

EXCEPTIONS

חריגים - Exceptions

- כאשר מתודה נתקלת בשגיאה בזמן ריצה ואינה יכולה לסיים את פעולתה הנורמלית, היא יכולה לזרוק חריג (Exception).
- במקרה כזה המשך הקוד **לא ירוץ** (בדומה ל-return) **ולא יוחזר ערך** מהפונקציה.
- כאשר פונקציה שקראנו לה זורקת חריג, ניתן **לטפל בו** או לזרוק **אותו הלאה** אל המתודה הקוראת.
- במקרה והמתודה זורקת את החריג הלאה, יש להצהיר עליו בחתימת המתודה, אלא אם מדובר בחריג היורש מ-`RuntimeException`.

חריגים - Exceptions

- החריגים בג'אווה מיוצגים ע"י היררכיה של מחלקות כאשר החריג הכללי ביותר מיוצג ע"י המחלקה Exception.
- סוג נוסף של שגיאה – Error – על שגיאות מסוג זה אין צורך להצהיר, ובד"כ לא נרצה לתפוס אותן
- **דוגמא:**
- התכנית הבאה קוראת מספרים מקובץ מסוים ושומרת אותם ברשימה.

חריגים - דוגמא

```
public class NumberList extends ArrayList<Integer>{
```

```
public static void main(String[] args) {
    NumberList list = new NumberList();
    try {
        list.parseFrom("list.txt");
        System.out.println("done: " + list);
    } catch (FileNotFoundException e) {
        System.out.println("FileNotFoundException");
    } catch (IOException | InputMismatchException e) {
        System.out.println("IOException or InputMismatchException");
    }
}
```

בלוקי ה- catch מטפלים בשגיאות שנזרקות מתוך בלוק ה- try

כל חריג מטופל בבלוק הראשון שיודע לטפל בטיפוס שלו או בהורה שלו

החל מג'אווה 7 ניתן לטפל ביותר מטיפוס אחד באותו בלוק

```
private void parseFrom(String fileName) throws IOException {...}
}
```

חריגים

```
public class NumberList extends ArrayList<Integer>{
    public static void main(String[] args) {...}
```

הצהרה על זריקת חריג
עבור כל חריג שדורש זאת.
IOException כולל גם את
,FileNotFoundException
שירש ממנו

```
private void parseFrom(String fileName) throws IOException {
    Scanner scanner = new Scanner(new FileInputStream(fileName));
```

```
try {
    while (scanner.hasNext()) {
        add(scanner.nextInt());
    }
```

זורקת FileNotFoundException

```
} finally {
    System.out.print("Finally...");
    scanner.close();
}
```

זורקת InputMismatchException
שירש מ- RuntimeException

```
if (size() < 3) {
    throw new RuntimeException("Size is too small: " + size());
}
```

finally – ירוץ תמיד, בין אם מתקבלת
שגיאה ובין אם לא

```
}
}
```

זריקת חריג חדש (runtime)

מה יודפס עבור הקלט...

Finally...done: [134, 2342, 433]

134 2342 433 מכיל list.txt •

Finally...Exception in thread "main"
NumberList\$1: Size is too small: 2
 at NumberList.parseFrom(NumberList.java:35)
 at NumberList.main(NumberList.java:14)

134 2342 מכיל list.txt •

Finally...IOException or
 InputMismatchException

134 2342 seven מכיל list.txt •

FileNotFoundException

list.txt אינו קיים •