

# Görsel Programlama

DERS 05

# İstisnai Durumlar (Exceptions)

İstisnai durumlar, programın çalışması sırasında meydana gelen beklenmeyen durumlardır.

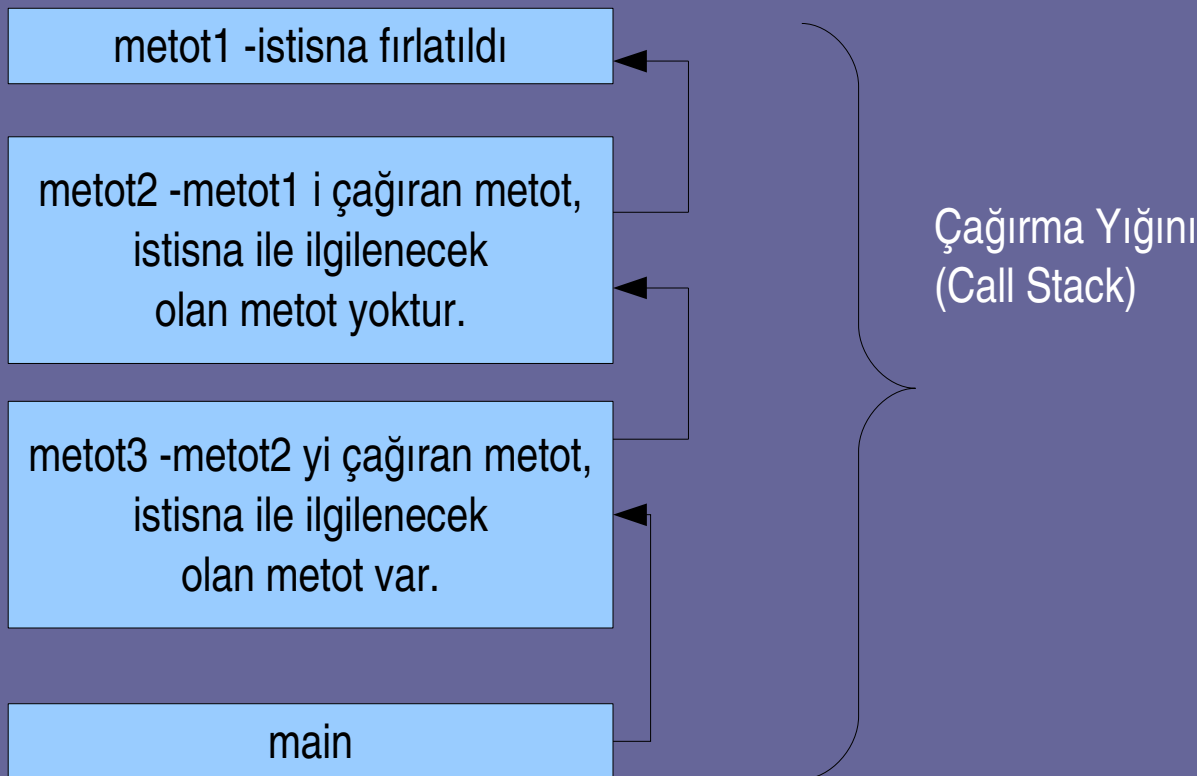
Örneğin; sıfıra bölme işlemi, dizinin indisini sınırından farklı bir indis numarası kullanılması, olmayan bir url adresine erişmeye çalışmak, olmayan bir nesnenin metotlarına erişmeye çalışmak,...

Bu gibi durum ile karşılaşıldığı zaman Java Sanal Makinesi (JVM) bu hatayı tanımlayan bir nesne oluşturur. Bu nesneye “**istisna nesnesi**” (**Exception Object**) denilir.

Bu nesnede hatanın tanımı, tipi ve bu hata oluştuğunda programın durumu saklanır. Bu nesnenin program çalışırken oluşturulmasına ve işlenmesine “**istisna fırlatma**” (**throwing an exception**) denilir.

# İstisnai Durumlar (Exceptions)

Bir istisna fırlatılınca çalışan sistem bu istisna ile ilgilenecek olan bir metot aramaya başlar ve bunu “**çağırma yığını**” (**call stack**) içinde yapar.



# İstisnai Durumlar (Exceptions)

İstisna ile ilgilenecek olan metota “**istisna yöneticisi**” (**exception handler**) denilir. İstisna yöneticisinin bu istisna ile ilgilenmesine **istisnayı yakalamak**(**catch exception**) denilir.

Program içinde fırlatılan istisnaları yakalayabilmek için aşağıdaki kod bloğu kullanılır:

```
try{  
  
}catch(istisna_durum_adı istisna_degiskeni){  
  
}
```

# İstisnai Durumlar (Exceptions)

```
try{  
  
}catch(istisna_durum_adı istisna_degiskeni){  
  
}
```

1. İstisnai durumun denetlenmesi için çalışan kodun try bloğu içine yazılması gereklidir.
2. “try” bloğundan sonra en az 1 “catch” bloğu gelmelidir.
3. Birden fazla “catch” bloğu tanımlanabilir.

# İstisnai Durumlar (Exceptions)

```
SifiraBolme.java
package gorsel_prog.ders05;

public class SifiraBolme {
    public static void main(String[] args) {
        int i=12, j=0;
        try {
            int sonuc = i/j;
            System.out.println("Sonuc:"+sonuc);
        } catch (ArithmeticException e) {
            System.out.println("Payda sıfır Olamaz");
        }
        System.out.println("Program bitti");
    }
}
```

catch blokları birden fazla tanımlanabilir. Sanal makine tarafından yukarıdan aşağıya doğru bloklar taranır. Fırlatılan istisnayı işleyebilecek olan “catch” bloğu varsa o blok çalıştırılır.

# İstisnai Durumlar (Exceptions)

```
UrlKullanim.java
package gorsel_prog.ders05;

import java.io.IOException;
import java.net.MalformedURLException;
import java.net.URL;
import java.net.URLConnection;

public class UrlKullanim {
    public static void main(String[] args) {
        try {
            URL u = new URL("http://127.0.0.1");
            URLConnection cc = u.openConnection();
            cc.connect();
        } catch (MalformedURLException e) {
            System.out.println("Hatalı url"+e);
        } catch (IOException e) {
            System.out.println("Baglanti kurulamadi "+e);
        }
    }
}
```

# finally bloğu

finally bloğu istisnai durum oluşsa da oluşmasa da daima çalışan kod bloğudur. Açılan kaynakları (**resources**) {**dosya, veritabanı, vb.**} kapatmak için kullanılır. Bu blok sadece “**System.exit();**” komutu çalıştırılırsa çalıştırılmaz.



# finally bloğu

```
FinallyKullanimi.java
package gorsel_prog.ders05;

public class FinallyKullanimi {
    public static void bolme(String a, String b){
        try {
            int i = Integer.parseInt(a);
            int j = Integer.parseInt(b);
            System.out.println("Sonuc :"+i/j);
        } catch (NumberFormatException e) {
            System.out.println("Tamsayıya dönüştürme hatası");
            return ; //finally e engel olmaz
        } catch (ArithmeticException e) {
            System.out.println("Sifira Bölme hatası");
            System.exit(1); //finally e engel olur
        } finally{
            System.out.println("Finally bloğu çalıştı");
        }
        System.out.println("method sonu");
    }

    public static void main(String[] args) {
        FinallyKullanimi.bolme("12", "4");
        System.out.println("-----");
        FinallyKullanimi.bolme("12", "karakter");
        System.out.println("-----");
        FinallyKullanimi.bolme("12", "0");
    }
}
```

# throw deyimi

Bu ifade ile programımızda istediğimiz metotdan bir istisna oluşturup fırlatabiliriz.

# throw deyimi

SayiDizisi.java x

```
package gorsel_prog.ders05;

public class SayiDizisi {
    private int[] sayilar;
    public SayiDizisi(int boyut) {
        sayilar = new int[boyut];
    }
    public int getDeger(int indeks)
        throws IndexOutOfBoundsException{
        if (indeks<0 || indeks>=sayilar.length){
            throw new IndexOutOfBoundsException("yanlis indeks");
        }
        return sayilar[indeks];
    }
    public void setDeger(int indeks,int sayi)
        throws IndexOutOfBoundsException{
        if (indeks<0 || indeks>=sayilar.length){
            throw new IndexOutOfBoundsException("yanlis indeks");
        }
        sayilar[indeks]=sayi;
    }
}
```

# throw deyimi

DiziKullanimi.java ✕

```
package gorsel_prog.ders05;

public class DiziKullanimi {
    public static void main(String[] args) {
        SayiDizisi sd = new SayiDizisi(3);
        sd.setDeger(2, 10);
        try {
            System.out.println(sd.getDeger(4));
        } catch (IndexOutOfBoundsException e) {
            System.err.println(e.getMessage());
        }
    }
}
```

# finally ornek

FinallyOrnek.java

```
package gorsel_prog.ders05;

import java.io.FileWriter;
import java.io.IOException;
import java.io.PrintWriter;

public class FinallyOrnek {
    private int[] dizi;
    public FinallyOrnek(int boyut) { dizi = new int[boyut]; }
    public int getEleman(int indeks){ return dizi[indeks]; }
    public void setEleman(int indeks,int sayi){ dizi[indeks]=sayi;}
    public void dosyayaYaz(String dosyaAdi){
        PrintWriter out = null;
        try {
            out = new PrintWriter(new FileWriter(dosyaAdi));
            for (int i = 0; i < dizi.length; i++) {
                out.println(dizi[i]);
            }
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("Dizi indis aşımı");
        } catch (IOException e) {
            System.err.println("Yazma/Okuma hatası");
        } finally{
            if (out!=null) out.close();
            else{
                System.err.println("PrintWriter kapatılamadı");
            }
        }
    }
}
```

# finally ornek

```
FinallyOrnek.java x
package gorsel_prog.ders05;

import java.io.FileWriter;

public class FinallyOrnek {
    private int[] dizi;
    public FinallyOrnek(int boyut) { dizi = new int[boyut]; }
    public int getEleman(int indeks){ return dizi[indeks]; }
    public void setEleman(int indeks,int sayi){ dizi[indeks]=sayi;}
    public void dosyayaYaz(String dosyaAdi){

    public static void main(String[] args) {
        FinallyOrnek f = new FinallyOrnek(3);
        f.setEleman(0, 4);
        f.setEleman(1, 8);
        f.setEleman(2, 12);
        f.dosyayaYaz("dosya.txt");
    }
}
```

# Kendi istisnalarımızı oluşturmak

Tüm istisnalar **Exception** sınıfından türetilmişlerdir. Kendi istisnalarımızı üretmek için bizler de bu sınıftan kalıtım ile yeni sınıflar oluşturmalıyız.

# Kendi istisnalarımızı oluşturmak

```
CiftSayiException.java ✕  
  
package gorsel_prog.ders05;  
  
public class CiftSayiException extends Exception {  
    private static final long serialVersionUID = 1L;  
  
    public CiftSayiException(String mesaj) {  
        super(mesaj);  
    }  
}
```



# Kendi istisnalarımızı oluşturmak

```
CiftSayi.java x
package gorsel_prog.ders05;

public class CiftSayi {
    private int sayi;
    public int getSayi(){return sayi;}
    public void setSayi(int sayi)
        throws CiftSayiException{
        if (sayi%2==0) {
            this.sayi=sayi;
        }else{
            throw new CiftSayiException("Cift sayi olmalidir");
        }
    }
}
```

# Kendi istisnalarımızı oluşturmak

```
CiftSayiKullanim.java X
package gorsel_prog.ders05;

public class CiftSayiKullanim {

    public static void main(String[] args) {
        CiftSayi c = new CiftSayi();
        try {
            c.setSayi(12);
            System.out.println("Sayi:"+c.getSayi());
            c.setSayi(11);
            System.out.println("Sayi:"+c.getSayi());
        } catch (CiftSayiException e) {
            System.err.println(e.getMessage());
        }
    }
}
```

# Görsel Programlama

DERS 05