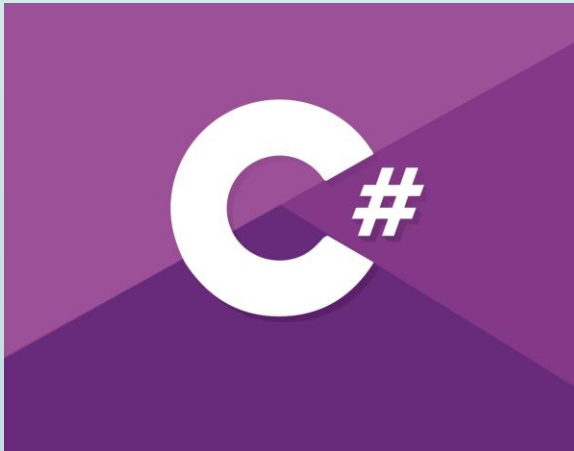


# C#da Sınıf Tanımlama ve Sınıf Çeşitleri

1



Hazırlayan : **Eray Nakışcı 2011404003**

Tarih : 31/0/2022

Sürüm : v1

Ders Yürütücüsü : Doç. Dr. İsmail KIRBAŞ

**Nesne nedir?**

**Nesne Tabanlı Programlama nedir?**

**Neden C#?**

**C#da Sınıf (Class) kavramı.**

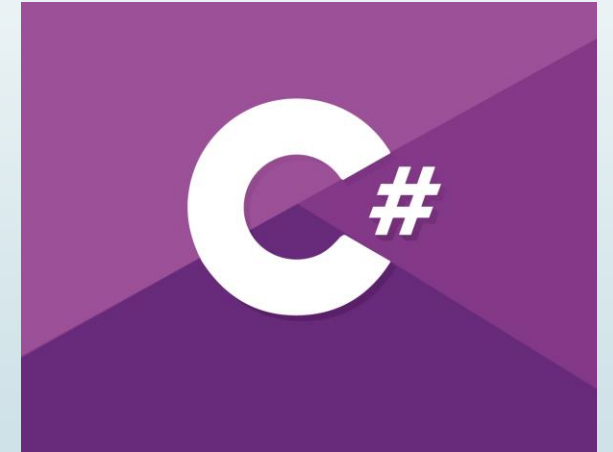
**Sınıf (Class) Çeşitleri.**

**Sınıf (Class) Nasıl Tanımlanır?**

**Classlarda Adlandırma Kuralları**

**Classlarda Kurucular**

**C#da Sınıf (Class) Tanımlama Uygulaması.**



## Nesne Nedir?

Nesne, içinde veri ve bu veriler üzerinde işlem yapacak olan metodları (fonksiyon) bulunduran yazılım bileşenidir. Nesne bu tanıma uygun olarak kendi işlevselliğini de içinde taşır. Nesneler her uygulamada tekrar tekrar kullanılabilir.

# Nesne Tabanlı Programlama Nedir?

**Object Oriented Programming** yani Nesne Yönelimli Programlama, her işlevin nesnel olarak soyutlandığı bir programlama şeklidir.

Bir nesnenin rengi, durumu, ismi, üretim yılı gibi birçok özelliklerin bilgisayar ortamında gösterilmesi buna örnek olarak verilebilir.

1960'lı yılların sonuna doğru ortaya çıkan bu programlama şekli, o dönemlerde yazılım dünyasının yaşadığı sıkıntının sonucudur.

Yazılımların karışıklığı ve boyutlarının artması, belirli bir nitelik düzeyini korumak için gereken maliyeti, zamanı ve çabayı arttırıyordu. **OOP** bu soruna karşı çözüm getiren özelliği yazılımdaki birimselliği yüksek oranda benimsemesidir.

## Neden C#?

C#, sunduğu gelişmiş **hata giderme (debugger)** özelliği sayesinde de yazılımcıların işlerini bir hayli kolaylaştırıyor.

C#, "syntax" hatalarını yazılımcıya ayrı bir ekranda göstererek yazılımcının bu **hataları kolayca tespit** etmesini sağlıyor.

Ayrıca yazılımcılar tarafından oldukça çok tercih edildiği için de internette kaynak yani C# dersleri oldukça fazla.

# C#da Class Kavramı

Sınıflar nesne yönelimli programlamanın en önemli ögesidir. Sınıflar sayesinde programlarımızı parçalara bölüp karmaşıklığını azaltırız.

C# Sınıf (Class) Yapısı nedir sorusuna vereceğimiz en iyi cevap ise şöyledir. Bir metot, fonksiyon veya davranış tanımlayabiliriz, bu metotlar, fonksiyonlar veya davranışlar, bunları bir araya getirerek bir bütün oluşturmak için ortak ve benzer özelliklere sahiptir. Sınıfı programlama ortamına aktarmak için kullanılan proje taslağı ile karşılaştırabiliriz. Şablon, verilerin nasıl saklanacağını, işleneceğini ve bunlara nasıl erişileceğini tanımlar.



İlk olarak bir sınıf tanımlamamız lazım burada bir arabadan yola çıkalım, bir arabanız var. Arabanızın rengi yani (dış görüntüsü boyası ) bir sınıftır çünkü her arabanın bir rengi vardır. Üstünde barındırdığı mavi yeşil siyah veya beyaz bunlar bir değişken tipidir değişkeni temsil eder veya motor gücü aracın kaç kapılı olduğu vs. metotları temsil eder. genel olarak bir araç ise sınıfı temsil eder. çünkü bu özelliklerin hespi tek bir araçta yani sınıfta barındırılabilir.

# Class Çeşitleri

2 çeşit class vardır;

Static class

Instance class

# C# da Class Nasıl Tanımlanır?

C#da class tanımlamadan önce, oluşturacağımız class'a nerelerden erişilebileceğini belirlememiz lazım. Nesne tabanlı programlamada 5 çeşit erişim türü bulunmaktadır. Bunlar;

- public
- private
- protected
- internal
- protected internal

public: Bu erişim türü ile tanımlanmış sınıf üyelerine her yerden erişim izni verilmiş olur.

private: Bu erişim türünde tanımlanmış üyelere sadece aynı sınıf içerisinde tanımlanmış diğer üyeler tarafından erişim izni verilmiş olur.

protected: Bu erişim türünde tanımlanmış üyelere tanımlandık sınıf ve bu sınıftan türetilmiş diğer sınıflardan erişim izni verilmiş olur.

Internal: Bu erişim türünde tanımlanmış üyelere aynı assembly içinden erişim izni verilmiş olur.

protected internal: Bu erişim türü protected ve internal erişim türlerinin veya işlemiyle birleştirilmiş halidir.

Oluşturmak istediğimiz classın erişim türünün adını yazdıktan sonra, "class" yazıyoruz ve belirlediğimiz class adını yazıyoruz. Şablon oluşturmak gerekirse;

```
<erişim türü> class <class adı> {  
    içerik  
}
```

# Classlarda Adlandırma Kuralları

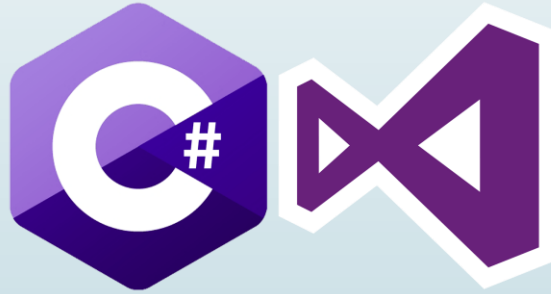
Erişilebilirlik kuralları yazılan kod içerisinde daha kolay görebilmek için aşağıdaki adlandırma kurallar yardımcı olur:

- Public alan ve metod adları büyük harfle başlamalıdır. Örneğin, VitesDegistir metodu public olduğundan “V” ile yazılmıştır. Bu adlandırma kuralına ilk defa Pascal programlama dilinde kullanıldığı için PascalCase adı verilmektedir.
- Private tanımlayıcılar küçük harfle başlamalıdır. Mesela vites alanı private olduğu için “v” ile yazılmıştır.

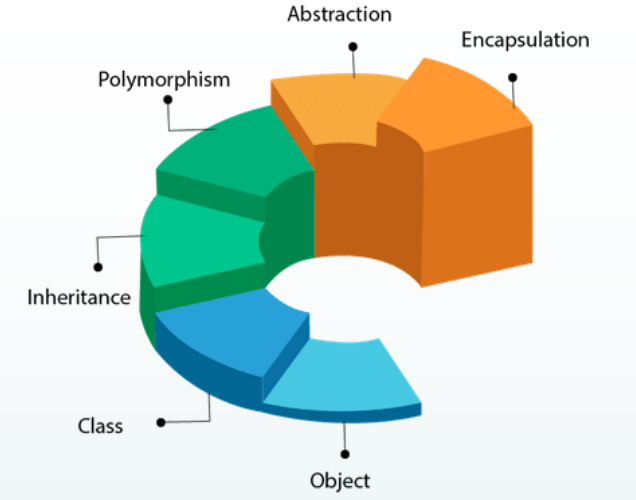
# Classlarda Kurucular

Kurucu (Constructor ) bir sınıfın örneği oluşturulduğunda kendiliğinden çalışan özel bir metottur. Sınıf ile aynı ada sahip olmak zorundadır, parametreler alabilir ama geri değer döndürmez. Tüm sınıfların bir kurucusu olmak zorundadır.

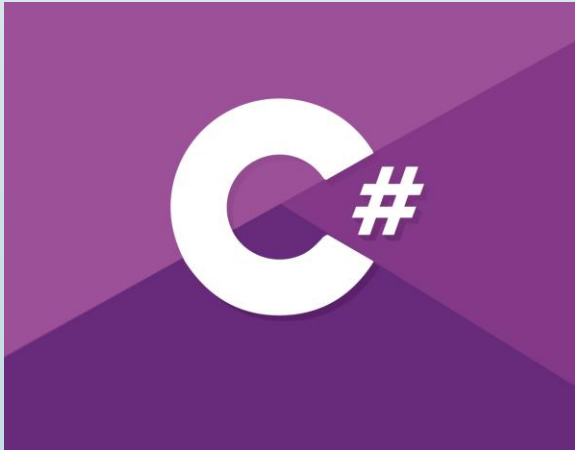
# C#da Class Tanımlama Uygulamaları







# İlginiz için teşekkürler...



Hazırlayan  
E-posta

: Eray Nakışcı  
: eraynksc@icloud.com

Tarih

: 31/05/2022

Ders Yürütücüsü

: Doç. Dr. İsmail KIRBAŞ