

# תוכנה 1

תרגול 1: סביבת העבודה ומבוא ל-Java  
ליאור שפירא ומתי שמרת

# כירוקרטיה

## מתי שמרת

שעת קבלה: שני 10-11, נא לתאם באימייל  
המשרד שלי: בניין הנדסת תוכנה, חדר 209

## ליאור שפירא

שעת קבלה: שלישי 16-17, נא לתאם באימייל  
המשרד שלי: שרייבר 002 (קומת קרקע מעבדת גרפיקה)

אתר הקורס: <http://courses.cs.tau.ac.il/software1/0809a>

סביבת המחשוב באוניברסיטה היא Linux

■ תנאי קדם: פתיחת חשבון אישי במחשבי האוניברסיטה

■ הנחיות לפתיחת חשבון והכרת סביבת העבודה באתר הקורס.

# צף הקורס

- קורס תכנות בסיסי
- מלמד תכנות מונחה עצמים באמצעות בשפת  
Java
- הקורס דורש לימוד עצמי של פרטים

# סביבת פיתוח והרצה Java-f

■ גרסת ה-Java שעמה נעבוד:

[Java SE \(Standard Edition\) 6.0 \(update 7\)](#)

■ חבילת סביבת ההרצה:

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://java.sun.com/javase>

# סביבת פיתוח שלוקה

**IDE** = Integrated Development Environment ■

סביבה המשלבת רכיבי/כלי פיתוח עצמאיים: ■

עורך טקסט (editor) ■

סייר הקבצים (browser) ■

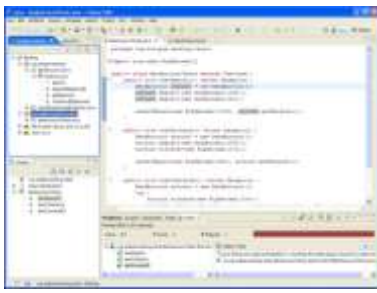
מהדר (compiler) ■

סביבת זמן ריצה (JRE) ■

מנפה השגיאות (debugger) ■

ועוד... ■

**Eclipse** – ה- IDE בו נשתמש בקורס. ■



# 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 התכנה (גרסא classic 3.4.1) כקובץ zip (הוראות התקנה ב-[הכרת סביבת המחשוב](#) באתר הקורס)
- אוסף גדול של מאמרים
- הכרות: [דפי עבודה ללימוד Eclipse](#) באתר הקורס
- דוגמא: פיתוח והרצת תכנית "Hello World" ב Eclipse

# טיפוסי השפה

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



# הטיפוסים הפרימיטיביים

■ בזיכרון המחשב נשמר המידע בפורמט בינארי

■ **סיבית (bit):** ספרה בינארית ('0' או '1')

■ **בייט (octet, byte):** קבוצה של 8 סיביות

	Type	Size	Value Range
שלמים	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	$-128 \rightarrow 127$
ממשיים	double	64 bits	beyond the scope of the discussion
	float	32 bits	
תווים	char	16 bits	most alphabet languages
ערכים לוגיים	boolean	"1 bit"	true, false



# הארת סיבוסים פרימיטיביים

```
public static void main(String[] args) {
    long l = 20000000000+20000000000; // l == -294967296
    int i = (int) 1.999999999; // i == 1
    float f = (float) 1.999999999; // f == 2
    f = 5/2; // f == 2
    f = (float) (5/2); // f == 2
    f = (float) 5/2; // f == 2.5
    f = 5 / (float) 2; // f == 2.5
    short a = 2;
    short c = a*a; // compilation error: cannot convert from int to short
}
```

עוד על המרות ב-

[http://java.sun.com/docs/books/jls/third\\_edition/html/conversions.html](http://java.sun.com/docs/books/jls/third_edition/html/conversions.html)

# טיפוסים לא פרימיטיביים

ישנם אלפי טיפוסים לא פרימיטיביים בספריה התקנית של Java.

■ תיעוד: Java API (Application Programming Interface)

<http://java.sun.com/javase/6/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

עצם (object)

# מתכונות

■ אופרטור שרשור:

- "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 ב-

<http://java.sun.com/javase/6/docs/api/java/lang/String.html>

# מחרוזות

■ "המרה" ממחרוזת למספר:

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

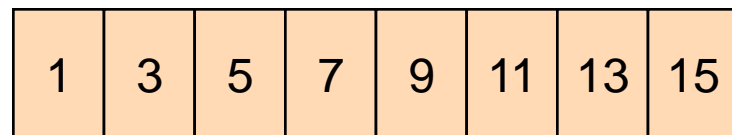
```
public static void main(String[] args) {  
    int i = Integer.parseInt("1"); // i==1  
    double d = Double.parseDouble("-12.45e2"); // d== -1245.0  
}
```

# מצרכים בקצרה

- מבנה נתונים פשוט
- אוסף של פריטים מאותו טיפוס
- גישה באמצעות אינדקס
- נשתמש ב [] לציין מערך
- מערך של int בשם odds.

```
int odds [ ] ;
```

אינדקס (מתחיל מ-0) → 0 1 2 3 4 5 6 7



odds.length == 8

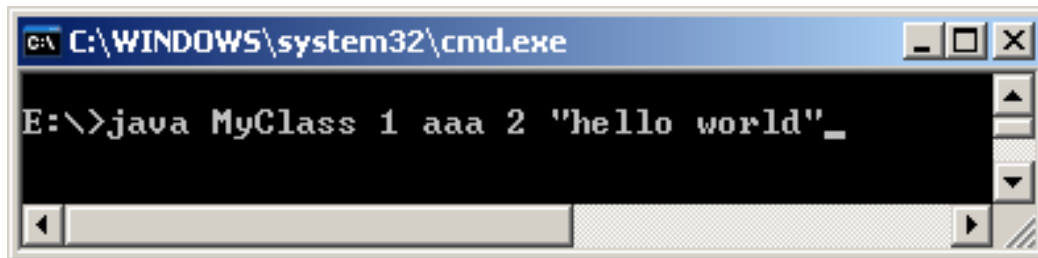
■ הרחבה על מערכים בתרגול הבא

# הצגת ארגומנטים לתוכנית

מערך המכיל את הארגומנטים  
שהועברו לתוכנית עם הרצתה

```
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? ■



The screenshot shows a Windows command prompt window titled "C:\WINDOWS\system32\cmd.exe". The prompt is "E:\>". The command entered is "java MyClass 1 aaa 2 'hello world'". The cursor is at the end of the command line.

command line ■

Run → Run Configurations → Arguments :Eclipse ■

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

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

פתרון א':

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

פתרון ב':

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

פתרון ג':



# תחום ההכרה (Scope) fe משתנה מקומי

- בלוק = תחום סגור ע"י סוגריים מסולסלים
- תחום ההכרה של משתנה מקומי = הבלוק בו הוא מוגדר

הבלוק הראשי של main

```
public static void main(String[] args) {  
    int i=1;  
    {  
        😊 i++;  
        ❌ int i=2;  
        😊 int j=2;  
        😊 System.out.println(i+j);  
    }  
    😊 i++;  
    ❌ System.out.println(j+1);  
}
```

בלוק פנימי

...פיו