

BLM0364 OYUN PROGRAMLAMA – HAFTA 3 RAPORU

Yasin Ekici

21360859029

Rapor İeriđi;

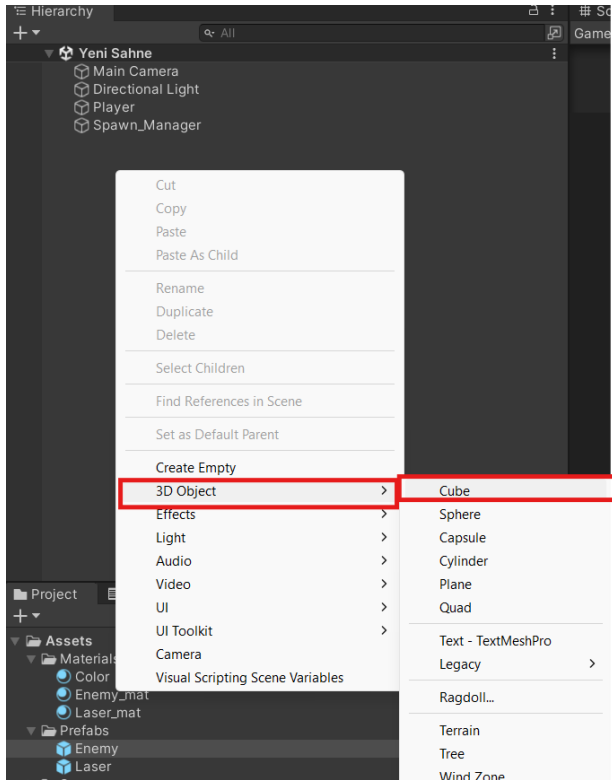
- Enemy prefab oluřturma
- Enemy hareketini sađlayan kodun yazılması
- Rigidbody ve Collision component'lerinin eklenmesi
- OnTriggerEnter fonksiyonu ve arpıřmaların tespiti ile gerekli aksiyonların alınması
- Oyuncu hasar alma iřlemlerinin yapılması

Bu raporda anlatılan kodlar iin: <https://github.com/YasinEkici/game-programming>

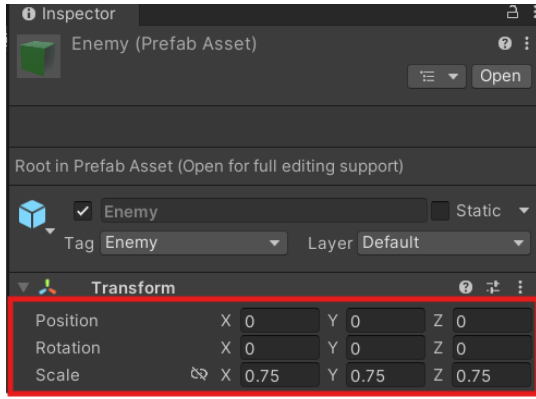
1. Enemy Nesnesi Oluřturma ve Prefab Yapma:

Nesne Oluřturma:

- Sahneye yeni bir Cube ekleyin.



- Konumunu (0, 0, 0) olarak ayarlayın.
- öleđini (0.75, 0.75, 0.75) yapın.
- İřmini "Enemy" olarak deđiřtirin.

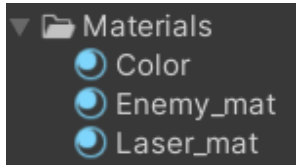


- “Enemy” nesnesini “Prefabs” klasörüne sürükleyerek prefab haline getirin.



Düşman Materyali Oluşturma:

- “Materials” klasöründe “Enemy_mat” adında yeni bir materyal oluşturun.
- Rengini kırmızı olarak ayarlayın.
- Bu materyali “Enemy” nesnesine uygulayın.



2. Enemy Davranışını Tanımlama ve Çalıştırma:

Düşman Davranışını Tanımlama:

- “Enemy_sc” adında yeni bir C# script’i oluşturun.
- Bu script’i “Enemy” prefab’ına ekleyin.

Hareket:

- Düşman nesnesi sahneye geldiğinde aşağı doğru hareket etsin. Bunun için şunları sağlamanız lazım;
- Bir speed değişkeni tanımlayın;
`float speed = 3.0f;`
- Şimdi Update fonksiyonu içerisinde şu kodu yazalım;
`transform.Translate(Vector3.down * speed * Time.deltaTime);`

Ekran Dışına Çıkma:

- Eğer düşman nesnesi y ekseninde -5.4f’ten küçük bir değere ulaşırsa, onu ekranın üst kısmında (y = 7f) -9f ile 9f arasında bir x değeriyle yeniden konumlandırın.

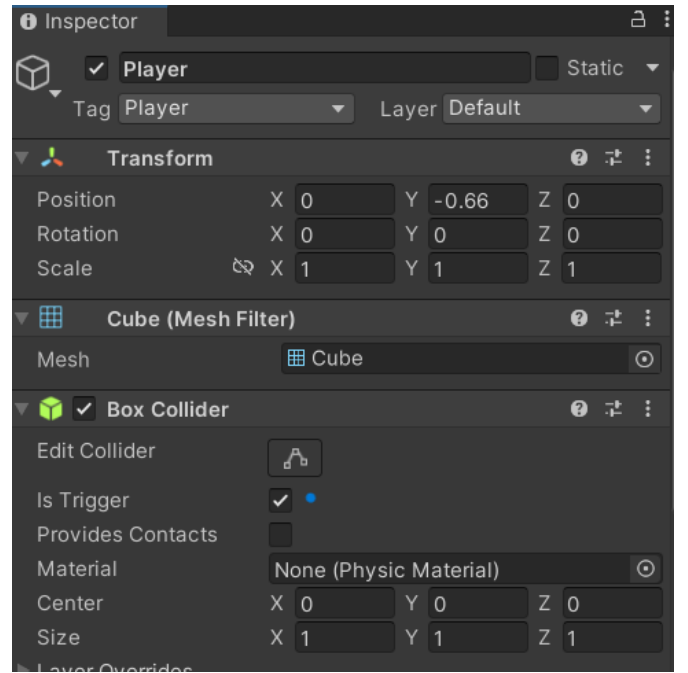
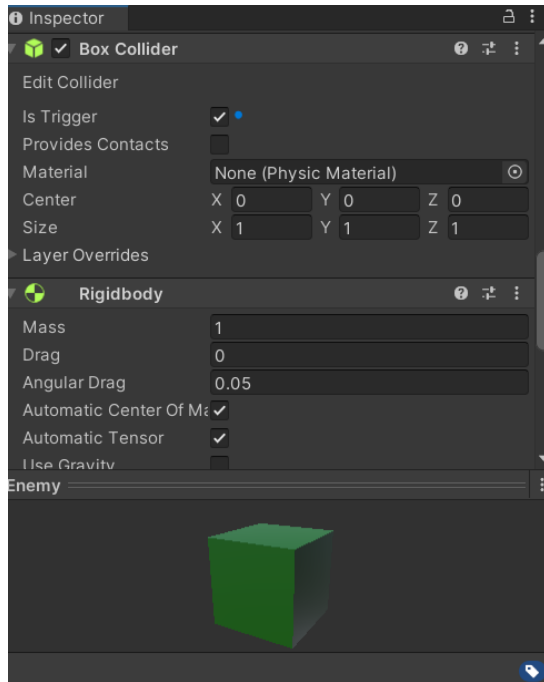
- Update fonksiyonu içersine;

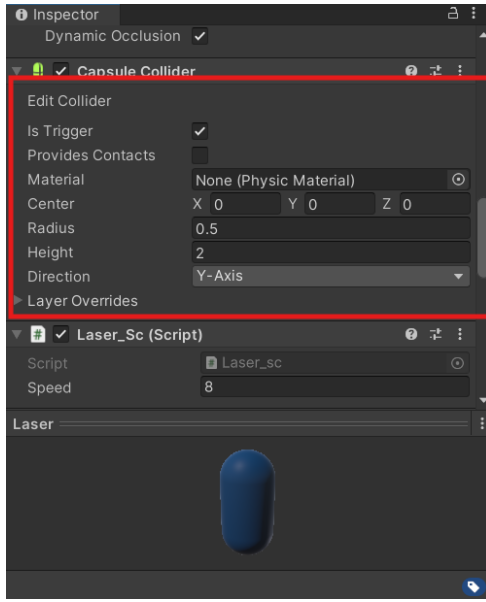
```
if(transform.position.y < -5.4f){  
    float randomX = Random.Range(-9f, 9f);  
    transform.position = new Vector3(randomX, 7f, 0);  
}
```

3. Çarpışmalar:

Collider Ekleme;

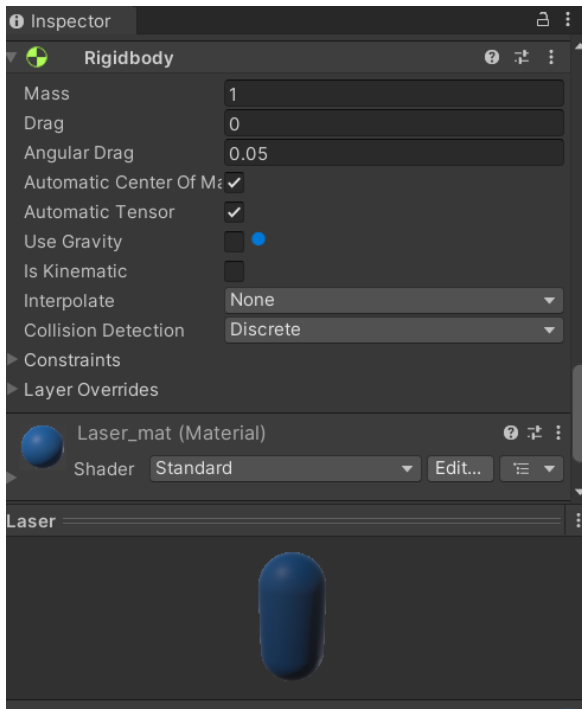
- Çarpışmaları algılamak için Collider kullanın.
- Düşman, Player ve Laser nesneleri arasında sert yüzey çarpışmalarını istemediğimiz için “isTrigger” özelliğini aktif hale getirin.





Laser-Enemy'den hangisi Rigidbody alacak?

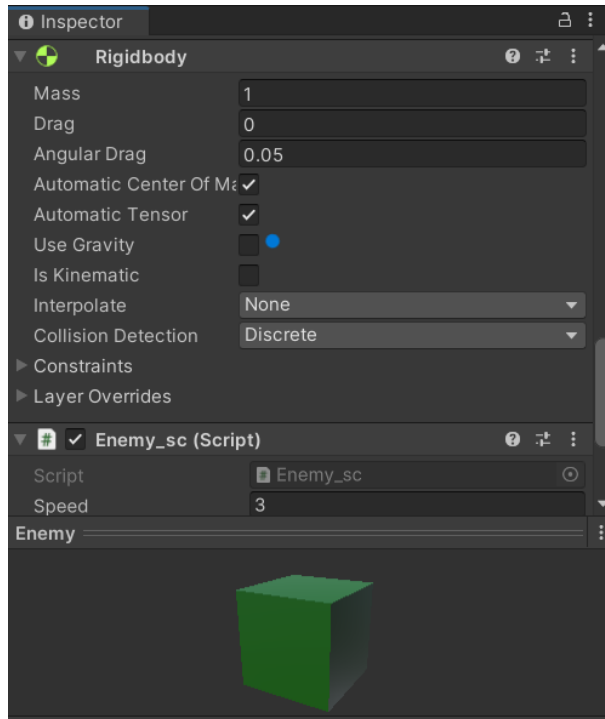
Performans nedenlerinden dolayı sadece “Laser” prefab’ine Rigidbody ekleyin ve Apply Gravity’yi kapatın.



Trigger otomatik olarak algılanabilir. Hangi objelerin birbiriyle collide olduğunu anlayabiliyoruz.

Player-Enemy'den hangisi Rigidbody alacak?

Performans nedenlerinden ötürü “Enemy” prefab'ine Rigidbody ekleyin ve Apply Gravity'yi kapatın.



Eğer “Enemy” “Player” ile çarpışırsa;

- Player’a Damage vermeniz gerekir.
- Enemy’i yok etmeniz gerekir.

Eğer “Enemy” “Laser” ile çarpışırsa;

- Laser’i yok etmeniz gerekir.
- Enemy’i yok etmeniz gerekir.

Bunun için yeni bir fonksiyon tanımlayacağız;

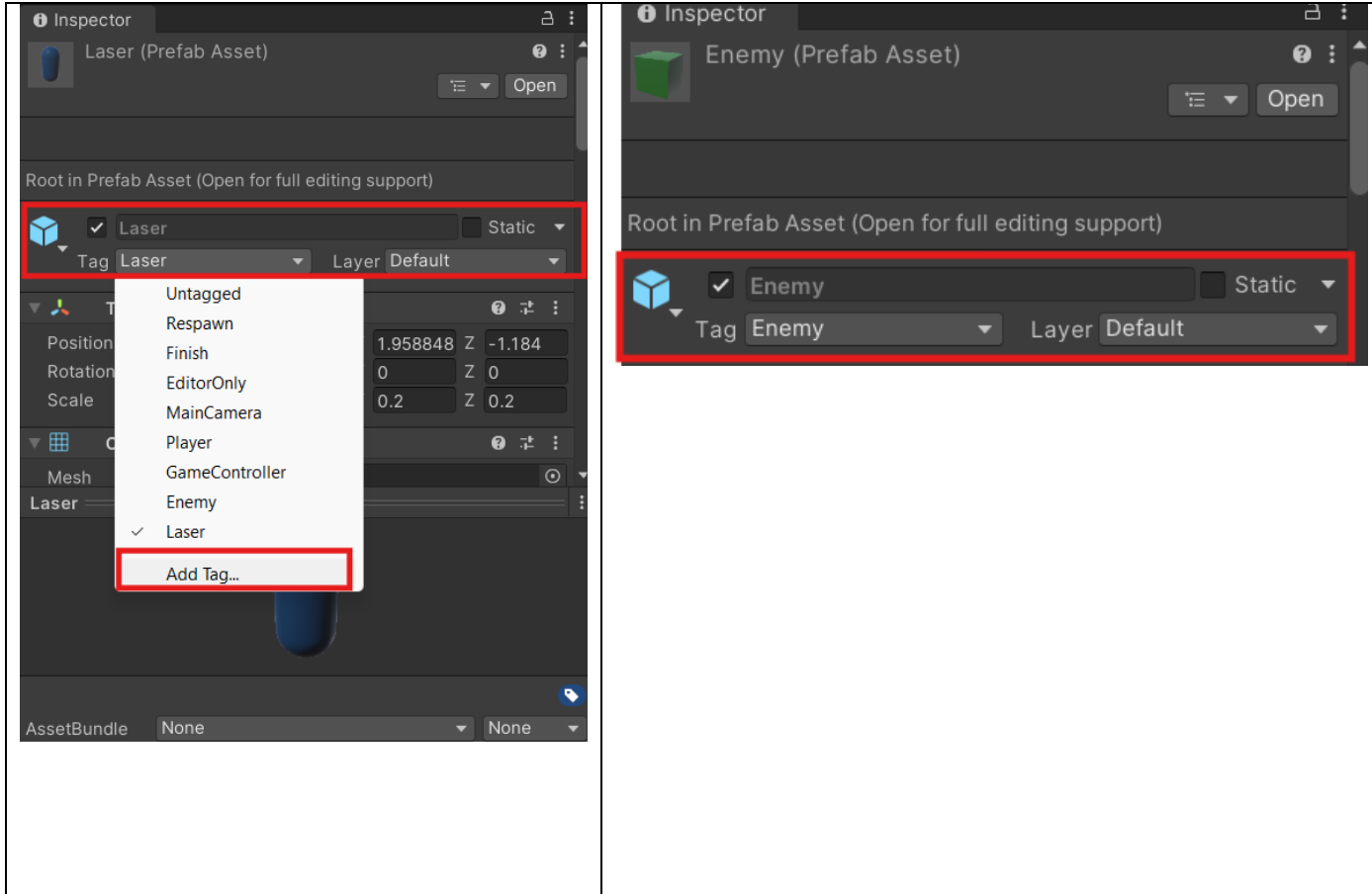
```
void OnTriggerEnter(Collider other){  
    if (other.tag == "Player"){  
        Destroy(this.gameObject);  
    }  
    else if (other.tag == "Laser"){  
        Destroy(other.gameObject)  
        Destroy(this.gameObject)  
    }  
}
```

}

}

Enemy, Laser ve Player objelerini nasıl ayırt edeceğiz?

2 yeni Tag eklememiz gerekiyor: Enemy ve Laser. Unity hali hazırda Player Tag'ini içeriyor.



Oyuncunun canı da bir değişkendir;

- Player'ın can değişkenini Player script'inde tanımlayın ve başlangıç değerini 3 olarak ayarlayın.
- Can değeri private olarak ayarlanmalı, böylece diğer programlar tarafından değiştirilemez.

```
private int lives = 3;
```

- "Damage" adında bir fonksiyon tanımlayıp Player objesinin yok edilme kısmını halledeceğiz

```
void Damage(){
```

```
lives--;
```

```
if (lives < 1)
```

```
    Destroy(this.gameObject);
```

```
}
```

```
}
```

- Enemy, Player'a çarptığında zarar fonksiyonunu çağırmak için script iletişimini sağlayın.
- other.transform üzerinden doğrudan nesnenin transform bileşenine erişin ve GetComponent fonksiyonunu kullanarak diğer bileşenlere ulaşın.
- OnTriggerEnter kodunu değiştirmemiz gerekir;

```
void OnTriggerEnter(Collider other){
```

```
    if(other.tag == "Player"){
```

```
        Player_sc player = other.transform.GetComponent<Player_sc>();
```

```
        if(player != null){
```

```
            player.Damage();
```

```
        }
```

```
        Destroy(this.gameObject);
```

```
    }
```

```
    else if (other.tag == "Laser"){
```

```
        Destroy(other.gameObject);
```

```
        Destroy(this.gameObject);
```

```
    }
```

```
}
```