# 1. Abstract Class (Soyut Sınıf) Nedir?

**Abstract class** (soyut sınıf), yalnızca miras alınarak kullanılabilen ve doğrudan nesne oluşturulamayan bir sınıf türüdür.

✅ **Abstract class**, kalıtım için bir temel sağlar.

✅ **Soyut metotlar**, türeyen sınıflar tarafından **ezilmek zorundadır (override)**.

✅ **Somut metotlar**, alt sınıflar tarafından olduğu gibi kullanılabilir.

✅ **Doğrudan nesne oluşturulamaz**, sadece türeyen sınıflardan nesne türetilebilir.

Bu yapı sayesinde **kod tekrarını önleyerek**, nesne yönelimli programlamada **daha düzenli ve genişletilebilir** bir yapı oluşturabiliriz.

# 2. Abstraction (Soyutlama) Nedir?

C# dilinde **Abstraction (Soyutlama)**, nesne yönelimli programlamanın (OOP) dört temel prensibinden biridir ve gereksiz ayrıntıları gizleyerek yalnızca önemli özellikleri göstermeyi amaçlar.

Soyutlama, bir nesnenin dış dünyaya sadece gerekli olan özelliklerini açığa çıkarırken, gereksiz ayrıntıları gizlemesidir. Örneğin, bir otomobil kullanırken motorun nasıl çalıştığını bilmek zorunda değiliz; sadece gaz, fren ve direksiyon gibi temel özellikleriyle ilgileniriz.

C#’ta soyutlama, **abstract class (soyut sınıf)** ve **interface (arayüz)** kullanılarak gerçekleştirilir.

✅ **Soyutlama**, bir nesnenin yalnızca önemli yönlerini gösterip, iç detaylarını gizlemektir.

✅ Abstract sınıflar veya interface'ler ile gerçekleştirilir.

# 3. Polymorphism (Çok Biçimlilik) Nedir?

Nesne yönelimli programlamada (OOP) bir nesnenin farklı şekillerde davranmasını sağlayan özelliktir. Yani **aynı metot, farklı şekillerde çalışabilir**.

Polymorphism, **Method Overriding (Metot Ezme)** ve **Method Overloading (Metot Aşırı Yükleme)** olmak üzere iki şekilde uygulanır.

## **1. Method Overriding (Metot Ezme) – Runtime Polymorphism**

* **Kalıtım (inheritance) kullanılır.**
* **Virtual (sanal) metotlar** kullanılır.
* **Base (ana) sınıftaki metot, türetilmiş sınıfta farklı bir şekilde yeniden yazılabilir.**
* **Çalışma zamanında (runtime) belirlenir.**

## **2. Method Overloading (Metot Aşırı Yükleme) – Compile-Time Polymorphism**

* **Aynı isimde birden fazla metot tanımlanabilir.**
* **Parametre sayısı veya tipi farklı olmalıdır.**
* **Derleme zamanında (compile-time) belirlenir.**

## **Polymorphism’in Avantajları**

✅ **Kod tekrarını azaltır.**  
✅ **Esneklik sağlar (Farklı nesneler aynı metotları farklı şekillerde kullanabilir).**  
✅ **Bakımı ve genişletmesi kolaydır.**