

# פרק 11 – מחלקות ועצמים: הרחבה והעמקה

## הצגת הפרק

### מטרת הפרק

בפרקים הקודמים הצגנו כי בשפת Java ניתן להשתמש בטיפוסים מורכבים הנקראים מחלקות (עצמים). בפרק 9 ראינו שימוש בעצמים מהמחלקה String. בפרק זה נגדיל את אוצר הטיפוסים שעומדים לרשותנו, כלומר נגדיר מחלקות חדשות ונשתמש בעצמים ממחלקות אלו.

### מושגים

- מחלקה
- עצם
- תכונות
- פעולות
- הרשאות גישה
- פרמטרים
- החזרת ערך מפעולה
- פעולה בונה
- פעולות גישה
- יצירת עצם
- שימוש בעצם

## דגשים ודידקטיקה

- כל סעיף בפרק מוסיף ומעמיק את הבנת התלמיד בנושא המחלקות ואופן השימוש בהן. מומלץ ללמד את הפרקים לפי סדרם, ולהדגיש לתלמידים בכל פרק מה התוספת שהם מקבלים מפרק זה.
- יש להדגיש ולחדד את ההבדל בין מחלקה, ומופע של מחלקה (עצם). יש להדגיש כי מחלקה הינה הגדרה כללית של טיפוס נתונים חדש, ואילו עצם הינו המופע של טיפוס זה. מומלץ להדגים יצירה של מספר עצמים מאותה המחלקה, ולהראות כי שינוי של עצם אחד אינו משפיע על עצם אחר.
- מומלץ להציג לתלמידים את המוסכמות לכתיבה (כגון: שם מחלקה יתחיל באות גדולה, שם תכונה יתחיל באות קטנה) ולהדגיש את החשיבות שבשמירה על מוסכמות אלו.
- יש להסביר לתלמידים את החשיבות של הרשאות הגישה בכלל, ואת שמירת תכונות המחלקה כפרטיים בפרט. בכיתות מתקדמות ניתן להסביר את עקרון הסתרת המידע (הכמסה).

## פתרונות

### שאלה 11.1

א. הוסיפו למחלקה Time פעולה בשם increment המקדמת את הזמן בדקה. למשל לאחר הפעולה increment הזמן 10: 52 ישתנה ל- 10: 53, והזמן 23: 59 ישתנה ל- 0: 0.  
ב. כתבו פעולה ראשית הקולטת מהמשתמש נתוני זמן (שעה ודקה) ומספר שלם k. התוכנית תקדם את הזמן ב-k דקות (באמצעות הפעולה שהוגדרה בסעיף הקודם) ותציג את השעה שהתקבלה.

### תשובה 11.1

א. המחלקה Time בתוספת הפעולה increment המקדמת את הזמן בדקה:

```
/* מחלקת זמן */
public class Time
{
    //גדרת התכונות
    private int hour = 0; // שעות
    private double minute = 0; // דקות
    //פעולה לעדכון הזמן
    public void setTime(int h, int m)
    {
        hour = h;
        minute = m;
    }
    //פעולה להחזרת הייצוג הסטנדרטי של הזמן
    public String getTime()
    {
        return hour + ":" + minute;
    }
    public void increment()
    {
        minute++;
        if (minute >= 59)
        {
            minute=0;
            hour++;
            if (hour>=24)
                hour=0;
        }
    }
} // class Time
```

ב. הפעולה הראשית:

```

/* המחלקה הראשית המשתמשת במחלקה זמן */
import java.util.Scanner;
public class UseTime
{
    public static void main(String[] args)
    {
        Time tm;
        int h, m;
        int k;
        Scanner in = new Scanner(System.in);
        tm = new Time();
        System.out.print("Enter the hour (0..23): ");
        h = in.nextInt();
        System.out.print("Enter the minute (0..59): ");
        m = in.nextInt();
        tm.setTime(h, m);
        System.out.print("Enter number of minutes to add: ");
        k = in.nextInt();
        for (int i=0; i<k; i++)
            tm.increment();
        System.out.println("The time is: " + tm.getTime());
    } // main
} // class UseTime

```

### שאלה 11.2

הגדירו וממשו את המחלקות הבאות בשפת Java:

המחלקה	תכונות	פעולות
חתול	- שם - סוג - אוהב או לא אוהב מים - אוהב או לא אוהב לייכל	- פעולה בונה המאתחלת את תכונות העצם. - האם צמא: עבור חתול שאוהב מים הפעולה תחזיר <b>true</b> אחרת יוחזר <b>false</b> . - יללה: עבור חתול שאוהב לייכל הפעולה תחזיר את המחרוזת "מיאו מיאו מיאו". עבור חתול שאינו אוהב לייכל הפעולה תחזיר את המחרוזת "מיאו". - בירור פרטי החתול: הפעולה תחזיר את המחרוזת: שמי <שם החתול> והסוג שלי הוא <סוג החתול>.
ילד	- שם - גיל	- פעולה בונה המאתחלת את תכונות העצם. - בירור פרטים אישיים: הפעולה תחזיר את המחרוזת: שמי <שם הילד> וגילי הוא <גיל הילד>. - יום הולדת: הפעולה תקדם את גיל הילד באחד.
מעגל	- רדיוס	- פעולה בונה המאתחלת את תכונות העצם. - שטח: הפעולה תחזיר את שטח המעגל. - היקף: הפעולה תחזיר את היקף המעגל.

## תשובה 11.2

### מחלקת חתול

```
/* המחלקה חתול */
public class Cat
{
    private String name;
    private String species;
    private boolean likeWater;
    private boolean likeMoan;

    public Cat(String name, String species, boolean likeWater,
               boolean likeMoam)
    {
        this.name = name;
        this.species = species;
        this.likeWater = likeWater;
        this.likeMoan = likeMoam;
    }
    public boolean isThirsty()
    {
        return likeWater;
    }
    public String moan()
    {
        if (likeMoan)
            return "Myau Myau Myau";
        return "Myau";
    }
    public String getDetails()
    {
        return "My name is " + name + " and my species is " +
               species;
    }
}
```

מחלקת ילד

```
/* המחלקה ילד */
public class Child
{
    private String name;
    private int age;

    public Child(String name, int age)
    {
        this.name = name;
        this.age = age;
    }
    public String getDetails()
    {
        return "My name is " + name + " and my age is " + age;
    }
}
```

```

public void birthday()
{
    age++;
}
}

```

מחלקת מעגל. במחלקה זו נשתמש בקבוע  $\text{PI}$  של המחלקה `Math`:

```

/* המחלקה מעגל */
public class Circle
{
    private double radius;

    public Circle(double radius)
    {
        this.radius = radius;
    }
    public double getArea()
    {
        return Math.PI * radius * radius;
        // או return Math.PI * Math.pow(radius, 2)
    }
    public double getPerimeter()
    {
        return 2 * Math.PI * radius;
    }
}

```

### שאלה 11.3

כתבו תוכנית המשתמשת במחלקת החתול שהוגדרה בשאלה הקודמת. התוכנית תקלוט מהמשתמש נתונים עבור חתולים (שם, סוג, האם הוא אוהב מים והאם הוא אוהב ליילל). עבור כל חתול, יוצג כפלט: פרטי החתול (שם וסוג), היללה שלו והאם הוא צמא. התוכנית תסתיים כאשר המשתמש יקליד מחרוזת ריקה עבור שם החתול.

### תשובה 11.3

```

/* המחלקה הראשית המשתמשת במחלקת חתול */
import java.util.Scanner;
public class UseCat
{
    public static void main(String[] args)
    {
        String name;
        String kind;
        boolean likeWater;
        boolean likeMoan;
        Scanner in = new Scanner(System.in);
        System.out.print("Enter the cat name: ");
        name = in.next();
        while (!name.equals(""))
        {
            System.out.print("Enter the cat kind: ");
            kind = in.next();
            System.out.print("Is the cat like water? (Y/N): ");

```

```

        char w = in.next().charAt(0);
        likeWater = w == 'Y';
        System.out.print("Is the cat like to moan?(Y/N):");
        char m = in.next().charAt(0);
        likeMoan = m == 'Y';
        Cat cat = new Cat(name, kind, likeWater, likeMoan);
        System.out.println(cat.getDetails());
        System.out.println("My moan: " + cat.moan());
        System.out.println("Am I thirsty? " +
                            cat.isThirsty());
        System.out.print("Enter the cat name: ");
        name = in.next();
    }
}
}

```

### שאלה 11.5

כתבו מחלקה המגדירה את הטיפוס מעגל. תכונות של עצם מהמחלקה הן צבע המעגל ורדיוס המעגל. הפעולות שניתן לבצע על עצם מהמחלקה הן חישוב שטח, חישוב היקף והחזרת צבע המעגל.

כתבו תוכנית עזר לציור מעגלים. התוכנית תקלוט מהמשתמש את צבע המעגל ואת רדיוסו ותציג כפלט:

שטח המעגל הוא <שטח המעגל> היקף המעגל הוא <היקף המעגל> וצבעו הוא <צבע המעגל> התוכנית תסתיים כאשר הצבע הנקלט יהיה שחור.

### תשובה 11.5

```

/* המחלקה מעגל */
public class ColorCircle
{
    private double radius;
    private String color;

    public ColorCircle(double radius, String color)
    {
        this.radius = radius;
        this.color = color;
    }

    public double getArea()
    {
        return Math.PI * radius * radius;
    }

    public double getPerimeter()
    {
        return 2 * Math.PI * radius;
    }

    public String getColor()
    {
        return color;
    }
}

```

```

    }
}

/* המחלקה הראשית המשתמשת במחלקה מעגל צבע */
import java.util.Scanner;
public class UseColorCircle
{
    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        System.out.print("Enter the circle color: ");
        String color = in.next();
        while (!color.equals("Black"))
        {
            System.out.print("Enter the circle radius: ");
            double radius = in.nextInt();
            ColorCircle cc = new ColorCircle(radius,color);
            System.out.println("The circle area is: " +
                cc.getArea());
            System.out.println("The circle perimeter is: " +
                cc.getPerimeter());
            System.out.println("The circle color is: " +
                cc.getColor());
            System.out.print("Enter the circle color: ");
            color = in.next();
        }
    }
}

```

### שאלה 11.7

במשחק "צבעי קלפים" מקבל כל משתתף 5 קלפים מכל צבע. הצבעים הם כחול, אדום וירוק. במשחק משתתפים 4 שחקנים. כל משתתף בתורו, זורק קלף בצבע כלשהו ולוקח קלף אחר מהקופה. המנצח הוא המשתתף הראשון שצבר 10 קלפים מאותו צבע. פתחו וממשו תוכנית לניהול משחק "צבעי הקלפים". התוכנית תשתמש במחלקה "שחקן". בכל תור התוכנית תפעיל על השחקן התורן את פעולת זריקת קלף, תגריל עבורו את צבע הקלף שיקבל מהקופה, ותעדכן את קלפיו בהתאם. התוכנית תסתיים כאשר אחד השחקנים ינצח.

**הדרכה:** לעצם מטיפוס המחלקה "שחקן" יש שלוש תכונות: עבור כל אחד משלושת הצבעים יש תכונה השומרת את מספר הקלפים שיש לשחקן מצבע זה. לעצם ממחלקה זו יש שתי פעולות: הוספת קלף וזריקת קלף. פעולת הוספת הקלף תקבל כפרמטר את צבע הקלף שיש להוסיף ותעדכן את התכונה המתאימה. פעולת זריקת קלף תחזיר את צבע הקלף שבחר השחקן לזרוק. הפעולה תבחר את הצבע שממנו יש לשחקן הכי פחות קלפים (אם יש יותר מצבע אחד כזה, יוחזר אחד מהם). יש כמובן צורך גם בפעולות גישה, כדי שהתוכנית תוכל לדעת את מצב הקלפים של השחקנים כדי שניתן יהיה לקבוע ניצחון. חשבו: מה צריכה לבצע הפעולה הבונה של עצם ממחלקה זו?

## תשובה 11.7

הגדרת המחלקה שחקן :

עלינו ליישם את המחלקה "שחקן" בדיוק לפי ההדרכה הנתונה. בנאי המחלקה יגדיל את מספר הקלפים הראשוני מכל צבע באופן הבא :  
מספר הקלפים בצבע כחול (blueCards) יוגרל כמספר בין 0 ל-10.  
מספר הקלפים בצבע אדום (redCards) יוגרל כמספר בין 0 ל-(10-blueCards)  
מספר הקלפים בצבע ירוק יהיה 10-(blueCards+redCards)  
בנוסף, נוסיף למחלקה פעולה הבודקת האם מספר הקלפים באחד משלושת הסוגים הוא 10.

יישום המחלקה :

```
import java.util.Random;
/* המחלקה שחקן */
public class Player
{
    private int blueCards;
    private int redCards;
    private int greenCards;
    public Player()
    {
        Random r = new Random();
        blueCards = r.nextInt(11);
        redCards = r.nextInt(11-blueCards);
        greenCards = 10 - (blueCards + redCards);
    }
    public void addCard(int color)
    {
        if (color == 1)
            blueCards++;
        else if (color == 2)
            redCards++;
        else //(color == 3)
            greenCards++;
    }
    public int removeCard()
    {
        if (blueCards<=redCards && blueCards<=greenCards)
        {
            blueCards--;
            return 1;
        }
        else
            if (redCards<=blueCards && redCards<=greenCards)
            {
                redCards--;
                return 2;
            }
        greenCards--;
        return 3;
    }
}
```



```

public int getNumOfBlueCards()
{
    return blueCards;
}
public int getNumOfRedCards()
{
    return redCards;
}
public int getNumOfGreenCards()
{
    return greenCards;
}
public boolean has10SameColor()
{
    return blueCards==10 || redCards==10 || greenCards==10;
}
}

```

הגדרת המחלקה הראשית:

רעיון הפתרון:

התבנית המתאימה לפתרון שאלה זו היא **האם קיים ערך בסדרה המקיים תנאי?** כאשר התנאי הוא האם שחקן ניצח, והסדרה היא נתוני הקלפים של ארבעת משתתפי המשחק. לפיכך, נגדיר ארבעה שחקנים, ונשחק את המשחק עד אשר אחד מהשחקנים ינצח. נשתמש במשתנה מטיפוס שלם אשר יאותחל ל-0 וישתנה למספר השחקן המנצח כאשר אחד מהשחקנים יקיים את התנאי לניצחון.

תבניות:

האם קיים ערך בסדרה המקיים תנאי?

בחירת משתנים:

player1, player2, player3, player4 – ארבעה עצמים מטיפוס המחלקה שחקן.  
winNum – שלם, ישמור את מספר המנצח במשחק.

יישום האלגוריתם:

```

/* המחלקה הראשית המשחקת את המשחק צבעי קלפים */
import java.util.Random;
public class CardColors
{
    public static void main(String[] args)
    {
        Player player1 = new Player();
        Player player2 = new Player();
        Player player3 = new Player();
        Player player4 = new Player();
        int winNum = 0;
        Random r = new Random();
        while (winNum == 0)
        {
            player1.removeCard();
            player1.addCard(r.nextInt(3)+1);
            player2.removeCard();

```

```

        player2.addCard(r.nextInt(3)+1);
        player3.removeCard();
        player3.addCard(r.nextInt(3)+1);
        player4.removeCard();
        player4.addCard(r.nextInt(3)+1);
        if (player1.has10SameColor())
            winNum = 1;
        else if (player2.has10SameColor())
            winNum = 2;
        else if (player3.has10SameColor())
            winNum = 3;
        else if (player4.has10SameColor())
            winNum = 4;
    }
    System.out.println("The winner is: " + winNum);
}
}

```

### שאלה 11.10

בתחרות קפיצה לגובה כל משתתף קופץ 5 קפיצות. כתבו מחלקה המגדירה קופץ לגובה. עצם מהמחלקה ישמור את שמו של הקופץ ואת תוצאות הקפיצות שלו. פתחו וממשו תוכנית לאימון בקפיצה לגובה. התוכנית תקלוט את שמו של הקופץ. לאחר מכן תבקש התוכנית מהקופץ להכניס בכל פעם את תוצאת קפיצתו. (שימו לב שתוצאה של קפיצה לגובה היא מספר ממשי, למשל 1.15 מטר). התוכנית תציג כפלט את שמו של הקופץ יחד עם תוצאת הקפיצה הטובה ביותר שלו.

### תשובה 11.10

```

/* מחלקת קופץ לגובה */
class Jumper
{
    private String name;
    private double[] jumpHeights;
    private final int NUM_OF_JUMPS = 5;

    public Jumper (String name)
    {
        this.name = name;
        jumpHeights = new double[NUM_OF_JUMPS];
        for (int i=0; i<jumpHeights.length; i++)
            jumpHeights[i]=0;
    }
    public String getName()
    {
        return this.name;
    }
    public int getNumOfJumps()
    {
        return this.NUM_OF_JUMPS;
    }
}

```

```

public double getJumpBestHeight ()
{
    double bestJump=jumpHeights[0];
    for (int i=1; i<jumpHeights.length; i++)
        if (jumpHeights[i] > bestJump)
            bestJump = jumpHeights[i];
    return bestJump;
}
public void addJumpHeight(double jumpHeight)
{
    for (int i=0; i<jumpHeights.length; i++)
        if (jumpHeights[i] == 0)
        {
            jumpHeights[i] = jumpHeight;
            return;
        }
}
}

/* לגובה קופץ במחלקת המשתמשת ראשית מחלקה */
import java.util.Scanner;
public class BestJump
{
    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        System.out.print("Enter Jumper Name: ");
        String jumperName = in.next();
        Jumper jumper=new Jumper(jumperName);
        for (int i=0; i<jumper.getNumOfJumps(); i++)
        {
            System.out.print("Enter jump height: ");
            double jumpHeight = in.nextDouble();
            jumper.addJumpHeight(jumpHeight);
        }
        System.out.println("Jumper name: " + jumper.getName());
        System.out.println("Best jump: " +
            jumper.getJumpBestHeight());
    } // main
} // class BestJump

```

#### שאלה 11.14

כתבו תוכנית למציאת הילדים שלהם יש שמות יחודיים בכיתה. התוכנית תקלוט את מספר הילדים בכיתה. לאחר מכן, תקלוט התוכנית עבור כל ילד את שמו הפרטי, את המקצוע האהוב עליו ואת גילו. התוכנית תציג את הנתונים של הילדים אשר שמותיהם ייחודיים בכיתה. ילד הוא בעל שם ייחודי אם בכיתה אין עוד ילד בשם זהה.

#### תשובה 11.14

הגדרת המחלקה תלמיד בכיתה :

הגדרת התכונות :

–name מחרוזת, שם התלמיד  
– favoriteLesson מחרוזת, מקצוע אהוב של התלמיד  
–age שלם, גיל התלמיד

הגדרת הפעולות:

פעולה בונה – פעולה זו מקבלת את שלושת נתוני התלמיד ומעדכנת את המאפיינים  
getName – פעולת גישה לקבלת שם התלמיד  
getDetails – פעולה זו מחזירה את נתוני התלמיד

יישום המחלקה:

```
/* המחלקה תלמיד בכיתה */
public class KidInClass
{
    private String name;
    private String favoriteLesson;
    private int age;

    public KidInClass(String name, String favoriteLesson, int age)
    {
        this.name = name;
        this.favoriteLesson = favoriteLesson;
        this.age = age;
    }
    public String getName()
    {
        return name;
    }
    public String getDetails()
    {
        String st;
        st = "My name is " + name;
        st += " my favorite lesson is " + favoriteLesson;
        st += " my age is " + age;
        return st;
    }
}
```

הגדרת המחלקה הראשית:

רעיון הפתרון:

לפתרון שאלה זו עלינו להשוות כל אחד משמות התלמידים, לשמות שאר תלמידי הכיתה. התבנית המתאימה לכך היא **האם קיים ערך בסדרה המקיים תנאי?** כאשר התנאי הוא האם קיים תלמיד בעל שם זהה לשם התורן אותו אנו בודקים, והסדרה היא שאר תלמידי הכיתה. לפיכך, את תבנית זו נבצע עבור כל אחד מתלמידי הכיתה. יש לתת תשומת לב מיוחדת לאתחול משתנה התבנית לפני תחילת הביצוע עבור כל אחד מתלמידי הכיתה, ולא בתחילת האלגוריתם. המבנה התיכנותי המתאים לביצוע משימה זו הוא לולאה מקוננת. יש לשים לב שלא נשווה שם תלמיד לעצמו.

יישום האלגוריתם:

```
/* המחלקה הראשית המשתמשת במחלקה תלמיד בכיתה */
import java.util.Scanner;
public class UniqueNameInClass
```

```

{
    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        System.out.print("Enter number of kids in class: ");
        int kidsNum = in.nextInt();
        KidInClass[] kids = new KidInClass[kidsNum];
        for (int i=0; i<kids.length; i++)
        {
            System.out.print("Enter kid name: ");
            String name = in.next();
            System.out.print("Enter kid favorite lesson: ");
            String lesson = in.next();
            System.out.print("Enter kid age: ");
            int age = in.nextInt();
            kids[i] = new KidInClass(name, lesson, age);
        }
        for (int i=0; i<kids.length; i++)
        {
            boolean unique = true;
            String name1 = kids[i].getName();
            for (int j=0; j<kids.length && unique; j++)
            {
                String name2 = kids[j].getName();
                if (i!=j && name1.equals(name2))
                    unique = false;
            }
            if (unique)
                System.out.println(kids[i].getDetails());
        }
    }
}

```

## שאלה 11.15

א. נתונה התוכנית הבאה:

```

public class Animal
{
    private String kind = ""; // סוג
    public void setKind (String Kind)
    {
        this.kind = Kind;
    }
    public String getKind ()
    {
        return kind;
    }
} // class Animal

public class MyAnimals
{
    public static void main(String[] args)
    {
        Animal ani = new Animal();
    }
}

```

```

Animal [] myAnimals = new Animal[3];
ani.setKind("dog");
myAnimals[0] = ani;
ani.setKind("cat");
myAnimals[1] = ani;
ani.setKind("fish");
myAnimals[2] = ani;
System.out.println("My animals are "+myAnimals[0].getKind()+
" " + myAnimals[1].getKind()+ " " + myAnimals[2].getKind());
} // main
} // class MyAnimals

```

מה הפלט שיוצג בסוף התוכנית? השלימו :

My animals are \_\_\_\_\_

ב. תקנו את הפעולה הראשית (אך לא את הוראת הפלט) כך שבסוף התוכנית תוצג ההודעה :

My animals are dog cat fish

### תשובה 11.15

א. הפלט שיוצג בסוף התוכנית :

My animals are fish fish fish

ב. הפעולה הראשית כך שבסוף התוכנית תוצג ההודעה החדשה :

```

public class MyAnimals
{
    public static void main(String[] args)
    {
        Animal[] myAnimals = new Animal[3];
        for (int i=0; myAnimals.length; i++)
            myAnimals[i] = new Animal();
        myAnimals[0].setKind("dog");
        myAnimals[1].setKind("cat");
        myAnimals[2].setKind("fish");
        System.out.println("My animals are "+myAnimals[0].getKind()+
" " + myAnimals[1].getKind()+ " " + myAnimals[2].getKind());
    } // main
} // class MyAnimals

```

### שאלה 11.16

נתון קטע התוכנית הבא :

```

int[] intArray = new int[1];
int i = 10;
intArray[0] = i;
System.out.println("intArray[0] is " + intArray[0]);
i = 5;
System.out.println("intArray[0] is " + intArray[0]);

```

כתבו מהו הפלט של קטע התוכנית.

### תשובה 11.16

הפלט של קטע התוכנית

intArray[0] is 10

```
intArray[0] is 10
```

### שאלה 11.17

א. נתונות המחלקות הבאות:

```
public class Doubler
{
    public static void doubleIt(int num)
    {
        num = num * 2;
        System.out.println("Doubled: num = " + num);
    }
} // Doubler

public class DoublerTest
{
    public static void main(String[] args)
    {
        int num = 2;
        System.out.println("num is " + num);
        Doubler.doubleIt(num);
        System.out.println("num after double is " + num);
    } // main
} // DoublerTest
```

מה הפלט שיוצג כתוצאה מביצוע הוראת הפלט האחרונה בתוכנית? השלימו:

num after double is \_\_\_\_\_

זיכרו שפרמטרים ב-Java מועברים על פי ערך (call by value) ולכן ערכו של num לא ישתנה כתוצאה מביצוע הפעולה.

ב. הדרך היחידה שבה הפעולה doubleIt יכולה להכפיל את ערכו של num היא על ידי החזרת הערך num\*2 והשמת הערך המוחזר במשתנה num. שנו את הפעולה doubleIt (ואת אופן הפעלתה), כך שהפעולה הראשית תקבל את ערכו של num מוכפל ב-2. כלומר, כתוצאה מביצוע הוראת הפלט האחרונה בתוכנית תוצג ההודעה:

num after double is 4

### תשובה 11.17

א. הפלט שיוצג כתוצאה מביצוע הוראת הפלט האחרונה בתוכנית:

num after double is 2

ב. הפעולה doubleIt:

```
public static int doubleIt(int num)
{
    num = num * 2;
    System.out.println("Doubled: num = " + num);
    return num;
}
```

הפעלתה:

```
num = Doubler.doubleIt(num);
```

### שאלה 11.19

פתחו וממשו תוכנית למשחק בול פגיעה. התוכנית תגדיר מספר בן 4 ספרות (ללא חזרות) והמשתמש ינסה לנחש את המספר. עבור כל ניחוש התוכנית תציג כמה ספרות הן "בול" וכמה ספרות הן רק "פגיעה".

נגדיר "בול" כשהמשתמש ניחש את הספרה במקומה הנכון ו"פגיעה" כשניחש ספרה המופיעה במספר שהוגרל אך לא במיקומו הנכון.

לדוגמא: המספר שבחרה התוכנית הוא 3456

המספר שניחש המשתמש הוא 2465

לכן הפלט יהיה: בול אחד (הספרה 4) ו-2 פגיעות (הספרות 5 ו 6)

התוכנית תמשיך לקלוט מהמשתמש מספרים עד אשר ינחש את המספר, ורק אז תסתיים.

**הדרכה:** כתבו מחלקה המגדירה את המספר הסודי שרוצים לנחש. הפעולה הבונה תגדיר מספר בן 4 ספרות ותשמור אותו במערך של שלמים בגודל 4 – כל תא במערך ייצג ספרה במספר. הגדירו במחלקה, פעולה המקבלת כפרמטר מספר בן 4 ספרות, בודקת כמה מהספרות הן בול וכמה קליעה ומחזירה מחרוזת עם התוצאה. במקרה שהניחוש נכון תוחזר המחרוזת "finished".

## תשובה 11.19

```
/* מחלקת בול פגיעה */
import java.util.Random;
public class DirectHit
{
    private int[] digits;

    public DirectHit()
    {
        Random rnd = new Random();
        int theNum = rnd.nextInt(9000)+1000;
        digits = setNumInArr(theNum);
    }
    private int[] setNumInArr(int num)
    {
        int[] arr = new int[4];
        for (int i=0; i<arr.length; i++)
        {
            arr[i] = num % 10;
            num /= 10;
        }
        return arr;
    }
    public String takeGuess(int guess)
    {
        if (guess<1000 || guess>9999)
            return "Illegal number";

        int direct=0, hit=0;
        int[] guessArr = setNumInArr(guess);
        for (int i=0; i<digits.length; i++)
            if (digits[i] == guessArr[i])
                direct++;
        if (direct == digits.length)
            return "finished";
        String ret = "Number of directs: " + direct;
    }
}
```



```

        for (int i=0; i<digits.length; i++)
            for (int j=0; j<digits.length; j++)
                if (i!=j && digits[i]==guessArr[j])
                    hit++;
ret += "\nNumber of hits: " + hit;
return ret;
    }
}

/*המחלקה הראשית המשתמשת במחלקה בול פגיעה*/
import java.util.Scanner;
public class PlayDirectHit
{
    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        DirectHit game = new DirectHit();
        System.out.print("Enter your 4 digits number: ");
        int guess = in.nextInt();
        String result = game.takeGuess(guess);
        while (!result.equals("finished"))
        {
            System.out.println(result);
            System.out.print("Enter your 4 digits number: ");
            guess = in.nextInt();
            result = game.takeGuess(guess);
        }
        System.out.println("correct!");
    }
}

```

### שאלה 11.20

בכל בוקר מוציא מר גרבון גרב ממגירת הגרביים ומתקשה למצוא לו בן-זוג תואם. פתחו תוכנית שתעזור למר גרבון למצוא בן-זוג תואם. התוכנית תקלוט ממר גרבון את הגרב (גודל, צבע ודוגמה) ולאחר מכן תקלוט את בן-הזוג שהוא מנסה להתאים לגרב. התוכנית תסתיים כאשר יימצא גרב תואם. הגדירו מחלקה בשם "גרב" בעלת התכונות גודל, צבע ודוגמה. הגדירו פעולה במחלקה שתקבל גרב אחר ותחליט אם הגרביים הם זוג. עליכם להחליט לפי שיקול דעתכם אילו טיפוסים נתונים מתאימים לתכונות גודל, צבע ודוגמה.

### תשובה 11.20

```

/* המחלקה גרב */
public class Sock
{
    private int size;
    private String color;
    private int design;

    public Sock (int size, String color, int design)
    {
        this.size = size;
        this.color = color;
    }
}

```

```

        this.design = design;
    }
    public boolean isEqual(Socket second)
    {
        return (size==second.size && color.equals(second.color)
                && design == second.design);
    }
}
/*התוכנית למציאת בן-זוג של גרבוני*/
import java.util.Scanner;
public class FindSock
{
    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        System.out.print("Enter Garbon size: ");
        int size = in.nextInt();
        System.out.print("Enter Garbon color: ");
        String color = in.next();
        System.out.print("Enter Garbon design (1 2 or 3): ");
        int design = in.nextInt();
        Socket garbon = new Socket(size, color, design);
        System.out.print("Enter other sock size: ");
        int otherSize = in.nextInt();
        System.out.print("Enter other sock color: ");
        String otherColor = in.next();
        System.out.print("Enter other sock design (1 2 or 3): ");
        int otherDesign = in.nextInt();
        Socket other = new Socket(otherSize, otherColor, otherDesign);
        while (!garbon.isEqual(other))
        {
            System.out.print("Enter other sock size: ");
            otherSize = in.nextInt();
            System.out.print("Enter other sock color: ");
            otherColor = in.next();
            System.out.print("Enter other design (1 2 or 3):");
            otherDesign = in.nextInt();
            other = new Socket(otherSize, otherColor,
                                otherDesign);
        }
        System.out.println("Found!");
    }
}

```

---

### שאלה 11.23

נתונה המחלקות Doubler ו-DoublerTest הבאות:

```
public class Doubler
{
    private double num;
    public Doubler(double num)
    {
        this.num = num;
    }
    public double getNum()
    {
        return num;
    }
    public void doubleNum()
    {
        num = num * 2;
        System.out.println("Doubled = " + num);
    }
    public void doubleIt(Doubler it)
    {
        it.doubleNum();
        System.out.println("Doubled: it = " + it.getNum());
    }
} // Doubler

public class DoublerTest
{
    public static void main(String[] args)
    {
        Doubler doubler1 = new Doubler(5);
        Doubler doubler2 = new Doubler(4);
        System.out.println("doubler1 is " + doubler1.getNum());
        doubler1.doubleNum();
        System.out.println("doubler1 after doubleNum is " +
                           doubler1.getNum());
        System.out.println("doubler2 is " + doubler2.getNum());
        doubler1.doubleIt(doubler2);
        System.out.println("doubler2 after doubleIt is " +
                           doubler2.getNum());
    } // Main
} // DoublerTest
```

מה הפלט שיוצג בסוף התוכנית? השלימו:

doubler1 after doubleNum is \_\_\_\_\_  
doubler2 after doubleIt is \_\_\_\_\_

### תשובה 11.23

הפלט שיוצג בסוף התוכנית:

```
doubler1 after doubleNum is 10.0
doubler2 after doubleIt is 8.0
```

### שאלה 11.25

כתבו מחלקה המגדירה ספר טלפונים אלקטרוני (כמו בטלפון נייד). ספר טלפונים מכיל מערך של מחרוזות. כל מחרוזת מורכבת משם וממספר טלפון. הניחו שהמערך ממוין לפי השמות. הגדירו פעולה המקבלת מחרוזת ומחזירה מערך המכיל את כל המחרוזות שמתחילות במחרוזת זו. לדוגמא: נניח שבמערך המחרוזות 3 מחרוזות:

"דונלד 4444"

"מיקי 1234"

"מיני 998877"

כאשר הפרמטר לפעולה יהיה "מיק" תחזיר הפעולה מערך המכיל את המחרוזות:

"מיקי 1234"

כאשר הפרמטר לפעולה יהיה "מ" תחזיר הפעולה מערך המכיל את המחרוזות:

"מיקי 1234"

"מיני 998877"

### תשובה 11.25

```
/* מחלקת ספר טלפונים */
public class PhoneBook
{
    private String[] namesArr;

    public String[] getNames(String search)
    {
        //search how many Strings will be returned
        int count=0;
        for (int i=0; i<namesArr.length; i++)
            if (namesArr[i].indexOf(search)==0)
                count++;
        String[] ret = new String[count];
        for (int i=0, j=0; i<namesArr.length; i++)
        {
            if (namesArr[i].indexOf(search)==0)
            {
                ret[j]=namesArr[i];
                j++;
            }
        }
        return ret;
    }
}
```