

# Java'da Exceptions ve Errors

## Exceptions Nedir?

### Java'da Exception Kavramı

Exception, programın çalışması sırasında oluşan beklenmedik durumları temsil eder. Java'da, bu durumlar try-catch blokları ile yönetilir.

Örnek: NullPointerException, bir nesne referansı null olduğunda ortaya çıkar.

### Checked ve Unchecked Exceptions

Checked exceptions, derleme zamanında kontrol edilir ve genellikle IOException gibi dosya işlemleri sırasında ortaya çıkar.

Unchecked exceptions, çalışma zamanında ortaya çıkar ve genellikle RuntimeException sınıfından türetilir.

## Error Nedir?

### Java'da Error Kavramı

Error, programın çalışmasını durdurabilecek ciddi sorunları temsil eder. Genellikle JVM (Java Virtual Machine) tarafından atılır.

Örnek: OutOfMemoryError, JVM'nin belleği tükendiğinde ortaya çıkar.

### Error ve Exception Arasındaki Farklar

Exceptions, programın çalışmasını durdurmadan yönetilebilirken, Errors genellikle programın sonlanmasına neden olur.

Errors, genellikle sistem kaynakları veya donanım sorunları ile ilgilidir.

## Exception Handling

### Try-Catch Blokları

Try-catch blokları, exceptions'ı yakalamak ve yönetmek için kullanılır. Try bloğu içinde hata oluşursa, catch bloğu çalışır.

Örnek: ArithmeticException, bir sayının sıfıra bölünmesi durumunda yakalanabilir.

### Finally Bloğu

Finally bloğu, try-catch blokları sonrasında her durumda çalıştırılır. Kaynakların serbest bırakılması için kullanılır.

Örnek: Dosya okuma işlemi sonrası dosyanın kapatılması.

## Özel Exception Sınıfları

### Kendi Exception Sınıfını Oluşturma

Java'da, kendi exception sınıflarınızı oluşturabilirsiniz. Bu, uygulama ihtiyaçlarına özel hata yönetimi sağlar.

Örnek: InvalidUserInputException, kullanıcıdan alınan geçersiz girdileri yönetmek için kullanılabilir.

### Özel Exception Kullanımı

Özel exception sınıfları, daha anlamlı hata mesajları ve özel hata yönetimi sağlar.

Örnek: Uygulama içinde belirli bir iş kuralının ihlali durumunda özel exception fırlatılması.

## Exception Propagation

### Exception Yayımlama

Exception propagation, bir exception'ın bir metoddan çağıran metoda iletilmesi sürecidir. Bu, exception'ın üst seviyelerde ele alınmasına olanak tanır.

Örnek: Bir metod içinde IOException oluştuğunda, bu exception üst metoda yayımlanabilir.

### Throws Anahtar Kelimesi

Throws anahtar kelimesi, bir metodun belirli türde exceptions fırlatabileceğini belirtir. Bu, exception propagation için gereklidir.

Örnek: public void readFile() throws IOException, bu metodun IOException fırlatabileceğini belirtir.

Hata mesajlarının daha anlamlı ve açıklayıcı olması

Spesifik hata türleri ile daha esnek hata yönetimi

Hataların daha iyi sınıflandırılması ve analizi

Hata ayıklama sürecinde daha hızlı çözüm bulma

Proje gereksinimlerine özel kuralların uygulanması