



## 1. Tujuan

- Mendefinisikan exception
- Menangani exception menggunakan blok try-catch-finally

## 2. Latar Belakang

Dalam bagian ini, kita akan mempelajari teknik yang dipakai dalam Java dalam menangani kondisi yang tidak biasa dalam menjalankan operasi normal dalam program. Teknik ini dinamakan **exception handling**.

Exception adalah sebuah peristiwa yang menjalankan alur proses normal pada program. Peristiwa ini biasanya berupa kesalahan(error) dari beberapa bentuk. Ini disebabkan program kita berakhir tidak normal.

Beberapa contoh dari exception yang Anda mungkin jumpai pada latihan-latihan sebelumnya adalah: exception `ArrayIndexOutOfBoundsException`, yang terjadi jika kita mencoba mengakses elemen array yang tidak ada, atau `NumberFormatException`, yang terjadi ketika kita mencoba melalui parameter bukan angka dalam method `Integer.parseInt`.

Untuk menangani exception dalam Java, kita gunakan blok try-catch-finally. Apa yang kita lakukan dalam program kita adalah kita menempatkan pernyataan yang mungkin menghasilkan exception dalam blok ini.

## 3. Percobaan

### Percobaan 1 Menangani Exception handling:

```
public class ExceptionExample {  
    public static void main( String[] args ) {  
        try {  
            System.out.println( args[1] );  
        } catch( ArrayIndexOutOfBoundsException exp ) {  
            System.out.println("Exception caught!");  
        } } }  
}
```

### Hasil Percobaan 1 Output Exception Handling :

```
Output - JavaApplication1 (run-single)  
init:  
deps-jar:  
compile-single:  
run-single:  
Exception caught!  
BUILD SUCCESSFUL (total time: 0 seconds)
```

Output