

Orijinal Kaynak: <http://forum.unity3d.com/threads/unity-lesson-1-draft.103421/>

# Unity 3D İle Oyun Programlama

## Bölüm 3: Script Yazmaya Giriş



**ÇEVİRİ: Süleyman Yasir KULA**

<http://yasirkula.com>

Ferdinand Joseph Fernandez

Chief Technological Officer, Dreamlords Digital  
Inc.

Admin and Co-founder, Unity Philippines Users  
Group

September 2011



All code snippets are licensed under CC0 (public domain)



This document except code snippets is licensed with  
Creative Commons Attribution-NonCommercial 3.0 Unported

Bu bölümde Unity'de kod yazmaya başlayacağız. Unity'de yazılan kodlar diğer oyun motorlarına göre biraz farklıdır çünkü Unity component bazlı bir sistem kullanmaktadır.

## Scriptler

Unity'de yazdığınız her script aynı zamanda tıpkı Transform, Rigidbody ya da Light gibi birer component'tir. Scriptin çalıştırılması için onu, çalışmasını istediğiniz objeye verirsiniz.

Derse temiz bir proje üzerinden devam edelim. "Lesson3" adında yeni bir proje oluşturun.

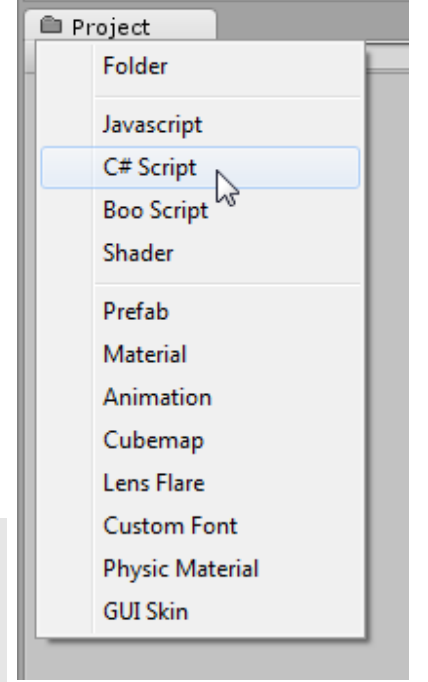
Şimdi bir script oluşturalım. Project panelindeki "Create" butonuna tıklayın ve "C# Script"i seçin.

Scriptin ismini "Rotate" yapın.

Rotate scriptine çift tıklayın. Varsayılan olarak MonoDevelop açılacaktır. MonoDevelop vasıtasıyla kod yazabilir/düzenleyebilirsiniz.

Şimdi ilk kodumuzu yazalım. Sarı renkle vurgulanmış kodu siz de scriptinizin Update fonksiyonuna yazın:

```
01 using UnityEngine;
02 using System.Collections;
03
04 public class Rotate : MonoBehaviour
05 {
06     // Use this for initialization
07     void Start()
08     {
09
10     }
11
12     // Update is called once per frame
13     void Update()
14     {
15         transform.Rotate(2, 0, 0);
16     }
17 }
```



Her yeni C# scripti bazı hazır kodlar yazılmış halde gelir.

Bu hazır kodlardan ilki en baştaki satır. Bu satırda "UnityEngine" kütüphanesini scriptimizde kullanacağımızı belirtiyoruz. Böylece kütüphanenin sunduğu fonksiyonlardan, değişkenlerden faydalanabileceğiz.

Scriptimizle aynı isimli bir class'ımız yer almakta. Class isminin script ismiyle birebir aynı olması çok önemli yoksa Unity istenmeyen hatalar verebilir.

Bizim class'ımız varsayılan olarak MonoBehaviour class'ını extend ediyor. MonoBehaviour, UnityEngine kütüphanesinde yer alan bir class. Unity'de çok kullandığımız fonksiyonlar, değişkenler hep bu class'te tanımlanmış vaziyette.

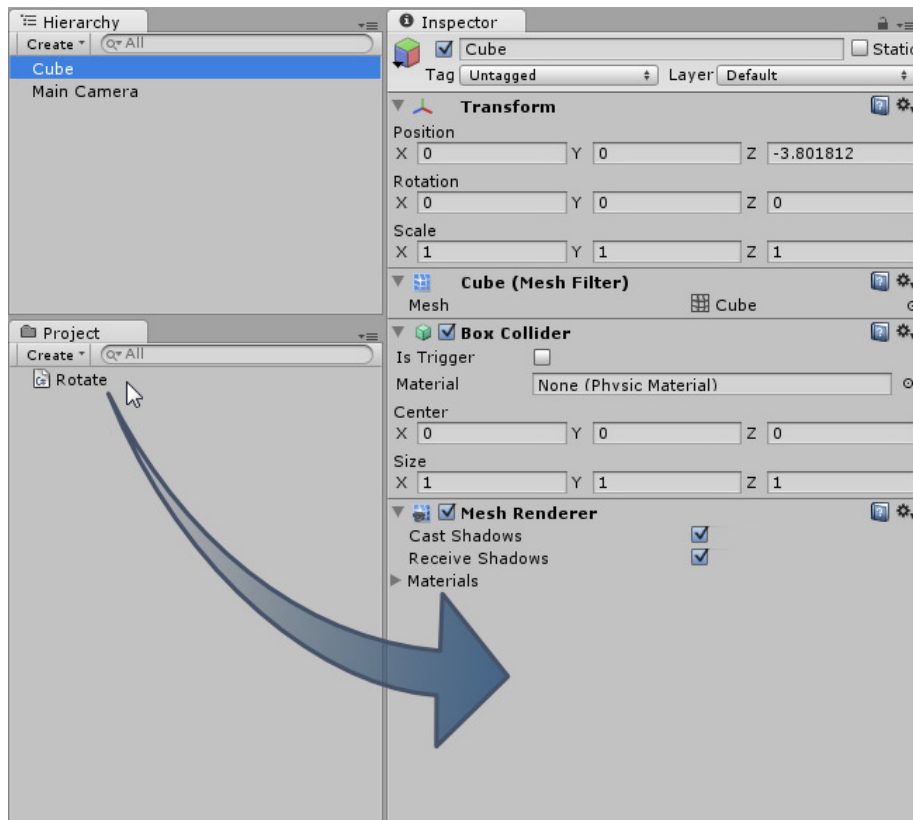
Update fonksiyonunda eriştiğimiz "transform", MonoBehaviour'un değişkenlerinden birisi. "transform" ile bir objenin pozisyonunun, eğiminin ve boyutunun depolandığı Transform component'ine erişiriz. Ardından script vasıtasıyla Transform component'indeki bu değerleri değiştirebiliriz.

Yazdığımız kod çok basit: Update fonksiyonu her çağrıldığında kod hangi objedeyse o objeyi X ekseninde 2 derece döndürüyoruz.

Update fonksiyonu her frame'de (kare) otomatik olarak çağrılır.

**GameObject > Create Other > Cube** ile sahnede bir küp oluşturun. Objeyi kameranın görebileceği bir yere yerleştirin.

Küp seçiliyken Rotate scriptini Project panelinden sürükleyerek Inspector panelinin altındaki boşluğa bırakın. Script, objeye component olarak atanacak.



Oyunu başlattığınızda küpün durmadan döndüğünü göreceksiniz. Küpü daha iyi görmek için sahneye bir ışık koyabilirsiniz (**GameObject > Create Other > Point Light**).

Rotate scriptini istediğiniz kadar objeye verebilirsiniz. Birkaç tane daha obje oluşturun. Mesela bir kapsül oluşturun (**GameObject > Create Other > Capsule**). Rotate scriptini kapsüle de verip oyunu başlatın. Hem küp hem de kapsül sürekli döneceklerdir.

Unity'de scriptlerin işleyiş mantığı böyle. Hangi objelerin scriptte verdiğiniz komutları yerine getirmesini istiyorsanız o objelere o scripti verirsiniz. Eğer bir scripti hiçbir objeye vermezseniz o script bir işe yaramaz. Bu duruma bazı istisnalar var tabi (static class'lar).

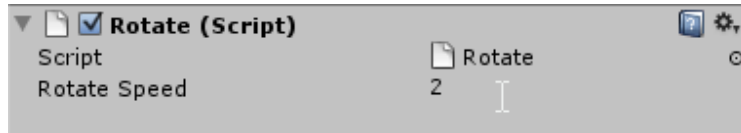
## Değişkenlerle Çalışmak

Az önceki kod objeyi her Update çağrılışında 2 derece döndürüyordu. Buradaki 2 değeri bir sabit sayıydı. Ama genelde böyle sabit sayılarla çalışmazsınız. Şimdi 2 dereceyi bir değişken ile değiştirelim:

```
01 using UnityEngine;
02 using System.Collections;
03
04 public class Rotate : MonoBehaviour
05 {
06     public float _rotateSpeed = 2;
07
08     // Use this for initialization
09     void Start()
10     {
11
12     }
13
14     // Update is called once per frame
15     void Update()
16     {
17         transform.Rotate(_rotateSpeed, 0, 0);
18     }
19 }
```

Aklınızda bulunsun, bu derste değişken isimlerinin başına hep \_ işareti koyacağım.

Scripti kaydedip Unity'e geri dönün. Inspector'da Rotate Speed değeri belirecek.



İşin güzel yanı Rotate Speed'i direkt Inspector üzerinden değiştirebilmeniz. Yani scriptte herhangi bir değişiklik yapmanız gerekmiyor.

Inspector'daki değerin scriptteki varsayılan değere göre üstünlüğü bulunmakta. Eğer scripti açıp \_rotateSpeed'i 2'ye değil de 3'e eşitleseniz bile Rotate Speed'in değeri yine de Inspector'daki 2 değeri olarak işleme sokulacaktır.

## Private Değişkenler

Scriptlerde public değişkenlerle çalışmak genel olarak iyi bir şey değildir. Örneğin \_rotateSpeed'in private olması script yazma prensiplerine daha uygun. Private değişkenlerde yaşanan sorun ise değerin Inspector'da gözükmemesi. Peki bu sorunu nasıl aşarız?

Neyse ki Unity'de private bir değişkenin değerini Inspector'da görüntülemeye zorlayan bir komut mevcut. \_rotateSpeed değişkeninin tanımlandığı (declaration) kısmı şöyle değiştirin:

```
01 [SerializeField]
02 float _rotateSpeed = 2;
```

SerializeField, bir deęişkenin deęerini Inspector'da görünmeye zorlayan bir anahtar kelimedir. Deęişkenin başındaki "public" takısını sildiğimiz için de artık deęişkenimiz private olmuş oldu.

## Oyun Hızına (Frame Rate) Adapte İşlemi

Aslına bakarsanız Update fonksiyonunda objeyi belli bir derece döndürmek iyi birşey deęildir. Oyunu yavaş bir bilgisayarda oynarken Update fonksiyonu daha az kez çağrılacaktır. Çünkü yavaş bilgisayarın gücü bir saniyede daha fazla Update çağırmaya yetmemektedir.

Bu durumda daha az Update çağrıldığı için de obje daha az kez dönecektir (Rotate). Hızlı bilgisayarlarda ise bunun aksine bir saniyede daha fazla kez Update çağrıldığından obje daha çok dönecektir. Yani objenin dönme miktarı bilgisayarın hızıyla doğru orantılı olacaktır.

Bu doğru orantı sorununu çözmek ve objenin her türlü sistemde (hızı ne olursa olsun) aynı sürede hep aynı miktar dönmesini sağlamak için adapte işlemi yapacağız. "her karede 2 derece döndür" demek yerine "her saniyede 2 derece döndür" diyeceğiz. Kodda yaptığımız deęişiklik şöyle:

```
01 void Update()  
02 {  
03     transform.Rotate(_rotateSpeed * Time.deltaTime, 0, 0);  
04 }
```

Burada \_rotateSpeed'in deęerini Time.deltaTime ile çarpıp objeyi o kadar derece döndürüyoruz. Time class'ı statiktir ve UnityEngine kütüphanesinde yer alır. **Bu class'taki deltaTime deęişkeni bir float'tur ve en son iki kare (frame) arasında kaç saniye geçtiğini depolar.**

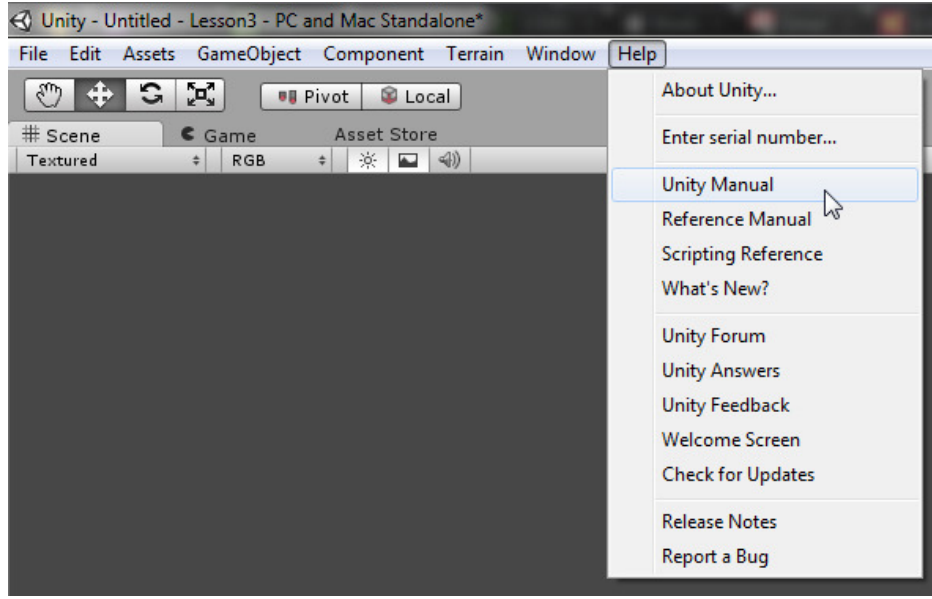
Artık her karede deęil her saniyede rotateSpeed kadar döneceğimizden rotateSpeed'in deęerini Inspector'dan artırmayı düşünebilirsiniz.

## Online Destek Hakkı

Oyunumuzu yapmaya başlamadan önce faydalı bulduğum bazı kaynakları paylaşacağım.

### *Unity'nin Kendi Kullanma Kılavuzu*

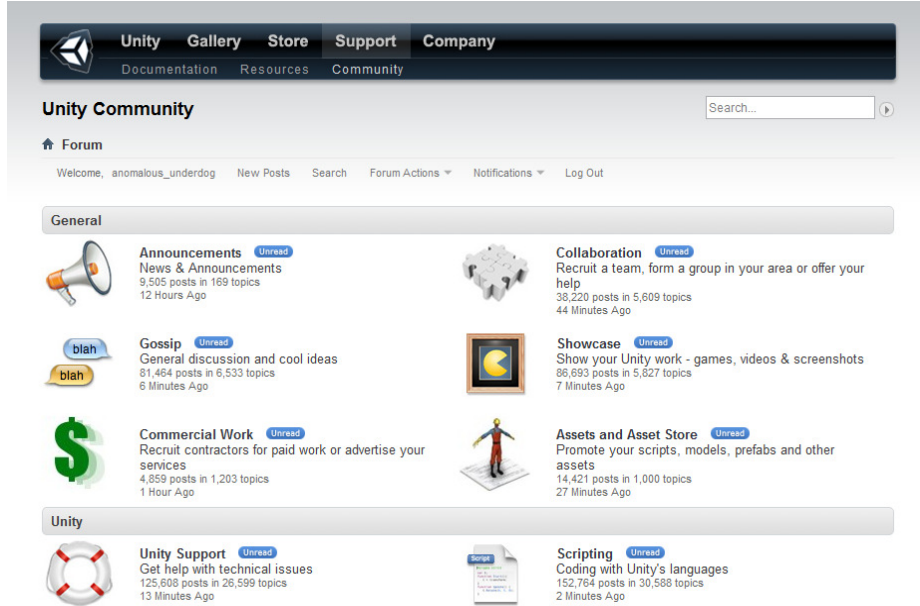
Unity, internet gerektirmeden okunabilen bir kullanma kılavuzu ve script referans arşivi ile birlikte gelmektedir.



Kullanma kılavuzuna **Help > Unity Manual** yoluyla, script referans arşivine **Help > Scripting Reference** yoluyla ulaşabilirsiniz.

## Resmî Forum Sayfası

Unity hakkında sorularınıza şurada cevap arayabilirsiniz: <http://forum.unity3d.com/>.



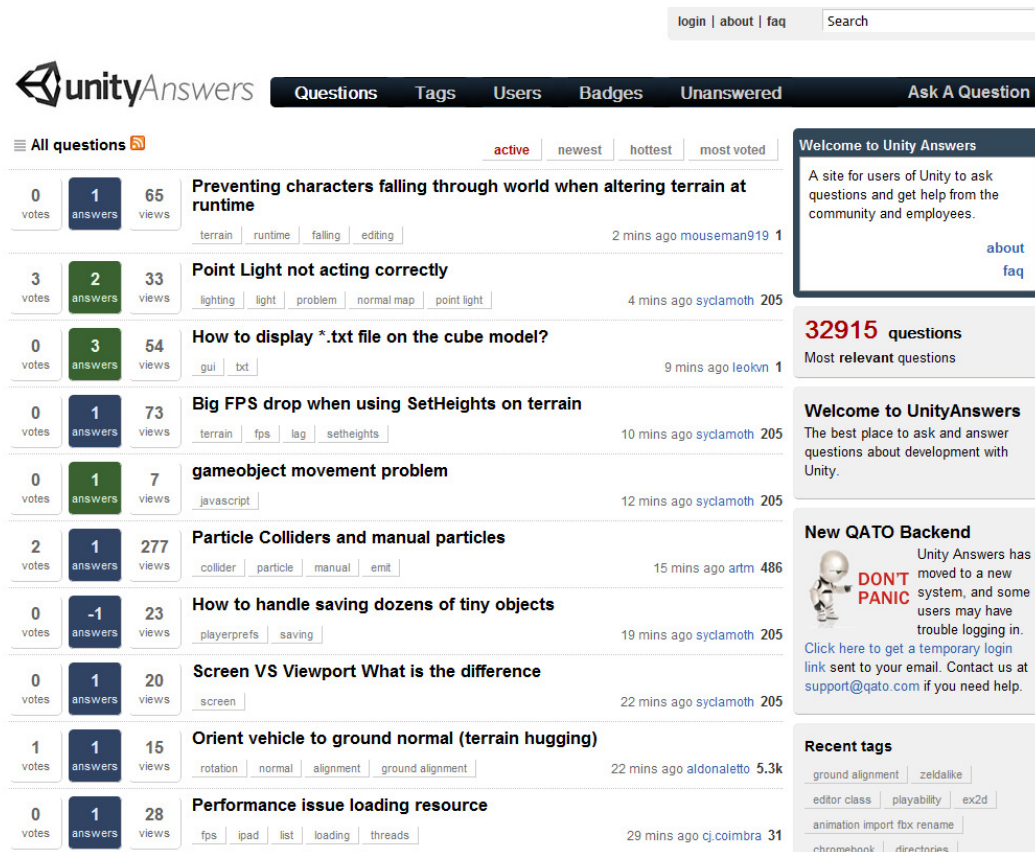
## Wiki Sayfası

Unity'nin hayranları tarafından yapılmış wikipedia tarzı bir destek sitesi olduğunu biliyor muydunuz? Bu siteye şuradan ulaşabilirsiniz: <http://www.unifycommunity.com/wiki/>. Burada bulduğunuz kodlar ticarî kullanıma açık olabilir de olmayabilir de. İlgilendiğiniz kodun açıklamasını dikkatle okumanız yararınıza olur.



## Soru&Cevap Sitesi

Unity'nin Stack Exchange sitesine benzer bir sitesi var: <http://answers.unity3d.com/>.

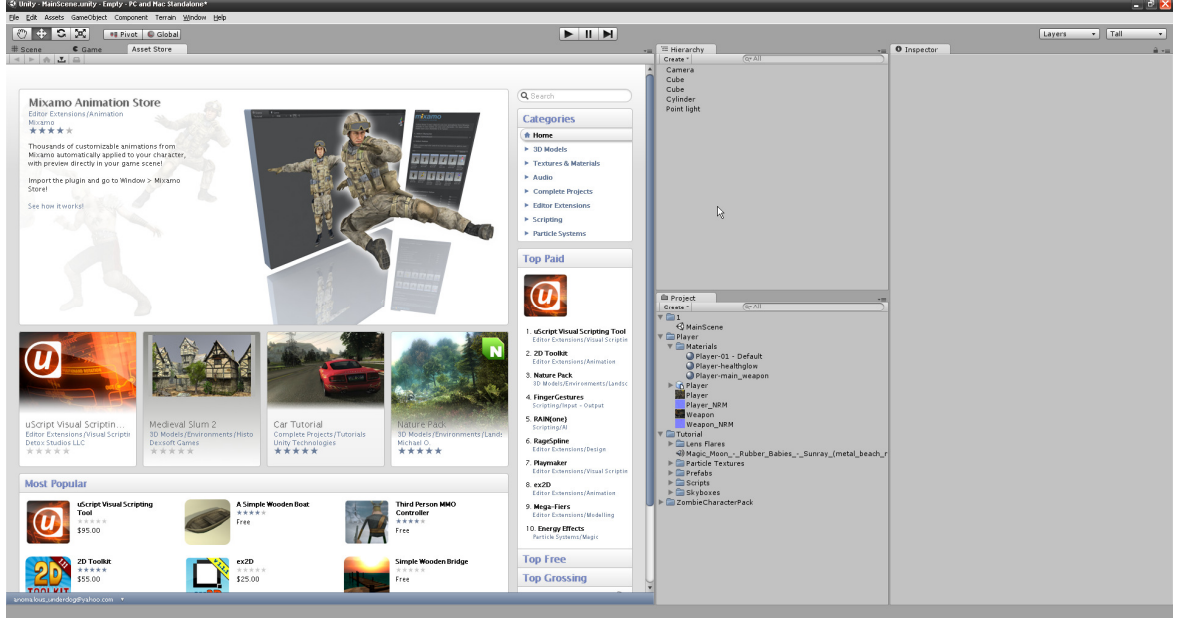




## Unity Asset Store (Dükkan)

Asset Store, Unity'de kullanmak üzere her türlü asset'i alıp satabileceğiniz bir mekandır. Bu asset'ler 3D modeller, hazır scriptler, plug-inler, müzik ve ses efektleri gibi çeşitli kategorilere ayrılmaktadır. Dükkan'daki bazı asset'ler tamamen ücretsizdir.

Window > Asset Store ya da Ctrl + 9 kombinasyonu ile Asset Store'a Unity içinden erişebilirsiniz.



## Özet Gececek Olursak...

Unity'de kod yazmaya bu dersle başlamış olduk. Private da olsa bir değişkenin değerini Inspector'da göstermeyi öğrendiniz.

Ayrıca sorunlarınıza şifa olabilecek birkaç site öğrendiniz.