

מטלה 1 :

הערה: לקרוא עד הסוף מסמך זה

מגשים:

רואי סדומי -305170490.

עדיאל יצחק-313485286.

הסבר איך להריץ:

פתח פרויקט באקליפס לא משנה מה השם.

גרור את תוכן כל ה git לשם ותריץ.

הסבר על המטלה:

על מנת להריץ לך לקובץ הג'אווה בשם TEST ותריץ משם. והמתן על להדפסה של FINISHED!

מה התוכנה עושה?

בתור התחלה היא תקח את כל הקבצי CSV בתקיית CSV ותהפוך אותם לקובץ אחד בשם TEST1.(איתו היא תתחיל לעבוד.

לאחר מכן היא שואלת עם איזה סוג מיון תרצה לעבוד.

במידה ובחרת מיון לפי ID לדוגמא היא תכין תקובץ בשם ORGNIZED שממיון

אך ורק לפי סוגי ID שונים , בכל שורה יהיו עד 10 הכי חזקות (כל כתובת מאק תופיע פעם אחת בלבד בעוצמתה הכי חזקה)

משם הפילטר יכנס לפעולה ויעבור על הקובץ ויוציא משם אך את הסינון הרצוי.

במידה ובחרת לדוגמא את המילה Bezeq , הפילטר יעשה בשיטת CONTAIN ויציג לך כל רשת שמכילה את הID Bezeq.

התוכנה יוצרת קובץ CSV במיוחד שנקרא FILTERD שבו ניתן לפקח ולראות שקיבלנו את תוצאות הסינון הרצוי.(יהיה לך נוח לבדוק)

משם הKML נכנס לפעולה וממיר את כל מה שבקובץ FILTERD .ל.KML.

נעזרנו בספריית JAX.(הסיבה:מטעמי נוחות ופשטות ההתקנה)

ולכן על מנת להריץ ללא תקלות תדאג שהיא תהיה מותקנת לך באקליפס.

התוצאות : מסמך KML (בשם:googleearth-ver2) אשר מציג את התוצאה הרצויה וגם כולל כמובן את הTIMESPAN.

(בשימוש בגוגל EARTH פרו ניתן לראות את הציר זמן מופיע.)

פרטים נוספים על המטלה

-המטלה עברה שינוי מאסיבי(מעבר מ3 מחלקות לקרוב ל10) בהמלצת ספי אשר אהב את התוצאות אבל המליץ לפתוח ליותר מחלקות ולעבוד באופן יותר מדויק לפי מונחה עצמים.

-כל המידע שנקרא או נכתב עובד לפי אובייקטים והמחלקה WIFI היא זו שאחראית בין היתר על בחינת תקינות התוכן(טריק אשר עזר בבדיקת JUNIT).

-המטלה נבנתה בשיטת "תחנות" הפונקציות עובדות בשיתוף פעולה על ידי יצירת מסמכים זמנים.

(פונקציה א' מכינה קובץ זמני עבור פונקציה ב').-אשר צורת העבודה של הכתיבה והקריאה נעשית כמובן לפי אובייקטים ושימוש ב<LIST array.

מה שהפך את הקוד שלנו לדינאמי ביותר מבחינת שינויים !.

-לפני בדיקת JUNIT יש להריץ את הקוד לפחות פעם אחת.

הסבר על הJUNIT :

כל הפונקציות שנבדקו בנפרד הגיעו אך ורק ל3 סוגי תקלות אפשריות.(כל עוד אלו נפתרות הכל ירוץ חלק).

-קובץ לא נמצא.

-תוכן לא תקין.

-בחירת משתמש לא תקינה.

ולכן ה junit שלנו מכיל 3 סוגי בדיקות אשר עוברות על הכל ולמעשה מוודאות את תקינות כל הפונקציות.

קובץ לא נמצא:

מאחר ויש שימוש במסמכים זמניים (שנוצרים אחרי ההרצה הראשונית).

עשינו טסט לבדיקת קבצים שמאמתת שכל קובץ אשר חיוני למטלה קיים ובכך פתרנו את כל השגיאות בכל הפונקציות אשר יכולות להיווצר מקובץ שחסר.

דרך לבדוק/לקבל שגיאה:

אחרי ההרצה הראשונית.

יווצרו קבצים זמניים. במחלקה TEST_JUNIT של הבדיקה מופיעים כל אותם שמות קבצים אלו.

לך לתקייה של הפרויקט והסר את אחד מהם ותריץ את הבדיקה במידה והתוכנה תזהה שאחד מהקבצים שהמחלקות והפונקציות עובדות איתם לא קיים היא תחזיר FAIL.

תוכן לא תקין :

מאחר ועשינו שימוש נכון בשיטת מונחה עצמים כל קריאה או כתיבה או עבודה עם כל סוג

של סטרינג עובר למעשה דרך המחלקה WIFI אשר מנתחת ומפרקת את התוכן לאובייקטים/משווה/וכו'.

לצורך הJUNIT הוספנו לה בדיקת תוכן שכוללים את השגיאות הנפוצות שמצאנו של תוכן לא תקין:

1.בודקת שאפשרי לפצל סטרינג ולחלק אותו לאובייקטים בכלל.

2.בודקת בכל מיקום בפסיקים שהתוכן הנכון נמצא(לדוגמא שבאיזור של הLOT לדוגמא לא יופיע אותיות אלא רק ערך מספרי וכו')

דרך לבדוק/לקבל שגיאה:

-לערוך את מסמך CSV שנמצא בתקיית CSV:

1.או לכתוב שורות לא תקינות שלא מכילות את כל הפרמטרים(נניח מכתובת המאק עד הפרמטר של הזמן שבכך בעצם יהיו פרמטרים חסרים לתוכנה לעבוד בהמשך כמו פרמטר של מיקום).

2.או לשים באזור שמחייב תוכן מספרי כמו LOT וה ILAT וכו' פרמטרים שמכילים טקסט.

בחירת משתמש לא תקינה :

השגיאות מהסוג השלישי הם שגיאות שבהם בבחירה הראשונית אשר המשתמש יבחר הוא ילחץ לא נכון.

דרך לבדוק/לקבל שגיאה:

1.בבחירת סוג המיון להכניס קלט שלא תואם את האופציות .

2.בבחירת הפילטור לפי סוג המיון(בחירת הID הרצוי /טווח השעות/טווח המיקום) להכניס במקום קלט מספרי קלט שמכיל אותיות.

