**תרגיל 3**

**תאריך הגשה: 23.12.20**

## מטרות התרגיל

* יצירת מחלקות, בנאים ושיטות ושימוש בהם ע"י יצירת אובייקטים והפעלתם
* תרגול הורשה, פולימורפיזם, הכלה והכמסה
* שימוש במחלקות אבסטרקטיות ובממשקים

### תיאור כללי

בתרגיל זה עליכם לתרגל שימוש מעשי בכלים שנלמדו בכיתה (אבסטרקציה, הכלה, הורשה, ממשקים וכו'). הנחיות העבודה מגדירות חלק מהרכב המחלקות, אך הן אינן מגדירות חלק נכבד מהשיטות, השדות והממשקים אותם יש להגדיר ולממש בכל מחלקה. כמו-כן ההנחיות לא מגדיות אילו מחלקות או שיטות צריכות להיות אבסטרקטיות. עליכם להציע פתרון שיתאים לסיפור המעשה עפ"י עקרונות פיתוח תוכנה מונחה עצמים, כפי שהוסברו והודגמו בכיתה.

בתרגיל זה עליכם לפתח חבילת תוכנה המדמה חברה לקורקינטים חשמליים. לחברה יש לקוחות ועובדים מסוגים שונים והיא מאפשרת ללקוחות לקבל תמיכה טכנית עבור קורקינטים חשמליים ולרכוש אותם.

עליכם לממש את המחלקות הבאות:

1. **class ElectricScooter**
2. **class QuickElectricScooter**
3. **class Employee**
4. **class SalesEmployee**
5. **class TechnicalEmployee**
6. **class ServiceCall**
7. **class Customer**
8. **class Company**

### סיפור המעשה

בעקבות הופעתם של הקורקינטים החשמליים בארץ, הוקמה חברה אשר מספקת שירותים לכלל סוגי הקורקינטים כגון מכירה, טיפולים ותיקונים. עליכם לממש מערכת שירות עבור החברה. על מנת להיות לקוח רשום של החברה, על הלקוח להיות בעלים של קורקינט אחד בלבד. לחברה עובדי מכירות ועובדי תמיכה טכנית אשר מטפלים בקורקינטים תקולים.

### האובייקטים במערכת

**עליכם לממש את כל האובייקטים הרשומים בהנחיות, ללא יוצא מן הכלל. מפורטים בסעיפים הבאים 8 אובייקטים. יש לממש את כולם עם שם זהה לשם המופיע כאן באנגלית. ניתן להוסיף אובייקטים נוספים (כולל מחלקות אבסטרקטיות וממשקים), כראות עיניכם.**

1. class ElectricScooter

מחלקה המייצגת קורקינט חשמלי "בסיסי". לחברה יש אפשרות לעדכן את מחירו של כל קורקינט כפי שיוגדר בהמשך. השוואה טבעית בין 2 קורקינטים נעשית על סמך מחיריהם, אשר חייב להיות מספר שלם וחיובי – במידה ולא כך הדבר, יש להגיב באגרסיביות. על המחלקה לכלול את הבנאי הבא:

public ElectricScooter (int serialNumber, int price, String model, int maxSpeed)

בנאי המקבל את המספר הסידורי של הכלי, את מחירו, את המודל שלו ואת המהירות המקסימאלית שלו.

על המחלקה להכיל את השיטה הבאה:

public Boolean update()

שיטה שמעדכנת את מחירו של כלי. אם הכלי הוא מהסוג "הפופולרי" בחברה בזמן העדכון, מחירו יעלה ב- 10% (שימו לב שעל המחיר להישאר שלם לאחר ההעלאה). במידה ולא, לא יבוצע עדכון\*. שימו לב שנתון זה יכול להשתנות עם הצטרפות לקוחות לחברה/קניית קורקינט חדש. אם קיים שוויון בין שני הסוגים, לא יתבצע עדכון. במידה ולא בוצע עדכון, השיטה תחזיר false.

1. class QuickElectricScooter

מחלקה המייצגת קורקינט מהיר וחזק יותר מהקורקינט הבסיסי. קורקינט זה הוא כמו הקורקינט הבסיסי אבל יש לו גם בקר חכם האפשר לבעליו להתחבר אליו באמצעות אפליקציה ולשלוט עליו באמצעות הטלפון. על המחלקה לכלול את הבנאי הבא:

public QuickElectricScooter (int serialNumber, int price, String model, int maxSpeed, double appVersion)

בנאי המקבל את המספר הסידורי של הכלי, את מחירו, את המודל שלו, את המהירות המקסימאלית שלו ואת גרסת האפליקציה אליה יכול להתחבר.

\* סוג הכלי "הפופולרי" בחברה בזמן כלשהו יהיה סוג הקורקינט שלרוב הלקוחות בחברה יש כזה בבעלותם לפי רישומי החברה, ולכן הכלי "הפופולרי" יכול להיות "בסיסי" או "מהיר".

1. class Employee

מחלקה המייצגת עובד בחברה. קיימים שני סוגי עובדים: עובד מכירות ועובד תמיכה טכנית. לכל עובד חדש בחברה, מעניקה החברה מענק ראשוני עבור רכישת ביגוד מתאים. עובד יכול להיקרא על ידי החברה לטיפול בקריאות שירות בהתאם לתפקידו ולצבור עמלות עבור כל קריאה שהוא מטפל בה. לחברה יש אפשרות לעדכן את שכרו של כל עובד בהתאם לסוגו וכפי שיוגדר בהמשך. השוואה טבעית בין שני עובדים נעשית על סמך סכום העמלות המצטבר של כל עובד בחברה. על המחלקה לכלול את הבנאי הבא:

public Employee (int ID, String fName, int age, char gender, char shirtSize)

בנאי המקבל את תעודת הזהות של המשתמש, שמו הפרטי, גילו, המגדר שלו (m עבור גבר, f עבור אישה ו-u עבור אנשים שאינם מגדירים עצמם מגדרית) ומידת החולצה שלו (s/m/l/x). במידה ומגדר העובד אינה בפורמט המתאים או במידה ומידת החולצה אינה בפורמט המתאים, יש להגיב באגרסיביות.

1. class SalesEmployee

מחלקה המייצגת עובד מכירות בחברה. מענק הביגוד של עובד זה מחושב באופן הבא:

|  |  |
| --- | --- |
| מחיר | מידה |
| 10$ | s |
| 15$ | m |
| 20$ | l |
| 25$ | x |

לעובדי מכירות יש עמלת מכירה עבור כל כלי שהם מוכרים. עובד מכירות יקבל בגובה X% ממחיר העסקה שבוצעה, כאשר X הוא גובה העמלה של העובד הספציפי. עובדי מכירות יכולים למכור קורקינטים ללקוחות החברה אך הם אינם מוכשרים לבצע עבודות טכניות על הכלים. על המחלקה לכלול את הבנאי הבא:

public SalesEmployee (int ID, String fName, int age, char gender, char shirtSize, double commRate)

בנאי המקבל את תעודת הזהות של המשתמש, שמו הפרטי, גילו, המגדר שלו (m עבור גבר, f עבור אישה ו-u עבור אנשים שאינם מגדירים עצמם מגדרית), מידת החולצה שלו (s/m/l/x) ואת עמלת המכירה שלו. במידה ומגדר העובד אינה בפורמט המתאים או במידה ומידת החולצה אינה בפורמט המתאים, יש להגיב על כך בחריפות.

על המחלקה להכיל את השיטות הבאות:

public ElectricScooter sellScooter

שיטה המדמה מכירה של קורקינט חשמלי ללקוח על פי התיאור הבא: עובד מכירות יחפש במאגר הקורקינטים של החברה את הכלי הזול ביותר מהסוג המבוקש. השיטה מחזירה את הקורקינט הנבחר לאחר שהתבצע חיוב (ייקרה בקומפני- עדכון אצל לקוח ועדכון ברווח החברה) והכלי הוסר ממאגר הקורקינטים למכירה של החברה.

public boolean update()

שיטה המעדכנת את עמלת המכירה של עובד מכירות. אם סך המכירות של אותו עובד לפחות הוכפלה מהעדכון הקודם, עמלת המכירות שלו תעלה ב-%2, אך בכל מקרה לא תעלה מעל 30). במידה ולא יבוצע עדכון השיטה תחזיר false.

1. class TechnicalEmployee

מחלקה המייצגת עובד תמיכה טכנית בחברה. מענק הביגוד של עובד זה מחושב באופן הבא:

|  |  |
| --- | --- |
| מחיר | מידה |
| 30$ | s |
| 35$ | m |
| 35$ | l |
| 40$ | x |

עובדים אלו יכולים לספק תמיכה טכנית ללקוחות החברה אך אינם יכולים למכור קורקינטים ללקוחות. עבור כל טיפול בקריאה לתמיכה טכנית יקבל העובד מהחברה עמלה בגובה 5$. על המחלקה לכלול את הבנאי הבא:

public TechnicalEmployee (int ID, String fName, int age, char gender, char shirtSize)

על המחלקה להכיל את השיטות הבאות:

public void technicalService (ElectricScooter es)

שיטה המדמה תמיכה טכנית ללקוח עבור הקורקינט שלו, על פי התיאור הבא: עבור קורקינט מתקדם (quick) יודפס למסך משפט פתיחה של העובד (Thank you for coming to us) מודל הקורקינט, מהירותו המקסימאלית ומשפט סיום (Hope not to see you again). עבור קורקינט פשוט, יש להדפיס אך ורק את מודל הקורקינט ואת מהירותו המקסימאלית.

Model :

Max speed :

חיוב לקוח על כל קריאת שירות טכנית יהיה על פי ההסתברות הבאה:

|  |  |
| --- | --- |
| הסתברות | סכום |
| p < 0.3 | 10$ |
| 0.3 <= p < 0.5 | 30$ |
| 0.5 <= p < 0.9 | 80$ |
| 0.9 >= p | 120$ |

כאשר: כלומר משתנה מקרי המתפלג אחיד רציף בין 0 ל-1 (Math.random()).

public boolean update()

שיטה המעדכנת את עמלת השירות של עובד תמיכה טכנית. אם סך העמלות של אותו עובד (סכום העמלות שהעובד צבר מעבודתו ולא סך המכירות/הטיפולים שביצע)הוכפלה לפחות פי 1.5 מהעדכון הקודם, עמלת השירות תעלה ב-2$ עבור כל קריאת שירות. במידה ולא יבוצע עדכון תחזיר השיטה false.

1. class ServiceCall

מחלקה המייצגת קריאת שירות בחברה. כל קריאות השירות מורכבות מלקוח, עובד המשרת אותו, סוג השירות והקורקינט עבורו התקבלה הפנייה. לכל קריאת שירות יש לשמור את הרווח המתקבל ממנה עבור החברה על פי החישוב הבא: הכנסות (תשלום הלקוח) בניכוי הוצאות (עמלת העובד). השוואה טבעית בין 2 קריאת מתבצעת על סמך הרווח המתקבל ממנה עבור החברה. על המחלקה להכיל את הבנאי הבא:

public ServiceCall (Customer c, Employee e, String serviceType, ElectricScooter es)

בנאי המקבל את הלקוח עבורו נוצרה קריאת השירות, העובד המשרת אותו, תוכן השירות ואת הקורקינט עבורו התקבלה הקריאה.

1. class Customer

מחלקה המייצגת לקוח בחברה. כל לקוח שנרשם לחברה, ישלם דמי רישום בסך (קומפני)30$. במידה וללקוח חדש שנרשם לחברה אין קורקינט, עליו לקנות קורקינט מיד עם הירשמו לחברה, דרך עובד מכירות בלבד, אך מובטח להם שיקבלו את הכלי הזול ביותר שקיים במאגרי החברה. לכל לקוח יכול להיות קורקינט אחד בלבד. במידה וברצונם לקנות קורקינט חדש, האחרון שקנה הוא זה שירשם ברישום החברה. במידה והלקוח מגיע עם קורקינט שקנה במקום אחר, יכול לקבל שירות גם על קורקינט זה. השוואה טבעית בין 2 לקוחות נעשית על סמך הרווח (סך התשלומים) שהתקבל עד כה מכל אחד מהם. בנוסף לכך, ניתן יהיה להשוות בין 2 לקוחות על סמך המהירות המקסימאלית של הקורקינטים שלהם. על המחלקה להכיל את הבנאים הבאים:

public Customer (int ID, String fname, int age, char gender, boolean helmet, boolean lock, ElectricScooter scooter)

בנאי המקבל את תעודת הזהות של הלקוח, שם פרטי, גיל הלקוח, האם ברשותו קסדה, האם ברשותו מנעול ואת הקורקינט שברשותו.

public Customer (int ID, String fname, int age, char gender, boolean helmet, boolean lock)

בנאי המיועד ללקוחות שאין בבעלותם עדיין קורקינט בעת הצטרפותם לחברה.

1. class Company

מחלקה המייצגת את החברה עצמה כולל הלקוחות, העובדים, מאגר הקורקינטים למכירה ותיעוד קריאות השירות שנוצרו במסגרת פעילותה. אין לשנות את חתימות השיטות והפונקציות הנתונות אך כמובן ניתן להוסיף שיטות ופונקציות. על המחלקה להכיל את השיטות/הפונקציות הבאות:

public Company (String ES, String customers, String employees, String serviceCalls)

בנאי המקבל כקלט 4 מחרוזות עם נתיבי 4 קבצי הנתונים שקיבלתם כחלק מהנחיות העבודה. בבנאי ייווצרו המופעים השונים בהתאם למבנה שהגדרתם. בעבודה זו תצטרכו לממש קריאה מקבצי הטקסט וליצור על סמך המידע בטקסט את האובייקטים השונים. שימו לב שלאחר יצירת האובייקטים ממחלקת ServiceCall, עליכם יהיה להזין לעובדים הרלוונטיים את המידע (כלומר – לעדכן את הרווח לכל אחד מהעובדים על סמך המידע שהתקבל, אלה קריאות שירות שכבר מומשו) כלל הקבצים נמצאים במודל ובנוסף, ישנו קובץ READ ME המספק הסבר לכל אחד מקבצי הנתונים.

public void addCustomer(Customer c)

שיטה המקבלת לקוח ומוסיפה אותו למאגר הלקוחות של החברה.

public void addElectricScooter(ElectricScooter es)

שיטה המקבלת קורקינט חשמלי למכירה ומוסיפה אותו למאגר הקורקינטים של החברה.

public void addEmployee(Employee e)

שיטה המקבלת עובד ומוסיפה אותו למאגר העובדים של החברה.

public boolean serviceForCustomer(int customerID, String serviceType, boolean quickES)

שיטה היוצרת ומבצעת שירות מבוקש ע"י לקוח לפי התיאור הבא:

* לקוח יזדהה על ידי מספר הזהות שלו
* לקוח שיהיה מעוניין בשירות מכירות יבקש סוג שירות "Sales", ולקוח שיהיה מעוניין בשירות תמיכה טכנית יבקש סוג שירות "Technical"
* לקוח יציין את סוג הקורקינט שברשותו, או לחלופין את סוג הקורקינט אותו הוא מעוניין לקנות (true עבור קורקינט quick)
* בחירת העובד לקריאת השירות תהיה לפי העובד בעל סך העמלות המצטברות הנמוך ביותר עד לקריאת שירות זו, ובתנאי שעובד זה רלוונטי לקריאת שירות זו (כך החברה תוכל לשמור על פערים קטנים יחסית בין העמלות של עובדיה והם לא ירגישו מקופחים)
* יצירת קריאת שירות חדשה:
  + מידה וזוהי קריאת שירות טכנית - יבוצע שירות תמיכה טכנית על ידי העובד שנבחר לטפל בקריאת השירות
  + במידה וזו קריאת מכירה - תבוצע מכירה של קורקינט ללקוח על ידי העובד שנבחר לטפל בקריאת השירות, ויעודכנו פרטי הלקוח במערכת
  + תיעוד קריאת השירות במאגר קריאות השירות של החברה

במקרים הבאים לא תיווצר קריאת שירות חדשה (השיטה תחזיר false):

* במידה והלקוח לא קיים במאגר הלקוחות
* במידה והתבקש שירות שהחברה לא מספקת (כלומר, ערך שונה מ-"Sales" / "Technical ").

public double purchasedQuickScooterRatio()

שיטה אשר מחזירה את אחוז קריאות השירות בהן נקנו קורקינטים מהסוג quick מתוך כלל קריאות השירות בהן נקנו קורקינטים בחברה.

public static double totalRevenues

פונקציה אשר מקבל רשימה/מערך של אובייקטים שניתן להפיק מהם רווח ומחזירה את סכום הרווחים מהאובייקטים.

public static Comparable getMin

פונקציה המקבלת מערך/רשימה של כל אובייקט הניתן להשוואה ומחזירה את האובייקט המינימאלי (למשל: עבור מערך של לקוחות – יוחזר הלקוח בעל הרווח הנמוך ביותר שהפיק לחברה) במקרה ויש יותר מאובייקט אחד בעל ערך המינימום, אין חשיבות לאופן בחירת האובייקט שייבחר, כל עוד הוא שייך לקבוצת האובייקטים המינימליים. (שימו לב שיש לוודא אפשרות של השוואה עבור כל קריטריון המתואר בעבודה).

public static int updatedRates

פונקציה אשר מקבל מערך/רשימה של אובייקטים שניתן לעדכן אותם ומחזירה את מספר האובייקטים שעודכנו.

## דגשים והנחיות נוספות

* **חשוב ביותר**: בעבודה זו ייבדקו עקרונות פיתוח תוכנה מונחה עצמים שנלמדו לאורך הקורס. שימו לב לגבי עקרונות פיתוח תוכנה מונחה עצמים, הכמסה, תכנות TOP DOWN. בנוסף, נסו לחשוב על המערכת במציאות וכיצד הייתה מתנהלת. **פתרונות שלא יעמדו בכך יאבדו נקודות.**
* באפשרותכם להוסיף מחלקות נוספות ככל שיידרש, כל עוד שומרים על עקרונות הקורס.
* **ישנן שיטות ופונקציות סטטיות בתיאור המחלקות בהן לא מצוינים ארגומנטים הנשלחים אליהן. עליכם להבין כיצד לממש את השיטות והפונקציות הללו בעצמכם.**
* **במידה וישנם מקרי קיצון שלא פורטו בעבודה אתם יכולים להניח מה שיהיה לכם נוח, ובתנאי שאין סתירה להנחיות שניתנו.**
* שימו לב, בניגוד לעבודות קודמות, עליכם לבנות **מערכת** שתענה על דרישות מסוימות ולא כוללת קובץ עם פונקציית **main**.
* ההנחיות לא מגדירות מחלקות או שיטות אבסטרקטיות ושימוש בממשקים – עשויים להיות מספר מימושים אפשריים שיענו על הנחיות העבודה. **עליכם להציע את הפתרון המתאים ביותר עפ"י העקרונות שנלמדו בקורס בדגש על עקרונות אלו.**
* עליכם להכין **שרטוט** של תכנון המערכת המתאר את המחלקות והממשקים בעבודה שלכם, תוך כדי תיאור הקשרים ביניהם כפי שנלמד בתרגולים )אין צורך לפרט שדות ושיטות(, נא להגיש שרטוט זה בקובץ **וורד/פאוורפוינט**.
* אין לשנות את החתימות של המחלקות והשיטות שהתבקשתם לכתוב פרט להוספת הממשקים אותם הן מממשות והמחלקות אותן הם יורשות. ניתן להוסיף משתנים נוספים ושיטות/פונקציות נוספות לפי הצורך.
* יש להוסיף תיעוד קוד (הערות) לאורך כל העבודה, בפרט בתחילת כל פונקציה. קוד בלי הערות לא ייבדק.
* בנוסף למצגת ולקבציי הנתונים מצורף לעבודה קובץ הנחיות אודות המחלקה Vector.
* עליכם לוודא שעבור כל קובץ Main שייכתב בהתאם להנחיות ולחתימות שהוגדרו העבודה תרוץ ללא שגיאות.
* עליכם לממש שיטות גישה **רק לשדות שצריך גישה אליהם**.
* אין צורך בבדיקת קלט יש להניח כי הקלט תקין, מלבד המקרים המצוינים במפורש בעבודה.
* הקפידו על כתיבה קריאה וברורה - שמות ברורים למשתנים, הזחות וכו'.

### סדר עבודה מומלץ

1. לקרוא את ההוראות.
2. לקחת את נשימה עמוקה 😊
3. לקרוא את ההוראות שוב.
4. חישבו היטב מהם הקשרים השונים בין האובייקטים.
5. בצעו תכנון מפורט של תהליך הפתרון שלכם. במהלך התכנות זכרו לתכנת Top Down. מומלץ מאוד לא לסיים לתכנת כל דבר עד הסוף אלא במהלך תכנות ה-Top Down להשאיר פונקציות ושיטות שלא מימשתם עד הסוף כדי להשלים קודם כל את המבנה הכללי של התהליך.
6. התחילו בבניית שלד העבודה- ממשו בצורה מלאה כפי שאפשר את אובייקטים 1-7 ברשימה מעלה. המשיכו ליצירת המחלקה Company.
7. סיימו לממש פונקציות מורכבות של כל האובייקטים.
8. בנו Main לבדיקת העבודה עד כה.

### הוראות הגשה ומידע כללי

* הגשה בזוגות. ביחידים – באישור של אחראי הקורס רועי זיוון בלבד.
* **יש להוסיף תיעוד קוד (הערות) לאורך כל העבודה. קוד בלי הערות לא ייבדק.**
* את התרגיל יש להגיש כקובץ זיפ לתיבת ההגשה במודל.
* יש **להגיש את כל תיקיית הפרויקט** (**עם כל המחלקות שנדרשתם לממש)** כאשר שם התיקייה יהיה מספרי תעודת הזהות. לדוגמא: 11111111\_22222222. את תיקייה זו יש להכניס לזיפ.
* דחיות לעבודה יינתנו ע"י אחראי הקורס רועי זיוון בלבד.
* מתרגלת אחראית: נעם גאון
* תתקיים בדיקה פרונטלית על ידי בודקי התרגילים
* כל יום איחור גורר הורדה של 5 נקודות.
* שאלות יש לשאול בפורום בלבד, **ורק בנוגע להנחיות** **ולא לאופן מימוש העבודה שאלות לגבי מימוש העבודה לא יקבלו מענה – אנא כבדו זאת.**