

Çalışma Soruları 2

Soru 1: Aşağıdaki kod parçasında boxing işlemi yapıldığında bellekte neler olur? Açıklayın ve unboxing işlemini ekleyin:

```
int sayi = 42;
object kutu = sayi; // Boxing
int geriAlinanSayi = (int)kutu; // Unboxing
```

- kutu nesnesi üzerinde yapılan boxing işleminin bellek üzerindeki etkisini açıklayın.
- unboxing sırasında hangi hatalarla karşılaşılabilirini tartışın.

Soru 2: Aşağıdaki kodda boxing ve unboxing hatalarını bulun ve düzeltin:

```
object obj = 10; // Boxing
double sayi = (double)obj; // Hatalı unboxing!
```

- unboxing işlemi neden hatalıdır? Doğru şekilde nasıl yaparsınız?
- unboxing sırasında tür uyumsuzluğu olduğunda ne tür bir hata alınır?

Soru 3: Boxing işleminin performans üzerindeki etkisini analiz eden bir program yazın:

- Bir döngü kullanarak 1 milyon int değeri object türüne box edin.
- Aynı döngüyü unboxing işlemi için de yapın.

İşlem sürelerini ölçüp, boxing ve unboxing işlemlerinin performans üzerindeki etkisini yorumlayın..

Soru 4: Aşağıdaki kodda boxing ve unboxing işlemlerini doğru hale getirin ve hataları

```
Kodu kopyala
object kutulu = 123; // Boxing
float sayi = (float)kutulu; // Hatalı unboxing!
```

- Kodun çalışmasını sağlamak için unboxing işlemini doğru şekilde yapın.
- Bu tür hatalardan nasıl kaçınılacağı hakkında kısa bir açıklama yapın.

Soru 5: boxing ve unboxing işlemleri ile bir hesap makinesi uygulaması yazın:

- Kullanıcıdan iki int değeri alın.
- Bu değerleri object olarak box edin ve toplama işlemi yapın.
- unboxing yaparak sonucu ekrana yazdırın.
- unboxing yapılmadan direkt matematiksel işlem yapmanın getirdiği zorlukları tartışın.