

תרגול 1 - מחלקות ועצמים - class & object

תרגיל 1 - כתוב מחלקה המייצגת בן אדם בהתאם לתרשים הבא :

class Person

שם (string) מס תעודת זהות (long) גיל (int)	תכונות
פעולה בונה המקבלת את 3 התכונות פעולה בונה ריקה המאתחלת את השם ל John Doe פעולה המקבלת רק שם ות.ז. וקובעת את הגיל ל30	פעולות בונות
פעולות מאחזרות (getters) לכל המשתנים ופעולות קובעות (setters) רק לשם ולגיל	פעולות קובעות ומאחזרות (Getters & Setters)
public void addYear()	פעולה המוסיפה שנה לאדם הנוכחי
public boolean isBigger(Person other)	פעולה המחזירה אמת אם האדם הזה גדול מאדם אחר המועבר כפרמטר ושקר אחרת
public void print()	פעולת הדפסה

במחלקה הראשית הגדר שני בני אדם שונים באמצעות שימוש בשתי פעולות בונות שונות הוסף לאחד מהם שנה ואז הדפס מי משני האנשים הוא גדול יותר באמצעות שימוש בפעולות המתאימות.

תרגיל 2 - הגדר את המחלקה הבאה המייצגת עובד:

class Worker

שם (string) מס שנות נסיון (int) משכורת (int) גיל (int)	תכונות
פעולה בונה המקבלת את כל התכונות	פעולות בונות
לכל המשתנים	פעולות קובעות ומאחזרות (Getters & Setters)
public void addBonus(int amount)	פעולה המוסיפה בONUS לשכר
public void print()	פעולת הדפסה

במחלקה הראשית הגדר מערך של 3 עובדים והדפס את פרטי העובד עם המשכורת הגבוהה ביותר. כעת הוסף בONUS כרצונך לעובד הוותיק ביותר.

תרגיל 3 - הגדר את המחלקה הבאה המייצגת סטודנט:

class Student

תכונות	שם (string) מקצוע לימוד (string) מערך של ציונים (int[]) מספר ציונים מוזנים (int)
פעולות בונות	פעולה בונה המקבלת שם ומקצוע ומאתחלת מערך ריק של 10 ציונים. פעולה בונה (copy constructor) המקבלת סטודנט אחר ומעתיקה את הפרטים שלו כולל הציונים.
פעולות קובעות ומאחזרות (Getters & Setters)	פעולות הקובעות ומאחזרות את שם ומקצוע הלימוד.
פעולה המוסיפה ציון למערך הציונים רק אם יש מקום ומחזירה אמת אם ההוספה הצליחה ושקר אחרת	public boolean addGrade(int grade)
פעולה המחזירה את ממוצע הציונים של התלמיד	public double computeAvg()
פעולה המקבלת סטודנט אחר ומחזירה אמת אם ממוצע הציונים של הסטודנט הנוכחי גבוה משלו ושקר אחרת	public boolean compareAvg(Student other)
פעולה המדפיסה את פרטי הסטודנט	public void print()

במחלקה הראשית צור מערך של שלושה סטודנטים והוסף לכל אחד 10 ציונים רנדומלים באמצעות שימוש בלולאות מקוננות. הדפס את פרטי הסטודנטים השונים ולאחר מכן הדפס את פרטי הסטודנט עם הממוצע הגבוה ביותר באמצעות שימוש בפעולות שיצרתם.

תרגיל 4 - צור את המחלקה הבאה המייצגת שבר:

class Fraction

תכונות	מונה (int) מכנה (int)
פעולות בונות	פעולה בונה המקבלת מונה ומכנה ומייצרת שבר חדש
פעולות קובעות ומאחזרות (Getters & Setters)	פעולות הקובעות ומאחזרות את המונה והמכנה (שים לב שהמכנה חייב להיות שונה מאפס)
פעולה המחזירה את הייצוג העשרוני של השבר	public double realValue()
פעולה המקבלת שבר נוסף ומוסיפה אותו לשבר הנוכחי (שים לב לבצע חיבור שברים נכון בעזרת מכנה משותף).	public void add(Fraction other)
פעולה סטטית המקבלת שני שברים ומחזירה שבר חדש שהוא תוצאת החיבור של שני השברים	public static Fraction add(Fraction f1, Fraction f2)
פעולה המצמצמת את השבר הנוכחי במידה וזה	public void simplify()

	מתאפשר.
public void print()	פעולת הדפסה

במחלקה הראשית צור שני שברים. צמצם אותם הדפס אותם לפני ואחרי הצמצום, הוסף אחד מהם לשני, צמצם את התוצאה והדפס אותה, ואז צור שבר חדש שהוא תוצאת החיבור של שני השברים (לאחר ההוספה), צמצם אותו במידת האפשר והדפס אותו.

תרגיל 5 - תרגול מצביעים - הגדר את המחלקה הבאה המייצגת מזגן

class AirConditioner

תכונות	יצרן (String) טמפר' נוכחית (int) האם דולק (boolean)
פעולות בונות	פעולה בונה ריקה המייצרת מזגן כבוי
פעולות קובעות ומאחרזרות (Getters & Setters)	לכל המשתנים
לחיצה על הפעל / כבה הטמפר' ההתחלתית של המזגן היא 24 מעלות	public void onOff()
פונקציה שמעלה טמפר' במעלה אחת עד לעצמה מקסימלית של 32 מעלות	public void tempUP()
פונקציה שמורידה את הטמפר' במעלה אחת עד לעצמה מינימלית של 16 מעלות	public void tempDown()
פונקציה להדפסת מצב המזגן הנוכחי	public void print()

במחלקה הראשית יש לייצר מזגן אחד לבית ושלושה אנשים שלכל אחד יש שלט משל עצמו. דמה את התרחיש הבא בבית:

1. אבא מגיע הביתה ומדליק את המזגן. הוא מכוון אותו לטמפר' של 22 מעלות. הדפס מה מרגיש יואב היושב בסלון.
2. יואב הצטנן ומעלה את הטמפר' ל 26 מעלות. הדפס מה מרגישה אמא שהרגע נכנסה הביתה.
3. אמא חם לה מאוד והחליטה לשים סוף לסיפור המזגן. הדפס מה מרגיש אבא.

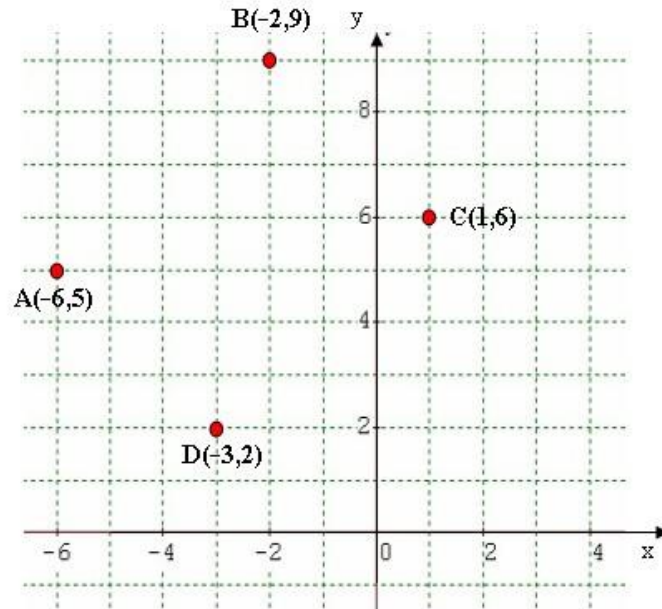
תרגיל 6 - מחלקה בתוך מחלקה - הגדר את המחלקות הבאות המייצגות נקודה וקו ישר

Class Point

x value (int) y value (int)	תכונות
פעולה בונה המקבלת את 2 הארגומנטים פעולה בונה מעתיקה המקבלת נקודה אחרת	פעולות בונות
לכל המשתנים	פעולות קובעות ומאחזרות (Getters & Setters)
public void print()	פונקציית הדפסה

class Line

נקודת מוצא (Point) נקודת סיום (Point)	תכונות
פעולה בונה המקבלת את 2 הארגומנטים פעולה בונה מעתיקה המקבלת קו אחר	פעולות בונות
לכל המשתנים	פעולות קובעות ומאחזרות (Getters & Setters)
public double distance()	פעולה המחשבת את האורך של הקו. לעזרתך נוסחת המרחק: $d = \sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2}$
public double slope()	פעולה המחזירה את השיפוע של הקו. לעזרתך נוסחת השיפוע: $m = \frac{y_2 - y_1}{x_2 - x_1}$



במחלקה הראשית צור את ארבעת הנקודות שבציור הגדר ביניהם מלבן באמצעות שימוש בארבעה קווים ישרים. כעת, באמצעות שימוש בנוסחת המרחק והשיפוע הוכח שזה מלבן. רמז: במלבן האורכים של כל 2 צלעות נגדיות שווים והצלעות מאונכות אחת לשניה (מכפלת השיפועים של כל שתי צלעות מאונכות שווה 1-).

תרגיל 7 - מחלקה בתוך מחלקה - הגדר את המחלקות הבאות המייצגות קרון רכבת ורכבת

class RailroadCar

תכונות	שטח במ"ר (int) מספר נוסעים מקסימלי (int) מספר נוסעים נוכחי (int)
פעולות בונות	פעולה בונה המקבלת שטח ומס' נוסעים מקסימלי ומייצרת קרון ריק פעולה בונה המקבלת את כל שלושת התכונות פעולה בונה המקבלת קרון ומעתיקה את ערכיו
פעולות קובעות ומאחזרות (Getters & Setters)	פעולות הקובעות ומאחזרות לכל התכונות (שים לב שהשטח ומספרי הנוסעים לא יכולים להיות שליליים)
פעולה המקבלת מספר אנשים ובודקת אם יש מקום בקרון	public boolean hasSpace(int amount)
פעולה המקבלת מספר אנשים ומוסיפה אותם לקרון, הפעולה מחזירה אמת אם היה מקום וההוספה הצליחה ושקר אחרת	public boolean addPassengers(int amount)
פונקציית הדפסה	public void print()

class Train

תכונות	מהירות מירבית (int) שם נהג (String) הקרונוט המחברים אליה (RailroadCar[]) מספר הקרונוט המחברים (int)
פעולות בונות	פעולה בונה המקבלת שם נהג ומהירות ויוצרת רכבת ריקה היכולה להכיל עד 10 קרונוט
פעולות קובעות ומאחרזות (Getters & Setters)	פעולות הקובעות ומחזירות את שם הנהג והמהירות.
פעולה המקבלת קרון ומוסיפה אותו לרכבת (בסופה) ומחזירה אמת אם היה מקום ושקר אחרת.	public boolean addCar(RailroadCar car)
פעולה המקבלת מס נוסעים מקסימלי ושטח מייצרת קרון ומוסיפה אותו לרכבת, הפעולה מחזירה אמת אם היה מקום ושקר אחרת.	public boolean addCar(int area, int passengers)
פעולה המחזירה את סה"כ הנוסעים ברכבת	public int numOfPassengers()
פעולה המחזירה את הקרון עם מס' הנוסעים המקסימלי	public RailroadCar maxPassengers()
פעולה המוסיפה מספר אנשים לרכבת ומחזירה את מספר הקרון הראשון אשר מכיל אותם או 0 אם אין מקום	public int addPassengers(int amount)
פעולה המקבלת מספר נוסעים ומחזירה מערך של הקרונוט בהם הם יכולים להיכנס	public RailroadCar[] freeCars(int amount)
פעולת הדפסה לרכבת	public void print()

במחלקה הראשית צור 5 קרונוט עם מספר נוסעים כרצונך (השאר מקום פנוי). צור רכבת וחבר לה את כל הקרונוט והדפס אותה. הדפס גם את מספר הנוסעים הכולל ברכבת. כעת הוסף אנשים כרצונך עד אשר תתמלא הרכבת לחלוטין. הדפס את מצב הרכבת בכל שלב.