

1. Üçlü Laser Nesnesi Oluşturma

- Laser prefab'inden iki kopya daha oluşturun. Her biri için pozisyonlarını ayarlayın, böylece bir tanesi merkeze, diğer ikisi sağa ve sola doğru açılı bir şekilde çıkacak.

```
void Shoot()
{
    if (Input.GetKeyDown(KeyCode.Space) && Time.time >= nextFireTime) // Zaman kontrolü
    {
        if (TripleShotActive)
        {
            Instantiate(bulletPrefab, transform.position + new Vector3(0, 1, 0), Quaternion.identity);
            Instantiate(bulletPrefab, transform.position + new Vector3(0.8f, 0.1f, 0), Quaternion.identity);
            Instantiate(bulletPrefab, transform.position + new Vector3(-0.8f, 0.1f, 0), Quaternion.identity);
        }
        else
        {
            Instantiate(bulletPrefab, transform.position + new Vector3(0, 1, 0), Quaternion.identity);
            // Instantiate(bulletPrefab, transform.position + new Vector3(0, 1, 0), Quaternion.identity);
            nextFireTime = Time.time + fireRate; // Bir sonraki atış zamanını ayarla
        }
    }
}
```

2. Üçlü Bonus Etkinleştirme Mantığını Kurma

- Bir üçlü bonus alınca veya belirli bir koşul sağlandığında üçlü bonusu etkinleştirecek bir bool değeri Player içinde tanımlayın.
- Bu değişkeni kontrol ederek, FireLaser fonksiyonunun normal laser mi yoksa üçlü laser mi oluşturacağını belirleyin.

3. Üçlü Bonus Sprite ve Etkinleştirme Süresi

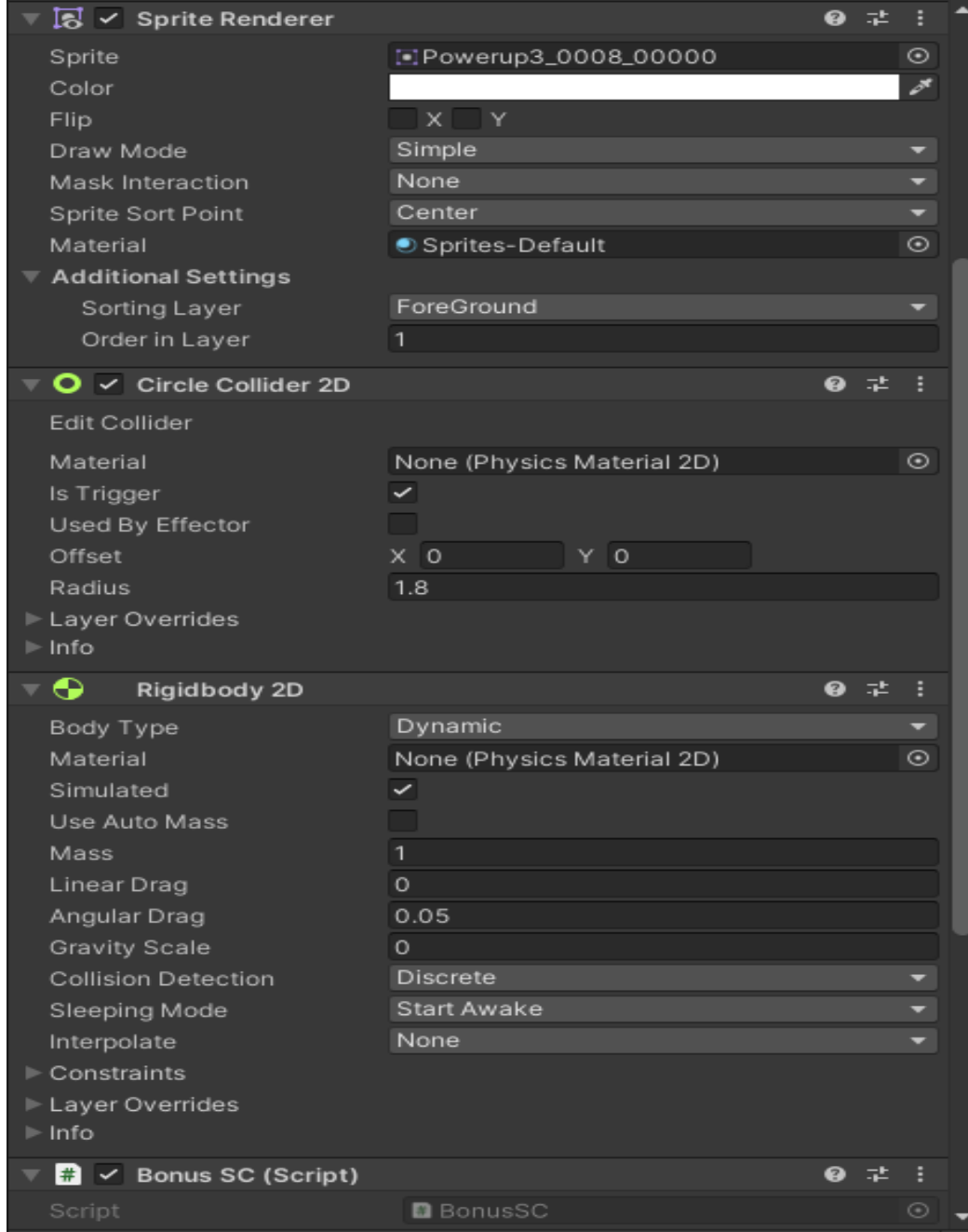
- Oyuna üçlü bonusu toplayabileceğiniz bir sprite ekleyin. Bu nesne toplandığında, üçlü atış modunu belirli bir süre etkin hale getirecek bir kod yazın. Örneğin, tripleShotDuration değişkeniyle bu sürede üçlü atış yapılmasını sağlayabilirsiniz.
- Bir coroutine (örneğin TripleShotPowerDownRoutine) kullanarak üçlü atış süresinin dolmasını sağlayın.

```
1 reference
public void TripleShotActiveMethod()
{
    TripleShotActive = true;
    // 5 saniye sonra TripleShotActive'i false yap
    StartCoroutine(TripleShotPowerDownRoutine());
}

//Coroutine
1 reference
IEnumerator TripleShotPowerDownRoutine()
{
    yield return new WaitForSeconds(5);
    TripleShotActive = false;
}
```

4. Üçlü Bonus Sprite'ı için Nesne Ayarlamaları

- Üçlü bonus sprite'ından bir nesne oluşturun ve oyun alanında toplamak için uygun bir boyuta ayarlayın.
- Nesneye Circle Collider 2D ve Rigidbody 2D ekleyin; Rigidbody'yi Kinematic moduna ayarlayın.
- Sorting Layer ayarlarını yaparak görsel olarak doğru katmanda görünmesini sağlayın.



5. Üçlü Bonus Script'i

- Üçlü bonus nesnesi için bir script oluşturun ve bonusu belirli bir hızla aşağı doğru hareket ettirin. Update fonksiyonunda hareket için transform.Translate kullanabilirsiniz.
- Nesne ekran dışına çıktığında Destroy komutuyla nesneyi yok edin.
- OnTriggerEnter2D fonksiyonunda Player ile çarpışmayı kontrol edin ve çarpıştığında üçlü atış bonusunu etkinleştirin.

```
1 reference
float mvspeed = 5;
// Update is called once per frame
0 references
void Update()
{
    transform.Translate(Vector3.down * mvspeed * Time.deltaTime);
    if (transform.position.y >= 3)
    {
        Destroy(this.gameObject);
    }
    else if (transform.position.y <= -11)
    {
        Destroy(this.gameObject);
    }
    if (transform.position.x >= 12)
    {
        Destroy(this.gameObject);
    }
    else if (transform.position.x <= -13)
    {
        Destroy(this.gameObject);
    }
}
```

6. Player Kodunda Üçlü Bonus'u Etkinleştirme

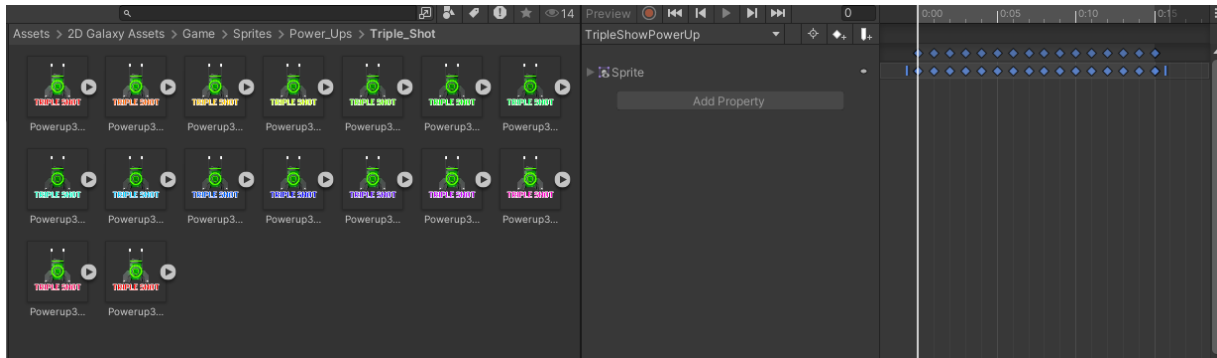
- Player script'inize bir referans ekleyerek, TripleShot etkin hale geldiğinde üçlü atış modunu açın. Çarpışma sonucu tetiklenen üçlü bonus kodu, burada etkinleştirilecektir.

```
void OnTriggerEnter2D(Collider2D other)
{
    Debug.Log(other.transform.name + " Çarpıştı");
    if (other.tag == "Player")
    {
        PlayerSC player = other.transform.GetComponent<PlayerSC>();
        player.TripleShotActiveMethod();

        Destroy(this.gameObject);
    }
    if (other.tag == "Bullet")
    {
        Debug.Log("Bullet Bonus ile Çarpıştı");
    }
}
```

7. Üçlü Atış Bonus Nesnesi için Animasyon Ekleyin

- Üçlü bonus sprite'ına bir animasyon ekleyin. Örneğin, parlama veya dönme animasyonu ekleyebilirsiniz.



8. SpawnManager Düzenlemeleri

- SpawnManager kodunu güncelleyerek üçlü bonus nesnesinin belirli zaman aralıklarıyla ortaya çıkmasını sağlayın. Powerup mantığını kullanarak üçlü bonus da rastgele olarak spawn edilecek nesneler arasına eklenebilir.

```
0 references
void Start()
{
    StartCoroutine(SpawnEnemies()); // Coroutine başlatılıyor
    StartCoroutine(SpawnPowerUps());
}

1 reference
IEnumerator SpawnPowerUps()
{
    // PowerUp'ları her spawnInterval sürede bir üretiyor
    while (spawnActive)
    {
        SpawnPowerUp();
        yield return new WaitForSeconds(spawnIntervalPowerUp);
    }
}

1 reference
void SpawnPowerUp()
{
    // PowerUp'ı oluştur ve parent nesnesinin altına yerleştir
    GameObject newPowerUp = Instantiate(powerUpPrefab, GetRandomSpawnPosition(), Quaternion.identity);
    newPowerUp.transform.parent = powerUpParent;
}
```

Github: <https://github.com/emirrdvn/OyunProgramlama>