

# 7. Ders: JAVA ile Nesne Yönelimli Programlama İstisna İşleme (Exception Handling)

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH112 Algoritma ve Programlama-II

Dr. Öğr. Üyesi Yaman Akbulut

# JAVA ile Nesne Yönelimli Programlama

- <http://www.kriptarium.com/algoritma.html> (Yardımcı kaynak)
- JAVA Programlama Dili (Ders: 67-72) video.
  - Ders 67: İstisna İşleme (Exception Handling) (izle)
  - Ders 68: Exception Handling Sınıflarının Hiyerarşisi (izle)
  - Ders 69: Exception Handling Sınıfı Anahtar Kelimeleri (izle)
  - Ders 70: try-catch Bloğunun İç Yapısı (izle)
  - Ders 71: İç İçe (Yuvarlanmış) try Bloğu (izle)
  - Ders 72: finally Bloğu Kullanımı (izle)

# Java Keywords

abstract

assert

boolean

break

byte

case

catch

char

class

const

continue

default

do

double

else

enum

extends

final

finally

float

for

goto

if

implements

import

instanceof

int

interface

long

native

new

package

private

protected

public

return

short

static

strictfp

super

switch

synchronized

this

throw

throws

transient

try

void

volatile

while

# İstisna İşleme (Exception Handling)

**İstisna işleme** (exception handling), bir programın istisnai durumlarla başa çıkmasına ve normal yürütülmesine devam etmesine imkan verir.

JVM, gerçekleştirilmesi imkansız bir işlem tespit ederse, bir program çalışırken çalışma zamanı hataları meydana gelir.

Örneğin, sınırların dışında bir indeks kullanarak bir diziye erişerseniz, **ArrayIndexOutOfBoundsException** ile bir çalışma zamanı hatası alırsınız.

Programınız bir tamsayı (**int**) beklediğinde **double** değer girerseniz, **InputMismatchException** ile bir çalışma zamanı hatası alırsınız.

# İstisna İşleme (Exception Handling)

Java'da çalışma zamanı hataları istisna olarak atılır.

İstisna, yürütmenin normal şekilde ilerlemesini engelleyen bir hatayı veya koşulu temsil eden bir nesnedir.

İstisna ele alınmazsa, program anormal şekilde sona erecektir.

İstisnayı, programın çalışmaya devam etmesi veya düzgün bir şekilde sona erdirilmesi için nasıl ele almalıyız?

# İstisna İşleme (Exception Handling)

İstisnalar, bir metottan atılır. Metodun çağırıcısı istisnayı yakalayabilir ve işleyebilir.

```
1 import java.util.Scanner;
2
3 public class Bolme {
4     public static void main(String[] args) {
5         Scanner giris = new Scanner(System.in);
6
7         System.out.print("Lutfen iki tam sayi giriniz: ");
8         int sayi1 = giris.nextInt();
9         int sayi2 = giris.nextInt();
10
11         System.out.println(sayi1 + " / " + sayi2 + " = " +
12             (sayi1 / sayi2) + " dir.");
13     }
14 }
```

Bolme.java

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni2>java Bolme
```

```
Lutfen iki tam sayi giriniz: 9 2
```

```
9 / 2 = 4 dir.
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni2>java Bolme
```

```
Lutfen iki tam sayi giriniz: 9 0
```

```
Exception in thread "main" java.lang.ArithmeticException: / by zero
    at Bolme.main(Bolme.java:11)
```

# İstisna İşleme (Exception Handling)

BolmeIfIle.java

```
1  import java.util.Scanner;
2
3  public class BolmeIfIle {
4      public static void main(String[] args) {
5          Scanner giris = new Scanner(System.in);
6
7          System.out.print("Lutfen iki tam sayi giriniz: ");
8          int sayi1 = giris.nextInt();
9          int sayi2 = giris.nextInt();
10
11         if (sayi2 != 0)
12             System.out.println(sayi1 + " / " + sayi2 + " = " +
13                                 (sayi1 / sayi2) + " dir.");
14         else
15             System.out.println("Bolen sifir olamaz ");
16     }
17 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni2>java BolmeIfIle
Lutfen iki tam sayi giriniz: 9 2
9 / 2 = 4 dir.
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni2>java BolmeIfIle
Lutfen iki tam sayi giriniz: 9 0
Bolen sifir olamaz
```

# İstisna İşleme (Exception Handling)

BolmeMetotIle.java

```
1  import java.util.Scanner;
2
3  public class BolmeMetotIle {
4      public static int bolme(int sayi1, int sayi2) {
5          if (sayi2 == 0) {
6              System.out.println("Bolen sifir olamaz");
7              System.exit(1);
8          }
9
10         return sayi1 / sayi2;
11     }
12
13     public static void main(String[] args) {
14         Scanner giris = new Scanner(System.in);
15
16         System.out.print("Lutfen iki tam sayi giriniz: ");
17         int sayi1 = giris.nextInt();
18         int sayi2 = giris.nextInt();
19
20         int sonuc = bolme(sayi1, sayi2);
21         System.out.println(sayi1 + " / " + sayi2 + " = " +
22             sonuc + " dir.");
23     }
24 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\
yeni2>java BolmeMetotIle
Lutfen iki tam sayi giriniz: 5 3
5 / 3 = 1 dir.
```

```
C:\Program Files\Java\jdk-15.0.1\bin\
yeni2>java BolmeMetotIle
Lutfen iki tam sayi giriniz: 5 0
Bolen sifir olamaz
```



# İstisna İşleme (Exception Handling)

```
1  import java.util.Scanner;
2
3  public class BolmeIstisnaIle {
4      public static int bolme(int sayi1, int sayi2) {
5          if (sayi2 == 0)
6              throw new ArithmeticException("Bolen sifir olamaz");
7
8          return sayi1 / sayi2;
9      }
10
11     public static void main(String[] args) {
12         Scanner giris = new Scanner(System.in);
13
14         System.out.print("Lutfen iki tam sayi giriniz: ");
15         int sayi1 = giris.nextInt();
16         int sayi2 = giris.nextInt();
17
18         try {
19             int sonuc = bolme(sayi1, sayi2);
20             System.out.println(sayi1 + " / " + sayi2 + " = " +
21                 sonuc + " dir.");
22         }
23         catch (ArithmeticException ex) {
24             System.out.println("İstisna: bir tam sayi " +
25                 "sifira bolunemez ");
26         }
27
28         System.out.println("Yurutme devam ediyor ...");
29     }
30 }
```

BolmeIstisnaIle.java

```
C:\Program Files\Java\jdk-15.0.1\bin\
yeni2>java BolmeIstisnaIle
Lutfen iki tam sayi giriniz: 5 3
5 / 3 = 1 dir.
Yurutme devam ediyor ...
```

```
C:\Program Files\Java\jdk-15.0.1\bin\
yeni2>java BolmeIstisnaIle
Lutfen iki tam sayi giriniz: 5 0
İstisna: bir tam sayi sifira boluneme
z
Yurutme devam ediyor ...
```

# try catch (dene yakala)

```
try {  
    // Çalıştırılacak kod;  
    // Bir istisna atabilecek bir ifade veya metot;  
    // Çalıştırılacak daha fazla kod;  
}  
catch (tip ex) {  
    // İstisnayı işlemek için kod;  
}
```

# try catch (dene yakala)

```
try {  
    // Bir istisna atabilecek ifadeler;  
}  
catch (Istisna1 exVar1) {  
    // İstisna1'i işlemek için kod;  
}  
catch (Istisna2 exVar2) {  
    // İstisna2'yi işlemek için kod;  
}  
catch (Istisna3 exVar3) {  
    // İstisna3'ü işlemek için kod;  
}  
  
...  
  
catch (IstisnaN exVarN) {  
    // İstisnayı işlemek için kod;  
}
```

# finally (sonunda)

**finally** cümlesi, bir istisna olup olmadığına bakılmaksızın her zaman yürütülür.

```
try {  
    // Bir istisna atabilecek ifadeler;  
}  
catch (Istisna ex) {  
    // İstisna'ı işlemek için kod;  
}  
finally {  
    // ifadeler;  
}
```

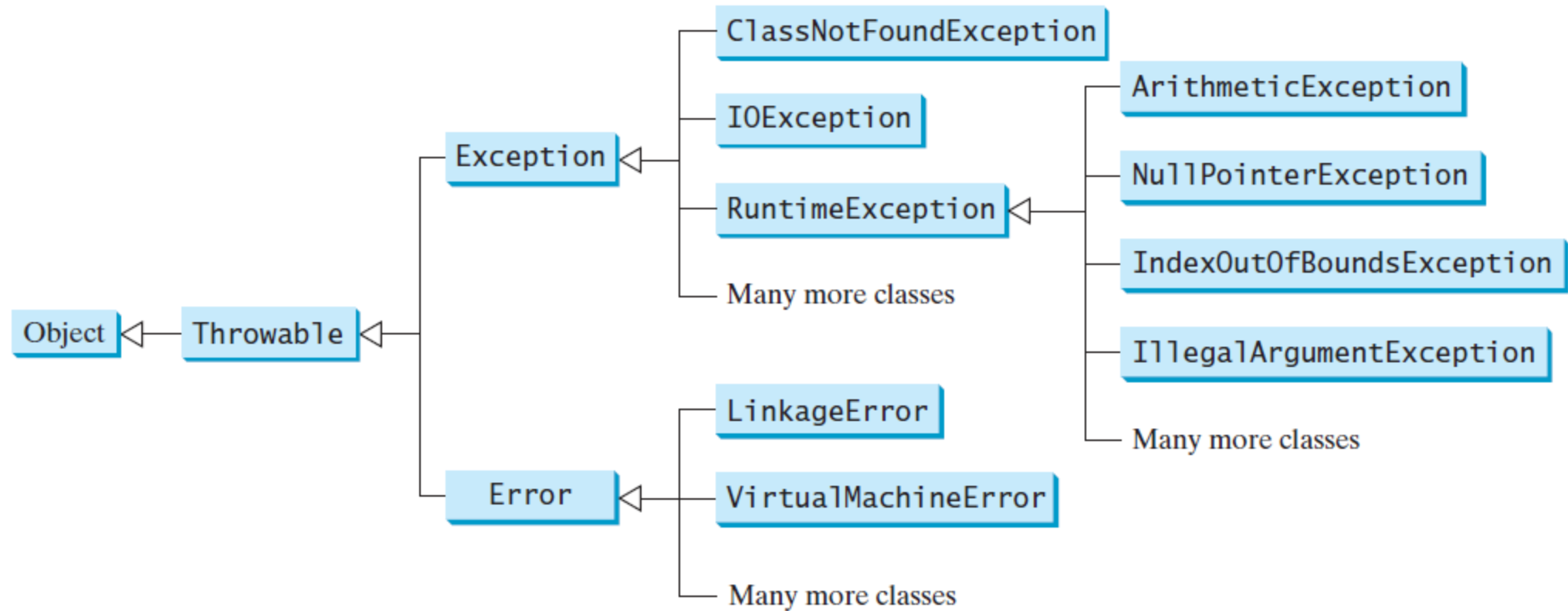
# InputMismatchException

```
1  import java.util.*;
2
3  public class InputMismatchExceptionDemo {
4      public static void main(String[] args) {
5          Scanner giris = new Scanner(System.in);
6          boolean giriseDevamEt = true;
7
8          do {
9              try {
10                 System.out.print("Bir tam sayi giriniz: ");
11                 int number = giris.nextInt();
12
13                 // Sonucu göster
14                 System.out.println(
15                     "Girilen sayi " + number + " dir.");
16
17                 giriseDevamEt = false;
18             }
19             catch (InputMismatchException ex) {
20                 System.out.println("Tekrar deneyin. (" +
21                     "Gecersiz giris: bir tam sayi gerekiyor)");
22                 giris.nextLine(); // Girisi sil
23             }
24         } while (giriseDevamEt);
25     }
26 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni2>
java InputMismatchExceptionDemo
Bir tam sayi giriniz: 3
Girilen sayi 3 dir.
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni2>
java InputMismatchExceptionDemo
Bir tam sayi giriniz: 3.0
Tekrar deneyin. (Gecersiz giris: bir tam sa
yi gerekiyor)
Bir tam sayi giriniz: 5.0
Tekrar deneyin. (Gecersiz giris: bir tam sa
yi gerekiyor)
Bir tam sayi giriniz: 4
Girilen sayi 4 dir.
```

# İstisna sınıfları (Exception classes)



**FIGURE 12.1** Exceptions thrown are instances of the classes shown in this diagram, or of subclasses of one of these classes.

# Çalışma zamanı istisnasının alt sınıfları (Subclasses of RuntimeException)

**TABLE 12.3** Examples of Subclasses of **RuntimeException**

<i>Class</i>	<i>Reasons for Exception</i>
<b>ArithmeticException</b>	Dividing an integer by zero. Note that floating-point arithmetic does not throw exceptions (see Appendix E, Special Floating-Point Values).
<b>NullPointerException</b>	Attempt to access an object through a <b>null</b> reference variable.
<b>IndexOutOfBoundsException</b>	Index to an array is out of range.
<b>IllegalArgumentException</b>	A method is passed an argument that is illegal or inappropriate.

## Örnek: İstisna tanımlama, atma ve yakalama (IllegalArgumentException)

```
1 public class CemberIstisnaIle {
2     /** Cemberin yarıcapı */
3     private double yarıcap;
4
5     /** Olusturulan nesne sayısı */
6     private static int nesneSayisi = 0;
7
8     /** 1 yarıcaplı cember yapilandir */
9     public CemberIstisnaIle() {
10         this(1.0);
11     }
12
13     /** Belirtilen yarıcap ile cember yapilandir */
14     public CemberIstisnaIle(double yeniYarıcap) {
15         setYarıcap(yeniYarıcap);
16         nesneSayisi++;
17     }
18
19     /** Yarıcapı dondur */
20     public double getYarıcap() {
21         return yarıcap;
22     }
23 }
```

```
24     /** Yeni bir yarıcap ata */
25     public void setYarıcap(double yeniYarıcap)
26         throws IllegalArgumentException {
27         if (yeniYarıcap >= 0)
28             yarıcap = yeniYarıcap;
29         else
30             throw new IllegalArgumentException(
31                 "Yarıcap negatif olamaz");
32     }
33
34     /** nesneSayisini dondur */
35     public static int getNesneSayisi() {
36         return nesneSayisi;
37     }
38
39     /** Cemberin alanini dondur */
40     public double alanHesapla() {
41         return yarıcap * yarıcap * 3.14159;
42     }
43 }
```



## Örnek: İstisna tanımlama, atma ve yakalama (IllegalArgumentException)

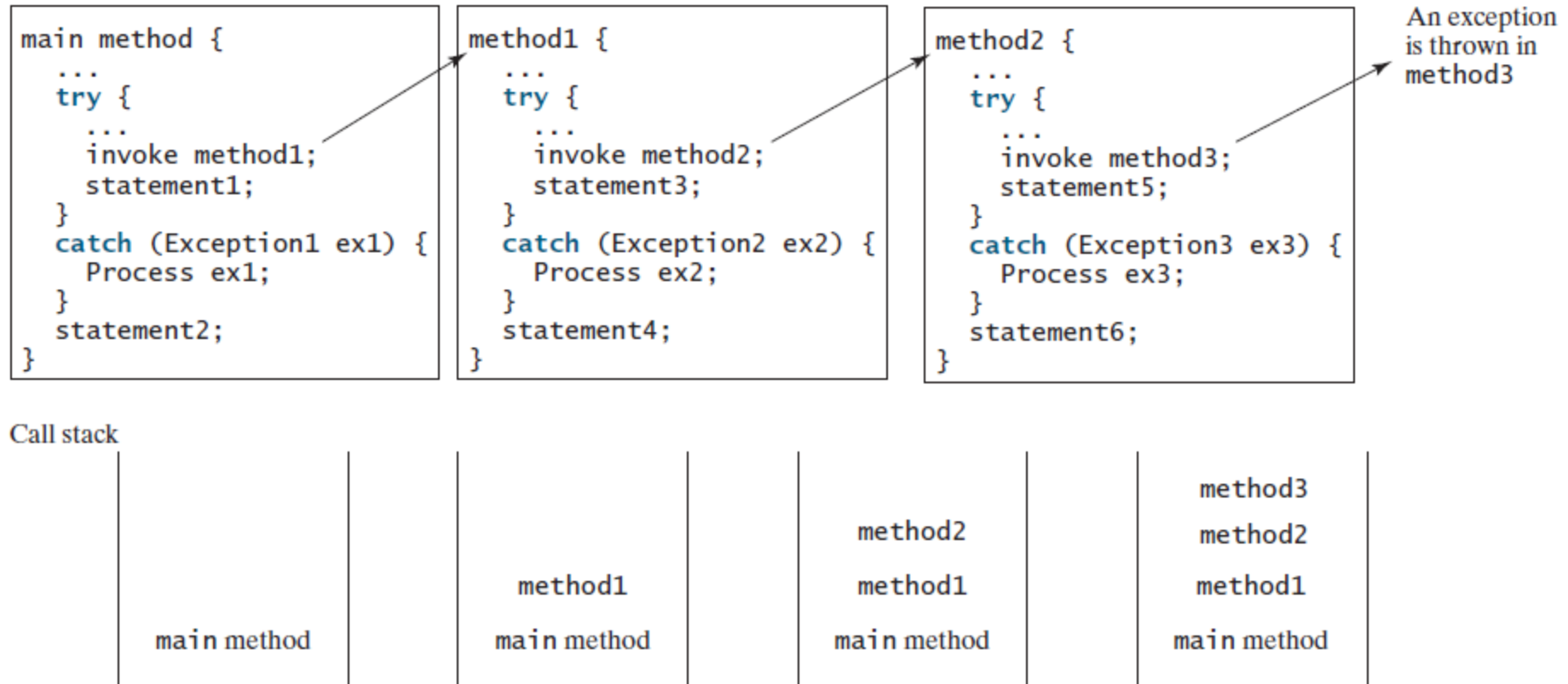
```
1 public class TestCemberIstisnaIle {
2     public static void main(String[] args) {
3         try {
4             CemberIstisnaIle c1 = new CemberIstisnaIle(5);
5             CemberIstisnaIle c2 = new CemberIstisnaIle(-5);
6             CemberIstisnaIle c3 = new CemberIstisnaIle(0);
7         }
8         catch (IllegalArgumentException ex) {
9             System.out.println(ex);
10        }
11
12        System.out.println("Olusturulan nesne sayisi: " +
13        CemberIstisnaIle.getNesneSayisi());
14    }
15 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni2>javac CemberIstisnaIle.java
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni2>javac TestCemberIstisnaIle.java
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni2>java TestCemberIstisnaIle
java.lang.IllegalArgumentException: Yaricap negatif olamaz
Olusturulan nesne sayisi: 1
```

# İstisnaları yakalamak



**FIGURE 12.3** If an exception is not caught in the current method, it is passed to its caller. The process is repeated until the exception is caught or passed to the `main` method.

# Ödev

```
1 public class TestException {
2     public static void main(String[] args) {
3         try {
4             System.out.println(sum(new int[] {1, 2, 3, 4, 5}));
5         }
6         catch (Exception ex) {
7             ex.printStackTrace();
8             System.out.println("\n" + ex.getMessage());
9             System.out.println("\n" + ex.toString());
10
11             System.out.println("\nTrace Info Obtained from getStackTrace");
12             StackTraceElement[] traceElements = ex.getStackTrace();
13             for (int i = 0; i < traceElements.length; i++) {
14                 System.out.print("method " + traceElements[i].getMethodName());
15                 System.out.print("(" + traceElements[i].getClassName() + ":");
16                 System.out.println(traceElements[i].getLineNumber() + ")");
17             }
18         }
19     }
20
21     private static int sum(int[] list) {
22         int result = 0;
23         for (int i = 0; i <= list.length; i++)
24             result += list[i];
25         return result;
26     }
27 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni2>java TestException
java.lang.ArrayIndexOutOfBoundsException: Index 5 out of bounds for length 5
    at TestException.sum(TestException.java:24)
    at TestException.main(TestException.java:4)
```

Index 5 out of bounds for length 5

```
java.lang.ArrayIndexOutOfBoundsException: Index 5 out of bounds for length 5
```

Trace Info Obtained from getStackTrace

```
method sum(TestException:24)
```

```
method main(TestException:4)
```