

# תוכנה 1

## תרגיל מספר 10

### GUI, enum, hashCode and equals

#### **הנחיות כלליות:**

קראו בעיון את קובץ נהלי הגשת התרגילים אשר נמצא באתר הקורס.

- הגשת התרגיל תעשה במערכת ה-moodle בלבד (<http://moodle.tau.ac.il/>).
- יש להגיש קובץ zip יחיד הנושא את שם המשתמש ומספר התרגיל (לדוגמא, עבור המשתמש aviv יקרא הקובץ aviv\_hw10.zip). קובץ ה-zip יכיל:
  - א. קובץ פרטים אישיים בשם details.txt המכיל את שמכם ומספר ת.ז.
  - ב. קבצי ה-java של התוכניות אותם התבקשתם לממש, כולל תיקיות החבילה.

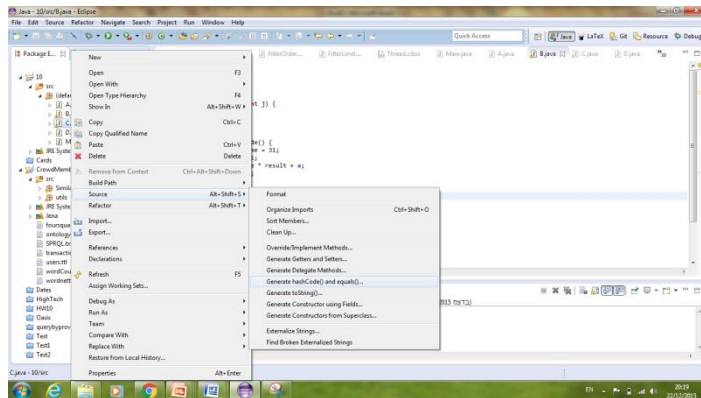
# שאלה 1 - חידת ג'אווה RIDDLES (20 נקודות)

בשאלה זו נשלים קוד ג'אווה, תוך תרגול של הבנת הקוד וחזרה על עקרונות שנלמדו בכיתה בנושאים שונים. באתר הקורס נתונה לכם החבילה riddles, ותחתיה המחלקה: Main. עליכם להשלים את המחלקות A, B, C, D, E, F מבלי לשנות את Main כלל, ומבלי להוסיף קבצים אחרים, כך ש:

1. כל הקוד יעבור קומפילציה ללא שגיאות וללא אזהרות.
2. אם נריץ את התכנית Main, היא תמיד תסיים את הריצה ללא שגיאות ותדפיס success!

הערות:

- יש להגיש רק את ארבעת קבצי A, B, C, D, E, F שכתבתם בתוך מבנה החבילות המתאים.
- אין מגבלה על הקוד של המחלקות, כל עוד הוא קוד ג'אווה תקין.
- מומלץ, בהינתן שגיאת קומפילציה, לחשוב תחילה בעצמכם כיצד לפתור אותה, לפני שתיעזרו בהצעות של eclipse. אם אתם משתמשים בהצעות אלה, היזהרו לא לשנות בטעות את Main, משום ששינוי זה עשוי לפגוע בנכונות הפתרון שלכם.
- בדקו את עצמכם ע"י הרצה של התכנית עם ערכים שונים.
- העזרו בהצעות של אקליפס כיצד לכתוב את הפונקציות hashCode ו-equals, וחשבו אילו שדות יש לקחת בחשבון בהשוואה. כדי לעשות זאת, עמדו על המחלקה הרצויה, לחצו על הכפתור הימני, ועקבו אחרי התמונה המצורפת:



## שאלה 2 - חידת ג'אווה – BUSTICKET (20 נקודות)

בשאלה זו נשלים קוד ג'אווה. באתר הקורס נתונה לכם החבילה `bus_tickets`, ותחתיה `Main`. עליכם להשלים את המחלקה `BusTicket` מבלי לשנות את `main` כלל, ומבלי להוסיף קבצים אחרים, כך ש:

1. כל הקוד יעבור קומפילציה ללא שגיאות וללא אזהרות.
2. אם נריץ את התכנית `Main`, היא תמיד תסיים את הריצה ללא שגיאות ותדפיס (ללא ירידות שורה מיותרות בין ההדפסות):

Available tickets:

Name: DAILY	Validity: 1 days	Price: 2.00 NIS	Description: Ticket valid for 24 hours
Name: WEEKLY	Validity: 7 days	Price: 13.30 NIS	Description: Ticket valid for 7 days
Name: MONTHLY	Validity: 30 days	Price: 54.00 NIS	Description: Ticket valid for 30 days
Name: QUARTERLY	Validity: 90 days	Price: 144.00 NIS	Description: Ticket valid for 3 month
Name: YEARLY	Validity: 365 days	Price: 474.50 NIS	Description: Ticket valid for 1 year

הערות:

- יש להגיש את קובץ `Main`.
- אין מגבלה על הקוד של המחלקה `BusTicket`, כל עוד הוא קוד ג'אווה תקין, והוא מממש את `enum`.
- זכרו שניתן להשתמש ב-`Switch-case` עבור ערכי ה-`enum`.
- ניתן להשתמש במתודות אברסטקיות ולממש באופן שונה עבור כל ערך.
- יש לשנות אך ורק את הקוד של המחלקה `BusTicket`.

## שאלה 3 - ג'אוה 8 - (10 נקודות)

השלם את השורה החסרה בmain של המחלקות החסרות Point,Point2.

בשתי השאלות יש להשלים את המחלקות באמצעות שורה אחת בלבד (5 נקודות לכל שאלה) ניקוד מלא יתקבל רק על ידי השורה הקצרה ביותר.

בשתי השאלות יש להוסיף שורה כך שהנקודות ברשימת הנקודות תהיינה ממוינות לפי ציר ה x.

את חלק זה של התרגיל יש לקמפל ולהריץ בג'אוה 8.

## שאלה 4 - משחק טריוויה - GUI (50 נקודות)

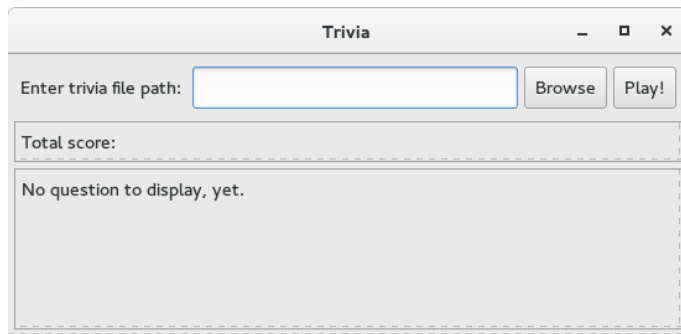
בתרגיל זה נכתוב ממשק משתמש גרפי אשר מתפקד כמשחק טריוויה בסיסי. נתרגל עבודה עם רכיבי GUI שונים והבנת קוד נתון - שלדי המחלקות והמחלקות אשר מופיעים באתר. אינכם נדרשים לבצע פעולות של עיצוב ה-GUI, אלא רק של תפעול האפליקציה לפי מהלך המשחק.

שימו לב, חלק מן החומר הדרוש לפתרון התרגיל יילמד בקורס רק בשבוע הבא, אבל כבר בשלב זה ניתן לממש את החלק הנוגע לטעינת שאלות טריוויה מקובץ ולבחירת שאלת טריוויה אקראית.

כמו כן, חלק זה בלבד (!) דורש שימוש ב-Java 8.

### תיאור השימוש בממשק הגרפי

כדי לפתוח את הממשק הגרפי יש להריץ את התכנית TriviaMain. כעת, ייפתח החלון הראשי של התכנית.

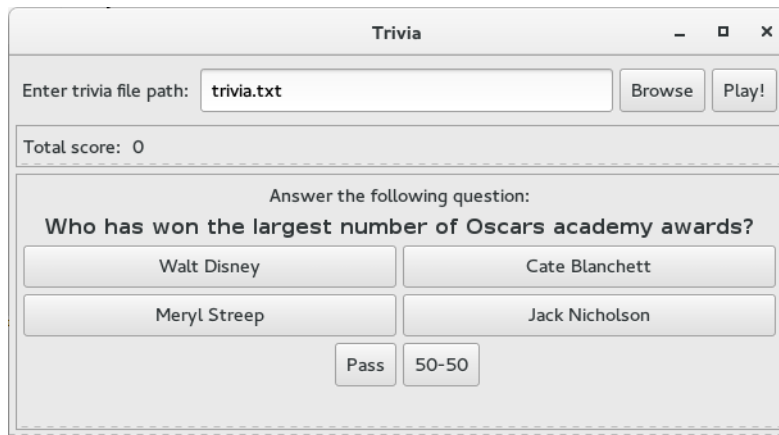


ניתן להקליד מסלול לקובץ בשדה הנתון, או ללחוץ על הכפתור Browse. במקרה הזה, ייפתח חלון לבחירת קובץ.

קבצי טריוויה הם קבצי טקסט בפורמט הבא: כל שורה מכילה שאלה ולאחריה 4 תשובות אפשריות, כאשר הראשונה מביניהן היא נכונה והיתר שגויות. בין השאלה לתשובות ובין התשובות מפריד טאב בודד. לדוגמא,

What is an apple?	Fruit	Vegetable	Fungus	Meat
When was the Beijing Summer Olympic Games?	2008	2014	2002	2000

לאחר בחירת מסלול הקובץ הרצוי ניתן ללחוץ על Play! בחלון הראשי. אם הקובץ תקין, הוא ייטען ע"י האפליקציה ותוצג למשתמש שאלה אקראית מתוכו ואת ארבע התשובות האפשריות עבורה בסדר אקראי. בשלב זה הניקוד הכולל של המשתמש הוא 0.



המשתמש יכול לבחור באחת התשובות או להשתמש באחד משני גלגלי הצלה: דילוג על השאלה (Pass) ו-50-50.

### חוקי המשחק:

- תשובה נכונה = +3 נקודות
- תשובה שגויה = -2 נקודות
- המשחק מסתיים בעקבות אחד מהאירועים הבאים:
  - 3 תשובות שגויות ברצף (פסילה)
  - נגמרו השאלות במאגר
- גלגלי הצלה זמינים לשימוש, מתחילת המשחק, ובכל שאלה מחדש תחת התנאים הבאים:
  - השימוש הראשון בכל גלגל הצלה (בנפרד) אפשרי ללא הגבלה או תלות במאזן הניקוד של השחקן
  - לאחר שימוש ראשון בגלגל הצלה מסוים, כל עוד מאזן הניקוד של השחקן קטן או שווה ל-0 לא ניתן לחזור ולהשתמש בו בשאלות נוספות.
  - לדוגמה, לאחר שימוש ראשון בכפתור 50-50, אם הניקוד הוא -2 הגלגל לא זמין לשימוש (לא ניתן ללחוץ עליו). ברגע שהשחקן יקבל ניקוד גדול מ-0 הגלגל יחזור לשימוש ויהיה זמין ללחיצה.
- לאחר שהשחקן צובר נקודות ומאזנו חיובי ניתן לפתוח את השימוש בכל אחד מגלגלי הצלה שוב
- ניקוד שימוש בכל אחד מגלגלי הצלה:
  - חינם בפעם הראשונה
  - כל שימוש נוסף מעבר לפעם הראשונה = -1 נקודה

כמו כן, תוצג למשתמש שאלה אקראית **חדשה** (המשתמש לעולם לא יקבל את אותה שאלה פעמיים באותו משחק). המשחק כאמור יסתיים לאחר 3 תשובות שגויות ברצף או כאשר סיימנו לעבור על כל השאלות בקובץ. בשלב זה תוצג למשתמש הודעה מסכמת.



### תכנית לדוגמא

נמצאת באתר הקורס. במקרה של ספק לגבי אופן פעולת המשחק במצב מסויים, ניתן להיעזר בדוגמא זו. יש להוריד את הקובץ Trivia.jar ולשמור אותו בתיקיה כלשהי במחשב. באותה תיקיה, שמרו עותק של swt.jar המתאים למערכת ההפעלה שלכם. ניתן להוריד את ה-jar מ- <http://www.eclipse.org/swt>. מומלץ להוריד ע"י לחיצה על Releases > Stable > more... , ולחיצה על השורה המתאימה למע' ההפעלה שלכם. קובץ הזיפ שתורידו יכיל בתוכו את swt.jar, ויש להוציא אותו מן הזיפ ולשמור באותה תיקיה יחד עם Trivia.jar. כעת, כדי להריץ את התכנית לדוגמא מ- command line היכנסו לאותה תיקיה (ב-Windows למשתמשי terminal, Linux למשתמשי command line), והקישו את הפקודה

```
java -jar Trivia.jar
```

(לחלופין, בחלק ממערכות ההפעלה, לחיצה כפולה על קובץ ה-jar תפעיל אותו) אם פעלתם לפי ההנחיות, ו- java 8 מותקנת על המחשב, ייפתח חלון עם משחק הטריוויה.

### מה עליכם לעשות

באתר הקורס נתונים לכם שלדים של מחלקות ה-GUI בחבילה `il.ac.tau.cs.sw1.trivia`. הוסיפו ושנו קוד לפי הצורך. יש להוסיף ל-`build path` של הפרויקט את `swt.jar` המתאים למערכת ההפעלה. באתר הקורס גם נתון קובץ טריוויה תקין לדוגמא, `trivia.txt`.

קיים פרוט הבא לשאלה:

## פירוט

**5] נק' הכפתור Browse:** הוסיפו מאזין לכפתור זה כך שלחיצה עליו תפתח חלון לבחירת קובץ, וכך שאם נבחר קובץ המסלול אליו יתעדכן בשדה הטקסט המתאים. היעזרו במתודה `GUIUtils.getFileFromFileDialog` הנתונה לכם.

**20] נק' הכפתור Play!** הוסיפו מאזין לכפתור כך שלחיצה עליו תגרום לטעינת השאלות והתשובות מהקובץ הנתון. ניתן להניח שפורמט הקובץ תקין.

שמרו את השאלות והתשובות במבנה נתונים לבחירתכם. ניתן להוסיף מחלקות כרצונכם. שימו לב, המזהה הייחודי של שאלה הוא הטקסט של השאלה בצירוף קבוצת התשובות שלה. כלומר, ייתכנו שתי שאלות עם אותו טקסט, אך עם קבוצת תשובות שונה (ראו דוגמא ב-`trivia.txt` שורות 6-7).

לבסוף, הציגו שאלה אקראית מן הקובץ ואת התשובות שלה בסדר אקראי. כדי ליצור את הכפתורים היעזרו במתודה `updateQuestionPanel`. ניתן להיעזר במתודה `nextInt` של המחלקה `java.util.Random` כדי לקבל מספר אקראי, ולא במתודה `Collections.shuffle` כדי לסדר את רשימת התשובות בסדר אקראי.

אתחלו את הניקוד להיות 0, ושדות אחרים לפי הצורך.

**15] נק' כפתורי התשובות:** הוסיפו מאזינים לכפתורים האלה כך שלחיצה על הכפתור הנכון תגדיל את הניקוד ב-3 ולחיצה על כפתור של תשובה שגויה תקטין את הניקוד ב-1 ותגדיל ב-1 את מספר התשובות השגויות.

בדומה לכפתור ה-Play!, יש להציג את השאלה הבאה. נסו לשתף קוד בין מאזיני הכפתורים. במידה והמשחק נגמר (התקבלו 3 תשובות שגויות ברצף או סיימנו את כל השאלות) יש להציג למשתמש הודעה מסכמת. לשם כך, היעזרו במתודה `GUIUtils.showInfoDialog`.

**5] נק' כפתור Pass:** הוסיפו לכפתור זה מאזין המתקדם לשאלה הבאה מבלי לעדכן את מספר השאלות שנענו (המצוין בהודעה המסכמת). זיכרו לעדכן את מצב הכפתור והניקוד בהתאם לחוקי המשחק. שוב, נסו לשתף קוד עם המאזינים לכפתור Play! וכפתורי התשובות.

**5] נק' כפתור 50-50:** הוסיפו לכפתור זה מאזין אשר מנטרל שני כפתורי תשובות לא נכונות ומשאיר רק כפתור עם תשובה נכונה ואחד עם תשובה לא נכונה באופן אקראי. זיכרו לעדכן את מצב הכפתור והניקוד בהתאם לחוקי המשחק. שוב, נסו לשתף קוד עם המאזינים לכפתור Play! וכפתורי התשובות.

## הערות

- לצורך פתרון התרגיל, חשובה יכולת קריאה והבנה של הקוד הנתון, תוך זיהוי החלקים הרלוונטיים לקוד שתוצו להוסיף. הקוד הנתון מתועד ברובו, אך ייתכן שתצטרכו להיעזר גם בתיעוד של SWT, בתיעוד של java ובאינטרנט כדי להבין חלקי קוד מסוימים.
- בחלק זה של התרגיל אתם רשאים לשנות את המחלקות והמתודות הנתונות לפי הצורך, מלבד פונקציית ה-main הראשית במחלקה `il.ac.tau.cs.sw1.trivia.TriviaMain`.
- לפני תחילת פתרון התרגיל, מומלץ לעבור על כל הקוד, ובפרט על המחלקה `GUIUtils`. ייתכן שתמצאו שירותים שיעזרו לכם בפתרון התרגיל (מומלץ אבל אין חובה להשתמש בהם).
- בדקו את עצמכם ע"י משחק בתכנית.
- יש להגיש את כל המחלקות (קבצי java), גם אם לא שיניתם אותן. אין צורך להגיש את `swt.jar`.

בהצלחה!