

## מאגר מעבדות לשפות התכנות החדשות

ניתן להשתמש בחומרים לצורך הוראה בלבד.

**לא ניתן לפרסם את החומרים או לעשות בהם כל שימוש מסחרי**

ללא קבלת אישור מד"ר תמר פז.

המעבדה בקובץ זה מיועדת לתלמידים הלומדים מדעי המחשב בשפת התכנות ג'אווה והיא מותאמת לסביבת אקליפס.

המעבדה מיועדת לשיעורי המעבדה והיא מבוססת על שיטת ההוראה לפיה הלימוד של כל נושא חדש יפתח בהתנסות אישית במעבדה. לאחריה, יבוא דיון כיתתי, שבעקבותיו ייפתרו משימות שונות.

המעבדה מתרכזת בלימוד נושא מרכזי אחד:

### הכרות ראשונית עם מדעי המחשב בסביבת ג'אווה

1 עמוד	הכרות ראשונה עם סביבת ג'אווה
3 עמוד	מבנה של מחלקה בשפת ג'אווה
7 עמוד	הוראות הדפסה
9 עמוד	שימוש בתווי בקרה
11 עמוד	סדר פעולות
14 עמוד	כתיבת מחלקה חדשה

## הכרות ראשונה עם סביבת ג'אווה

כתיבת תכנית בשפת ג'אווה  
ביצוע (הרצה) של תכנית

```
public class FirstProgram
{
    public static void main(String[] args)
    {
        System.out.println ("welcome to computer science");
        System.out.println ("*****");
        System.out.println (" this is our first program ");
    }
}
```

### משימה 1 – חלק א'

המסך שלפניכם מחולק לשלושה חלקים.

החלק העליון הוא העורך ומופיעה בו מחלקה בשפת ג'אווה.

נפתח בהרצת המחלקה (ביצוע התכנית). לשם כך:

• בחלון השמאלי, עימדו עם

העכבר על `FirstProgram.java`.

• לחצו על הלחצן הימני של העכבר.

• בחלון שנפתח, בחרו באפשרות `Run As` (עומדים על האפשרות הנבחרת ולוחצים על הלחצן השמאלי של העכבר).

• בחלון שנפתח כעת, בחרו באפשרות `Java application` (עומדים על האפשרות הנבחרת ולוחצים על הלחצן השמאלי של העכבר).

המחלקה תתבצע, בחלון התחתון (**חלון קלט-פלט**) יופיע הפלט של המחלקה. רישמו כאן את הפלט של המחלקה:

---




---



---

### משימה 1 – חלק ב'


• הריצו את המחלקה פעם נוספת, הפעם לחצו על כפתור  (כפתור חמישי משמאל בשורת הכפתורים בחלק העליון של המסך).

• המחלקה תתבצע בשנית. שימו לב שהפלט של המחלקה לא השתנה.




**הרצה (ביצוע) של מחלקה:****הרצה ראשונה:**

- בחלון השמאלי (Navigator), **עומדים עם העכבר על השם של המחלקה** (במקרה הנוכחי FirstProgram.java) ולוחצים על הלחצן הימני של העכבר.
- בחלון שנפתח, בוחרים בחרו באפשרות Run As (עומדים על האפשרות הנבחרת ולוחצים על הלחצן השמאלי של העכבר).
- בחלון שנפתח כעת, בוחרים באפשרות Java application .



**הרצות נוספות:**

- לאחר שמריצים מחלקה פעם אחת (לפי ההוראות לעיל), אפשר להריץ אותה גם באמצעות לחיצה על כפתור  (כפתור חמישי משמאל בשורת הכפתורים בחלק העליון של המסך).

**משימה 2 – חלק א'**



- טיילו (בעזרת החיצים) עם הסמן (הקו המהבהב) עד לסימן הפותח { הראשון ומחקו אותו.
- לאחר ששינינו את המחלקה, צריך לשמור אותה. לשם כך לחצו בו זמנית על המקשים Ctrl + S. המחלקה נשמרת. אבל משמאל לשורה הראשונה מופיע הסימן  בתוך עיגול אדום. משמעות הסימן היא כי יש בעיה במחלקה. הבעיה היא כי חסר \_\_\_\_\_
- החזירו את הסימן הפותח למקומו.
- שימרו שוב את המחלקה (Ctrl + S) ושימו לב כי הסימן  נעלם.
- הריצו שוב את המחלקה (לחצו על כפתור ) ובידקו כי הפלט לא השתנה.

**משימה 2 – חלק ב'**

- טיילו עם הסמן עד להוראה public static void main(String[] args) ומחקו אותה (לחצו בו זמנית על המקשים Ctrl + x).
- שימרו את המחלקה (Ctrl + S).
- הריצו את המחלקה (לחצו על כפתור ) . מופיעה הודעה **המציינת כי יש בעיה במחלקה!**
- החזירו את ההוראה שמחקתם למקומה (לחצו בו זמנית על המקשים Ctrl + v).
- שימרו שוב את המחלקה (Ctrl + S).
- הריצו שוב את המחלקה (לחצו על כפתור ) ובידקו כי הפלט לא השתנה.

```
public class FirstProgram
{
    public static void main(String[] args)
    {
        System.out.println ("welcome to computer science");
        System.out.println ("*****");
        System.out.println ("  this is our first program  ");
    }
}
```

**משימה 2 – חלק ג'**

- טיילו עם הסמן עד להוראה (`System.out.println ("welcome to computer science");`) ומחקו אותה (`Ctrl + x`).
- שימרו את המחלקה (`Ctrl + S`).
- הפעם לא הופיע הסימן  המציין כי יש בעיה במחלקה!
- הריצו שוב את המחלקה (לחצו על כפתור ) ובידקו כי התקבל פלט תקין.

❖ ללא סימני הסוגריים { } וללא ההוראה `public static void main(String[] args)` המחלקה

❖ `System.out.println` היא הוראת הדפסה (נפתחת ב S גדולה ולאחריה רק אותיות קטנות).

ללא הוראת הדפסה המחלקה

**הערה "טכניות"**


לאחר שמשנים מחלקה, יש לבצע שמירה שלה.

אם מריצים לפני ששומרים, מופיעה שאלה האם לשמור את המחלקה, אפשר לאשר, המחלקה תישמר ולאחר מכן היא תבצע את הנדרש.

**מבנה של מחלקה בשפת ג'אווה****משימה 3**

בחלון השמאלי מופיעים שמות של חמש מחלקות נוספות (`Sod1` , `Sod2` , `Sod3` , `Sod4` , `Sod5`) בכל המחלקות יש שגיאות. עליכם **למצוא** את השגיאות, **לתקן** אותן ו**למלא** את הטבלה. לשם כך, טענו כל אחת מהמחלקות לחוד (הקליקו עם המקש השמאלי של העכבר פעמיים על השם של המחלקה). שימו לב כי ברשימת המחלקות, כל מחלקה מופיעה פעמיים ברציפות: פעם ראשונה עם הסיומת `class` ופעם שניה עם הסיומת `.java`. למשל, `Sod1.class` ולאחריה `Sod1.java`. **עליכם להתייחס רק לשם שמסתיים ב `.java`**.

לאחר ההטענה של כל מחלקה:

- נסו למצוא מה יכולות להיות הסיבות לשגיאות וכתבו אותן במשבצת מספר 1 בטבלה.
- כתבו במשבצת מספר 2 מה צריך לעשות כדי לתקן את התכנית.
- תקנו את השגיאות במחשב.
- שמרו את המחלקה מספר פעמים עד שיעלמו סימני ה  המציינים כי יש בעיה במחלקה.
- הריצו את המחלקה (לחצו ימני-- Run As -- Java application).
- רישמו במשבצת מספר 3 מה הוצג כפלט של ההרצה.

המחלקה	1. הסיבות לשגיאה	2. התיקונים הנדרשים	3. הפלט
Sod1			
Sod2			
Sod3			
Sod4			
Sod5			

```
public class Sod1
(
    public static void main(String[] args)
    {
        System.out.println ("only one kind");
    }
)
```

```
Sod2
{
    public static void main(String[] args)
    {
        System.out.println ("sod2");
    }
}
```

```
public class Sod3
{
    public static void main(String[] args)
        System.out.println ("we learn java");
}
```

```
public class Sod4
{
    public static void main(String[] args)
    {
        System.out.println (".....")
        System.out.println (".....")
    }
}
```

```
public class Sod5;
{
    public static void main(String[] args);
    {
        System.out.println ("we learn java");
    }
}
```

## מבנה חוקי של מחלקה בשפת ג'אווה:

```

public class שם המחלקה           כותרת המחלקה - השורה הפותחת את המחלקה
{
    סימן לפתיחת המחלקה
    public static void main(String[] args)   כותרת הפעולה הראשית לביצוע
    {
        סימן לפתיחת הפעולה הראשית לביצוע
        הוראות לביצוע
    }
    סימן לסגירת הפעולה הראשית לביצוע
}
    סימן לסגירת המחלקה

```

- לאחר כל הוראה לביצוע, רושמים

כשמריצים מחלקה, מה ש"מופעל" זו הפעולה הראשית!

## משימה 4

שנו את המחלקה Sod5 כך שתקבל המחלקה הבאה:

```

public class Sod5
{
    public static void main(String[] args)
    {
    }
}

```

- שמרו, והריצו את המחלקה

**מסקנה** (מחקו את המיותר):

מחלקה תקינה *כן* / *לא* חייבת להכיל הוראות לביצוע.

**משימה 5 – חלק א'**

שנו את המחלקה Sod5 כך שתקבל המחלקה הבאה:

```
public class Sod5
{
    public static void main(String[] args)
    {
        /* this remark
           is only for us */
        System.out.println (" remarks");
    }
}
```

• שמרו, והריצו את המחלקה.

החלק שכתוב בין הסימנים `/*.....*/` מיועד לביצוע! אלה הערות המיועדות למתכנת ולמי שעוקב אחר מהלך הביצוע של המחלקה.

**משימה 5 – חלק ב'**

שנו את הפעולה הראשית במחלקה Sod5 כך שתקבל המחלקה הבאה:

```
public class Sod5
{
    public static void main(String[] args)
    {
        // this remark
        // is only for us
        System.out.println (" remarks"); //other remark;
    }
}
```

• שמרו, והריצו את המחלקה.

**הוספת הערות לתכנית** (מיועדות למתכנת ולמי שעוקב אחר מהלך הביצוע של המחלקה):

אפשרות א: הערה תפתח בסימנים `/*` ותסתיים בסימנים `*/` למשל,

```
/* this remark
   is only for us */
```

במקרה כזה ההערה יכולה להשתרע על פני מספר שורות.

אפשרות ב: הערה תיכתב מנקודה זו ועד לסוף השורה `//.....` למשל,

```
// this remark
```

במקרה כזה ההערה יכולה להשתרע רק על שורה אחת.

## הוראות הדפסה

הוראות שמדפיסות הודעות

### משימה 6 – חלק א'

```
public class Sod5
{
    public static void main(String[] args)
    {
        System.out.println ("we learn java");
    }
}
```

- שנו את המחלקה Sod5 כך שתתקבל המחלקה הבאה. שימרו, הריצו ובידקו שאין שגיאות.
- כעת מחקו את סימני המרכאות " " שימרו את המחלקה.
- מופיע הסימן המעיד כי יש בעיה בתכנית!

הפלט של המחלקה (החלק שיודפס) צריך להופיע בין סימני \_\_\_\_\_

### משימה 6 – חלק ב'

```
public class Sod5
{
    public static void main(String[] args)
    {
        System.out.println (-39);
        System.out.println (2+2);
        System.out.println ("we learn java");
        System.out.println ("8+7");
    }
}
```

- שנו את המחלקה Sod5 כך שתתקבל המחלקה הבאה:
- שימרו, הריצו והשלימו:

באמצעות הוראת ההדפסה System.out.println ניתן להדפיס מספרים, וניתן להדפיס תוצאות של ביטויים חשבוניים. כמו למשל, \_\_\_\_\_

ניתן גם להדפיס תווים. התווים מופיעים בין סימני המרכאות " " ומודפסים כפי שהם. ולכן בהוראת ההדפסה האחרונה מודפסת ההודעה 8+7 ולא המספר \_\_\_\_\_.

התווים המופיעים בין סימני המרכאות הם המחרוזת להדפסה.

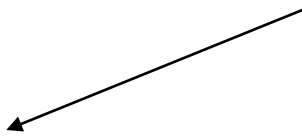


public class **Sod5**

```
{
  public static void main(String[] args)
  {
    System.out.println (-39);
    System.out.println (2+2);
    System.out.println ("w e l e a r n j a v a");
    System.out.println ("8+7");
  }
}
```

**משימה 6 – חלק ג'**

- הוסיפו לפני כל תו בהוראת ההדפסה השלישית מספר סימני רווח.
- שימרו, הריצו והשלימו:



התו רווח הוא תו כמו כל תו אחר! ולכן הוא גם \_\_\_\_\_ כמו כל תו אחר!

**משימה 6 – חלק ד'**

- **חידה:** למה תגרום הוראת ההדפסה `System.out.println ("");` ?

- הוסיפו את ההוראה למחלקה.
- שימרו, והריצו כדי לבדוק את השערתכם.

**משימה 7**

- שנו כעת את כל הוראות ההדפסה כך שתופיע בהן ההוראה `print` ולא ההוראה `println`. למשל כך: `System.out.print (-39);`
- שימרו, הריצו בידקו והשלימו:

כאשר משתמשים בהוראת ההדפסה **print** ההדפסה הבאה תודפס \_\_\_\_\_  
**println** משמעו: הדפס ולאחר מכן, עבור לתחילת השורה הבאה.  
**print** משמעו: הדפס והשאר במקומו.

## שימוש בתווי בקרה

שליטה על העיצוב של ההדפסה

### משימה 8 – חלק א'

```
public class Sod5
{
    public static void main(String[] args)
    {
        System.out.print ("we learn");
        System.out.println ("\n how \n to print ");
    }
}
```

- שנו את המחלקה Sod5 כך שתקבל המחלקה הבאה:
- שימרו את המחלקה.
- הריצו מספר פעמים והשלימו:

הסימן \n הוא תו בקרה ותפקידו: \_\_\_\_\_

### משימה 8 – חלק ב'

```
public class Sod5
{
    public static void main(String[] args)
    {
        System.out.print ("we learn \n");
        System.out.println (" \n how \n to print ");
    }
}
```

- העבירו כעת את תו הבקרה \n מתחילת הוראת ההדפסה השניה לסוף הוראת ההדפסה הראשונה.
- שימרו, הריצו את התכנית מספר פעמים והשלימו:

תו הבקרה \n יכול להופיע בתחילת המחרוזת להדפסה (רצף התווים להדפסה), ויכול גם להופיע באמצע המחרוזת להדפסה או ב \_\_\_\_\_  
התו יגרום להעברת "הראש המדפיס" ל \_\_\_\_\_

### משימה 9

בשפת ג'אווה קיימים תווי בקרה נוספים. אחד מהם הוא תו הבקרה \t. שנו את המחלקה כרצונכם. שימרו, הריצו והשלימו:

תו הבקרה \t גורם ל"הראש המדפיס" ל \_\_\_\_\_

**משימה 10**

- ```
public class Sod5
{
    public static void main(String[] args)
    {
        System.out.println ("we \\ learn");
        System.out.println ("how \" to\n print ");
    }
}
```
- שנו את המחלקה Sod5 כך שתקבל המחלקה הבאה:
  - שימרו את המחלקה.
  - הריצו מספר פעמים והשלימו:
  - שימרו, הריצו את התכנית מספר פעמים והשלימו:

- הסימון \\ בתוך מחרוזת ההדפסה דואג להדפסת התו \_\_\_\_\_
- הסימון \" בתוך מחרוזת ההדפסה דואג להדפסת התו \_\_\_\_\_

**משימה 11**

- שנו את המחלקה Sod5 כך שתציג כפלט עבור שלושה אנשים שונים את הפרטים הבאים:  
*ד"ר מספר אחים      שם פרטי      שם משפחה*
- יש לדאוג להדפסה מסודרת. כלומר, כל השמות הפרטיים יתחילו באותה עמודה, כל שמות המשפחה יתחילו באותה עמודה וכן הלאה. **השתמשו בתו הבקרה \t**
- יש לדאוג לכותרת מתאימה.
- הדפסה לדוגמה:

| First name | second name | no. brothers | age  |
|------------|-------------|--------------|------|
| Gil        | Paz         | 3            | 11.5 |
| Dan        | Daniel      | 3            | 20.3 |
| Yael       | Lev         | 5            | 17.0 |

- שימרו, הריצו ובדקו שהתקבל הפלט הרצוי.

**משימה 12 – חלק א'**

- ```
public class Sod5
{
    public static void main(String[] args)
    {
        System.out.println ("answer=" + 12);
    }
}
```
- שנו את המחלקה Sod5 כך שתקבל המחלקה הבאה.
  - שימרו את המחלקה, הריצו והשלימו:

הסימן + \_\_\_\_\_ (כן / לא) הודפס (אחזקו את המיותר).

**תפקיד הסימן + הוא לחבר (לשרשר) בין מספר הדפסות.**

למשל, משמעות ההוראה System.out.println("answer=" + 12); היא הדפסת רצף התווים

(המחרוזת) " answer=" ולאחריה הדפסת המספר \_\_\_\_\_.

## משימה 12 – חלק ב'

- בידקו האם ניתן בעזרת הסימן + לחבר גם בין 3 הדפסות ויותר.
  - בידקו איך אפשר להדפיס את הסימן + עצמו
- רישמו את מסקנותיכם: \_\_\_\_\_
- \_\_\_\_\_

## סדר פעולות

## משימה 13 – חלק א'

הוסיפו למחלקה את הוראות ההדפסה הבאות:

```
System.out.println (9+8);
System.out.println (" "+9+8);
System.out.println ("1+3"+9+8);
System.out.println (9+8+"1+3");
System.out.println (9+8+" hi "+2+4);
System.out.println (9+8+"1+3"+2+4);
```

- שימרו את המחלקה, הריצו והשלימו:

• משמעות ההוראה `System.out.println (9+8);` היא: הדפסת הערך המספרי המוחזר מהביטוי `9+8` ולכן יודפס \_\_\_\_\_

• כמו במתמטיקה פעולות החשבון מתבצעות לפי הסדר משמאל לימין.

• לאחר שהוראת החיבור + נתקלת במחרוזת, כל הערכים הבאים הופכים גם הם להיות מחרוזות, והוראת החיבור הופכת להיות חיבור (שירשור) של מחרוזת. לכן,

1. משמעות ההוראה `System.out.println ("1+3"+9+8);` היא הדפסת המחרוזת \_\_\_\_\_ (כי ה 9 וה 8 הופכים גם הם להיות מחרוזות ומתבצע שרשור של 3 מחרוזות).

2. אבל, הוראות החיבור שלפני המחרוזת הן הוראות חיבור חשבון רגילות.

לכן, בתגובה להוראה `System.out.println (9+8+"1+3");` יודפס \_\_\_\_\_ (כי קודם מחושבת תוצאת החיבור `9+8` ולאחריה מתבצע שרשור של שתי מחרוזות: המספר 17 והמחרוזת \_\_\_\_\_).

3. ובתגובה להוראה `System.out.println (9+8+"1+3"+2+4);` יודפס \_\_\_\_\_

(כי קודם מחושבת תוצאת החיבור \_\_\_\_\_ ולאחריה מודפסת מחרוזת שהיא שרשור של המחרוזת \_\_\_\_\_).

### משימה 13 – חלק ב'

רשמו ליד כל אחת מהוראות ההדפסה הבאות, מה יודפס כתוצאה מהפעלתה?  
שימו לב, חלק מההוראות שגויות!

`System.out.println (10-2+" hi "+9+9);` \_\_\_\_\_

הסבר שלכם: \_\_\_\_\_

`System.out.println (10-2+" hi "+9-9);` \_\_\_\_\_

הסבר שלכם: \_\_\_\_\_

`System.out.println (10*2+" hi "+9+9);` \_\_\_\_\_

הסבר שלכם: \_\_\_\_\_

`System.out.println (10*2+" hi "+9*9);` \_\_\_\_\_

הסבר שלכם: \_\_\_\_\_

`System.out.println (10*2+" hi "+(9-9));` \_\_\_\_\_

הסבר שלכם: \_\_\_\_\_

• הוסיפו את ההוראות למחלקה, שימרו, הריצו והשלימו

סדר ביצוע ההוראות כמו במתמטיקה:

ראשית מתבצעים סוגריים. אחריו הוראות כפל וחילוק, ואחריו הוראות חיבור ו \_\_\_\_\_  
לכן,

• גם אם יש הוראת כפל ולפניה חיבור למחרוזת, קודם יתבצע הכפל ולאחר מכן התשובה, תהפוך ל \_\_\_\_\_

• אם יש חיבור לפני חיבור, קודם מתבצעת הוראת החיסור.

• אם מנסים לחבר למחרוזת תוצאת חיסור בין מספרים, מופיעה שגיאה כי למחרוזת אפשר רק מחרוזות אחרות (או מספרים שניתן להפוך אותם ל \_\_\_\_\_).

## משימה 14

רשמו ליד כל אחת מהוראות ההדפסה הבאות, מה יודפס כתוצאה מהפעלתה?  
שימו לב, חלק מההוראות שגויות!

`System.out.println (\n+9+9);` \_\_\_\_\_

`System.out.println ("\n"+9+9);` \_\_\_\_\_

`System.out.println ("\n"+(9+9));` \_\_\_\_\_

`System.out.println (9+9+"\n"+9+9);` \_\_\_\_\_

• הוסיפו את ההוראות למחלקה, שימרו, הריצו והשלימו

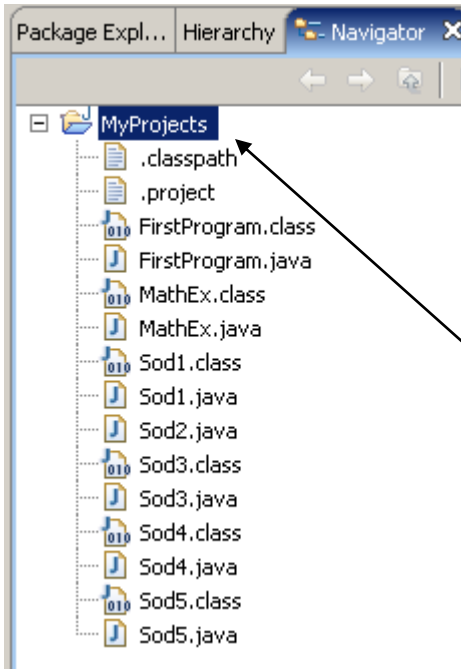
• תווי בקרה אפשר לשלב רק בתוך הדפסה של מחרוזת. לכן, ההוראה: `System.out.println (\n+9+9)` איננה \_\_\_\_\_.

• בהוראה `System.out.println ("\n"+9+9)` מופיעה מחרוזת, לכן הערכים המשורשרים **לאחר** המחרוזת הופכים למחרוזות, ולכן מודפסת המחרוזת \_\_\_\_\_.

• גם בהוראה `System.out.println ("\n"+(9+9))` מופיעה מחרוזת, וגם בה הערכים המשורשרים **אחריה** הופכים למחרוזות, אבל הפעם הודפס \_\_\_\_\_ כי קודם חושב ערך הביטוי שבסוגריים (9+9) ורק לאחר מכן הוא הפך למחרוזת.

• ההוראה `System.out.println (9+9+"\n"+9+9)` גרמה להדפסת שתי שורות: בשורה הראשונה הודפס \_\_\_\_\_ ובשורה השניה הודפס \_\_\_\_\_ כי רק הערכים שלאחר המחרוזת "\n" הפכו למחרוזות.

## כתיבת מחלקה חדשה



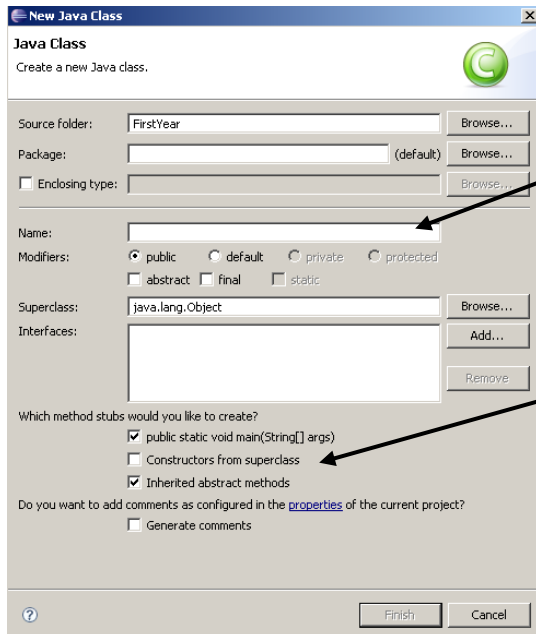
### משימה 15 – חלק א'

בחלון השמאלי של תכנת ג'אווה (חלון הקבצים) מופיעה רשימת הקבצים עימם אנו עובדים. בשלב זה החלון נראה כך:  
 MyProjects הוא פרויקט שמכיל את 6 המחלקות שהרצנו עד כה:  
 FirstProgram , Sod1 , Sod2 , Sod3 , Sod4 , Sod5  
 (הפרויקט מכיל גם מחלקה נוספת אליה נגיע בהמשך).

שימו לב כי שם של מחלקה (class) נפתח באות \_\_\_\_\_  
 (כמו Sod1 וכמו FirstProgram).

נכתוב כעת מחלקה חדשה.  
 לשם כך:

- עימדו עם העכבר על MyProjects , ולחצו על המקש הימני של העכבר.
- בחלון שנפתח בחרו ב- New. בחלון שנפתח עכשיו בחרו ב- Class.



- כעת, נפתח החלון הבא:
- בשורת Name , תנו שם כרצונכם (הקפידו שהאות הראשונה תהייה אות גדולה).
- שימו לב שמופיעים שני סימני  בהתאם לשרטוט הבא (לחצו על ריבוע ריק כדי שיופיע הסימן ).
- לחצו על Finish.

כעת בעורך(החלון העליון), נפתח קובץ חדש שיש בו כבר שש שורות:

```
public class השם שבחרתם ←
{
    public static void main(String[] args) ←
    {
        כאן תכתבו את הפעולה שלכם
    }
}
```

השורה הראשונה היא הכותרת של המחלקה בשורה השנייה מופיעות סוגריים לפתיחת המחלקה השורה השלישית היא פתיחה של הפעולה הראשית. נעסוק במשמעותה בהמשך. בשתי השורות הבאות מופיעות סוגריים שתוחמות את הבלוק בתוכו תכתבו את התכנית שלכם. ובשורה השישית והאחרונה מופיעות סוגריים לסגירת המחלקה.

### משימה 15 – חלק ב'

```
@@@@@
  @@@
   @
  @@@
@@@@@
```

- במחלקה החדשה שהגדרתם, כתבו פעולה שתציג כפלט את "שעון החול" הבא:
- שימרו את המחלקה.
- הריצו ובדקו שהתקבל הפלט הרצוי.