

1. Spawn Manager Boş Oyun Nesnesi ve Script'i Oluşturma

Spawn işlemlerini yönetecek bir oyun nesnesi ve script oluşturacağız.

Adımlar:

1. **Hierarchy** penceresine sağ tıklayın ve Create Empty seçeneğini seçin. Bu nesneye "SpawnManager" adını verin.
2. **Project** penceresinde "Scripts" klasörüne sağ tıklayın, Create > C# Script seçeneğini seçin, script'e SpawnManager adını verin.
3. **SpawnManager** script'ini açın ve aşağıdaki adımları ekleyin.

2. SpawnManager Script'i ile Her 5 Saniyede Bir Düşman Üretme

SpawnManager script'inde Coroutine kullanarak düşmanları belirli aralıklarla sahneye ekleyeceğiz.

SpawnManager Script'i:

```
public class SpawnManagerSC : MonoBehaviour
{
    1 reference
    public GameObject enemyPrefab; // Üretilecek düşman prefab'i
    1 reference
    public Transform enemyParent; // Düşmanların toplanacağı parent nesne
    1 reference
    public float spawnInterval = 5f; // Spawn aralığı
    2 references
    private bool spawnActive = true; // Spawn işlemini durdurmak için kullanılacak
    0 references
    void Start()
    {
        StartCoroutine(SpawnEnemies()); // Coroutine başlatılıyor
    }
    1 reference
    IEnumerator SpawnEnemies()
    {
        // Düşmanları her spawnInterval sürede bir üretiyor
        while (spawnActive)
        {
            SpawnEnemy();
            yield return new WaitForSeconds(spawnInterval);
        }
    }
    1 reference
    void SpawnEnemy()
    {
        // Düşmanı oluştur ve parent nesne (field) GameObject SpawnManagerSC.enemyPrefab
        GameObject newEnemy = Instantiate(enemyPrefab, GetRandomSpawnPosition(), Quaternion.identity);
        newEnemy.transform.parent = enemyParent;
    }
    1 reference
    Vector3 GetRandomSpawnPosition()
    {
        // Düşmanın üretileceği rastgele bir pozisyon belirleyin (örnek)
        float xPos = Random.Range(-10f, 10f);
        float zPos = Random.Range(-10f, 10f);
        return new Vector3(xPos, 0, zPos);
    }
    1 reference
    public void StopSpawning()
    {
        spawnActive = false;
    }
}
```

Açıklama:

- **enemyPrefab:** Üretilecek düşman prefab'ini belirtiyoruz.
- **enemyParent:** Tüm düşmanların toplanacağı parent nesne.
- **spawnActive:** Spawn işlemini durdurmak için kullanacağız.
- **SpawnEnemies Coroutine'i:** spawnActive true olduğu sürece her spawnInterval sürede bir düşman üretiyor.

3. Parent Container Oyun Nesnesi Oluşturma

Oluşturulan tüm düşmanları toplamak için bir parent container oluşturacağız.

Adımlar:

1. **Hierarchy** penceresinde sağ tıklayın ve Create Empty seçin.
2. Bu nesneye "Enemies" adını verin.
3. **SpawnManager** nesnesine gidin ve enemyParent alanına Enemies nesnesini sürükleyip bırakın.

4. Oyuncu Yok Olduğunda Spawn İşlemini Durdurma

Oyuncu yok olduğunda spawn işlemini durdurmak için SpawnManager script'inde bir fonksiyon yazdık: StopSpawning(). Şimdi oyuncu yok olduğunda bu fonksiyonu çağırmamız gerekiyor.

4.1 PlayerSC Script'i ile Oyuncu Yok Olduğunda Spawn İşlemini Durdurmak

Oyuncunun yok olma anında SpawnManagerdaki StopSpawning() fonksiyonunu çağıracağız.

```
public void Damage()
{
    health--;
    if (health <= 0)
    {
        Debug.Log("Oyuncu öldü!");
        spawnManager.StopSpawning(); // Spawn işlemi durduruluyor
        Destroy(gameObject); // Oyuncu yok ediliyor
    }
}
```

5. Hierarchy Ayarları

Unity Editor'de gerekli nesneleri ayarlamak için aşağıdaki adımları izle:

1. **SpawnManager** nesnesini seçin.
2. **Inspector** penceresinde:
 - Enemy Prefab alanına üretilecek düşman prefab'ini ekleyin.
 - Enemy Parent alanına Enemies nesnesini sürükleyip bırakın.

Bu adımlarla oyunda her 5 saniyede bir düşman üreten ve oyuncu öldüğünde bu işlemi durduran temel bir spawn yönetimi sistemini kurmuş olduk.

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