

## הנחיות כלליות של ביצוע כל השלבים של המיני-פרויקט

- אנא קחו לתשומת לבכם שמדובר במיני-פרויקט במבוא להנדסת תוכנה ולגרפיקה ממוחשבת. הדרישות הקשורות להנדסת תכנה – הינן דרישות חובה. אי עמידה בדרישות האלה תגרום ולהורדה משמעותית בציון בכל שלב ושלב. גם כאשר מדובר באותה הפרה – הציון ירד בכל שלב ושלב.

להלן הדרישות הכלליות לעבודה על כל השלבים של הפרויקט בקורס:

- חובה לעבוד בזוגות ע"פ עקרונות **Pair Programming** במסגרת **XP (Extreme Programming)** כפי שנלמד בקורס "מבוא להנדסת תכנה"

- חובה להקפיד על **מוסכמות שמות** כפי שנלמד במבוא לגיאוה הקלטה או בכיתה:

- חובה שמות בעלי משמעות ומפורמטים לפי **camelCase** | **PascalCase** בהתאם
- חובה שהבלוקים {} יהיו מעוצבים על פי **Java Style**
- חובה לטפל ב**פירמוט קוד** אחיד: הזחות, ריווח בתוך שורה ושורות רווח (אך לא רצף של שורות רווח)

- כבר מהשלב הראשון חובה לפתוח לפרויקט מאגר **git** מקומי ולסנכרנו (לאחר "פרסום" ראשוני) באתר **github**
  - המאגר באתר יוגדר כפרטי (**private**), לא יתקבלו נימוקים של קריסת מחשבוניסך או כל אסון טבע אחר
  - שמות הפרויקט והמאגר

■ **שם פרויקט == שם מאגר**

■ פורמט השם (יש להקפיד על אותיות גדולות ומיקום של "\_"): **ISE5785\_XXXX\_YYYY**

כאשר **XXXX** ו-**YYYY** – ארבע הספרות האחרונות של מספר ת"ז של השותפים לזוג

- יש להוסיף את המרצה כשותף (collaborator) למאגר ב- **github**:

אורית רוזנבליט	roorit
אליעזר גנסבורגר	eliezergensburger
גיא יחזקאל	GuyYech
דן זילברשטיין	dzilbers
יאיר גולדשטיין	yair-go

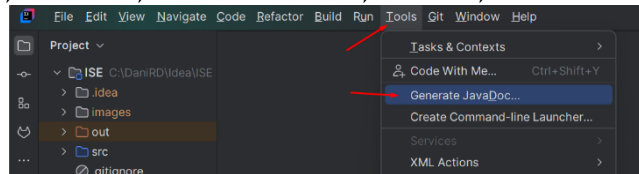
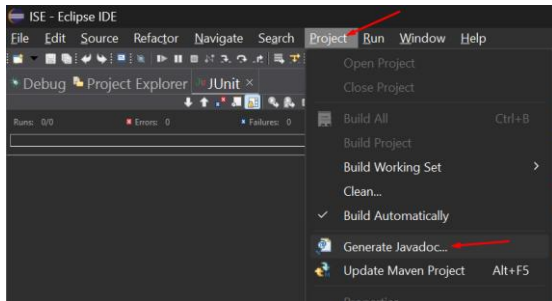
- הגשת שלבים:**

- לאחר השלמת שלב ושמירתו במאגר, יש להוסיף ל- **commit** האחרון של השלב **תג (tag)**
- שם התגים של השלבים: **PR01**, או **PR02**, וכדומה – לפי מספר השלב המוגש
- התג של שני השלבים האחרונים (הנקראים "מיני-פרויקט"יים) יהיו **MP01** ו-**MP02** בהתאם
- לאחר מכן יש לדחוף את **התג** למאגר המשותף של הפרויקט ב- **github**
- בתיבת הגשה יש להגיש בהגשה מקוונת בתיבת הטקסט את **הקישור לתג**

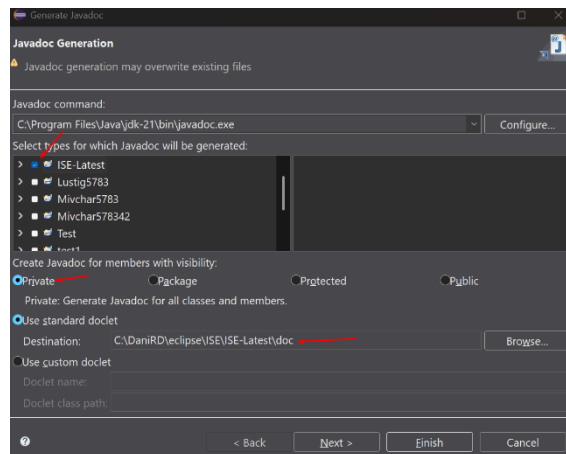
**ראה המשך בעמוד הבא**

• חובת תיעוד :

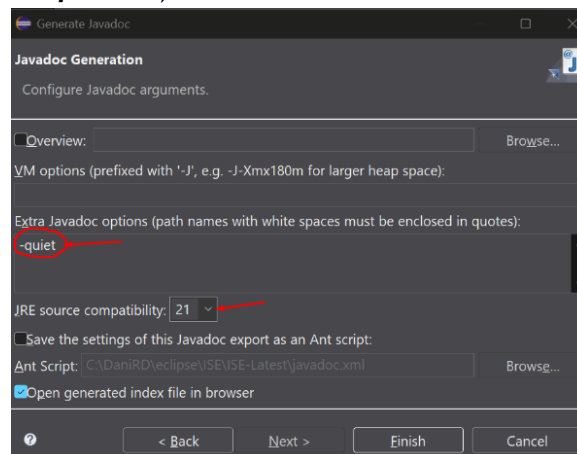
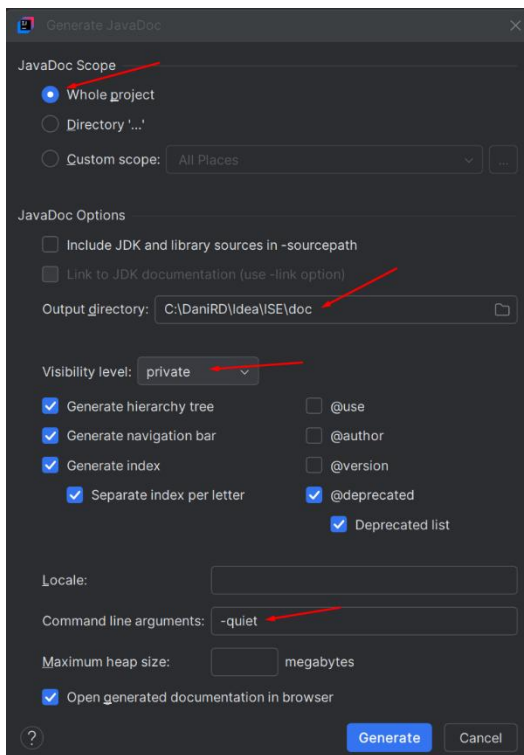
- כל התיעוד חייב להיות **באנגלית** תקינה (לא יורד הציון על טעויות כתיב, אבל **אסור** להשתמש בשפות חלופיות באותיות לטיניות – למשל כמו **mechane** במקום **denominator**)
- תיעוד חיצוני בפורמט **JavaDoc**
  - לפני הצהרת כל מחלקה, כל שדה, כל מתודה (עם כל הרשאת גישה – מ-**private** ועד **public**)
  - יחד עם זאת, **אסור** לתעד בפורמט זה דריסה מתודה שירשנו או מימוש מתודת ממשק
- התיעוד חייב להיות רק בממשק או במחלקת האב האבסטרקטית המתאימים
- חובה ליצור בפרויקט תת-תיקיה עם תיעוד הפרויקט בעזרת פקודה של סביבת פיתוח **Generate Javadoc**



- יש ליצור תת-תיקיה **doc** בתוך תיקיית הפרויקט
- יש להוסיף את תיקיית **doc** בקובץ **.gitignore**.
- הגדרות קונפיגורציה (פעולה חד-פעמית – בפעם הראשונה):
  - יש להגדיר תת-תיקיה **doc** כתיקיה יעד ליצירת התיעוד
  - יש להגדיר את הרשאת גישה מינימלית לתיעוד כ-**private**, למשל :



- יש לוודא גרסה נכונה של **JRE** ולהוסיף השקטת דיווחים שוטפים בעזרת אופציית השקטה **-quiet** :



- יש לוודא (לעיל) שמסומן כל הפרויקט ליצירת התיעוד (מופיע סימון "v" ליד שם הפרויקט)!
- יש לוודא שאין הערות על חוסרים ובעיות בתיעוד
- בקוד הממש מודל מתמטי / לוגיקה **שאינם** טריוויאליים – יש לתעד בתוך הקוד עם הסבר תמציתי אך ממצה