**Kalıtım (Inheritance )**

a) Temel sınıfta yer alan bir üyenin (metot, özellik, indeksleyici ya da olay), türeyen sınıf

tarafından nasıl değiştirileceğine dair bir yol sunmaktadır

b) Bir türetilen sınıfın (base class ya da parent class) özellik ve davranışlarını, bir alt sınıfın

miras alarak sanki kendi üyeleriymiş gibi kullanmasıdır.

c) Dilin nesne kullanıcısından gereksiz uygulama ayrıntılarını saklayabilme yeteneği olarak

ifade edilebilir.

d) Temel işlevi veri korumadır.

**Çok biçimlilik (Polymorphism)**

a) Temel sınıfta yer alan bir üyenin (metot, özellik, indeksleyici ya da olay), türeyen sınıf

tarafından nasıl değiştirileceğine dair bir yol sunmaktadır

b) Bir türetilen sınıfın (base class ya da parent class) özellik ve davranışlarını, bir alt sınıfın

miras alarak sanki kendi üyeleriymiş gibi kullanmasıdır.

c) Dilin nesne kullanıcısından gereksiz uygulama ayrıntılarını saklayabilme yeteneği olarak

ifade edilebilir.

d) Temel işlevi veri korumadır 5

Kapsülleme (Encapsulation)

a) Temel sınıfta yer alan bir üyenin (metot, özellik, indeksleyici ya da olay), türeyen sınıf

tarafından nasıl değiştirileceğine dair bir yol sunmaktadır

b) Bir türetilen sınıfın (base class ya da parent class) özellik ve davranışlarını, bir alt sınıfın

miras alarak sanki kendi üyeleriymiş gibi kullanmasıdır.

c) Dilin nesne kullanıcısından gereksiz uygulama ayrıntılarını saklayabilme yeteneği olarak

ifade edilebilir.

d) Dilin varolan sınıf tanımlamalarının üzerine yeni bir sınıf tanımlaması inşa edilmesine izin verme yeteneği olarak ifade edilebilir.

Virtual (Sanal) Metodu

Temel sınıfa ait bir metodun bazen türemiş sınıfta farklı bir şekilde kullanılması gerekebilir. Bu durumda temel sınıfın söz konusu metodu türemiş sınıfın değiştirip kullanabileceği yönünde izin vermesi gerekir. Bu izin, metodun temel sınıfta “*virtual*” olarak tanımlanması sonucu verilmektedir.

Override (Geçersiz Kılma) Metodu

Temel sınıfta “*virtual*” tanımlanmış bir metodun türemiş sınıfta yeniden yazılması için (geçersiz kılma) “*override*” sözcüğü kullanılır.

Mühürlenmiş (Sealed) Sınıflar

Sealed sınıflar kalıtım vermeyen sınıflardır. Bir sınıfın başka sınıflara kalıtım vermesini engellemek için **sealed** komutu kullanılır. Sealed classları bir çeşit güvenlik önlemi olarak düşünebiliriz. Yanlışlıkla türetilmeleri engellenmiş oluyor.

Bu komut kullanıldığında sınıfınızdan başka hiçbir sınıfa kalıtım alınamaz. Ancak, yeni türetilen bir değişken yardımıyla sınıfın metotları çağrılabilir.

New (Ad Gizleme)

Bir türetilmiş sınıfın temel sınıftaki bir üye ile aynı ismi taşıyan bir üye tanımlaması mümkündür. Böyle bir durum söz konusu olunca temel sınıfın üyesi türetilmiş sınıf içindeki gizlenmiş olur. Bu, C#'ta teknik olarak bir hata olmasa da derleyici bir uyarı mesajı verecektir. Bu uyarı, bir ismin gizlenmekte olduğu gerçeğine işaret edecektir. Eğer niyetiniz temel sınıf üyesini gizlemek ise, bu durumda bu uyarıyı önlemek için türerilmiş sınıf üyesinin öncesinde new anahtar kelimesini kullanmalısınız. new'un bu kullanımının bir nesne örneği oluştururkenki kullanımından tamamen ayrı ve de farklıolduğunu kavrayın.

Erişim Belirleyiciler

**“public”** programdaki öğeye heryerden erişim yetkisi verir,

**“private”** ilgili öğeye sadece tanımlandığı yerden (örneğin class içinden) erişim yetkisi verilir, **“protected”** ile işaretlenmiş öğeye sadece türetilmiş (inherit edilmiş) class içerisinden erişilebilir, **“internal”** ile belirlenen öğeye sadece ilgili assembly (uygulama) içerisinden erişilebilir ve bu öğeye uygulama dışından erişim engellenmiş olunur,

**“protected internal”** erişim belirleyicisi protected ve internal belirleyicilerinden herhangi birisinin şartları sağlanmışsa yani öğeye tanımlandığı class içerisinden veya türetilmiş classlar içerisinden ya da aynı assembly içerisinden erişilebilir.

Get-Set

Tüm programlama dillerinde, kendi yazacağınız metodlarla bu değerleri kontrol edebilirsiniz.

**GET metodu:** Değişken çağırıldığında çalışır.

**SET metodu:** Değişkene değer atandığında çağırılır. Değişken kontrolleri bu metod içinde yapılır.

Base

Türetilmiş sınıflarda temel sınıftan metodu veya değişkeni çağırmak için kullanılır.