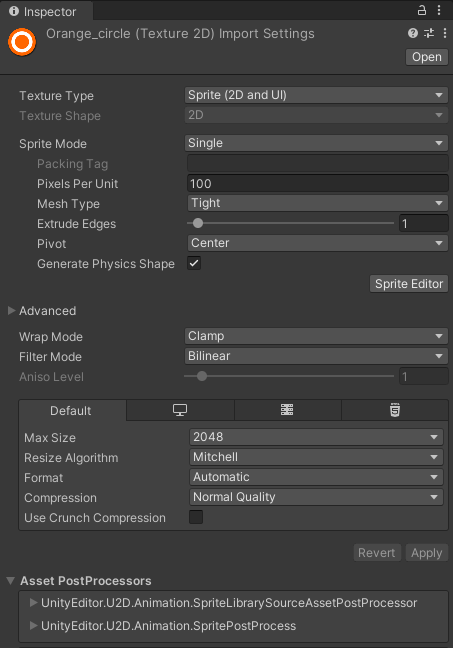
**Рис.21.4** – Orange Car.

5.Создание Circle Blue и Square Blue.



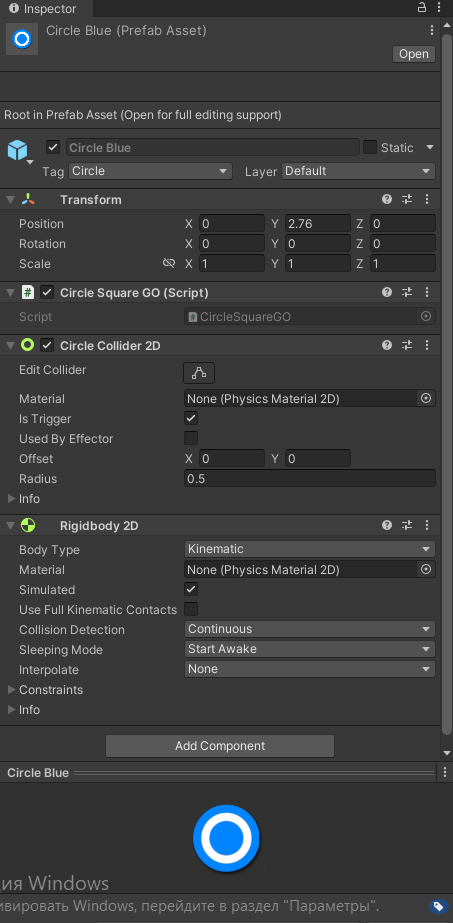
**Рис.21.5** – Circle Blue и Square Blue.

6.Создание Circle Orange и Square Orange.



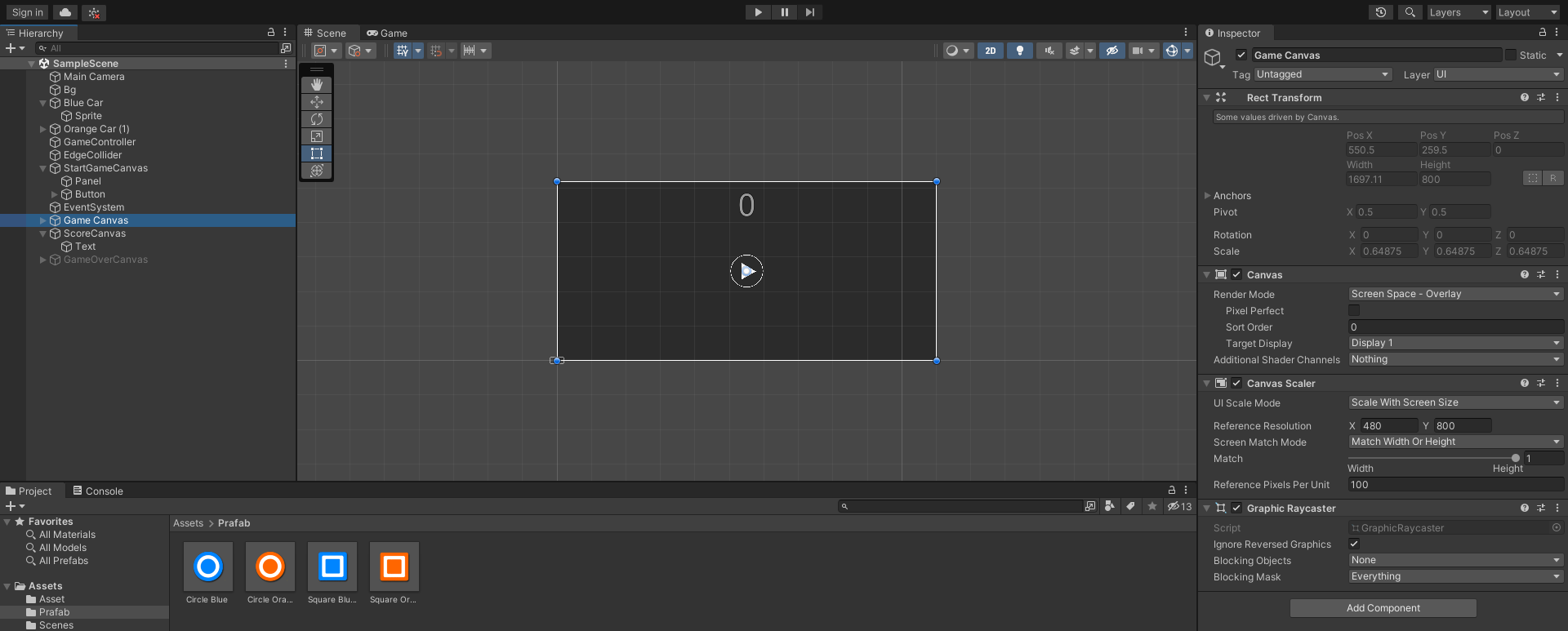
**Рис.21.6** – Circle Orange и Square Orange.

7.Присваиваем компоненты Rigidbody2D и Collider`s.



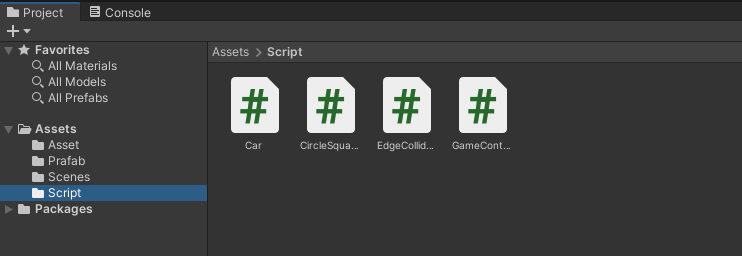
**Рис.21.7** – Компоненты Circle Blue.

8.Создание Меню начала игры и счёта.



**Рис.21.8** – Меню начала игры.

9.Создание скриптов.



**Рис.21.9** – Scripts.

**Скрипты:**

Car.cs:

using UnityEngine;

public class Car : MonoBehaviour

{

public bool FirstLaneBlueCar, FirstLaneOrangeCar;

public bool BlueCar;

public Vector2 Xpos;

// Start is called before the first frame update

void Start()

{

}

// Update is called once per frame

void Update()

{

if (Input.GetKeyDown("left"))

{

LeftButtonPressed();

}

if (Input.GetKeyDown("right"))

{

RightButtonPressed();

}

if (BlueCar)

{

if (FirstLaneBlueCar)

{

transform.position = Vector3.Lerp(transform.position, new Vector3(-Xpos.y, transform.position.y, 0), .1f);

}

else

{

transform.position = Vector3.Lerp(transform.position, new Vector3(-Xpos.x, transform.position.y, 0), .1f);

}

}

else

{

if (FirstLaneOrangeCar)

{

transform.position = Vector3.Lerp(transform.position, new Vector3(Xpos.y, transform.position.y, 0), .1f);

}

else

{

transform.position = Vector3.Lerp(transform.position, new Vector3(Xpos.x, transform.position.y, 0), .1f);

}

}

}

public void LeftButtonPressed()

{

if (FirstLaneBlueCar) { FirstLaneBlueCar = false; } else { FirstLaneBlueCar = true; }

}

public void RightButtonPressed()

{

if (FirstLaneOrangeCar) { FirstLaneOrangeCar = false; } else { FirstLaneOrangeCar = true; }

}

public void OnTriggerEnter2D(Collider2D collision)

{

if (collision.tag == "Circle")

{

//give Points

FindObjectOfType<GameController>().AddScore();

Destroy(collision.gameObject);

}

if (collision.tag == "Cube")

{

//gameOver

FindObjectOfType<GameController>().GameOver();

Debug.Log("playerGO");

}

}

}

CircleSquare.cs:

using UnityEngine;

public class CircleSquareGO : MonoBehaviour

{

int speed;

Rigidbody2D rgbd;

// Start is called before the first frame update

void Start()

{

speed = 10;

rgbd = GetComponent<Rigidbody2D>();

rgbd.velocity = new Vector2(0, -speed);

}

}

EdgeCollider.cs:

using UnityEngine;

public class EdgeCollider : MonoBehaviour

{

public void OnTriggerEnter2D(Collider2D collision)

{

if (collision.tag == "Circle")

{

//gameOver

FindObjectOfType<GameController>().GameOver();

Debug.Log("GameOver");

}

else if (collision.tag == "Cube")

{

//destroy go

Destroy(collision.gameObject);

}

}

}

GameController:

using System.Collections;

using UnityEngine.UI;

using UnityEngine;

using UnityEngine.SceneManagement;

public class GameController : MonoBehaviour

{ //Circle And Square GO

public GameObject[] BlueGO;

public GameObject[] OrangeGO;

//To instantiate squate and circle

public float startWait;

public float spawnWait;

//which GO to Instantiate

GameObject Blue, Orange;

//spawn point spawn randomly X pos

float[] XPosition = new float[2] { 1.5f, 4.6f };

bool GameOverBool;

int Score;

public Text ScoreText;

public GameObject GameOverCanvas;

// Start is called before the first frame update

void Start()

{

GameOverBool = false;

Time.timeScale = 1;

GameOverCanvas.SetActive(false);

}

// Update is called once per frame

void Update()

{

}

IEnumerator SpawnObjects()

{

yield return new WaitForSeconds(startWait);

while (true)

{

for (int i = 0; i < 50; i++)

{

//choosing Xpos

float OrangeXpos = XPosition[Random.Range(0, XPosition.Length)];

//setting position

Vector3 OrangePos = new Vector3(OrangeXpos, 10, 0);

//choosing between square or circle

Orange = OrangeGO[Random.Range(0, OrangeGO.Length)] as GameObject;

//Instantiate now

Instantiate(Orange, OrangePos, Quaternion.identity);

yield return new WaitForSeconds(spawnWait);

//choosing Xpos

float BlueXpos = -XPosition[Random.Range(0, XPosition.Length)];

//setting position

Vector3 BluePos = new Vector3(BlueXpos, 10, 0);

//choosing between square or circle

Blue = BlueGO[Random.Range(0, BlueGO.Length)] as GameObject;

//Instantiate now

Instantiate(Blue, BluePos, Quaternion.identity);

yield return new WaitForSeconds(spawnWait);

}

}

}

public void StartGame()

{

StartCoroutine(SpawnObjects());

}

public void GameOver()

{

GameOverBool = true;

Time.timeScale = 0;

GameOverCanvas.SetActive(true);

}

public void AddScore()

{

Score += 1;

ScoreText.text = Score.ToString();

}

public void Restart()

{

SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex);

}

}