

בירוקרטיה

- אני: ליאור שפירא
- שעת קבלה: יום ג' 15-16, נא לתאם באי-מייל
- המשרד שלי: שרייבר 002 (מעבדת גרפיקה)
- תא: 279 – שרייבר קומה 1 מול המעלית
- שעות התרגול
- יום חמישי 12-13 (חוץ מהשלמות ביום שישי)
- שיעורי בית: יש (הסברים באתר)
- **אתר הקורס:** <http://courses.cs.tau.ac.il/software1/0708b>
- סביבת המחשוב באוניברסיטה היא Linux
- **תנאי קדם:** פתיחת חשבון אישי במחשבי האוניברסיטה
- הנחיות לפתיחת חשבון והכרת סביבת העבודה באתר הקורס.

2

תוכנה 1

תרגול 1: סביבת העבודה ומבוא ל-Java
ליאור שפירא ואוהד ברזילי

1

סביבת פיתוח והרצה Java-f

- גרסת ה-Java שעמה נעבוד:
- [Java SE \(Standard Edition\) 5.0 update 13](#) (newest is 6)
- 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>

4

על הקורס

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

3



Eclipse

- IDE המתאים גם לפיתוח תוכנה ב Java
- ניתן להתקנה ב- Windows, Linux ועוד
- דורש התקנה בנפרד של JRE (או JDK)
- אתר הבית: www.eclipse.org
- הורדת התכנה (גרסא 3.3.2 classic) 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" באקליפס

6

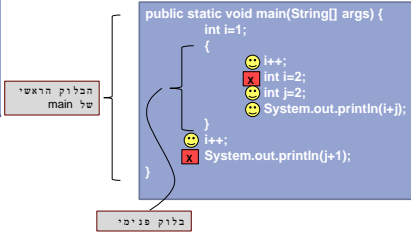
סביבת פיתוח עלובה

- 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

תחום ההכרה (Scope) fe משתנה מקומי

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



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]);
    }
}
    
```

איך מעבירים ארגומנטים לתוכנית? command line

```

C:\WINNT\system32\cmd.exe
C:\>java MyClass 1 aaa 2 "hello world"
    
```

Run...→Arguments→Program arguments :Eclipse

```

Output? 4 1 2 hello world
    
```

הטיפוסים הפרימיטיביים

- בזכרון 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 → 2 ⁶³ -1
int	32 bits	-2 ³¹ → 2 ³¹ -1
short	16 bits	-2 ¹⁵ → 2 ¹⁵ -1
byte	8 bits	-128 → 127
double	64 bits	beyond the scope of the discussion
float	32 bits	
char	16 bits	most alphabet languages
boolean	"1 bit"	true, false

שלמים: long, int, short, byte
ממשיים: double, float
תווים: char
ערכים לוגיים: boolean

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 טיפוס הפנייה חדשים
 - דוגמאות מיוחדות: מחרוזות ומערכים



טיפוסים לא פרימיטיביים

- ישנם אלפי טיפוסים לא פרימיטיביים בספריה התקנית של Java.
- תיעוד: Java API (Application Program Interface) <http://java.sun.com/j2se/1.5.0/docs/api/>
- אנו כמתכנתים ניצור טיפוסים לא פרימיטיביים נוספים
- דוגמאות: מערכים, מחרוזות (String)
- הפנייה (reference): משתנה מטיפוס שאינו פרימיטיבי

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

Non-primitive type

עצם (Object)

הצגת טיפוסים פרימיטיביים

```

public static void main(String[] args) {
    long l = 2000000000+2000000000 //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 הכרות ב-

http://java.sun.com/docs/books/jls/second_edition/html/conversions.doc.html

אחרונות

■ "המרה" ממחרוזת מספר:

- `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
}
```

14

אחרונות

■ 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/j2se/1.5.0/docs/api/java/lang/String.html>

13

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'));
}
```

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

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

16

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);
}
```

15

הסוף...

17