

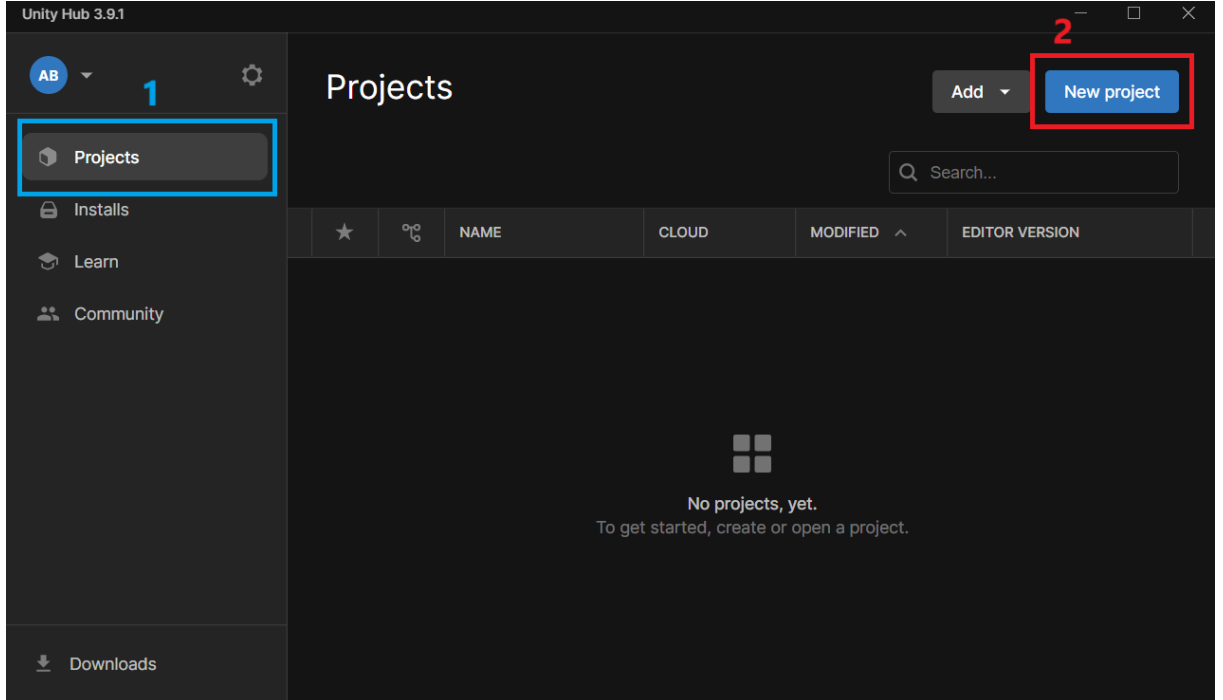
Öğrenci : Ahmet Yusuf Birdir

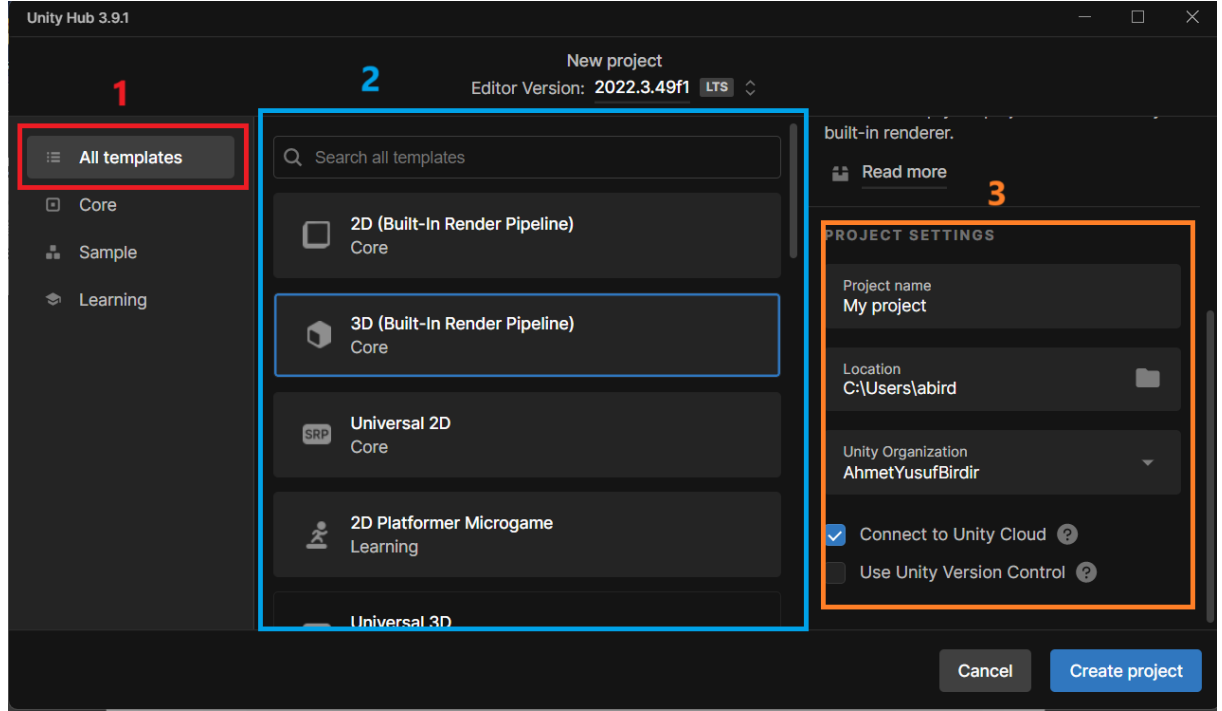
Numara : 21360859026

İlk olarak Unity Hub uygulamasını indirmelisiniz. Ardından uygulama ekranından Unity 2022 FTS sürümünü bilgisayarınızı kurmalısınız. Bu işlemleri tamamladıktan sonra devam eden işlemleri yapabilirsiniz.

### 1- Yeni Proje Oluşturma:

İlk olarak Unity Hub uygulamasınız açın, ardından sol paneldeki **Project** kısmına tıklayın. Bu kısımdan var olan projelerinizi görebilirsiniz. Sonra sağ üst kısımdaki **New Project** bölümüne tıklayın.



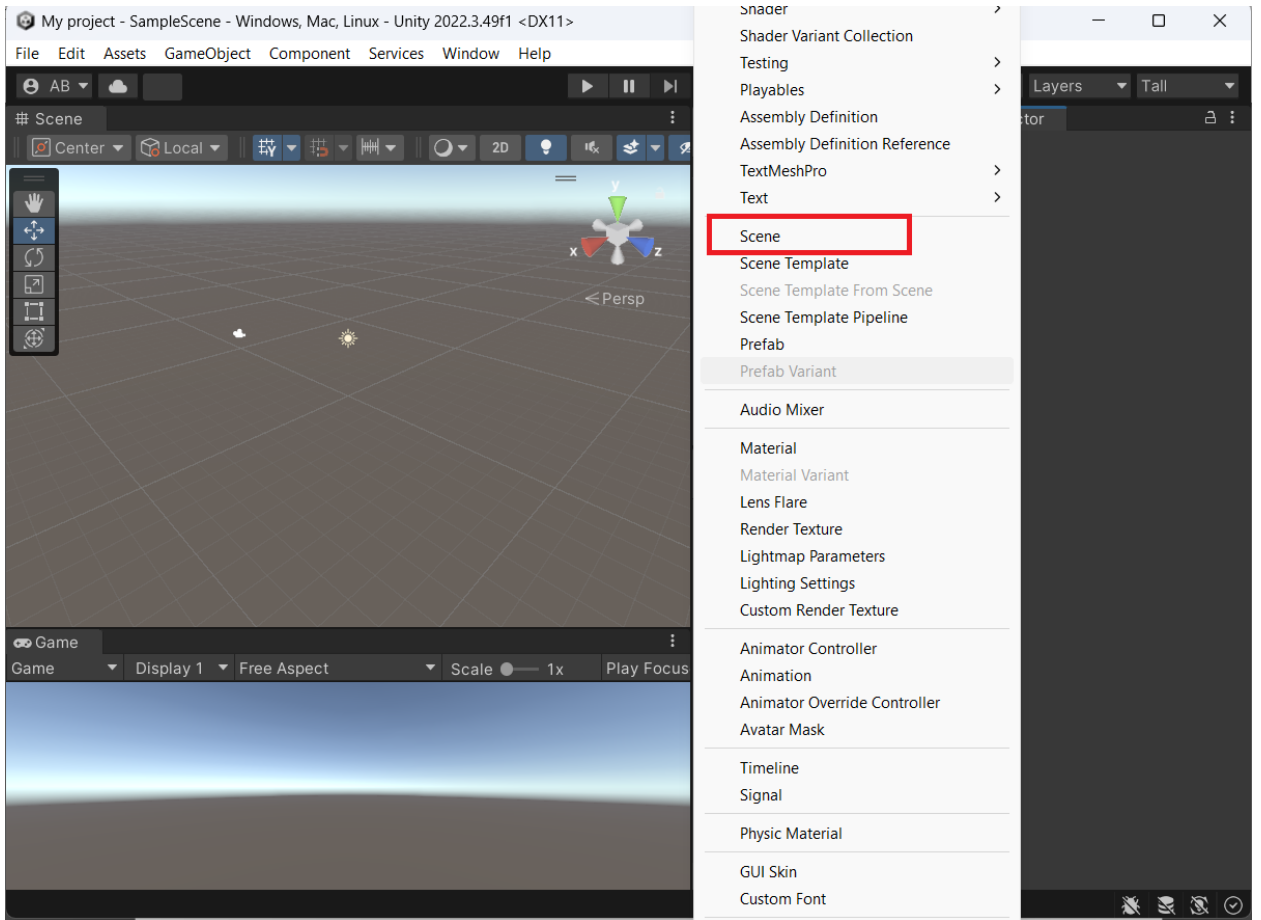
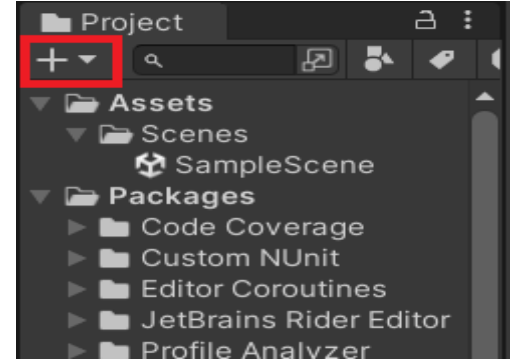


Karşınıza açılan pencerede 1. taraftaki panelden **all templates** kısmını seçtikten sonra 2. kısımdaki panelden çalışmak istediğiniz türü seçebilirsiniz. Son olarak ise 3. Kısımdaki bölümden projenize isim verebilir ve bilgisayarınızda nereye kaydedileceğini seçebilirsiniz.

## 2. Sahne Ekleme/Silme ve Sahneye Nesne Ekleme

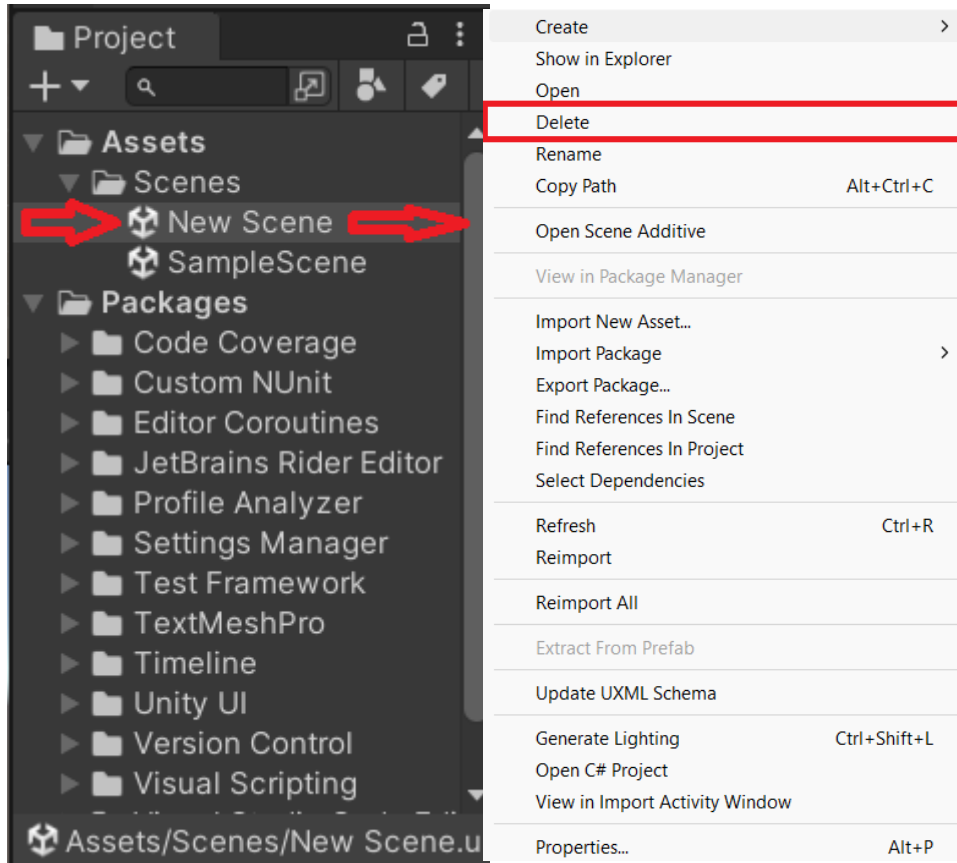
### Sahne Ekleme

Projenizi açtıktan sonra karşınıza gelen ekranda Project panelinin altındaki “+” tuşuna tıkladıktan sonra Scene tuşuna tıklayıp yeni bir sahne oluşturabilirsiniz.



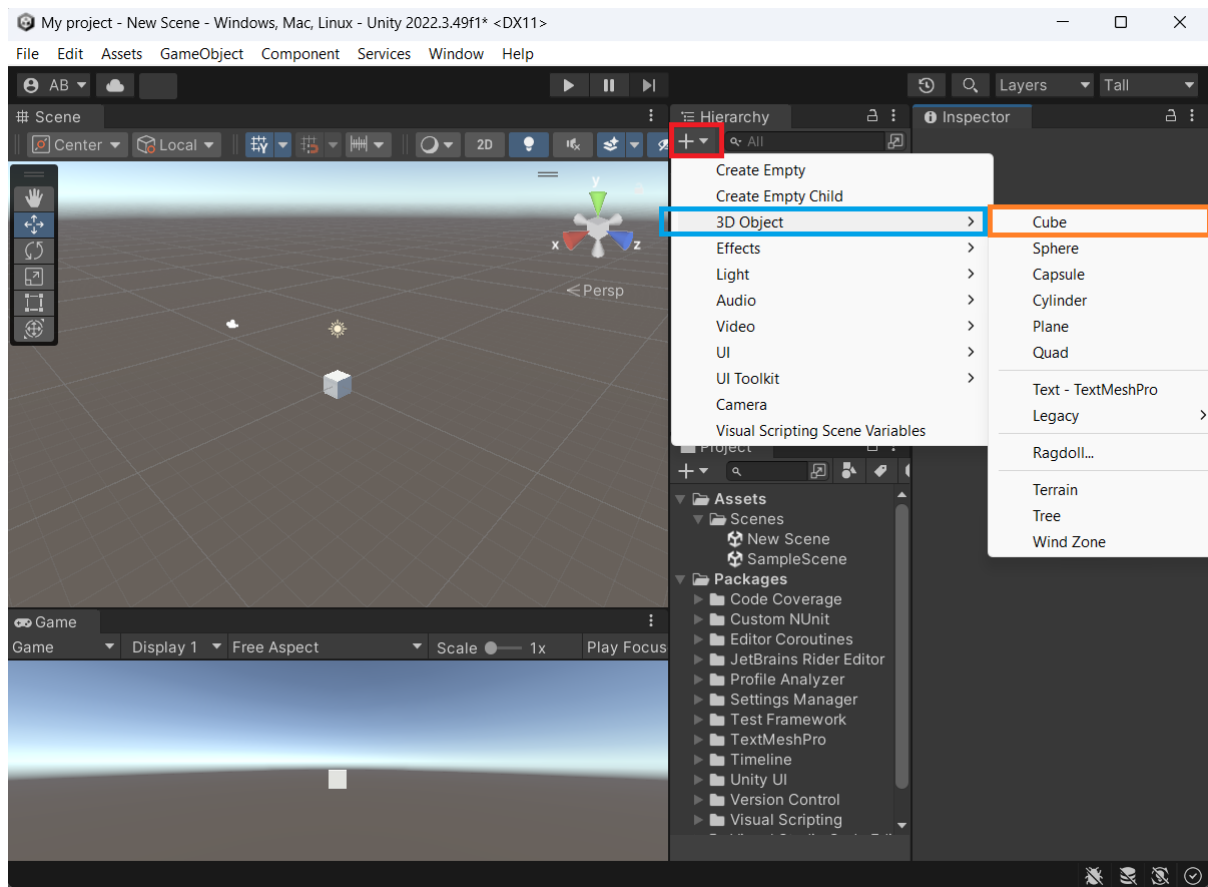
## Sahne Silme

Oluřturulan bu sahneyi silmek iin ise Project paneline gelip silmek istediėiniz sahnenin zerine saė tıklayıp Delete iřlemine tıklayarak silebilirsiniz.



## Nesne Ekleme

Sahneye nesne eklemek için ise Hierarchy paneline hemen altındaki “+” tuşuna tıklayarak açılan panelden 3D Object ‘i seçip açılacak diğer panelden Cube seçeneğini seçtikten sonra sahnenize bir küp nesnesi eklenecektir.

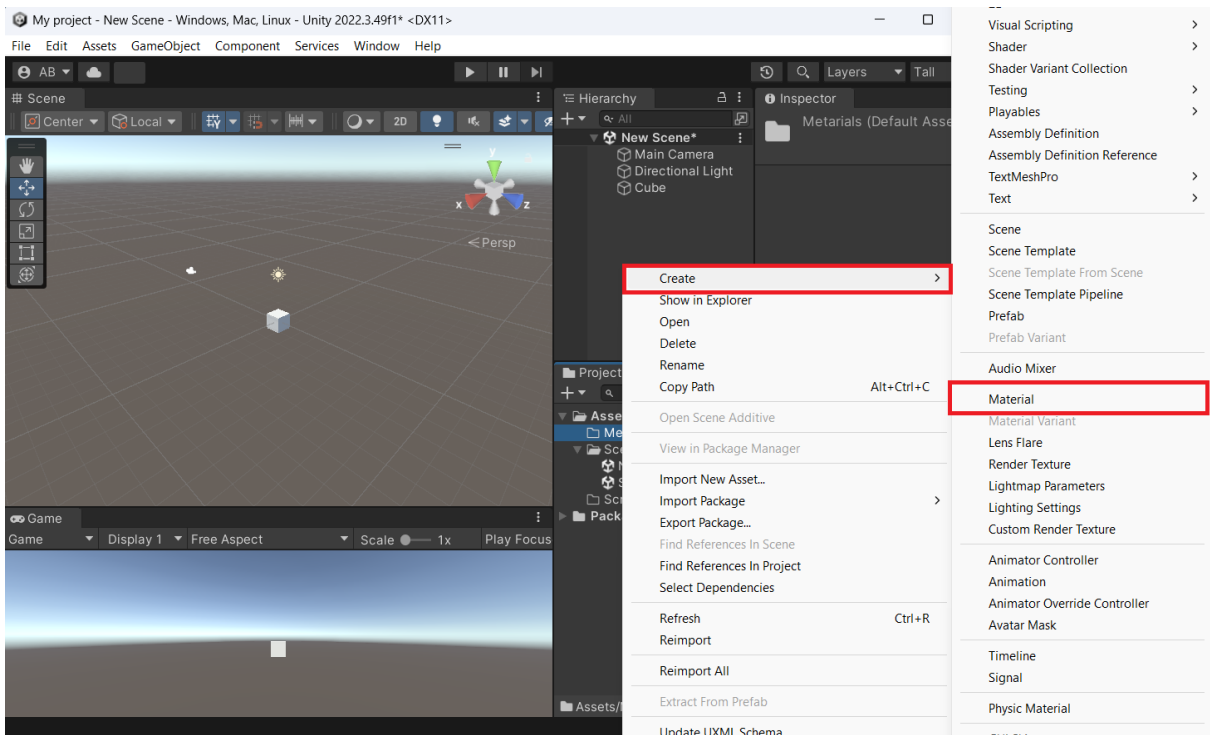


# Nesne İşlemleri

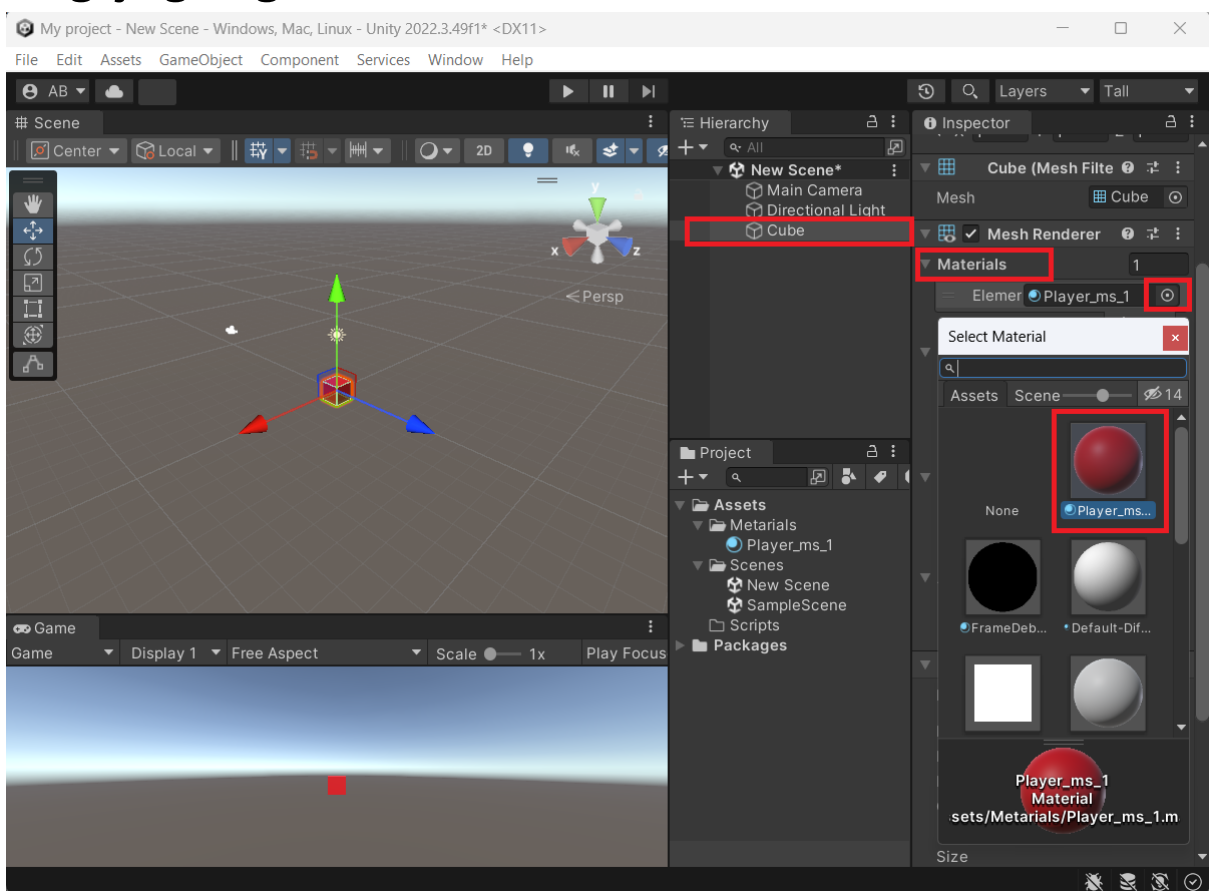
Öncelikle daha önceki Project paneline gelip Assets klasörünün içine Scripts ve Materials adında 2 dosya oluşturuyoruz.

## 1-Nesneye Materyal Ekleme

İlk önce Assets klasörüne oluşturduğumuz Materials dosyasına tıklayarak açılan panelden Create kısmını seçin. Ardından açılan panelden Material seçeneğini seçerek bir materyal oluşturun ve bu materyale bir isim verin.

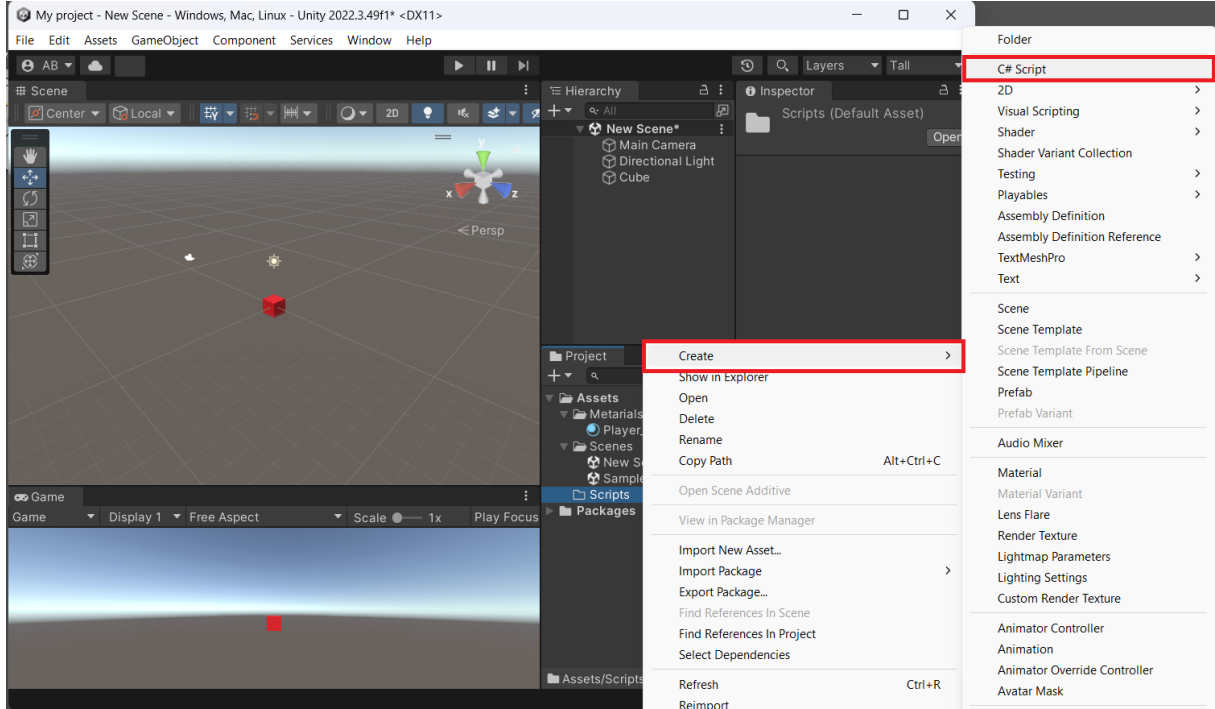


The image consists of two vertically stacked screenshots of a 3D environment. Both screenshots show a gray floor with a white grid pattern. In the background, there is a yellow sun and a white cloud. In the top screenshot, a blue cube is positioned in the center of the frame. In the bottom screenshot, a red cube is positioned in the same location.



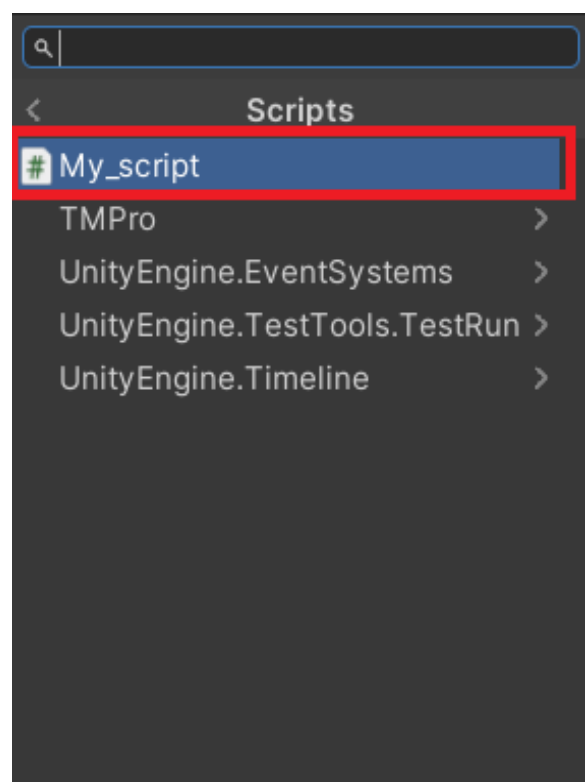
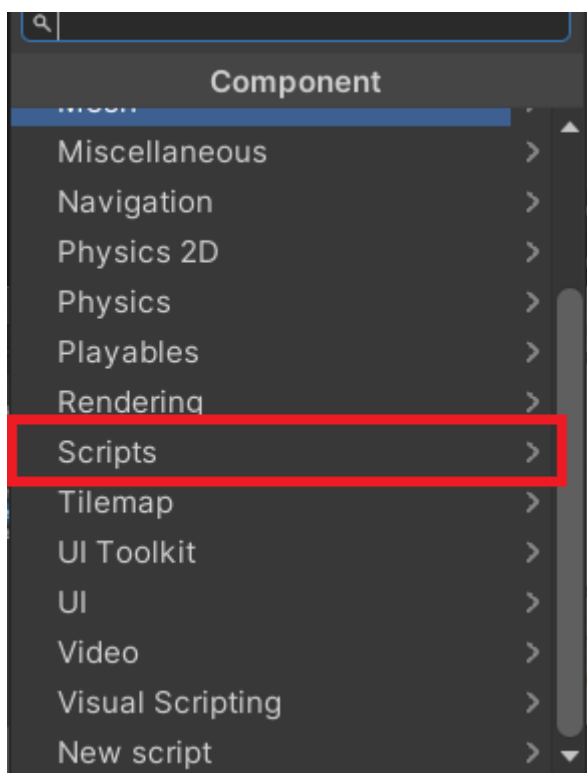
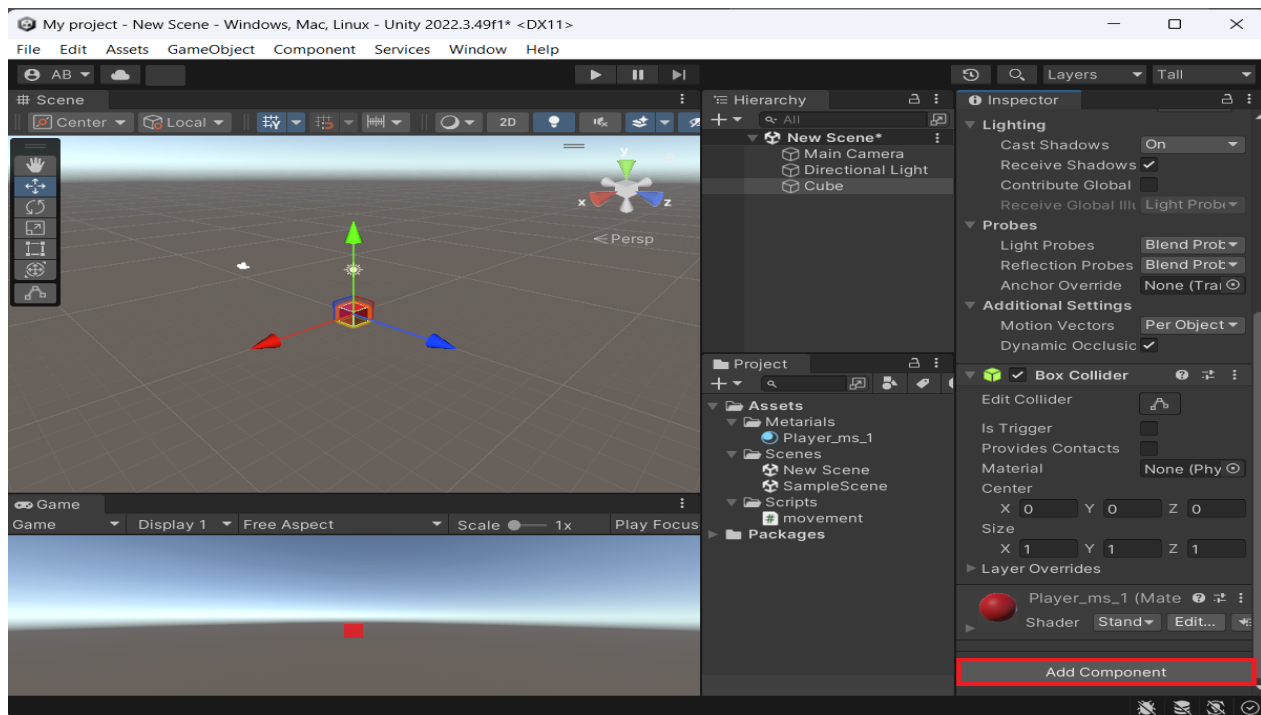
## 2- Nesneye Script Ekleme

Daha önceden oluşturduğunuz Script klasörüne tıklayarak açılan panelden Create seçeneğini seçip ardından açılan panelden de C# script kısmını seçerek bir script dosyası oluşturabilirsiniz.



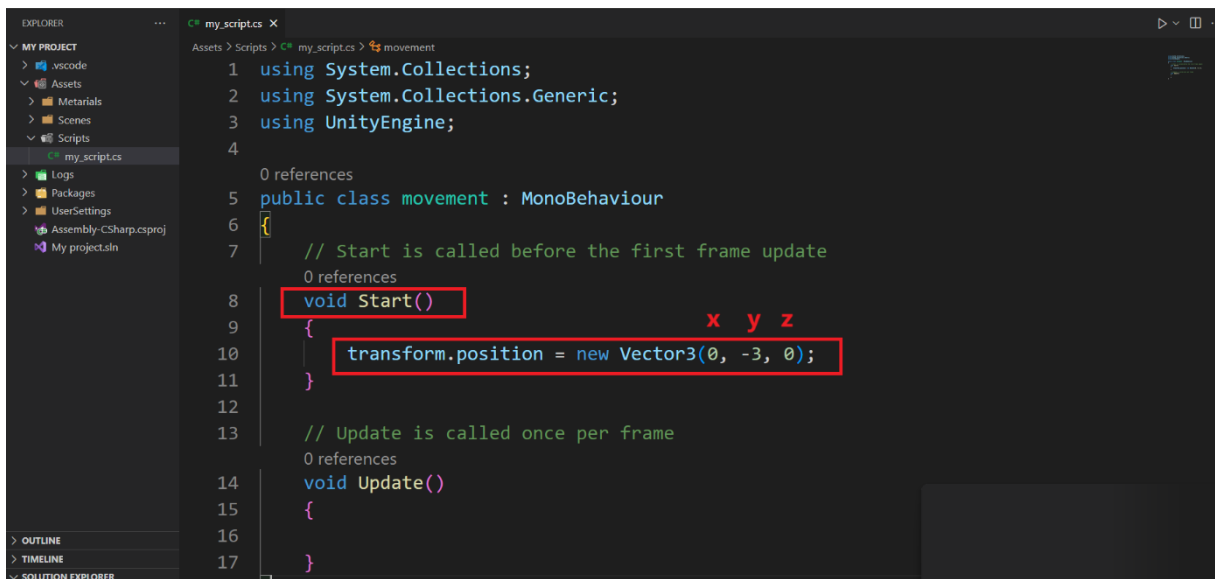
Bu işlemten sonra oluşturduğunuz dosyayı fare ile nesnenize sürükleyerek ekleyebilirsiniz ya da Hierarchy kısmındaki nesnenize tıklayarak Inspector panelinin en altındaki Add Component kısmından scripts bölümünü seçip açılan panelden eklemek istediğiniz script dosyanızı seçebilirsiniz.





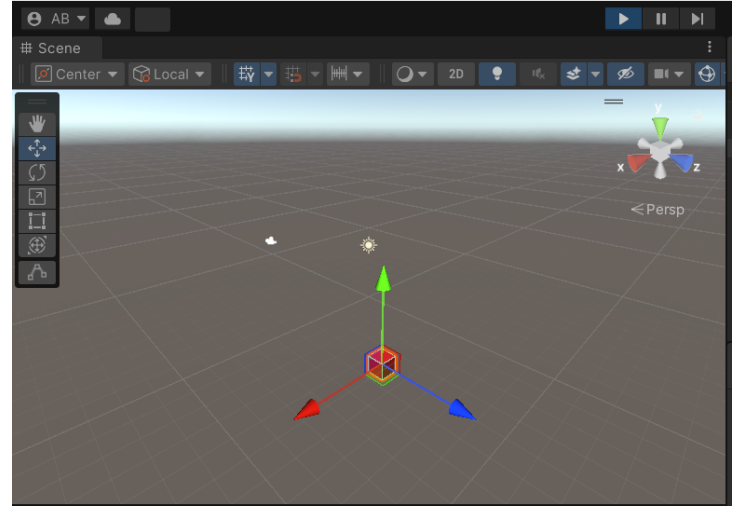
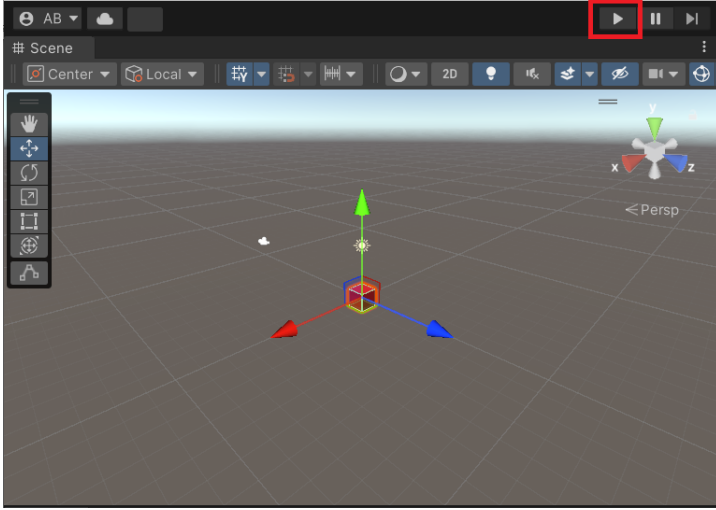
### 3-Script ile Nesnenin Konumunu 1 kez Değiştirme

Script dosyanızı açtığınızda istemcinizde karşınıza çıkacak olan kod bloğu içerisinde Start() fonksiyonu içerisine nesnenizin pozisyonuna erişip resimdeki şekilde Vector3(x, y, z) şeklinde seçtiğiniz koordinatları vererek nesnenizin başlamasını istediği yeri seçebilirsiniz.



```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 0 references
6 public class movement : MonoBehaviour
7 {
8     // Start is called before the first frame update
9     0 references
10    void Start()
11    {
12        x y z
13        transform.position = new Vector3(0, -3, 0);
14    }
15
16    // Update is called once per frame
17    0 references
18    void Update()
19    {
20    }
21 }
```

İşleminizi kaydettikten sonra nesnenizin bulunduğu sahneye döndüğünüzde yaptığınız değişiklikler otomatik olarak kaydedilecektir. Ardından oynat tuşuna bastığınızda nesnenizin yer değiştirdiğini göreceksiniz.



#### 4- Nesnenin Pozisyonun Sürekli Değiştirme

Nesnenizi sürekli hareket ettirmek için script dosyanızın içindeki Update fonksiyonu içerisine resimdeki ifadeyi girer ve gitmesini istediğiniz yönün koordinatlarını girerseniz nesneni o yöne doğru çok hızlı bir şekilde gittiğini görebilirsiniz.

```
// Update is called once per frame
0 references
void Update()
{
    transform.Translate(new Vector3(1,0,0));
}
}
```

## 5-Zaman Normalizasyonu

Nesnemizi bir yöne saniyede bir birim gitmesi durumu. Buna ulaşmak için Update fonksiyonu içerisindeki ifadeyi Time.deltaTime ifadesi ile çarpıp sahnenizi oynatmaya başladığınızda nesnenizin istediğimiz gibi saniyede 1 birim gittiğini görebilirsiniz.

```
// Update is called once per frame
0 references
void Update()
{
    transform.Translate(new Vector3(1,0,0) * Time.deltaTime);
}
```

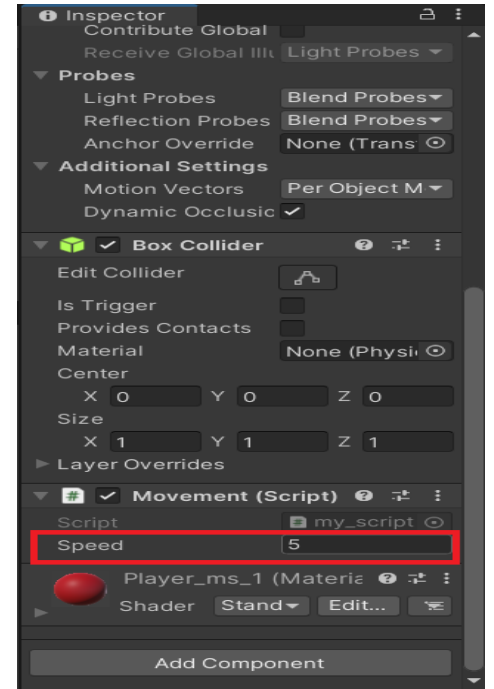
## 6-Bir Değişken Tanımlama Public/Private

Script dosyasının içindeki sınıfınızın içine nesnenizin görünürlüğü, tipi ve ismiyle birlikte resimdeki gibi tanımlayabilirsiniz.

```
public class movement : MonoBehaviour
{
    0 references
    public float speed = 5;
    // Start is called before the first frame update
    0 references
    void Start()
    {
    }
}
```

Resimdeki gibi public tanımlamak size bu değişkeninizi unity uygulaması üzerinde inspector panelinde görme ve değiştirebilme imkânı sağlar.

Ancak private tanımlarsanız bu özelliği yalnızca script dosyası üzerinden değiştirebilirsiniz.

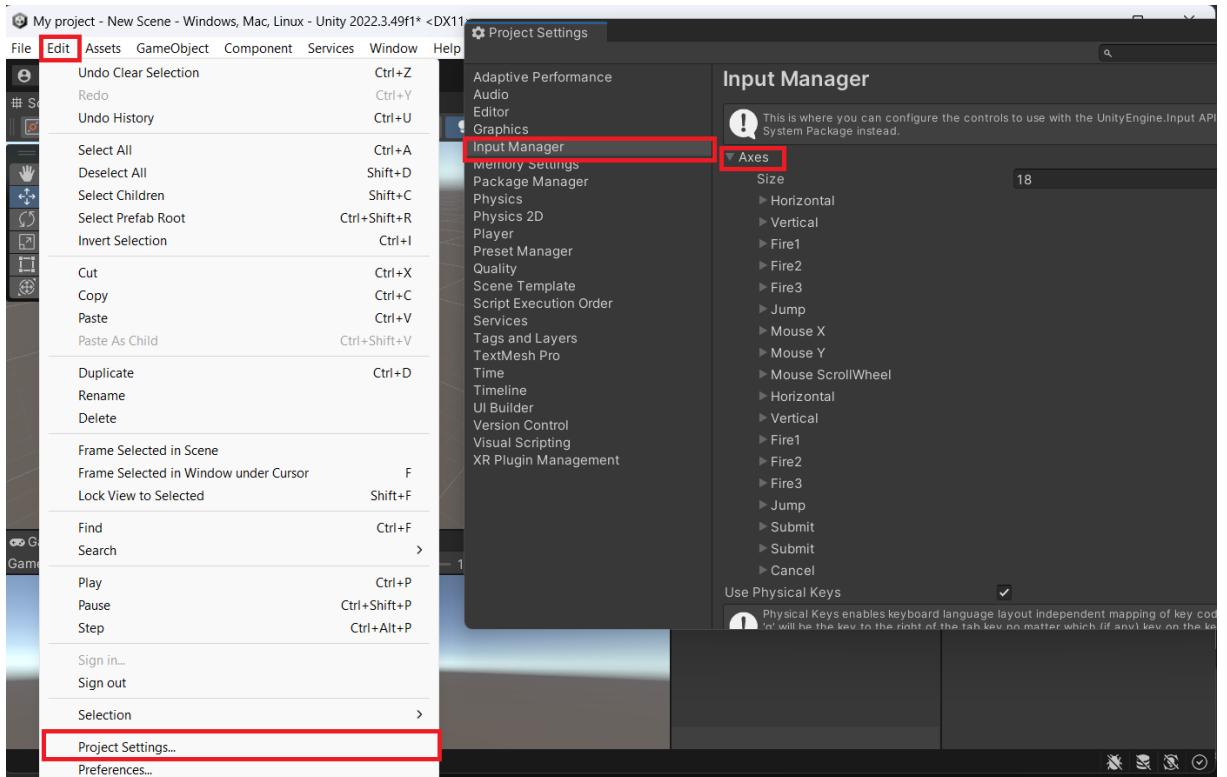


## 7- Klavyeden Tuşlar ile Yön Verme

Klavyenizden yön tuşları ile nesnenize yön vermek için öncelikle script dosyasının içine dikey(vertical) ve yatay(horizontal) olacak şekilde 2 değişken tanımlayın.

```
0 references
5 public class movement : MonoBehaviour
6 {
7     2 references
      public float speed = 5;
8     2 references
      public float horizontal;
9     2 references
      public float vertical;
10 // Start is called before the first frame update
```

Ardından Update fonksiyonu içerisinde bu değişkenlerinizi kontrol tuşları ile eşleştirmelisiniz. Kontrol tuşlarını öğrenmek için Unity uygulamasında Edit kısmından Project settings kısmını açın. Daha sonra Project settings kısmındaki Input Manager kısmından Axes bölümünün altındaki seçeneklerden istediğiniz tuş atamalarına bakabilirsiniz.



Bu tuşlara yandaki şekilde daha önceden oluşturduğunuz değişkenler ile eşleştirebilirsiniz.

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

Eşleştirme yaptıktan sonra aşağıdaki gibi değişkenlerinizi hareket işleminize eklediğinizde artık nesnenizi yön tuşları up, down, left, right veya w, a, s, d tuşları ile yönlendirebilirsiniz.

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

    transform.Translate(new Vector3(1,0,0) * Time.deltaTime * speed * horizontal);
    transform.Translate(new Vector3(1,0,0) * Time.deltaTime * speed * vertical);
}
```