**EXCEPTION**

**דף תרגול מס' 11– חלק א'**

1. נסי ובדקי אלו סוגי exceptions יזרקו במקרים הבאים:

הערה: שימי לב שלחיצה על קישורית כחולה בחלון הoutput תעמיד את הסמן בשורה הבעיתית.

* צרי אוסף מטיפוס List (List list=null) , "שכחי " לעשות לו הקצאת זיכרון (אתחלי אותו null) נסי להוסיף לו אלמנט .

סוג ה EXCEPTION \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

שימי את השורה בהערה ועברי לשלב הבא: (כך אחרי כל שלב)

* בצעי לאוסף הנ"ל הקצאה (new ArrayList()=), הוסיפי לו 2 אלמנטים, הפעילי את הפונקציה GET שלחי כפרמטר אינדקס שחורג מגבולות המערך מ 2 והלאה.

סוג ה EXCEPTION \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* צרי אובייקט מטיפוס HashSet , אתחלי אותו במערך הנ"ל . הוסיפי המרה ArrayList HashSet set=( HashSet) list;

סוג ה EXCEPTION \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* נסי לבצע את ההמרה הבאה: Float.parseFloat("456.7m");

סוג ה EXCEPTION \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* שלחי מפונקציה X לפונקציה Y ולהפך (כלומר, שוב מ Y ל X)

סוג ה ERROR \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* הגדירי משתנה פרימיטיבי מטיפוס שלם, בצעי לו חלוקה באפס.

סוג ה EXCEPTION \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* הגדירי 2 מערכים פשוטים, אחד מטפוס int והשני מטיפוס float. העתיקי מהמערך של ה float למערך של ה int , ע"י הפונקציה System.arraycopy

סוג ה EXCEPTION \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* צרי אובייקט של LocalDate.of , אתחלי את האובייקט בנתונים לא הגיוניים. כתבי סוגים שונים של חריגות
* דוגמא נוספת לException

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. שימי את כל השורות הנ"ל בתוך קטע של try (בטלי את ההערות) , הוסיפי מס' catch לבלוק ה tryעבור לכידת כל השגיאות הנ"ל .כל catch יקבל סוג אחר של חריגה, בכל catch הדפיסי הודעה מתאימה, המסבירה את השגיאה , הדפיסי גם את ex.getMessage() ..
2. בחרי 2 מתוך החריגות שבעבודה 1, אותם לא מערכת ההפעלה תזרוק אלא המתכנת

לדוגמא:

if (s==null)

throw new NullPointerException("אין אפשרות להפעיל על null");

שלחי הודעת שגיאה והדפיסי אותה ב catch ע"י ex.getMessage().

**דף תרגול – חלק ב'**

**ע"פ מצגת מס 1 (מצורפת) – משקופית 124**

**שקופית 125**

**בצעי, כמו בדוגמא קלט של 2 מספרים, הריצי את התרגיל פעמיים, הכנסי בכל פעם נתונים שגויים. הדפיסי את השגיאות**

**שקופית 126**

**מה החסרון בשימוש ב if ללא exception ?**

**שקופיות 128-129**

**פתרי את הבעיה על ידי try ו catch.**

**שקופית 130**

**הוסיפי לולאת while - עד שמתקבל קלט תקין.**

**שקופית 131 – 135**

**לקרוא**

**שקופית 136**

**מה המטרה של בלוק ה finally? הוסיפי לתרגיל.**

**שקופית 137**

**לקרוא**

**שקופיות 138- 141**

**גלגול חריגות : לעיתים נרצה שהפונקציה לא תטפל בעצמה בבעיה אלא תגלגל את החריגה לפונקציה שקראה לה.**

**קראי על כך בשקופיות הנ"ל. (נסביר שוב בשיעור הבא)**

**שקופיות 142-144**

**עקבי אחר תכנית הדוגמא , ששולחים הודעות משלנו על השגיאה. (כמו שעשינו בחלק א שאלה 3)**

**שקופית 145** והלאה - עוד תכניות דוגמא לשימוש ב exception . מומלץ מאוד לקרוא

חלק ג'

העתיקי מחלקה עם אובייקט של תאריך. (או ליצור עם 3 שדות – שם , קוד וכו' פונקציות בסיסיות)

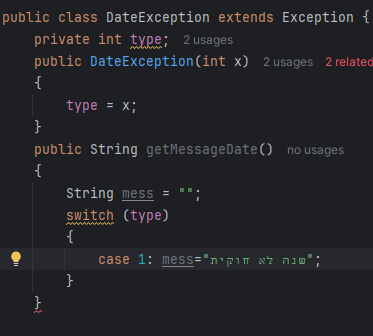
צרי מחלקה חדשה של exception בשם DateException (לא לשכוח לרשת מ..Exception.)

במחלקה תהיה פונקציה בונה שמקבלת מספר (וגם מדפיסה: נוצר אובייקט של חריגה), וכן פונקציה נוספת שלפי המספר תדפיס הודעה מתאימה באדום, לדוגמא 1- בעיה בשנה 2. בעיה בחודש3. בעיה ביום

שלחי מה main לפונקציה שתקרא לעוד פונקציה ושם יש ליצור אובייקט Person, לשאול על השדות של אובייקט של התאריך האם הן חוקיות, ליצור את החריגה ולשלוח מספר. גלגלי את החריגה אחרונית לפונקציה שקראה לה. שם טפלי ע"י try catch.

בבלוק ה catch הפעילי את הפונקציה שמדפיסה הודעה מתאימה. וכן את מחסנית הקריאות (printStackTrace)

קוד לעזרה:



יצירת מחלקה של exception חדש.

