Nesne tabanlı programlama aslında gerçek hayatta olan her şeyi bir nesne olarak tanıtacağımız ve bu nesneden yeni nesneler üretebileceğimiz bir alan. Nesne tabanlı programlama ihtiyaçlarımızı karşılayabileceğimiz bir çok alanda bize yardımcı olabilir. Bu trafik lambaları,kütüphane otomasyonu, obs vs. şeklinde genişleyip gider. Nesne tabanlı programlama hayatın her yerinde karşımıza çıkan bir yapıdır.

Nesne tabanlı programlamanın 4 önemli özelliğinden az da olsa bahsetmek istiyorum.

1- Soyutlama (Abstraction): Gerçekleşen olayın iç yüzünü görmeden sadece işlevselliğe odaklanma denilebilir.

Örneğin bir giyim dükkanınız olduğunu düşünün. Elinizde 100 tane manken bulunsun. Mankenleri üretimde kalıp olarak kullanıp seri üretime başlamadan önce 100 mankenin ilk başta cinsiyetine bakılır ve o mankenler artık birer cinsiyeti temsil eder. Kendi cinsiyetleri için model oluşturmak için mankenlikten cinsiyeti temsil etmesine soyutlandırıldılar. Daha sonra ayakkabı üreteceksiniz diyelim. Bu 100 manken oluşturdukları ayrı ayrı cinsiyet kümesini temsil ederler. Burada ayaklarının şekline göre yeni ayakkabı üretilmesinin modelliğini üstlenirler. Bu kısımda vücudun geri kalan hiçbir kısmına ihtiyaç yoktur. Sadece ayağın şeklinin alınması yeterli olacaktır. İşte burada soyutlama işlemi yaparak vücuttan sadece bize lazım olan bir yeri seçtik. Konuyu uzatmak istemediğim için bu kısmı burada sonlandırmak istiyorum. Buna benzer örnekler ile bu konu daha da detaylı bir şekilde işlenebilir.

2- Diğer nesneler tarafından bu sınıfta değişiklik yapılma yetkisi yoktur sadece genel işlevler ve yöntemler listesi çağrılabilir. Bu sayede veri bozulması ve program güvenliği daha iyi sağlanır ve istenmeyen değişimler bir nevi engellenmiş olur.

3- Nesneler arasındaki ilişkiler ve alt sınıflar atanabilir, böylece geliştiricilerin benzersiz bir hiyerarşiyi korurken ortak bir mantığı tekrar kullanmalarına izin verilir. OOP'ın bu özelliği daha kapsamlı bir veri analizini zorlar, geliştirme süresini azaltır ve daha yüksek bir doğruluk düzeyi sağlar.

4- Nesneler arasındaki ilişkiler ve alt sınıflar atanabilir, böylece geliştiricilerin benzersiz bir hiyerarşiyi korurken ortak bir mantığı tekrar kullanılmasına izin verilir yada şöyle anlatmak gerekirse kalıtımda aynı isimli nesnelerin farklı görevleri vardır. tabi bunları bir dil ile yazmak gerek bunlar Java, Javascript, Python, C++, Visual Basic . Net, Ruby, Scala, Php vs. dir.

Hiç bilmeyen birine NTP’yi anlatırken yukarıda verdiğim örneği verirdim. Buna ek olarak daha güzel örnek olması açısından çağımızın popüler oyunlarında kullanılan nesneler ile örnek verirdim. Örneğin futbol oyununda her oyuncu insandır ve normal bir insanın sahip olduğu uzuvlara sahiptirler. Bunlar birer insan nesnesidir. Daha sonra her bir oyuncu bu nesne sınıfından türetilir ve yeni oyuncuları oluşturur. Burada da görüldüğü gibi hepsi insan olarak bir nesneydi ve artık başka bir insanı temsil ediyorlar.

Uzman biriyle tartışacak olsam NTP’yi 7 yaşındaki bir çocuğa, 35 yaşındaki bir ev kadına ve 50 yaşındaki huysuz bir erkeğe nasıl anlatacağını sorardım. Ek olarak da kendisi için NTP de cevabını bulamadığı bir sorunu da söylemesini isterdim.

16541514

Mehmet Hüseyin Delipalta