

הכנות מונחה עצמים (שפת JAVA)

תרגיל בית אהבה מס' 3. (ארנון ברקת)

סמסטר א, 2020.

JavaFX

תאריך חלוקת התרגיל בכיתה: יום שלישי, 14/01/2020.
התאריך להגשת התרגיל: עד וכולל יום שלישי, 04/02/2020. (עד השעה 23:45).

יש להגיש אך ורק דרך תפריט המטלות שבאתר הקורס, כפי שהוסבר בתרגול.
שם הקובץ הראשי, שבו יש את `main`, יכיל את שם המגיש ויהיה אך ורק לפי הפורמט הבא: למשל:
עבור תרגיל בית מס' 1: `HW1_AviLevi.java`. עבור תרגיל בית מס' 2: `HW2_AviLevi.java`, וכך הלאה.
שם המחלקה יהיה אף הוא בהתאם. למשל: `public class HW1_AviLevi{`.
בנוסף, הקובץ הראשי יכלול בשורה הראשונה למעלה, הערה ובה השם המלא של המגיש/ה.
אם העבודה מכילה רק קובץ אחד, יש להגיש את קובץ ה-`java` היחיד הזה.
אחרת, יש לזפזף ולהגיש קובץ `zip`, שיכיל רק את קבצי ה-`Java`, ולא פרויקט של `Eclipse` וכו'.
בכל מקרה אין להגיש באתר יותר מקובץ אחד.
על הסטודנט חלה האחריות שהקובץ שהוא הגיש תקין, נשלח בצורה נכונה ומתאים לנדרש.
על התרגילים להיפתח בהצלחה בתוכנת ה-`Eclipse`, ללא שגיאות הידור או אזהרות.
יש לשלב בקוד הערות ותיעוד מתאים. (באנגלית בלבד!)
יש לבצע הזחה של הקוד כנדרש.
יש להקפיד על כללי הנדסת התוכנה ככל הניתן: קוד קצר, לא מסורבל ויעיל הן מבחינת כתיבתו והן מבחינת ריצת התוכנית. לא להשתמש במשתנים סטטיים שלא לצורך, תוכנית כללית שניתנת בקלות לשינויים והרחבות בעתיד, שימוש בקבועים, חלוקה מתאימה לפונקציות, פונקציות עצמאיות (כלומר שאינן תלויות בקוד /משתנה חיצוני), וכיוצ"ב. בנוסף, פתרון התרגיל צריך להיות גם כללי לכל שינוי של הנתונים, שינוי הגדלים של המערכים באם קיימים, וכו'.
כאשר הבדיקה תסתיים, תהיה על כך הודעה באתר. עד שאין את ההודעה, יש עוד עבודות שלא נבדקו.
לא יבדקו תרגילים שמוגשים באיחור ו/או שאינם עומדים בדרישות הנ"ל

בדיקת התרגילים מבוצעת ע"י גיא.

Guy.Kabiri@s.afeka.ac.il

1. כתוב תוכנית JavaFX כך שהמערכת מבצעת 'טיקים' / עדכון מצב בכל 1 שנייה.

התחלת ההפעלה של ה- 'טיקים' תתרחש כאשר מקליקים הקלקה כפולה עם העכבר במקום כלשהו על החלונית, או כאשר מקישים על האות S או s (שמיצגת את Start) בלוח המקשים.

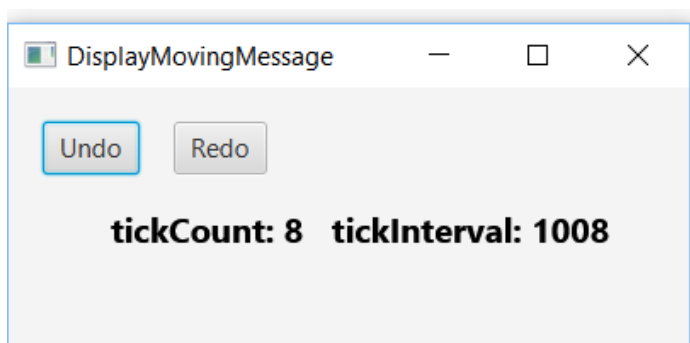
הסבר : כפי שניתן לראות בתמונה, במסך מוצג אובייקט של Text, שמציג את ספירת ה- טיקים (tickCount), וגם כיתוב נוסף באותו אובייקט Text, שנקרא tickInterval שמציג את ספירת ה- טיקים עם תוספת של 1000.

בכל 'טיק' כל ה- Text זו מעט ימינה, וגם הספירה שמוצגת מתעדכנת.

כאמור קודם, התחלת כל הפעולה כולה היא בהקלקה כפולה על המשטח עם העכבר, או לחיצה על S/s.

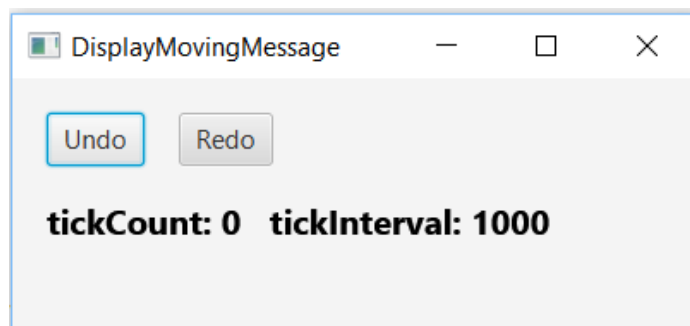
כאשר הטקסט כולו עובר את החלונית ונעלם בצד ימין, הוא ימשיך ויופיע מחדש מצד שמאל, ולאט לאט ייכנס חזרה כולו מחדש לתוך החלונית, תוך כדי שהספירה ממשיכה כרגיל וכך הלאה. (כלומר, אם החלק השמאלי של הטקסט כרגע נעלם בצד ימין, מייד בצד שמאל של החלונית, נתחיל לראות את התחלת החלק הכי ימני של ה- Text מתחיל להופיע, ומשם בהמשך כל ה- Text ייכנס לאט לאט, לתוך החלונית). שים לב : התנועה של ה- Text היא תמיד לצד ימין. (כלומר, כאשר הוא עובר את צד ימין הוא ממשיך 'באופן מעגלי' ונכנס מחדש בתנועה לצד ימין, החל מצד שמאל).

דוגמא להרצה, למשל למצב החלונית לאחר טיק מס' 8 :
(שים לב, שכל הטקסט כבר זו ימינה ביחס להתחלה במידה כלשהי.)



2. כפתור Undo :

מחזיר את מיקום הכתובית ואת מצב ה- tick להתחלה, ותנועת הכתובית תמשיך כעת מההתחלה :
דוגמא למצב החלונית מיד לאחר לחיצה על Undo, למשל בחלונית הני"ל :



ותנועת הכתובית תמשיך עכשיו כרגיל מכאן.

3. כפתור Redo :

מחזיר למצב האחרון (הן במיקום הטקסט והן בכיתוב שמופיע) שהיינו בו כאשר לחצנו על Undo האחרון,

ותנועת הכתובית תמשיך כרגיל משם. ניתן להניח שתמיד לוחצים על Redo רק לאחר שקודם לכן לחצנו על Undo.

4. יש לעבוד עם Lambda Expression ככל הניתן.

5. יש לעבוד עם KeyFrame , Timeline וכיוצ"ב לצורך מימוש התנועה.

בהצלחה!