

## עצור 9א. תכנות מונחה עצמי.

### פעולות קלט-פלט – המשך

#### קלט/פלט מקבצי טקסט

JAVA מספקת מחלקות רבות אשר משמשות לביצוע פעולות קלט ופלט. רוב המחלקות הן מחלקות שהאובייקטים מטיפוסן מייצגים streams.

תרגום המילה stream לעברית הוא זרם/נחל, ואכן, האובייקטים שמהווים streams מתארים זרימה של bytes אשר מהווה, למעשה, זרימה של נתונים. ניתן גם לראות ב-stream מקור של bytes, או לחלופין, יעד של bytes, אשר סדר ה-bytes שנשלח אליו או מגיע ממנו בעל משמעות. כך, למשל, תכנית שמקבלת קלט של נתונים מן המקלדת מבצעת זאת באמצעות stream (זרם) של בתים (byte) מהמקלדת. כיוון שקשה לדמיין במחשבותינו אובייקט שמייצג זרימה של נתונים ניתן לראות באובייקטים הללו ייצוג לצינורות שדרכם הנתונים עוברים. את ה-streams שמיוצגים על ידי המחלקות ששייכות ל-java.io ניתן לחלק לשתי קבוצות:

#### input streams

אלה הם מקורות שניתן לקלוט/לקרוא מהם בתים (byte). למקורות אלה לא ניתן לכתוב/לשלוח בתים (byte) בחזרה.

#### output streams

אלה הם יעדים שניתן לשלוח/לכתוב אליהם בתים (byte). מיעדים אלה לא ניתן לקרוא/לקבל בתים (byte) חזרה.

### היררכית מחלקות ה-I/O ב-JAVA

נציג את שתי ההיררכיות הבולטות ביותר:

```
java.lang.Object
  java.io.Reader
    java.io.BufferedReader
    java.io.InputStreamReader
    java.io.FileReader
java.lang.Object
  java.io.Writer
    java.io.PrintWriter
    java.io.OutputStreamWriter
    java.io.FileWriter
```

### דוגמה לכתיבה בקובץ ולקריאה מקובץ

```
import java.io.*;
public class ReadWriteOnly {
    public static void writeFile(String fileName) {
        // try write to the file
        try {
            FileWriter fw = new FileWriter(fileName);
            PrintWriter outs = new PrintWriter(fw);
            outs.println("the first string: java, C++ ");
            outs.println("the second string: geometria analitit");
            outs.println("the third string, other new string");
            outs.println("the fourth string");
            outs.close();
        }
    }
}
```

```

        fw.close();
    }
    catch(IOException ex) {
        System.out.print("Error writing file\n" + ex);
    }
}

public static void readFile (String fileName){
    // try read from the file
    try {
        FileReader fr = new FileReader(fileName);
        BufferedReader br = new BufferedReader(fr);
        String str;
        str = br.readLine();
        System.out.println(0+" "+str);
        for(int i=1; str!=null; i=i+1) {
            str = br.readLine();
            if (str != null){
                System.out.println(i+" "+str);
            }
        }
        br.close();
    }
    catch(IOException ex) {
        System.out.print("Error reading file\n" + ex);
        System.exit(2);
    }
}

public static void main(String[] args) {
    String name = "mySimpleFile.txt";
    writeFile(name);
    readFile(name);
}
}

```