

HAFTA 4 – RAPOR

DERS : OYUN PROGRAMLAMA

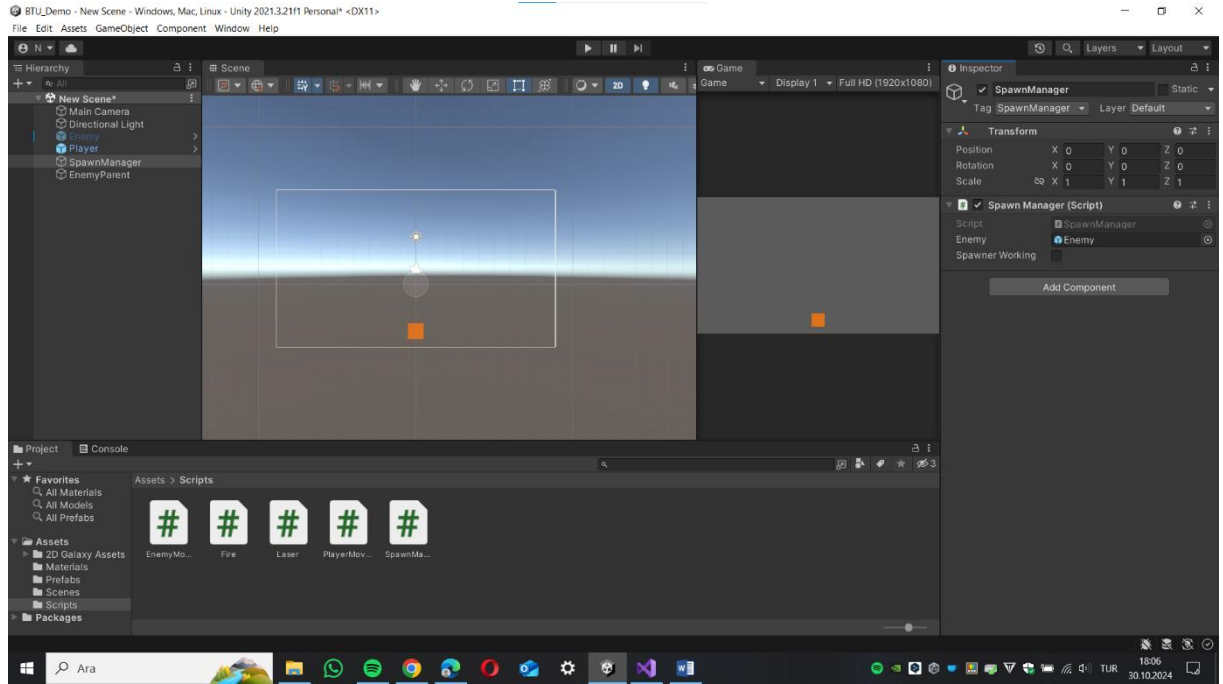
AD SOYAD : LÜTFÜ BEDEL

ÖĞRENCİ NO : 21360859030

GITHUB : https://github.com/lutfubedel/BTU_Oyun_Programlama_Kodlari/tree/main/Hafta_4

BÖLÜM 1 : Spawn Managerın Oluşturulması

Hierarchy Paneli üzerinde sağ tık ile “Create Empty” diyerek boş bir obje oluşturulur ve diğer scriptlerden daha rahat ulaşabilmek için tagını “SpawnManager” olarak verilir ve içerisine enemy oluşturma işlemlerini yapacağımız SpawnManager scripti eklenir.



BÖLÜM 2 : Enemy Objesi Oluşturma

Öncelikle GameObject türünde oluşturulacak olan enemy objesini tutan “enemy” ve enemy objesini oluşturan coroutine ni kontrol eden bool türünde “spawnerWorking” isminde bir değişken tanımlanır.

Ardında zaman ile ilgili işlemleri yapabilmek için “Spawner” adında ve IEnumerator türünde bir coroutine oluşturulur. Enemy oluşturulma işlemi tüm oyun boyunca devam edeceği için öncelikle bir while döngüsü oluşturulur ve “spawnerWorking” değişkeninin değeri true olduğu sürece enemy objesinin oluşturulması sağlanır

Öncelikle her oluşturulan enemy objesi hierarchy panelinde dağınık halde bulunmaması için bir parent objesi belirlenir. Bu parent objesi için sahne üzerinde yeni boş bir obje oluşturulur ve "EnemyParent" tagı verilir. Ardından `GameObject.FindWithTag()` fonksiyonu ile sahnedeki "EnemyParent" tagına sahip obje bulunur ve `parentObj` değişkeninde tutulur.

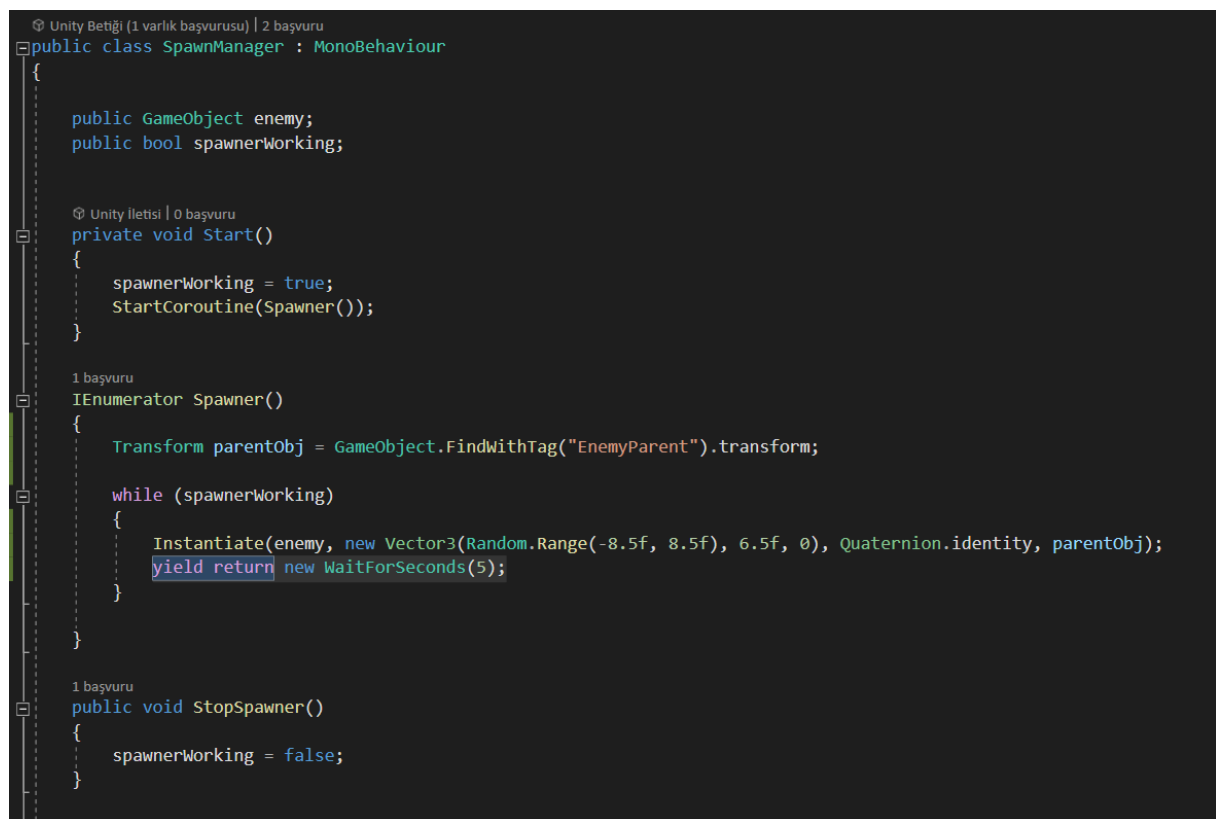
```
Transform parentObj = GameObject.FindWithTag("EnemyParent").transform;
```

Ardından `Instantiate` fonksiyonu ile enemy prefabı ekranın üst kısmında x konumu rastgele olacak şekilde ve parent objesi "parentObj" olacak şekilde oluşturulur.

```
Instantiate(enemy, new Vector3(Random.Range(-8.5f, 8.5f), 6.5f, 0),  
Quaternion.identity, parentObj);
```

Son olarak aynı anda birden fazla enemy objesi oluşturmasını önlemek adına "WaitForSeconds" ile bir sonraki enemy oluşturulana kadar 5 saniye beklemesi sağlanır.

```
yield return new WaitForSeconds(5);
```



```
Unity Betiği (1 varlık başvurusu) | 2 başvuru
public class SpawnManager : MonoBehaviour
{
    public GameObject enemy;
    public bool spawnerWorking;

    Unity İletisi | 0 başvuru
    private void Start()
    {
        spawnerWorking = true;
        StartCoroutine(Spawner());
    }

    1 başvuru
    IEnumerator Spawner()
    {
        Transform parentObj = GameObject.FindWithTag("EnemyParent").transform;

        while (spawnerWorking)
        {
            Instantiate(enemy, new Vector3(Random.Range(-8.5f, 8.5f), 6.5f, 0), Quaternion.identity, parentObj);
            yield return new WaitForSeconds(5);
        }
    }

    1 başvuru
    public void StopSpawner()
    {
        spawnerWorking = false;
    }
}
```

BÖLÜM 3 : Coroutini Durdurma

Enemy objesini oluşturulan coroutini durdurmak için “StopSpawner” adında bir fonksiyon oluşturulur. Bu fonksiyon Player içerisindeki bir script tarafından çağırılacağı için public olarak oluşturulur. “StopSpawner” fonksiyonu coroutine içerisindeki while döngüsünü kontrol eden “spawnerWorking” değişkenini false yaparak enemy oluşturulma işlemini durdurur.

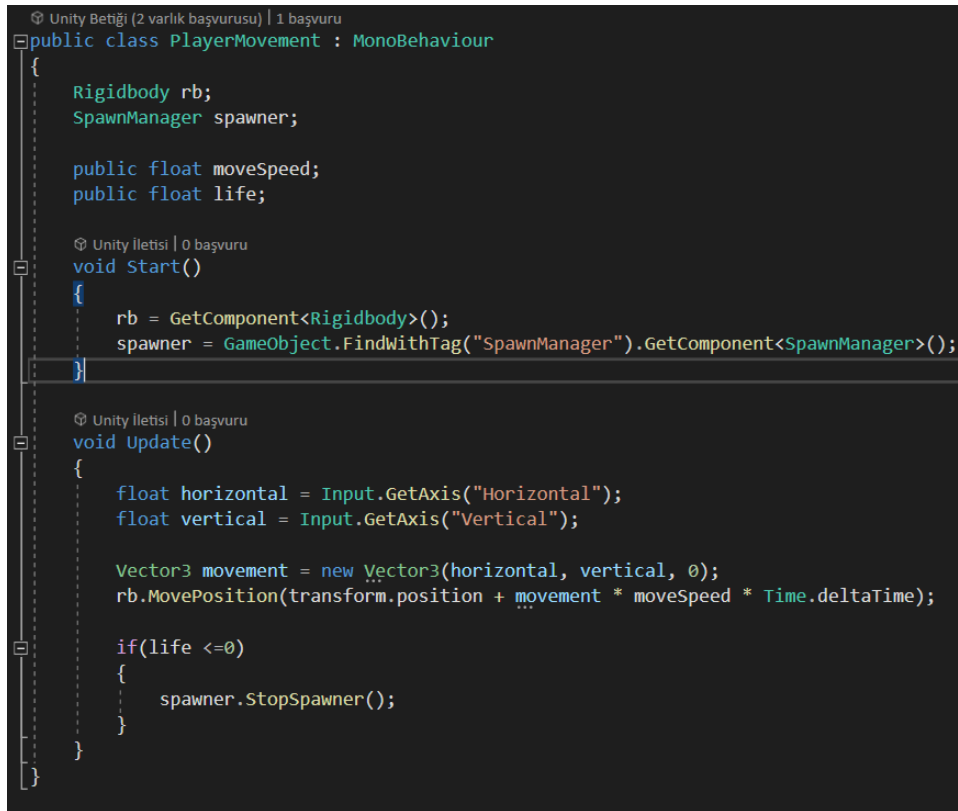
```
public void StopSpawner()
{
    spawnerWorking = false;
}
```

Bu fonksiyon oyuncunun “PlayerMovement” scripti içerisindeki “live” değeri 0 olduğu zaman yani oyuncu kaybettiği zaman çalışacak. O yüzden öncelikle “PlayerMovement” scripti içerisinde “SpawnManager” scripti “spawner” ismi ile tanımlanır ve oyun başlarken sahnede “SpawnManager” tagına sahip obje bulunup bu “spawner” değişkeninde tutulur.

```
spawner =
GameObject.FindWithTag("SpawnManager").GetComponent<SpawnManager>();
```

Ardından oyuncunun life değeri 0 dan küçük ise “spawner” içerisindeki “StopSpawner” fonksiyonu çalıştırılarak enemy objesi oluşturulma işlemi durdurulur.

```
if(life <=0)
{
    spawner.StopSpawner();
}
```



The screenshot shows the Unity Inspector with the PlayerMovement script selected. The code is as follows:

```
public class PlayerMovement : MonoBehaviour
{
    Rigidbody rb;
    SpawnManager spawner;

    public float moveSpeed;
    public float life;

    void Start()
    {
        rb = GetComponent<Rigidbody>();
        spawner = GameObject.FindWithTag("SpawnManager").GetComponent<SpawnManager>();
    }

    void Update()
    {
        float horizontal = Input.GetAxis("Horizontal");
        float vertical = Input.GetAxis("Vertical");

        Vector3 movement = new Vector3(horizontal, vertical, 0);
        rb.MovePosition(transform.position + movement * moveSpeed * Time.deltaTime);

        if(life <=0)
        {
            spawner.StopSpawner();
        }
    }
}
```

