

## SE Assignment 2

### Homework Submission Guidelines

1. **Due date: 20.12.2020 at 23:59**
2. הגשה בזוגות בלבד!
3. הקוד חייב להיכתב בהתאם למוסכמות כתיבת הקוד בקורס כולל תיעוד כנדרש. קוד שלא עומד בדרישות יגרור הורדת ניקוד
4. ההגשה מתבצעת ב-Moodle באזור המיועד על ידי אחד מהשותפים בלבד
5. כל איחור בהגשה יגרור הורדת 20% מהציון בכל יום
6. פורמט הגשת התרגיל נמצא בקובץ ההנחיות ב-Moodle. כל חריגה מפורמט זה תגרור ציון 0
7. שאלות והבהרות ייכתבו רק בפורום ייעודי שייפתח לתרגיל הבית ב-Moodle
8. מגישים קובץ zip אחד שהשם שלו בעל פורמט זהה ל: HW\_2\_firstID\_secondID  
למשל: HW\_2\_316666666\_203455556

### מטרת התרגיל

- עבודה עם מספר מחלקות
- כתיבת מתודות ומשתני מחלקות
- תרגול כימוס, רב צורתיות, ירושה, מחלקות אבסטרקטיות

### נושא התרגיל:

מערכת לניהול לקוחות מול ספקי אינטרנט ותשתיות

### חלק 1:

### תיאור מאגר המידע:

בתרגיל הזה נעשה שימוש במידע על לקוחות במדינת ישראל.  
במדינת ישראל קיימות ספקיות אינטרנט: Cellcom, Benleumi, Partner  
וספקיות תשתית: HOT, Bezeq

לכל חברה יש חיוב חודשי ( base charge ) אחר:

Provider	Base charge
Cellcom	100
Benleumi	150
Partner	90
Bezeq	70
Hot	100

ולא קיים ספק אחר שלא נזכר למעלה. ( שימו לב שמות הספקית case sensitive )

קיים קובץ csv המכיל מידע עבור כל לקוח : תעודת זהות id שזה מספר גדול מאפס, סוג הלקוח ( Regular, VIP, Government ) , ופעולות שהוא ירצה לעשות, למשל, להצטרף ל partner , ולהתנתק מ Hot .

Id של לקוח מסויים יכול להופיע כמה פעמים בקובץ, מותר להניח שה id מופיע תמיד עם אותו טיפוס לקוח

בחלק הזה של התרגיל לקוח שירצה להצטרף או להתנתק מ ספק תשתית או ספק אינטרנט יצטרך לגשת אליהם בעצמו ולבקש את הפעולה הנדרשת.

#### תיאור המערכת:

עם תחילת הריצה של התוכנה, מופעלת מתודת הבנייה (constructor) של המחלקה Main המצפה לקבל את הנתבי של קובץ ה csv .

בשלב הבא התוכנה תאתחל אובייקט של State עם הספקים שיש לה. ואז יהיה ניתן לקרוא מהקובץ ולבצע את הפעולות השונות שכתובות שם.

#### הכנות טרם תחילת התרגיל:

1. פתיחת פרוייקט חדש ב IntelliJ והוספת תקיית csv תחת התיקיה src
2. שמירת קובץ ה csv בתוך הפרויקט ( ליד תיקיית src )
3. הוספת הקבצים שסופקו לכם תחת src\part1

#### הוראות כלליות:

1. יש לקרוא את כלל ההנחיות וההסברים **לאט ומספר פעמים** טרם תחילת העבודה.
2. לצורך פתרון התרגיל מומלץ לחזור על התרגולים וההרצאות וכן להיעזר באינטרנט.
3. יש לבדוק שהקוד מתקמפל ללא שגיאות.

#### תיאור קובץ ה csv

הקובץ מחולק ל 4 עמודות, עמודה ראשונה היא id, שניה היא סוג הלקוח ועמודה שלישית ורביעית מכילות פעולות שהלקוח מבקש. למשל:

1	510	Regular	add Partner	remove infra Bezeq
2	5485	Regular	add Partner	add infra Bezeq
3	626	Government	add Benleumi	add infra Bezeq
4	4140	Regular	add Cellcom	remove infra Bezeq

בשורה הראשונה רואים לקוח בעל תעודת זהות 510, לקוח מסוג רגיל, הוא מבקש להצטרף ל Partner כספקית אינטרנט, להתנתק מספקית תשתית Bezeq. שימו לב להבדל בין פקודה שמדברת על ספק אינטרנט לספק תשתית, כאשר מדובר בספק תשתית יש את המילה infra.

#### ה-API איתו נשתמש בתרגיל

בתרגיל הזה נשתמש ב SEFileUtil ו- CSVParser שאותם כבר ראינו בתרגיל בית קודם

#### דרישת מימוש:

בתרגיל תדרשו לממש את המחלקות הבאות: Main, State, Customer, InfraStructureCustomer, GovernmentCustomer, VIPCustomer, InternetProvider, InfraStructureProvide

**שימו לב:** אולי תצטרכו לממש מחלקות נוספות שלא נאמרו לכם על מנת להשתמש בעקרונות הנדסת התוכנה (קריאות הקוד, טיפול טוב בשגיאות, code reuse ...)

#### מחלקה State

מחלקה זו מכילה השדות הבאים: שם המדינה (String), רשימה של ספקי אינטרנט, רשימה של ספקי תשתית, רשימה של סוגי לקוחות אפשריים. כל השדות הינם private \* שימו לב ייתכן ותצטרכו להוסיף שדות לייעול המימוש \* שימו לב ייתכן ותצטרכו להוסיף מתודות לייעול המימוש מתודות:

- בנאי: אשר מקבל כפרמטר שם המדינה ומאתחל שדות בהתאם.
- מתודת setInfraStructureProviders(List<InfraStructureProvider> lst) אשר מקבלת כפרמטר רשימה של ספקי תשתית ומוסיפה אותם לרשימת ספקי התשתית של המחלקה
- מתודת setInternetProviders(List<InternetProvider> lst) אשר מקבלת כפרמטר רשימה של ספקי אינטרנט ומוסיפה אותם לרשימת ספקי האינטרנט של המחלקה

- מתודת `InternetProvider getInternetProvider(String name)` אשר מקבלת כפרמטר שם של ספק אינטרנט `String` ומחזירה את אובייקט הספק הרוולנטי. בהנחה ואין ספק בעל שם זה, המתודה תחזיר `null`
- מתודת `InfraStructureProvider getInfraStructureProvider(String name)` אשר מקבלת כפרמטר שם של ספק תשתית `String` ומחזירה את אובייקט הספק הרוולנטי. בהנחה ואין ספק בעל שם זה, המתודה תחזיר `null`.

### מחלקת `InternetProvider`

- מחלקה זו מכילה בשדות הבאים: שם ספק (`String`), מחיר בסיס (`baseCharge (double)`), ורשימה של לקוחות. כל השדות `private`
- \*שימו לב ייתכן ותצטרכו להוסיף מתודות לייעול המימוש מתודות:
- בנאי: אשר מקבל כפרמטר שם הספק ומחיר בסיס ומאתחל אובייקט בהתאם.
- `String getName()` אשר מחזירה את השם של הספק
- `addCustomer(Customer customer)`, מקבלת כפרמטר לקוח מסוג `Customer` ומוסיפה אותו לרשימת הלקוחות שלה במידה ואינו קיים. אם הלקוח כבר מחובר לספק הזה אזי תדפיס הודעה בפורמט:

Customer "ID" already connected to "Provider name"

כאשר יש להחליף את המילים בין ה " " בערכים המתאימים

- `removeCustomer(Customer customer)`, מקבלת כפרמטר לקוח מסוג `Customer` ומוחקת אותו מהרשימה במידה והוא קיים. אם הוא לא קיים המתודה תזרוק `Exception` מסוג `CustomerNotListedException` שמספרת מה השתבש. הפורמט של הודעת השגיאה הוא:

Customer "customerId" is not listed in "provider name" customers list

כאשר יש להחליף את המילים שבין ה " " בערכים המתאימים.

על המתודה שקוראת ל `removeCustomer` לטפל ב `Exception` ע"י הדפסת הודעת השגיאה

### מחלקת `InfrastructureProvider`

- מחלקה זו מכילה בשדות הבאים: שם ספק (`String`), מחיר בסיס (`baseCharge (double)`), ורשימה של לקוחות. כל השדות `private`
- \*שימו לב ייתכן ותצטרכו להוסיף מתודות לייעול המימוש מתודות:
- בנאי: אשר מקבל כפרמטר שם הספק ומחיר בסיס ומאתחל אובייקט בהתאם.
- `String getName()` אשר מחזירה את השם של הספק

- `void addCustomer(Customer customer)` , מקבלת כפרמטר לקוח מסוג `Customer` ומוסיפה אותו לרשימת הלקוחות שלה במידה ואינו קיים. אם הלקוח כבר מחובר לספק הזה אזי תדפיס הודעה בפורמט :

Customer "ID" already connected to "Provider name"

כאשר יש להחליף את המילים בין ה " " בערכים המתאימים

- `void removeCustomer(Customer customer)` , מקבלת כפרמטר לקוח מסוג `Customer` ומוחקת אותו מהרשימה במידה והוא קיים. אם הוא לא קיים המתודה תזרוק `Exception` מסוג `CustomerNotListedException` שמספרת מה השתבש. הפורמט של הודעת השגיאה הוא :

Customer "customerId" is not listed in "provider name" customers list

כאשר יש להחליף את המילים שבין ה " " בערכים המתאימים.

על המתודה שקוראת ל `removeCustomer` לטפל ב `Exception` ע"י הדפסת הודעת השגיאה

### מחלקות `Customer`, `GovernmentCustomer`, `RegularCustomer`, `VIPCustomer`

מחלקת `Customer` היא מחלקה אבסטרקטית אשר ממנה יורשים שלושת המחלקות של סוגי הלקוחות, עליכם לבד להחליט מה נכנס לתוך המחלקה הזו (שדות + מימוש) ומה צריך להכניס למחלקות הבנים. לכן בחלק הזה יוסבר איזה שדות ופונקציונליות צריכים להתקיים בין הקלאסים ביחד ואתם תחליטו איך לחלק אותם.

שדות: `id` (int) , רשימה של ספקי תשתית (`ArrayList<InfrastructureProvider>`) , רשימה של ספקי אינטרנט (`ArrayList<InternetProvider>`) .

\*שימו לב: ייתכן ותצטרכו להוסיף מתודות לייעול וקריאות הקוד מתודות :

- בנאי: אשר מקבל `id` (int) ומאתחל אובייקט בהתאם

- `int getId()`: אשר מחזיר את ה `id` של הלקוח

- `void handleInternetProvider(String [] command, InternetProvider provider)` :

מתודה זו מקבלת כפרמטר פקודה של לקוח (נקרא לו `command`) , אשר מבקש להצטרף או להתנתק מספק מסוים לפי הפורמט שהוסבר קודם בקובץ. הפקודה יכולה להיות מסוג `String` או מערך של `String` (איך שנוח לכם) ובנוסף מקבלת כפרמטר ספק מסוג `InternetProvider` , המתודה תפענח את הטקסט ותפעל בהתאם. דוגמאות :

נניח שלקוח 1 ביקש `add Partner` אזי על המתודה להבין שצריך להוסיף את הלקוח 1 למאגר

הלקוחות של `Partner` ולהוסיף `Partner` ברשימת הספקים המתאימה של הלקוח.

לחלופין אם לקוח 1 ביקש `remove Partner` אזי על המתודה לפענח את הטקסט ולהבין שצריך

למחוק את הלקוח ממאגר הלקוחות של `Partner` ושצריך למחוק את `Partner` מרשימת

הספקים של הלקוח.

**\*שימו לב :** ייתכן שהלקוח יבקש להתחבר לספק שהוא כבר מחובר אליו, יש להודיע לו על כך עם

הודעה מתאימה, למשל: Customer 1 already connected to provider Partner

ייתכן גם שיבקש להתנתק מספק שאינו מחובר אליו, ואז יש להודיע לו על כך בהתאם, למשל:

Customer 1 does not have Partner as Internet Provider

ייתכן גם שתקבלו פקודה שגויה, כלומר פקודה שהיא לא add או remove ואז צריך לזרוק

שגיאה IllegalOperationRequest, הודעת השגיאה תהיה:

Only add or remove requests

- void handleInfrastructureProvider(String [] command, InfrastructureProvider

provider): מתודה זו מקבלת כפרמטר פקודה של לקוח (נקרא לו command), אשר מבקש

להצטרף או להתנתק מספק מסוים לפי הפורמט שהוסבר קודם בקובץ. הפקודה יכולה להיות

מסוג String או מערך של String (איך שנוח לכם) ובנוסף מקבלת כפרמטר ספק מסוג

InfrastructureProvider, המתודה תפענח את הטקסט ותפעל בהתאם (לפי אותו היגיון של

מתודה קודמת וגם היא זורקת IllegalOperationRequest)

- double calculateMonthlyPayment(): מתודה זו צריכה לחשב כמה הלקוח הזה אמור

לשלם בסוף כל חודש, נזכר כי כל ספק מוגדר לו baseCharge.

לקוח שהוא ממשלה: הממשלה מחייבת כל חברה שהיא ספקית אינטרנט לשלם באופן חודשי

לממשלה 150 ₪, ומכל חברה שהיא ספקית תשתית לשלם באופן חודשי 200 ₪.

לכן לקוח ממשלה מקזז מהחייבים שהחברות חייבות אליו כמה הוא חייב אליהם, ומה שנשאר

משלם. המתודה תחזיר כמה הלקוח הזה חייב במצטבר לכל הספקים.

לקוח שהוא לקוח רגיל: לקוח רגיל ישלם לכל ספק את ה baseCharge פלוס 20%.

המתודה תחזיר כמה הלקוח הזה חייב במצטבר לכל הספקים.

לקוח שהוא לקוח VIP: לקוח VIP ישלם לכל ספק מספר אקראי בין 0 ל baseCharge של

אותו ספק. המתודה תחזיר כמה הלקוח הזה חייב במצטבר לכל הספקים.

## מחלקת Main

זאת המחלקה הראשית שרצה ראשונה בתוכנה שלנו, המחלקה הזו מצפה לנתיב קובץ ה csv בשביל

שתרוץ. שם בקובץ מועבר כפרמטר הראשון בשורת הפקודה.

למחלקה יש שדה HashMap<Integer, Customer> customers ופונקציית main מוכנה.

- ממשו פונקציית void parseData(String path, State state) אשר מקבלת כפרמטרים את

ה path של ה csv ואובייקט State,

הפונקציה קוראת נתונים מהקובץ ומתחילה לעשות את הפעולות הנדרשות ממנה לפי הקובץ עבור

כל לקוח. כלומר היא קוראת שורה שורה, עבור כל שורה היא צריכה לפענח באיזה לקוח

מדובר(לקוח קיים או חדש), מהו סוג הלקוח ומהן הפעולות הנדרשות ללקוח זה, כך למשל אם

ללקוח 1 היה:

add Partner , remove infra Hot

אזי מתודה זו צריכה להבין כל פקודה בשורה הזו ולבצע אותה,

למשל עבור פקודה ראשונה צריך להבין שמדובר בספק אינטרנט ולא ספק תשתית, ולחבר את הלקוח לספק במידה ואינו מחובר.

\*שימו לב: ייתכן והספק שרשום בפקודה אינו ספק חוקי למשל add blabla אזי המתודה צריכה להבחין שאין ספק בשם הזה, נזכר שרק ה state יודע איזה ספקים הם חוקיים ואיזה לא.

\*שימו לב: ייתכן והפקודה תהיה בפורמט אחר לגמרי, כלומר לא add nameOfProvider או add infra NameOfProvider וכנ"ל ל remove.

במידה ואחד מהמקרים האלה קורה צריך לזרוק שגיאה של IllegalOperationRequest הודעת השגיאה למקרה הראשון תהיה:

"provider name given in command" is not an internet/ infrastructure provider

ההודעה תכיל או Internet או infrastructure בהתאם למקרה

הודעת השגיאה למקרה השני תהיה:

Command not legal

- שימו לב: עבור כל מתודה שזורקת, exception המתודה שקראה לה תטפל ב exception ע"י הדפסת הודעת השגיאה.
- שימו לב: את מחלקות ה exceptions יש לשים תחת package : src/exceptions

## חלק 2:

הוחלט על חוק חדש בחסות משרד התקשורת, החוק אומר שלקוח לא יכול לדבר עם ספקי התשתית אף פעם, אלא הוא רק יכול לתקשר עם ספקי האינטרנט.

כלומר נניח ולקוח מסוים רוצה להתחבר לספק תשתית X או להתנתק מספק תשתית Y (X ו-Y הם רק לדוגמה ואינם שמות ספקי תשתית חוקיים) אזי הלקוח יצטרך רק לפנות לאחד מספקי האינטרנט שלו (לא משנה איזה מהם) ויבקש מהם והם יטפלו לו בנושא.  
אם אין ללקוח ספק אינטרנט אזי צריך להודיע על כך, למשל:

Customer 1 does not have Internet Provider to handle Infrastructure

דרישות מימוש:

פתחו package חדש תחת src תקראו לו part2 וממשו את אותן מחלקות עם השינויים המתאימים לחוק החדש.

20 שורות ראשונות של הפלט עבור הדוגמה שסופקה לכם:

Customer 510 does not have Bezeq as Infra provider

Customer 510 is not listed inBezeq customers list

Customer 4140 does not have Bezeq as Infra provider

Customer 4140 is not listed inBezeq customers list

Customer 9 does not have HOT as Infra provider

Customer 9 is not listed inHOT customers list  
Customer 2496 does not have HOT as Infra provider  
Customer 2496 is not listed inHOT customers list  
Customer 5176 does not have HOT as Infra provider  
Customer 5176 is not listed inHOT customers list  
Customer 4420 does not have Benleumi as Internet provider  
Customer 4420 is not listed inBenleumi customers list  
Customer 4420 does not have HOT as Infra provider  
Customer 4420 is not listed inHOT customers list  
Customer 4282 does not have Partner as Internet provider  
Customer 4282 is not listed inPartner customers list  
Customer 4282 does not have Bezeq as Infra provider  
Customer 4282 is not listed inBezeq customers list  
Customer 5273 does not have Partner as Internet provider  
Customer 5273 is not listed inPartner customers list

אתם מתבקשים לחשוב על עוד מקרים ודוגמאות חוץ מהקובץ שסופק לכם ולבדוק את התוכנית שלכם.

מגישים קובץ זיפ אחד שמכיל את תיקיית ה: src

Contents of the Zip file ( eVeryTHINg is case sensitive):

src

part1

part2

exceptions

csv

NOTE: The checking program will try to run src.part1.Main.main() so you MUST comply to this requirement. (same goes for part2)