

Pemogramaman Java

Yoannita, S.Kom

Argumen Command Line
Exception Handling
Package

Argumen Command Line

- `public static void main(String[] args)`
- Pemakaian :
- `javac <namafile.java>`
- `java namaclass arg[0] arg[1]`

Contoh

```
public class CommandLineExample
{
    public static void main( String[] args ){

        System.out.println("Number of arguments=" + args.length);

        System.out.println("First Argument="+ args[0]);

    }
}
```

Exception Handling

Penanganan eksepsi diperlukan untuk alasan berikut :

- User umumnya tidak mengerti dan tidak peduli bagaimana cara suatu program bekerja atau dieksekusi sehingga menampilkan pesan kesalahan yang tidak dimengerti. Anda mungkin ingin mengganti pesan kesalahan dengan pesan yang lebih mudah dimengerti.

Contohnya apabila user mengetikkan huruf untuk harga barang, maka akan muncul pesan error berikut: *...java.lang.NumberFormatException: For input string:....*, dan anda dapat menggantinya dengan *"harga barang harus berupa angka"*

Exception Handling (ii)

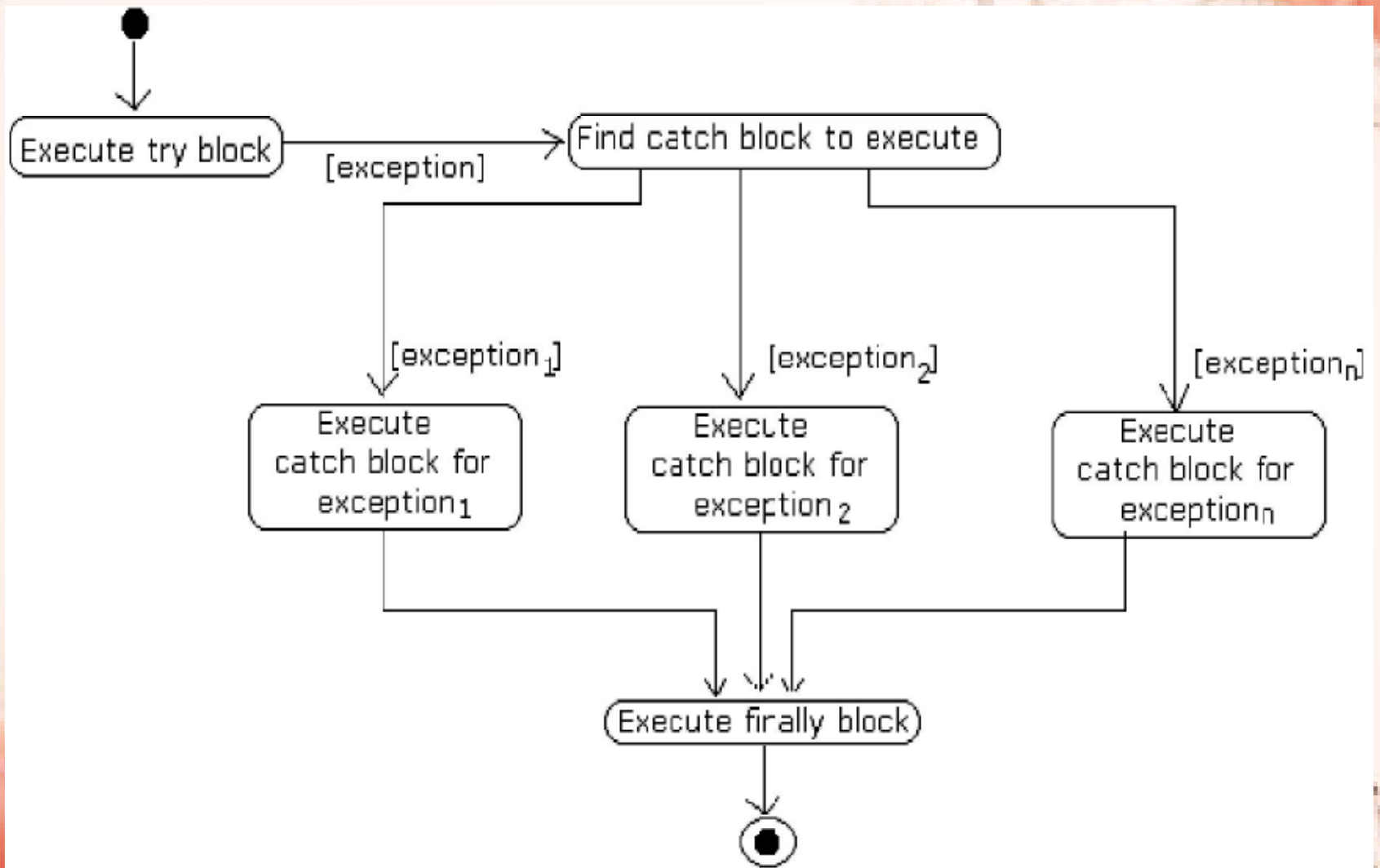
- Anda ingin program memperbaiki kesalahan akibat eksepsi dan terus melanjutkan eksekusi seolah-olah tidak pernah terjadi eksepsi.
- Anda ingin program terus berjalan dan bukannya dihentikan pada saat eksepsi terjadi.

Try... Catch... Finally...

```
try {  
    //tulis pernyataan yang dapat mengakibatkan exception  
    //dalam blok ini  
}  
catch( <exceptionType1> <varName1> ){  
    //tulis aksi apa dari program Anda yang dijalankan jika ada  
    //exception tipe tertentu terjadi  
}  
...  
catch( <exceptionTypen> <varNamen> ){  
    //tulis aksi apa dari program Anda yang dijalankan jika ada  
    //exception tipe tertentu terjadi  
}  
finally{  
    //tambahkan kode terakhir di sini  
}
```

Try... Catch... Finally...

- Exception dilemparkan selama eksekusi dari blok *try* dapat ditangkap dan ditangani dalam blok *catch*.
- Kode dalam blok *finally* selalu di-eksekusi.
- aspek kunci tentang sintak dari konstruksi try-catch-finally:
 - Notasi blok bersifat perintah
 - Setiap blok *try*, terdapat satu atau lebih blok *catch*, tetapi hanya satu blok *finally*.
 - Blok *catch* dan blok *finally* harus selalu muncul dalam konjungsi dengan blok *try*, dan diatas urutan
 - Blok *try* harus diikuti oleh **paling sedikit** satu blok *catch* ATAU satu blok *finally*, atau keduanya.
 - Setiap blok *catch* mendefinisikan sebuah penanganan exception. Header dari blok *catch* harus membawa satu argumen, dimana exception pada blok tersebut akan ditangani.
 - Exception harus menjadi class pelempar atau satu dari subclassesnya.



Gambar 1: Alur kejadian blok try-catch-finally

Exception Handling : Contoh

- Contoh : terdapat kode yang mencetak argumen kedua ketika menjalankan kode menggunakan argumen command-line.
- Perkirakan, tidak ada pengecekan dalam kode Anda untuk angka dari argumen dan hanya mengakses argumen kedua `args[1]`, maka akan didapatkan exception berikut :
 - Exception in thread "main"
 - `java.lang.ArrayIndexOutOfBoundsException: 1`
 - `at ExceptionExample.main(ExceptionExample.java:5)`

Exception Example(i)

```
public class ExceptionExample
{
    public static void main( String[] args ){

        try {
            System.out.println( args[1] );

            // mencetak argumen command line

        } catch( ArrayIndexOutOfBoundsException exp ) {

            System.out.println("Exception caught!");

            /* kesalahan akan dicetak apabila tidak ada argumen command line
            kedua yang diberikan */

        }

    }
}
```

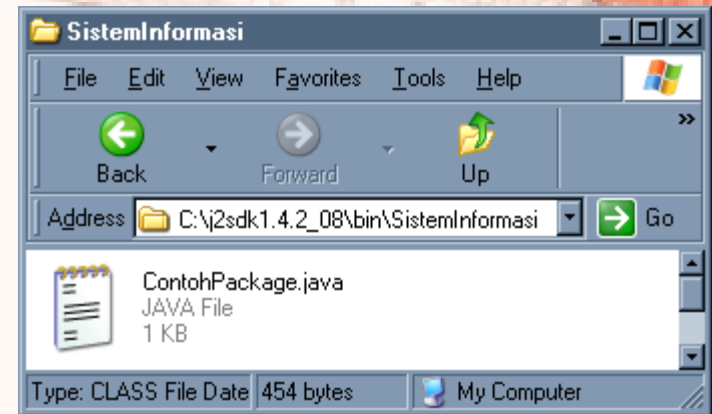
Exception Example(ii)

```
class DemoEksepsi {  
    public static void main(String[] args){  
        try {  
            int[] arr = new int[1];  
            System.out.println(arr[1]); /* ingat bahwa index array dimulai dari nol  
            shg baris ini akan memunculkan pesan kesalahan karena arr yg  
            didefinisikan hanya memuat satu index(arr[0]) */  
            System.out.println("baris ini tidak akan pernah dieksekusi");  
        } catch (ArrayIndexOutOfBoundsException e)  
        {  
            System.out.println("terjadi eksepsi karena index diluar kapasitas array");  
            System.out.println("Kesalahan yang ditangkap :" + e);  
        } finally{ /* finally akan selalu dicetak */  
            System.out.println("percobaan selesai");  
        }  
        System.out.println("Good bye");  
    }  
}
```

Package

package SistemInformasi;

```
class ContohPackage {  
    public static void main (String [ ] args) {  
        System.out.println("penggunaan package");  
    }  
}
```



Keterangan : File ContohPackage.java berada di dalam folder SistemInformasi

Compile:

```
C:\j2sdk1.4.2_08\bin>javac sisteminformasi/contohpackage.java
```

Execute:

```
C:\j2sdk1.4.2_08\bin>java SistemInformasi.ContohPackage
```

Latihan (i) : *Mencetak Argumen*

- Dapatkan input dari user dengan menggunakan argumen command line dan **cetak semua argumen ke layar**. Sebagai contoh, jika user memasukkan :

```
java Hello world that is all
```

- program Anda haruslah mencetak :

```
Hello  
world  
that  
is  
all
```

Latihan (ii) : *Operasi Aritmatik*

- Dapatkan dua buah bilangan yang diinputkan user dari command line dengan argumen command line, kemudian cetak hasil penjumlahannya, hasil pengurangannya, hasil perkalian, dan juga hasil pembagiannya.

```
java ArithmeticOperation 20 4
```

- program anda akan mencetak :

```
sum = 24  
difference = 16  
product = 80  
quotient = 5
```

Latihan (iii)

- Diberikan kode berikut:

```
public class TestExceptions{
    public static void main( String[] args ){
        for( int i=0; true; i++ ){
            System.out.println("args["+i+"]="+
                args[i]);
        }
    }
}
```

- Compile dan jalankan program TestExceptions. Keluarannya akan tampak seperti ini:

```
javac TestExceptions one two three
args[0]=one
args[1]=two
args[2]=three
Exception in thread "main"
java.lang.ArrayIndexOutOfBoundsException: 3
at TestExceptions.main(1.java:4)
```

Next slide

Latihan (iii)

- Ubah program TestExceptions untuk menangani exception, keluaran program setelah ditangkap exception-nya akan seperti ini:

```
javac TestExceptions one two three
```

```
args[0]=one
```

```
args[1]=two
```

```
args[2]=three
```

```
Exception caught:
```

```
java.lang.ArrayIndexOutOfBoundsException: 3
```

```
Quiting...
```