# Exercise 1

Add a README.md file to your repo with your names and (edu.hac.ac.il) emails.

You must submit an IntelliJ project ready to compile and run (right click -> run on the main class should work). שימו לב להערות בסוף שהוספתי כתשובה לשאלות שקיבלתי

* **נביאים**:
* <https://classroom.github.com/a/tvA-HQ9N>

תאריך הגשה: 8/4/2024 23:59

* **שטראוס נשים**:
* <https://classroom.github.com/a/EdSrtTnd>

תאריך הגשה: 11/4/2024 23:59

* **שטראוס גברים**:
* <https://classroom.github.com/a/yslLnU58>

תאריך הגשה: 8/4/2024 23:59

המטרה: כתיבת תוכנית שקוראת פקודות מקלט סטנדרטי ומדפיסה לפלט סטנדרטי. המטרה היא תרגול ראשוני עם Java OOP : מחלקות, ממשקים, ירושה, חריגות.

יש להשתמש בתבנית Composite ולהגדיר ממשק / interface (לא מחלקה אבסטרקטית) למחלקה הבסיסית. המטרה של תבנית Composite הינה הרכבת אובייקטים לאובייקט מורכב יותר כך שהאובייקט המורכב והאובייקטים המרכיבים מצייתים לאותו ממשק. התבנית מאפשרת ללקוח להתייחס לאובייקטים אינדיבידואלים ולהרכבות של אובייקטים בצורה אחידה.

קראו על תבנית זו. קישור: <https://sourcemaking.com/design_patterns/composite>

התוכנית מאפשרת בניית מבנה מערכת קבצים שבו ניתן לחבר קבצים וספריות (component / רכיב) להיררכיה דמוית עץ. מחלקת הבסיס לרכיב אמורה לאפשר הוספה והדפסה של נתיב הקובץ. קובץ הוא "עלה" (לא יכול להכיל כלום) וספרייה היא "מורכבת" (מכילה ספריות או קבצים). עלים (קבצים) מזוהים על ידי סיומת (תבנית שם הקובץ היא \*.\*)

אנו מציעים להשתמש במחלקת Java ArrayList כדי לאחסן רשימות של רכיבים ברכיב הספרייה.

התוכנית מבוססת על שורת פקודה. המשתמש מקליד רשימה של נתיבים מלאים, והקבצים או הספריות מתווספים למבנה. אם נתיב מכיל ספרייה לא קיימת, יש ליצור אותה באופן אוטומטי. הפקודה "exit" מעבדת את כל הנתיבים ומדפיסה את מבנה הספריות. יש לאתחל את המבנה עם ספריית שורש "/root".

הערות נוספות בתשובה לשאלות שקיבלתי:

* המשתמש מכניס את הקלט המלא קודם לכן, ורק בסיומו מקבל פלט
* אין לשבור את הפולימורפיזם של התבנית (אין שימוש ב-casting או instanceof )
* אם מכניסים שורה ריקה/רווחים, מתעלמים משורה זו
* אם מכניסים קלט שכבר הוכנס מתעלמים ממנו או רושמים הודעה שכבר קיים
* הדפסת השגיאות מתבצעת תוך כדי בנייה
* ההדפסת המבנה הינה טיול בעץ הבנוי (אחרי עיבוד של כל השורות הקלט)

Sample command line execution:

Input:

Enter file/directory paths (or 'exit' to quit):

/home/user/documents

/home/user/documents/file1.txt

/home/user/documents/folder1

/home/user/documents/folder1/file2.txt

/home/user/pictures

abc/def

exit

Output:

Directory: /root

Directory: /root/home

Directory: /root/home/user

Directory: /root/home/user/documents

File: /root/home/user/documents/file1.txt

Directory: /root/home/user/documents/folder1

File: /root/home/user/documents/folder1/file2.txt

Directory: /root/home/user/pictures

Directory: /root/abc

Directory: /root/abc/def

Another example:

Input:

Enter file/directory paths (or 'exit' to quit):

/home/user

/home/user/documents

/home/user/documents/file1.txt

/home/user/documents/file1.txt/invalid

/home/user/pictures

/home/user/pictures/photo.jpg

/home/user/pictures/photo.jpg/album

exit

Output:

Cannot add /root/home/user/documents/file1.txt/invalid to /root/home/user/documents/file1.txt

Cannot add /root/home/user/pictures/photo.jpg/album to /root/home/user/pictures/photo.jpg

Directory: /root

Directory: /root/home

Directory: /root/home/user

Directory: /root/home/user/documents

File: /root/home/user/documents/file1.txt

Directory: /root/home/user/pictures

File: /root/home/user/pictures/photo.jpg