

Generic Class'lar ve Generic Method'lar (Boxing & Unboxing)

Generic yapılar, kodunuzu farklı veri tipleriyle çalışacak şekilde esnek ve yeniden kullanılabilir hale getirir. Bu sayede, aynı işlevi gören kod bloğunu birden fazla veri tipiyle çalıştırabilirsiniz. Örneğin, bir "Box" sınıfı düşünün; bu sınıfın içine int, string veya başka bir tür veri koyabilirsiniz. Böylece her tür için ayrı ayrı sınıf yazmak yerine, tek bir generic sınıf kullanarak işimizi kolaylaştırmış oluruz.

Ayrıca, generic metotlar da aynı şekilde çalışır. Metot çağrılırken hangi veri tipi kullanılacağı belirlenir. Bu, kodunuzun tip güvenliğini sağlar; yani yanlış tipte veri gönderilmesini önler.

Boxing ve Unboxing nedir?

Boxing: Değer tipini (örneğin int), referans tipine (object) dönüştürmektir.

Unboxing: Box edilmiş (yani object'e dönüştürülmüş) değer, tekrar orijinal değer tipine çevrilmesidir.

Boxing ve unboxing işlemleri ek maliyet oluşturabilir, çünkü her iki işlem de ek işlem yapmayı gerektirir. Generic yapılar sayesinde bu işlemlerden kaçınarak performansı artırabilirsiniz.

C#'ta Garbage Collector Nedir?

Garbage Collector (GC), C#'ta bellek yönetimini otomatik olarak yapan bir sistemdir. Program çalışırken belleğe yerleştirdiğimiz nesnelerin, artık kullanılmadığı zamanları algılayıp, bu nesnelerin kapladığı belleği otomatik olarak temizler. Bu sayede, geliştiricilerin elle bellek temizleme işlemleriyle uğraşması gerekmez. GC, nesnelerin kullanım süresine göre onları farklı nesil (generation) gruplarına ayırır ve en çok kullanılmayanları öncelikle temizleyerek uygulamanın performansını ve verimliliğini artırır. Böylece bellek sızıntıları önlenmiş olur ve sistem kaynakları daha verimli kullanılır.