

# תוכנה 1

תרגול 1: סביבת העבודה ומבוא ל-Java

1

## כירוקרטיה

דבר נתמלי  
שעת קבלה: רביעי 18:00-19:00, בתיאום מראש

נר אטיאם  
שעת קבלה: בתיאום מראש

- אתר הקורס: <http://courses.cs.tau.ac.il/software1/1213b>
- סביבת המחשוב באוניברסיטה היא Linux
- תנאי קדם: פתיחת חשבון אישי במחשבי האוניברסיטה
- הנחיות לפתיחת חשבון והכרת סביבת העבודה באתר הקורס.

2

## עוד כירוקרטיה

- נוהל הגשת תרגילים (פרטים מלאים באתר)
  - מועד ההגשה
  - שיטת חישוב הציון
  - הגשה באיחור
  - הגשה דרך ה-moodle
  - פורום הקורס (גם ב-moodle)
- הגשת תרגיל מספר 1
  - ביום ד' הבא
  - פרטים באתר

3

## סביבת פיתוח והרצה Java-f

- גרסת ה-Java שעמדה נעבוד:
  - חבילת סביבת ההרצה: [Java SE \(Standard Edition\) 7.0](http://www.oracle.com/technetwork/java/javase/downloads/index.html)
  - JRE (Java Runtime Environment) that includes:
    - JVM (Java Virtual Machine)
    - Standard Class Library
  - חבילת ערכת הפיתוח: JDK (Java Development Kit) that includes:
    - JRE
    - Command line tools: compiler, debugger etc.
- הורדה ותיעוד ב-  
<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

4

## סביבת פיתוח IDE

- IDE = Integrated Development Environment
- סביבה המשלבת רכיבי/כלי פיתוח עצמאיים:
  - עורך טקסט (editor)
  - סייר הקבצים (browser)
  - מהדר (compiler)
  - סביבת זמן ריצה (JRE)
  - מנפה השגיאות (debugger)
  - ועוד...
- Eclipse – ה-IDE בו נשתמש בקורס.

5

## Eclipse



- IDE המתאים גם לפיתוח תוכנה ב-Java
- ניתן להתקנה ב-Linux, Windows ועוד
- דורש התקנה בנפרד של JRE (או JDK)
- אתר הבית: [www.eclipse.org](http://www.eclipse.org)
- הורדת התוכנה כקובץ zip (הוראות התקנה ב-הכרת סביבת המחשוב באתר הקורס)
- אוסף גדול של מאמרים
- הכרות: דפי עבודה ללימוד Eclipse באתר הקורס
- דוגמה: פיתוח והרצת תכנית "Hello World" ב-Eclipse
- הסבר מפורט לגבי השמשת סביבת העבודה מהבית:  
<http://courses.cs.tau.ac.il/software1/1213b/misc/workenv.pdf>

6

## טיפוסים בסיסיים

- טיפוסים בסיסיים (פרימיטיביים): 8 טיפוסים מוגדרים בשפה שמועדים להכיל ערכים פשוטים:
  - מספרים שלמים: `byte`, `short`, `int`, `long`
  - מספרים ממשיים: `float`, `double`
  - תווים: `char`
  - ערכים בוליאניים: `boolean`
- טיפוס הפנייה: טיפוסים מורכבים היכולים גם להכיל מידע וגם לספק שיחותים (יוסבר בהמשך)
  - המתכנת יכול להגדיר טיפוס הפנייה חדשים
  - דוגמאות מיוחדות: מחרוזות ומערכים



7

## טיפוסים פרימיטיביים

- בזיכרון המחשב נשמר המידע בפורמט בינארי
- סיבית (bit): ספרה בינארית ('0' או '1')
- בייט (byte) (octet): קבוצה של 8 סיביות

Type	Size	Value Range
long	64 bits	$-2^{63} \rightarrow 2^{63}-1$
int	32 bits	$-2^{31} \rightarrow 2^{31}-1$
short	16 bits	$-2^{15} \rightarrow 2^{15}-1$
byte	8 bits	$-128 \rightarrow 127$
double	64 bits	beyond the scope of the discussion
float	32 bits	beyond the scope of the discussion
char	16 bits	most alphabet languages
boolean	"1 bit"	true, false

8

## טיפוסים לא פרימיטיביים

- ישנם אלפי טיפוסים לא פרימיטיביים בספרייה התקנית של Java.
  - תיעוד: Java API (Application Programming Interface)
  - <http://docs.oracle.com/javase/7/docs/api/>
- אנו כמתכנתים ניצור טיפוסים לא פרימיטיביים נוספים
  - דוגמאות: מערכים, מחרוזות (String)
- הפניה (reference): משתנה מטיפוס שאינו פרימיטיבי
 

```
String myString = "Hello World"
```

Non-primitive type

קובץ (object)

9

## מחרוזות

- אופרטור שרשרת:

- "Hello " + "World" is "Hello World"
- "19" + 8 + 9 is "1989"

- דוגמאות לפונקציות מהמחלקה String:

```
String str1 = "Hello";
char c = str1.charAt(0); // c == 'H'
String str2 = str1.toUpperCase(); // str2 == "HELLO"
int strLength = str1.length(); // i == 5
```

עוד ->

<http://docs.oracle.com/javase/7/docs/api/index.html?java/lang/String.html>

10

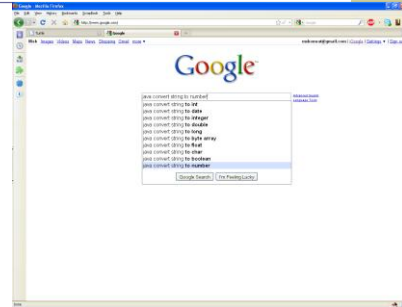
## המרה לאספרים

- `Long.parseLong`
- `Integer.parseInt`
- `Short.parseShort`
- `Byte.parseByte`
- `Double.parseDouble`
- `Float.parseFloat`
- `Boolean.parseBoolean`

```
public static void main(String[] args) {
    int i = Integer.parseInt("1"); // i==1
    double d = Double.parseDouble("-12.45e2"); // d==-1245.0
}
```

11

## איך נדע...?



12

## התשובה



## מצרכים בקצרה

- מבנה נתונים פשוט
  - אוסף של פריטים מאותו טיפוס
  - גישה באמצעות אינדקס
  - נשתמש ב [] לציין מערך
  - מערך של int בשם odds.
- ```
int[] odds;
```
- (0-מ) אינדקס (מתחיל מ-0)
- |   |   |   |   |   |    |    |    |
|---|---|---|---|---|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5  | 6  | 7  |
| 1 | 3 | 5 | 7 | 9 | 11 | 13 | 15 |
- odds.length == 8  
 ■ הרחבה על מערכים בתרגול הבא

## הצגת ארגומנטים לתכנית

```

public class MyClass {
    public static void main(String[] args) {
        System.out.print(args.length + "\t");
        System.out.print(args[0] + "\t");
        System.out.print(args[args.length - 1]);
    }
}
    
```

■ כיצד מעבירים ארגומנטים לתוכנית?



■ Eclipse: Run → Run Configurations → Arguments

Output: 4 1 hello world

## אחרונות ותווים

- כתוב תוכנית שמקבלת תו כארגומנט ומדפיסה: את התו
- את התו העוקב לו

```

public static void main(String[] args) {
    char c = args[0].charAt(0);
    char c1 = (char) (c + 1);
    System.out.println(c + "\t" + c1);
}
    
```

## אחרונות ותווים

- כתוב תוכנית המקבלת תו מ- {a,b,...,z} ומדפיסה את ה-Uppercase שלו
- נחשב את המיקום של התו בabc ונמיר אותו לאותו תו (אותו מיקום) ב ABC

```

public static void main(String[] args) {
    char c = args[0].charAt(0);
    System.out.println((char) (c - 'a' + 'A'));
}
    
```

## דרכים נוספות?



## אחרונות ותווים

String.toUpperCase() בעזרת

```
public static void main (String[] args) {  
    System.out.println (args [0].toUpperCase ());  
}
```

Character.toUpperCase() בעזרת

```
public static void main (String[] args) {  
    char c = args [0].charAt (0);  
    System.out.println (Character. toUpperCase (c) );  
}
```

19



20

הסוף...

21