

תרגילי חזרה

תרגיל 1 - יחס מורכב

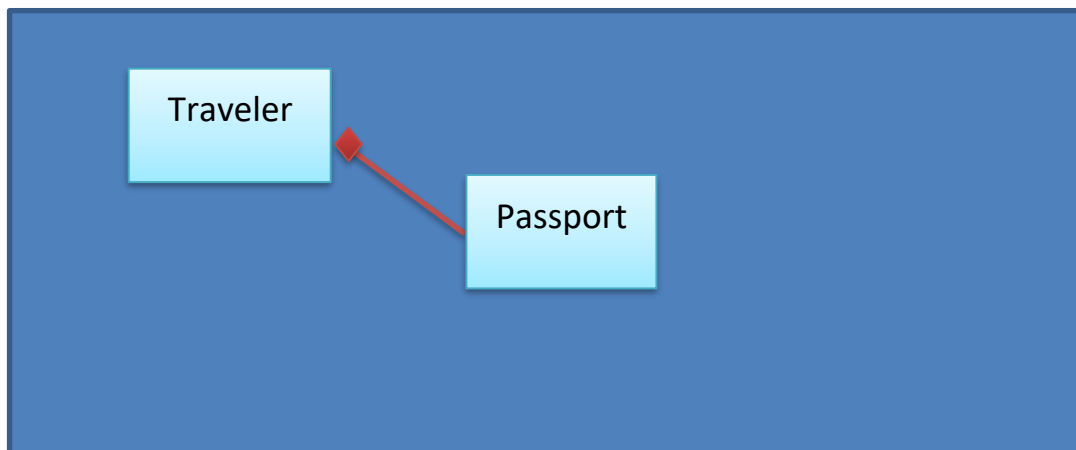
עליכם לממש את המחלקות הבאות:

Passport – מייצגת דרכון של נוסע

Traveler – מייצגת נוסע כללי

כדי לממש את המחלקות השתמשו במחלקה Date, שכתבתם. עליכם להוסיף לה שיטה בונה מעתיקה.

היחסים והקשרים בין המחלקות מתוארות באיור הבא:



מייצג קשר של הרכבה



(a מורכב מ- b, כלומר: לפחות אחת מהתכונות של a היא מטיפוס b)

המחלקה Passport

- מחלקה זו מייצגת דרכון של נוסע. לשם מימושה עליכם להשתמש במחלקה Date.
- עליכם לממש את ממשק המחלקה הבא:

חתימת השיטה	תיאור השיטה
Passport (String name, int number, Date expiryDate)	שיטה בונה היוצרת דרכון חדש ע"פ הנתונים המועברים אליה: שם, מספר דרכון ותאריך תפוגה.
Passport (Passport passport)	שיטה בונה מעתיקה
String toString()	מחזירה מחרוזת המייצגת את הדרכון בפורמט: Name: <name> Pass. Num: <number> Exp Date:<expiry date>
Boolean isValid (Date dateChecked)	מחזירה true אם ורק אם הדרכון בתוקף בתאריך הנתון.
void setExpiryDate (Date newExpiryDate)	מעדכנת את תאריך התפוגה של הדרכון

המחלקה Traveler

מחלקה זו מייצגת נוסע. פרטיו של הנוסע הם הדרכון שלו והמידע האם הנוסע שילם או לא. שימו לב, ע"פ היחסים שהוגדרו המחלקה Traveler מורכבת מהמחלקה Passport.

על המחלקה לממש את הממשק הבא:

תיאור השיטה	חתימת השיטה
שיטה בונה היוצרת נוסע חדש ע"פ הנתונים	Traveler(passport passport, Boolean isPaid)
מאפשרת לנוסע לשלם עבור הנסיעה	void pay()
מחזירה true אם ורק אם הנוסע שילם עבור הנסיעה	boolean isPaid()
מחזירה מחרוזת המייצגת את הנוסע. המחרוזת תהיה זהה לחלוטין למחרוזת המייצגת את דרכון הנוסע.	String toString()
מחזירה true אם ורק אם הנסיעה אפשרית בתאריך הנתון. הנסיעה אפשרית אם דרכון הנוסע תקף בתאריך הנתון, ואם הנוסע שילם	boolean checkTravel(Date travelDate)

בדיקת התוכנית:

כדי לבדוק את התוכנית, כתבו מחלקת בדיקה קצרה שתיצור מספר עצמים מטיפוס נוסע, ותדפיס אותם רק אם הנסיעה שלהם אפשרית בתאריך 1 בינואר שנת 2023.

תרגיל 2

חברה לשיווק צעצועים פיתחה פרויקט לרישום מאגר הצעצועים המשווקים בחברה.

לפניך הגדרה חלקית של המחלקות ב-Java:

צעצוע - Toy, צעצוע לגיל הרך – BabyToy, צעצוע חשיבה – ThinkToy, צעצוע הרכבה – PazelToy

```
public class Toy
{
    protected String nameToy;    // צעצוע שם
    protected String state;      // הייצור ארץ
    protected int age;           // מינימום גיל
    protected double price;      // עלות מחיר
    protected int q;             // במלאי כמות

    public Toy()
    {
    }

    public Toy ( String nameToy, String state, int age, double
                                                price, int q)
    {
    }

    // הצעצוע של המלאי עלות הכול סך את מחזירה הפעולה
    public double total()
    {
        return this.price*this.q;
    }
}
```

```

public class BabyToy extends Toy
{
    private String mat; //הצעצוע עשוי ממנו החומר
}

public class ThinkToy extends Toy
{
    private int part;      // החלקים מספר
    private String[] mat;  // המשחק חלקי עשויים מהם החומרים

    public double total()
    {
        return this.price*this.q + 10;
    }
}

public class PazelToy extends Toy
{
    private int part;      // בפאזל החלקים מספר
    private String mat;    // הפאזל עשוי ממנו החומר
    private String type ;  // קומיקס, צילום, בדיוני מדע - התמונה סוג
אמנות
}

```

א. ממש ב- java פעולה בונה במחלקה BabyToy, שתשתמש בפעולה הבונה המקבלת פרמטרים במחלקה Toy.

ב. ציין איזה מנגנון של תכנות מונחה עצמים בא לידי ביטוי בהגדרת הפעולה total() במחלקה Toy והפעולה total() במחלקה ThinkToy.

ג. האם אפשר לממש את הפעולה total() במחלקה ThinkToy בעזרת הפעולה total() במחלקה Toy ? אם כן – ממש אותה, אם לא- נמק מדוע

ד. המחלקה Store מייצגת חנות מחסן צעצועים המכיל צעצועים מכל הסוגים (BabyToy,

ThinkToy,PazelToy) במחסן יש עד 30 סוגי צעצועים

ממש ב- Java את המחלקה Store מימוש המחלקה צריך לכלול את הסעיפים שלפניך

1. כותרת המחלקה והתכונות שלה

2. פעולה המחזירה את מספר צעצועי התינוקות במחסן

3. פעולה המקבלת צעצוע חדש שהגיע לחברה ומוסיפה אותו למחסן

ה. המחלקה WareHouse, מייצגת 10 מחסנים של צעצועים הממוספרים מ-1 עד 10

ממש ב- Java את המחלקה WareHouse. מימוש המחלקה צריך לכלול את הסעיפים 1-3 שלפניך

1. כותרת המחלקה, והגדרת התכונות שלה

2. פעולה המקבלת מספר מחסן 1-10, ומחזירה כמה צעצועי תינוקות יש במחסן

זה.

3. פעולה המקבלת מספר מחסן 1-10, וצעצוע ומוסיפה את הצעצוע למחסן. הנח

שיש מקום לצעצוע במחסן.

תרגיל 3

קראו בעיון את קוד המחלקות המופיעות בשאלה זו וענו על כל הסעיפים:

- א. שרטטו תרשים UML המתאר את יחסי המחלקות, אין צורך לפרט את איברי המחלקות.
- ב. עבור כל אחת מ 5 השורות הממוספרות בקוד כתבו אם היא:
תגרום לשגיאה או לא, הסבירו במשפט את טענתכם.
- ג. עקבו אחר התוכנית ב CarShop וציינו מהו הפלט ? (הניחו שכל השגיאות מסעיף קודם תוקנו).

```
public class Car {
    private int speed;
    private String manufacturer; // שם היצרן
    private static String color = "WHITE";
    private static final int SPEED_UP=5;

    public Car(String m, int s) {
        this.mauufacturer = m;
        this.speed = s;
    }

    public void changeColor(String color) {
        this.color = color;
    }

    public void speedUp() {
        if (this.speed + SPEED_UP <= 120)
            this.speed += SPEED_UP;
    }

    public String getMauufacturer() {
        return this.mauufacturer;
    }

    public String toString() {
        return "Car " + this.mauufacturer +
            " " + this.color + " " + this.speed;
    }
}
```

```

public class SportCar extends Car{
    private boolean hasTurbo = false;

    public SportCar(String m, int s, boolean t) {
        super(m,s);
        this.hasTurbo = t;
    }

    public void speedUp() {
        super.speedUp();
        super.speedUp();
        super.speedUp();
    }

    public String toString() {
        return "Sport " + super.toString() +
            " Turbo:" + this.hasTurbo;
    }
}

public class TowTruck extends Car{
    private Car carToTow;
    public TowTruck(String m, int s, Car c) {
        super(m,s);
        this.carToTow = c;
    }

    public String toString() {
        return "TowTruck: " + this.getManufacturer() + ...line1
            " " + this.color + " " + this.speed; ...line2
    }
}

public class CarShop {

    public static void main(String[] args) {
        Car[] ourCars = new Car[4];
        ourCars[0] = new Car("Ford",55);
        ourCars[1] = new SportCar("Ferari",80,true);
        ourCars[2] = ourCars[1]; .....line3
        ourCars[3] = new TowTruck("Isuzu",80,null);
        SportCar myCar = ourCars[1]; .....line4
        for (int i = 0; i < ourCars.length; i++) {
            ourCars[i].speedUp();
        }
        ourCars[2].changeColor("BLUE"); .....line5
        for (int i = 0; i < ourCars.length; i++) {
            System.out.println(ourCars[i]);
        }
    }
}

```


תרגיל 4

א. נתונים ממשקי המחלקות הבאים:

המחלקה Point - מגדירה נקודה במישור.

Point (int x, int y)	בונה נקודה במישור
void setY (int y)	מעדכנת את ערך קואורדינטת ה-y להיות הערך המועבר
String toString()	מחזירה מחרוזת המייצגת את ערך ה-x וה-y של הנקודה

המחלקה Vector - מגדירה טיפוס נתונים בגודל משתנה, בו ניתן לאחסן עצמים מטיפוס

Object. שים לב, העצם הראשון מאוחסן במקום **שמספרו 0** בוקטור.

Vector()	בונה וקטור ריק, בגודל 0
void addElement (Object obj)	מוסיפה את האובייקט המועבר, למקום האחרון בוקטור, ומגדילה את גודל הוקטור באחד
Object elementAt (int index)	מחזירה את האובייקט המצוי בוקטור באינדקס הנתון
int size()	מחזירה את מספר העצמים המאוחסנים בוקטור

כתוב תוכנית בשם MyPoints הכוללת שיטת main(...) בלבד, והמבצעת את הפעולות הבאות:

- התוכנית תיצור וקטור המכיל **לפחות** את שלוש הנקודות הבאות:
(1,1), (2,2), (3,3). אין חשיבות לסדר הנקודות בוקטור.
- הנח שהאורך המדויק של הוקטור **אינו ידוע** לך מהשלב הקודם.
התוכנית תדפיס את כל הנקודות הנמצאות בוקטור. **הנקודות יודפסו כך שערך ה-x שלהן ישמר וערך ה-y יתאפס**. (נקודות מהצורה ((x,0).
שים לב: על התוכנית להשתמש בכל השיטות המופיעות בממשקים שלעיל.

ב. סמן את התשובה המציגה את הפלט הנכון של התוכנית Bank. נמק בקצרה את תשובתך.
(תשובה ללא נימוק לא תזכה בנקודות כלל.)

```
public class Bank {
    public static void main(String[] args) {
        AccountA a1 = new AccountA();
        AccountA a2 = new AccountA();
        AccountB b1 = new AccountB();
        AccountB b2 = new AccountB();
        System.out.print(a1.getNum()+" "+a2.getNum()+" "+
            b1.getNum()+" "+b2.getNum());
    }
}
```

```

public class AccountA {
    private int accountNumber = 0;
    private static int currentAccountNumber = 1000;

    public AccountA(){
        this.currentAccountNumber--;
        this.accountNumber = currentAccountNumber;
    }
    public int getNum(){
        return this.accountNumber;
    }
}

public class AccountB {
    private int accountNumber = 0;
    private int currentAccountNumber = 1000;

    public AccountB(){
        this.currentAccountNumber--;
        this.accountNumber = currentAccountNumber;
    }
    public int getNum(){
        return this.accountNumber;
    }
}

```

התשובות האפשריות:

- A. 1000 1000 999 999
- B. 999 998 999 999
- C. 999 998 0 0
- D. התוכנית לא תעבור את שלב ההידור
- E. 999 999 999 998