

2) איך מתחילים ללמד עצמים ומחלקות?

לחשוף את התלמידים לשימוש במחלקות מוכנות ולהזכיר שזו מחלקה ממנה יוצרים עצמים.

דוגמאות: Scanner, Random לעומת מחלקת שירות Math

להביא דוגמאות של שימוש במחלקות הקיימות ב java.awt ובמחלקות שיצרו במחלקה להוראת המדעים בירושלים: Bucket, Turtle

(3) השיעור הראשון שבו יעשה שימוש בנושא "עצמים"
מיקום השיעור יהיה לפני או אחרי לולאות.

דוגמאות:

- שימוש במחלקות הנמצאות `java.awt`
- שימוש במחלקות הגרפיות `JPanel`, `JFrame`
- שימוש במחלקות שיצרו באוניברסיטת ירושלים, המחלקה להוראת המדעים
`Turtle`, `Bucket`

דוגמה ראשונה:

דף עבודה מספר 4 , פרק 2, ספר עיצוב תכנה
שימוש במחלקה `Point` מתוך `javaAPI`

המחלקה `Point` (מתוך `Java API`)

בתרגיל זה נשתמש באחת המחלקות המוכנות ב-`JavaAPI` ששמה `Point`. המחלקה `Point` מייצגת נקודה דו ממדית. אנו נביא כאן ממשק חלקי שלה. ממשק מלא קיים ונגיש דרך סביבת העבודה.

<code>Point (int x, int y)</code>	הפעולה בונה נקודה בהתאם לפרמטרים <code>x</code> ו- <code>y</code>
<code>void move (int x, int y)</code>	הפעולה מקבלת כפרמטרים ערכי <code>x</code> ו- <code>y</code> חדשים ומזיזה את הנקודה בהתאם
<code>void translate (int dx, int dy)</code>	הפעולה מוסיפה לערך ה- <code>x</code> של הנקודה את הפרמטר <code>dx</code> ולערך ה- <code>y</code> של הנקודה את הפרמטר <code>dy</code>
<code>String toString()</code>	הפעולה מחזירה מחרוזת המתארת את הנקודה

על מנת להשתמש במחלקה זו עלינו "לייבא" אותה. המחלקה `Point` נמצאת בחבילה הנקראת `awt`. זו היא חבילה המכילה את כל המחלקות הגרפיות של ג'אווה. על מנת "לייבא" אותה עלינו לרשום בתחילת הקובץ (לפני הגדרת המחלקה):

```
import java.awt.Point;
```

מה עליכם לעשות?

א. עליכם ליצור שתי נקודות שונות, להדפיס אותן. לאחר מכן עליכם להזיז את הנקודות למקום חדש: פעם אחת באמצעות הפעולה move, ופעם אחת באמצעות הפעולה translate ולהדפיס אותן אחרי כל הזזה. הגדירו במילים מה השוני בין שתי פעולות ההזזה.

ב. 1. מה לדעתכם יקרה בעקבות ההרצה של שתי השורות הבאות:

Point p;

p.move (3, 4);

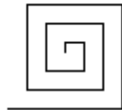
2. בדקו בעזרת הפעולה הראשית האם התוצאה היא כפי שציפיתם.

דוגמה מספר 2:עבודה עם המחלקה הגרפית Turtle מתוך המאמר unit4 . (עמודים 90-92, יסודות 1)

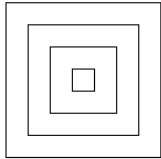
(1) צייר רבוע לפי עמוד 91. שם המחלקה UseTurtle1

(2) צור מחלקה UseTurtle2 , העתק אליה את תוכן מחלקה UseTurtle1 ושנה כך שהציור יתבצע באמצעות לולאת for

(3) צור מחלקה UseTurtle3 .
 מטרת התוכנית לצייר מצולע לפי מספר צלעות שייקלט מהמשתמש.
 • התוכנית תקלוט את מספר הצלעות למשתנה n
 • התוכנית תחשב את זווית הפנייה לפי $\text{angle} = 360.0/n$
 • התוכנית תצייר את המצולע.



(4) צור מחלקה UseTurtle4 .
 מטרת התוכנית ליצור שבלול ריבועי ←
 • קלוט מספר הסיבובים
 • קבע את שיעור הגדלת הריבוע (למשל 10)



(5) צור מחלקה UseTurtle5 .
 מטרת התוכנית לצייר ריבוע בתוך רבוע, כל רבוע יותר גדול מקודמו. ←
 קלט: מספר ריבועים, גודל רבוע ראשוני, מרווח בין ריבועים.

(6) צור מחלקה UseTurtle6 .
 מטרת התוכנית לצייר דיאגרמת מקלות בעזרת הצב. אורכי המקלות יהיו מספרים אקראיים בין 50-100 .
 התוכנית תחשב את גודל הקו הארוך ביותר, הקצר ביותר ואת האורך הממוצע.
 התוכנית תציג כפלט את האורכים ותצייר בצבע שונה את הקו הארוך ביותר, הקצר ביותר והממוצע.

(*) יש להשתמש במחלקה Color . לשם כך ייבא אותה: `import java.awt.Color;`

דוגמה מספר 3:**תרגילים ראשונים בשימוש מימשק משתמש גראפי – Gui**

מטרת התרגיל: יצירת חלון, הוספת רכיב גרפי פאנל, הוספת רכיבים גרפיים לפאנל ותצוגתם.
הנחיה: הרץ אחרי כל הוראה, נסה לשנות את הפרמטרים ולבדוק מה נראה על המסך לאחר השינוי.

(1) צור פרויקט בשם GUI

(2) צור מחלקה בשם First מסוג Java Main Class

יצירת חלון JFrame

(3) צור חלון (מסגרת) בעל כותרת: First

```
JFrame frame = new JFrame("First");
```

(4) קבע את מימדי החלון להיות בעל רוחב 500 פיקסלים ואורך 300 פיקסלים.

```
frame.setSize(500,300);
```

(5) הרץ ובדוק, האם רואים את החלון על המסך?

```
frame.setVisible(true);
```

(6) הפוך את החלון לנראה:

(7) אפשר סגירת חלון בלחיצה על x בפינה עליונה של החלון.

```
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

(8) קבע ניהול סידור הרכיבים על החלון בעזרת מנהל הפריסה LayerManager

באפשרותך לנסות סוגים נוספים.

```
frame.setLayout(new FlowLayout());
```

יצאת פאנל JPanel

(9) צור רכיב מסוג פאנל

```
JPanel panel1 = new JPanel();
```

(10) קבע לו גודל קבוע:

```
panel1.setPreferredSize(new Dimension(300,100));
```

רוחב 300 ואורך 100.

(11) קבע צבע רקע: ירוק.

(12) מחק את שורת קביעת הגודל של המסגרת.

(13) הוסף את הפנאל למסגרת

(14) ארוז את הרכיבים למסגרת.

(15) הרץ ובדוק את התוצאה.

(16) שנה את הפרמטרים ובדוק איך הם משפיעים על התצוגה.

הוספת רכיבים פאנל JPanel

(17) הוסף אזור טקסט מסוג JTextArea וכתוב בו משהו.

```
JTextArea jtArea = new JTextArea("This is a text area", 3, 20);
```

הפעולה מקבלת טקסט לכתיבה,

מספר שורות ומספר עמודות.

(18) הוסף את אזור הטקסט לפאנל.

```
panel.add(jtArea);
```

(19) הרץ ובדוק אם נוסף.

(20) כדי לראות את הרכיב שנוסף יש לבקש מהמסגרת לצייר את כל הרכיבים שלה.

זאת עושים בעזרת הפעולה paintAll אשר מקבלת אובייקט מטיפוס Graphics.

אפשר להתייחס ל Graphics כאל מברשת צבע שבעזרתה מציירים את הרכיבים.

```
frame.paintAll(frame.getGraphics());
```

(21) הרץ מחדש . האם עכשיו רואים את הרכיבי שנוסף?

בוא נשנה קצת את מאפייני הרכיב של משטח הטקסט, למשל שינוי גופן , שינוי צבע הגופן.

(22) קבע את פונט הכתב ל David , מסוג מודגש ובגודל 14

```
jtArea.setFont(new Font("David", Font.BOLD, 14));
```

נסה לבחור סוגים שונים ובדוק את המראה.

```
jtArea.setForeground(Color.red);
```

(23) שנה את צבע הכתב , למשל לאדום.

(24) צור תווית שכתוב בה Label וצור תיבת טקסט למילוי על ידי המשתמש לפאנל.

```
JLabel jLabel1 = new JLabel("Label1");
```

```
JTextField jtf = new JTextField(4);
```

```
// add Components to panel
```

הוסף אותם לפאנל.
הרץ ובדוק את התוצאה.

(25) הוסף 3 כפתורים. כפתור הוא מטיפוס JButton. כפתור שכתוב עליו אדום וצבע הרקע שלו אדום. כפתור שכתוב עליו כחול וצבע הרקע שלו כחול. כפתור שכתוב עליו צהוב וצבע הרקע שלו צהוב.. הוסף אותם לפאנל והרץ.

[דוגמה מספר 4:](#)

להציג את המחלקה String עם API חלקי ולתת משימות הקשורות במחלקה זו.

4) השיעור הראשון שבו יתחיל הדיון ב"מחלקות".
מיקום, אחרי לולאות ואפילו אחרי מערכים.

דוגמאות למחלקות ראשונות:

- א- המחלקה "שעון" Clock המכילה כתונות: שעה, דקה, שנייה
- ב- מחלקה Date המכילה כתונות: יום, חודש, שנה
- ג- המחלקה Point לממש לבד. התכונות שיעור x ושיעור y של נקודה במישור.
- ד- מחלקה Box המכילה כתונות: אורך, רוחב, עומק של קופסה.
- ה- מחלקה תלמיד Student המכיל כתונות: שם, ציון 1, ציון 2

דיונים לכתיבת מחלקות:

נושא מידול בעיה, מתי באמת כדאי לכתוב מחלקה, טיפוס חדש שיעזור בפתירת הבעיה.

[פתרון לדוגמה מספר 2:](#)

```
/**
 * Use class Turtle: draw square
 * @author Evi
 * @version 15/9/08
 */
import java.awt.Color;
import java.util.Random;
import unit4.turtleLib.Turtle;
public class UseTurtle
{
    /**
     * ex1: draw rectangle
     */
    public static void ex1()
    {
        Turtle t1 = new Turtle();
        t1.tailDown();
        t1.moveForward(100);
        t1.turnRight(90);
        t1.moveForward(100);
        t1.turnRight(90);
        t1.moveForward(100);
        t1.turnRight(90);
        t1.moveForward(100);
        t1.turnRight(90);
    }
    /**
     * ex2: draw rectangle using for
     */
    public static void ex2()
    {
        Turtle t1 = new Turtle();
        t1.tailDown();
        for (int i=1; i<=4; i++)
        {
            t1.moveForward(100);
            t1.turnRight(90);
        }
        t1.setVisible(false);
    }
    /**
     * draw poligon
     * @param n : int , number of sides
     */
    public static void ex3(int n)
    {
        double angle = 360.0/n;
        Turtle t1 = new Turtle();
        t1.tailDown();
        for (int i=1; i<=n; i++)
```

```
{
    t1.moveForward(100);
    t1.turnRight(angle);
} // end for
t1.setVisible(false);
}
public static void ex6()
{
    final int N= 4; // no., of lines
    int x;          // size of line
    int max, min;
    double avg;
    Random rnd = new Random();
    Turtle t1 = new Turtle();
    avg = 0;
    min = 101;
    max = 49;
    for (int i=1; i<=N; i++)
    {
        x = rnd.nextInt(51)+50;
        System.out.println("line1: " + x);
        avg += x;
        if (x>max)
            max = x;
        if (x<min)
            min = x;
        t1.tailDown();
        t1.moveForward(x);
        t1.tailUp();
        t1.moveBackward(x);
        t1.turnRight(90);
        t1.moveForward(20);
        t1.turnLeft(90);
    } // end for
    avg = avg/N;
    t1.setTailColor(Color.GREEN);
    t1.turnRight(90);
    t1.moveForward(40);
    t1.turnLeft(90);
    t1.tailDown();
    t1.moveForward(min);
    t1.moveBackward(min);
    t1.tailUp();
    t1.setTailColor(Color.RED);
    t1.turnRight(90);
    t1.moveForward(20);
    t1.turnLeft(90);
    t1.tailDown();
    t1.moveForward(max);
    t1.moveBackward(max);
    t1.tailUp();
}
```

```

t1.setTailColor(Color.ORANGE);
t1.turnRight(90);
t1.moveForward(20);
t1.turnLeft(90);
t1.tailDown();
t1.moveForward(avg);
t1.moveBackward(avg);
t1.setVisible(false);

System.out.println("maximum line is: " + max);
System.out.println("minimum line is: " + min);
System.out.println("average line is: " + avg);
}
/**
 * draw square form of snail
 * @param n: int, number of turns
 * @param w : width of expand
 */
public static void ex4(int n, int w)
{
//    System.out.println("Enter no of steps");
//    int n=in.nextInt();
//    int delta = 5;    // size to add
    Turtle t1 = new Turtle();
    t1.tailDown();
    for (int i=1; i<=n; i++)
    {
        t1.moveForward(w);
        t1.turnRight(90);
        w +=5;
    }// end for
    t1.tailUp();
    t1.setVisible(false);
}
/**
 * draw square inside square
 * @param n : int, number of squares
 */
public static void ex5(int n)
{
    int w=20;
    int d=20;
    Turtle t1 = new Turtle();
    for (int index = 1; index<=n; index++)
    {
        t1.tailDown();
        for (int i=1; i<=4; i++)
        {
            t1.moveForward(w);
            t1.turnRight(90);
        }// end for
    }
}

```

```
t1.tailUp();
t1.moveBackward(d);
t1.turnLeft(90);
t1.moveForward(d);
t1.turnRight(90);
w = w+2*d;
}
t1.setVisible(false);
}
/-- ex 6
/**
 * draw square with Turtle t with size
 * @param t: Turtle
 * @param size
 */
public static void square(Turtle t, int size)
{
    for (int i=1; i<=4; i++)
    {
        t.moveForward(size);
        t.turnRight(90);
    }
}
public static void ex7(int n)
{
    int delta = 30;
    Turtle t1 = new Turtle();
    t1.tailDown();
    for (int i=1; i<=n; i++)
    {
        square(t1, delta);
    }
    t1.setVisible(false);
}

public static void main(String[] args)
{
    // ex3(6)
    // ex4(12,5);
    // ex5(4);
    // ex6(6);
}
}
```

[פתרון דוגמה 3:](#)

```

/*
 * First Gui Application using SWING
 */

package GUI;

import java.awt.*;
import javax.swing.*;

/**
 * @version 15/9/11
 * @author EVI
 */
public class First
{
    public static void main(String[] args)
    {
        JFrame frame; // frame , Upeer Container
        frame = new JFrame("First");
        frame.setSize(500,300);
        frame.setVisible(true);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLayout(new FlowLayout());
        // create panel
        JPanel panel = new JPanel();
        panel.setPreferredSize(new Dimension(300,100));
        panel.setBackground(Color.green);
        // create text area, label and textfield
        JTextArea jtArea = new JTextArea("This is a text area", 3, 12);
        jtArea.setFont(new Font("David", Font.BOLD, 14));
        jtArea.setForeground(Color.red);
        JLabel jlabel1 = new JLabel("Label1");
        JTextField jtf = new JTextField(4);
        // add buttons
        JButton jb1 = new JButton("red");
        jb1.setBackground(Color.red);
        JButton jb2 = new JButton("blue");
        jb2.setBackground(Color.blue);
        JButton jb3 = new JButton("yellow");
        jb3.setBackground(Color.yellow);

        //add components to panel
        panel.add(jtArea);
        panel.add(jlabel1);
        panel.add(jtf);
        panel.add(jb1);
        panel.add(jb2);
        panel.add(jb3);
    }
}

```



```
// create another panel
JPanel panel2 = new JPanel();
panel2.setPreferredSize(new Dimension(250, 150));
panel2.setBackground(Color.PINK);
// create more components
String[] strList = { "one", "two", "three", "four" };
JList jList = new JList(strList);
// create scroll bar and add it to list
JScrollPane jScrollPane = new JScrollPane(jList);
// add components to panel2
panel2.add(jScrollPane);
panel2.add(jList);
// add panel to frame
frame.add(panel);
frame.add(panel2);
//frame.pack();
frame.paintAll(frame.getGraphics() );
}
} // end class
```