

Öğrenci: Ahmet Yusuf Birdir

Numara: 21360859026

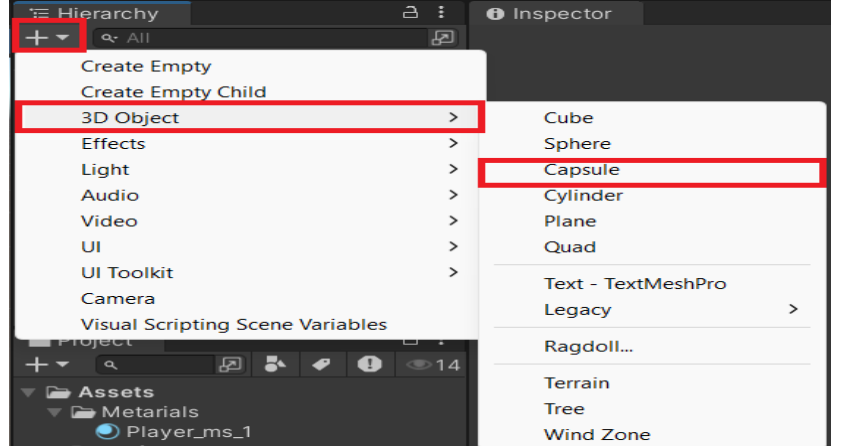
## Laser Prefab Oluşturma / Capsule Ekleme / Ölçeklendirme / Materyal Ekleme

Öncelikle Asset klasörünün içerisine bir Prefabs kalsörü oluşturuyoruz.

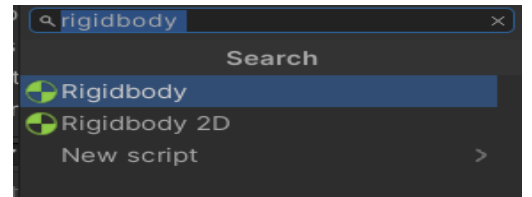
Ardından Hierarch paneline bir capsule nesnesi oluşturuyoruz ve Laser ismini veriyoruz.

Daha sonra Hierarchy penceresinden laser nesnemize tıklayarak inspector panelinin transform kısmından laser nesnemizi ölçeklendiriyoruz.

Bundan sonra Materials klasörünün içerisine bir materyal oluşturuyoruz ve bunu Laser nesnemize sürükleyerek bağlıyoruz.



Son olarak oluşturduğumuz Laser prefab'ı Project panelinden seçiyoruz. Ardından Inspector panelinden en alt kısmından add components'e tıklayarak açılan pencereden prefab'a rigidbody parçasını ekliyoruz.



## Laser Prefab'ı Hareket Ettirme / Laser Ateşleme

Öncelikle script klasörümüze bir adet Laser\_sc adında bir script klasörü oluşturuyoruz. Bu nesnemize ait bir hız özelliği oluşturuyoruz ve update fonksiyonunun içine yandaki şekilde ekleme yapıyoruz. Bu eklemede if bloğu nesnemiz ekran dışına çıktığında nesneyi yok etmektedir.

```
public class Laser_sc : MonoBehaviour
{
    [SerializeField]
    1 reference
    float speed = 10;
    0 references
    void Start()
    {
    }

    // Update is called once per frame
    0 references
    void Update()
    {
        transform.Translate(new Vector3(0, 1, 0) * Time.deltaTime * speed);
        if(transform.position.y > 6){
            Destroy(this.gameObject);
        }
    }
}
```

Ardından bu nesnemizi oyuncumuzla bağlamalıyız. Bunu yapmak için oyuncu script dosyamıza ekleme yapmamız gerekiyor. Bu ekleme şekildeki gibi olmalı:

Bir laser nesnesi .

```
public GameObject laserPrefab;
```

İleride karışıklık olmaması açısından hareketleri kontrol eden ayrı bir fonksiyon.

Update fonksiyonu içerisinde her space tuşuna basıldığında bir laser nesnesi oluşturan bir if bloğu ve update içinde oluşturduğumuz hareket fonksiyonunu çağırmak.

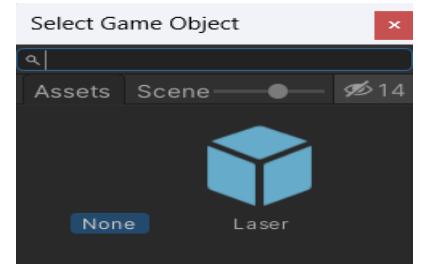
```
void Update()
{
    CalculateMovement();
    if(Input.GetKeyDown(KeyCode.Space))
    {
        Instantiate(laserPrefab, transform.position + new Vector3(0, 0.8f, 0), Quaternion.identity);
    }
}

1 reference
void CalculateMovement()
{
    horizontal = Input.GetAxis("Horizontal");
    vertical = Input.GetAxis("Vertical");
    space = Input.GetAxis("Jump");

    Vector3 direction = new Vector3(horizontal, vertical, 0);
    transform.Translate(direction * Time.deltaTime * speed);
}
```

Bunların ardından unity'de Hierarchy kısmından nesnemizi seçiyoruz. Inspector panelinde Player\_sc dosyamız ve içinde oluşturduğumuz laserPrefab nesnesi gözükmektedir. Bu nesnenin yanındaki tuşa tıklayarak açılan pencereden assets kısmına gelip Laser prefab'ini seçiyoruz ve nesnemiz ile prefab bağlanmış oluyor.

Laser Prefab None (Game Object)



## Seri Atışlar Arası Bekleme Süresi Ekleme

Bunun için iki ekleme yapamız gerekiyor;

Atış hızı ve sonraki atış süresi için iki değişken. Ardından player script dosyasındaki update fonksiyonu içinde bulunan if bloğuna ufak bir değişim.

```
float fireRate = 0.5f;  
2 references  
float nextFire = 0f;
```

```
if(Input.GetKeyDown(KeyCode.Space) && Time.time > nextFire)  
{  
    Instantiate(laserPrefab, transform.position + new Vector3(0, 0.8f,  
    0), Quaternion.identity);  
    nextFire = Time.time + fireRate;  
}
```

Burada if koşulu içindeki ekleme bir sonraki atış için gerekli zamanı kontrol eder, böylelikle kullanıcı belirli bir süre zarfı içinde yeniden ateş edemez. İç işlemlerde ise her defasında sıradaki atış zamanı güncellenerek atış mantığı korunmuş olur.