

מדריך התקנות - תכנות מונחה עצמים

מדריך זה נכתב כמחווה לתלמידי הקורס על ידי הדס נחשון

מאי 2021

Most good programmers do programming not because they expect to get paid or get adulation by the public,
but because it is fun to program. - Linus Torvalds, creator of Linux



פרק 1: הורדת ג'אווה	1
מה להוריד?	2
איך אדע אם כבר יש לי ג'אווה במחשב?	2
הגדרת משתני סביבה	2
פרק 2: הורדת עורך קוד פשוט	4
פרק 3: הוראות עבודה עם הטרמינל	5
ניווט בקונסול	5
עבודה עם java ו-javac דרך הקונסול	6
יצירת קובץ ריק	7
פרק 4: הוראות הורדה והתקנה של IntelliJ IDEA	8
הורדה והתקנה	8
יצירת פרויקט	9
הגדרות נראות של IntelliJ	10
פרק 5: קיצורי מקשים של ה-IDE	10
פרק 6: הוראות הורדה והתקנה של DanoGameLab	11
פרק 7: הוראות ייצוא ל-jar	12

פרק 1: הורדת ג'אווה

מה להוריד?

ישנן שתי אפשרויות הורדה:

1. דרך [Oracle](#) - תהליך פשוט מאוד. היכנסו לאתר, הורידו את ה-exe המתאים, הוא יפתח, לחצו Next על כל מה שרוצים מכם וזהו. בסוף התהליך תיווצר תיקיה ששמה מתחיל ב-jdk, בתוך תיקייה בשם Java, בנתיב שהופיע לכם בזמן ההורדה (לרוב C:\Program Files\Java).
2. דרך [OpenJDK](#) - גם תהליך פשוט מאוד! מורידים קובץ zip. עושים extract. מקבלים תיקיה בשם jdk עם מספר הגרסה, למשל jdk-16.01.1. ניתן לשים אותה היכן שרוצים לפני שעוברים לשלב הבא, רצוי בתיקיה בשם Java תחת C:\Program Files למשל.

אז מה ההבדל, אתם בטח תוהים. OpenJDK הוא קוד פתוח (כלומר קוד המקור פתוח ונגיש לכולנו). המימוש של Oracle, לעומת זאת, הוא יותר מהיר, שזה מצוין למשחקים שנפתח בהמשך הקורס. גם הוא חינמי, אך ייתכן שיצריך מכם יצירת חשבון. (למה רק ייתכן? כי זה תלוי ביום שבו ניסיתם להוריד, בכיוון הרוח ובשאלה האם מרקורי בנסיגה). בשני המקרים, **יש להוריד לכל הפחות גרסה 10**, אחרת חלק מהקוד שנספק לכם פשוט לא יעבוד.

איך אדע אם כבר יש לי ג'אווה במחשב?

ראשית, אם הרצת הפקודה java -version כפי שמוסבר [למטה](#) מניבה את התוצאה הרצויה, סימן שכבר עברתם את כל המסע הזה מתישהו. מזל טוב! אחרת, בכל זאת יש סיכוי שהתקנתם ופשוט לא סיימתם את התהליך. בדקו אם ג'אווה מופיעה ברשימת התוכנות המותקנות אצלכם (Start -> Control Panel -> Add/Remove Programs).

הגדרת משתני סביבה

זהו? סיימתי? כמעט! נטפל בדבר אחרון שיקל על העבודה שלנו בהמשך. לפעמים נרצה להיות מסוגלים להריץ תוכנות משורת הפקודה של המחשב. חפשו "terminal" במחשב שלכם. אנשי חלונות מוזמנים כעת להגליד "calc" וללחוץ על Enter. הפלא ופלא, נפתח מחשבון!

נשאלת השאלה האם אתם יודעים לומר היכן שמורה התוכנה של המחשבון במחשב שלכם? כנראה שלא. איכשהו המחשב שלכם כבר מכיר את הנתיב מראש! נרצה לאפשר זאת גם לתוכנות אחרות, למשל ג'אווה שכרגע הורדתם.

במקום שנצטרך להקליד את הנתיב של התוכנה כל פעם מחדש, נשמור אותו מראש. בדיוק בשביל זה הומצא משתנה סביבה (environment variable) בשם PATH!

1. מצאו את הנתיב שבו התקנתם את ה-JDK. בתוך תיקיית Java תופיע תיקיה עם ה-JDK שהתקנתם, למשל jdk-12.0.2, כמו אצל הלא מעודכנים כמוני. בתוכה תמצאו את התיקיה bin שמכילה את כל הקבצים שצריך כדי להריץ את התוכנה. העתיקו לכם את הנתיב לתיקיית ה-bin.

2. במק:

a. **אפשרות ראשונה:** להתקין את Command Line Tools של XCode. לא רק שזה יטפל בכל עניין ה-PATH, זה גם יתקין לכם כלים רבים למפתח כמו gcc, cmake, java ועוד.

- אם החלטתם להתקין, לאחר ההתקנה פתחו את הטרימינל וכתבו java. אם הפקודה מזוהה, סיימתם את התקנת ג'אוה!

- אם מסיבה כלשהי אתם מעדיפים שלא להתקין את הכלי, או אם לאחר ההתקנה קיבלתם הערה בסגנון bash: javacc: command not found, נסו את האפשרות השניה (b).

b. **אפשרות שניה:** נגדיר את משתנה הסביבה באופן ידני. בצעו את הפקודות הבאות:

i. פתחו טרימינל. כתבו ~ cd ולחצו Enter.

ii. כתבו which java והעתיקו לכם את הנתיב שמופיע כדי שיהיה לכם. (זה אמור להיות משהו כמו /usr/bin/java.)

iii. הריצו "vim .bash_profile". (לסקרנים מביניכם, Vim הוא עורך טקסט.)

iv. לחצו על המקש i (זה יכניס אתכם ל-Insert Mode ב-Vim, כלומר מצב כתיבה).

v. אתרו את השורה:

Export PATH=/usr/local/bin:/usr/bin:/bin:<other file path>: ecc...

vi. הוסיפו את הנתיב ששמרתם לכם בצד. זה יראה כך:

Export PATH=/usr/bin/java:/usr/local/bin:/usr/bin:/bin:<other file path>: ecc...

vii. הקישו Esc (לצאת מ-Insert Mode) ואז "wq" (לשמור ולצאת מ-Vim). לא צריך Enter.

viii. התחילו מחדש את ה-terminal וכתבו java כדי לראות שהכל עובד.

ix. לחילופין, אל תעשו אף אחד מהשלבים האלו וחזרו להערה הראשונה.

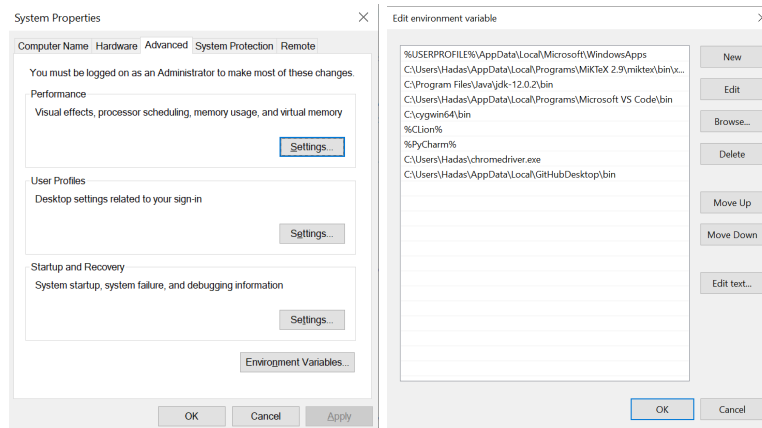
3. בחלונות 10:

a. חפשו Environment variables במחשב שלכם ולחצו על Edit the system environment variables. (למי שזה לא מוצא, המסלול הארוך הוא: Control Panel → System ואז לחפש שם בצד.)

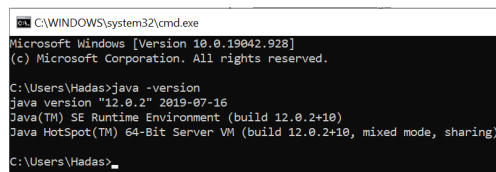
b. בתחתית החלון שנפתח יהיה כפתור Environment Variables.

c. בחלון התחתון, תחת System variables, לחצו על Path ואז על Edit.

d. בשורה הריקה הראשונה הדביקו את הנתיב לתיקיית ה-bin.



כדי לבדוק שהכל עובד, נסו להריץ בשורת הפקודה (ה-terminal ממקודם) את הפקודה הבאה: `java -version`. וודאו שמופיע משהו כזה:



פרק 2: הורדת עורך קוד פשוט

לפני שנשלוף את התותחים הכבדים, נתחיל עם עורך קוד פשוט. הנה ההמלצות שלנו, כתלות במערכת ההפעלה שלכם.

1. חלונות: Notepad++. (שימו לב ל-++! לא מדובר ב-Notepad הרגיל, שאינו מיועד לקוד).
2. לינוקס: Notepadqq או Geany.
3. מק: TextWrangler.



פרק 3: הוראות עבודה עם הטרמינל

ניווט בקונסול

הגיע הזמן לדבר על עבודה בסיסית עם ג'אווה והקונסול. בשלב זה יש לאנשים נטייה להיבהל ולחשוב שכל היקום קורס לתוך עצמו. אם יורשה לי לצטט את דאגלס אדמס, [DON'T PANIC](#). התרגלנו לעולם שבו לכל תוכנה יש מסך ידידותי למשתמש, עם ציורים וצבעים וכפתורים חמודים. אבל בסופו של דבר, הרי מדובר בפקודות! אז למה לא לכתוב אותן ישירות למחשב? לדבר תכלס? זאת שורת הפקודה. פשוט תכלס.

פתחו את הקונסול במחשב שלכם:

- אנשי חלונות: לחצו WinKey + R ואז כתבו cmd ולחצו Enter.
- אנשי מק: חפשו Terminal.
- אנשי לינוקס: אתם לא צריכים אותי.

בחלון שיפתח, שורת הפקודה תראה לכם את מה שנקרא ה-current working directory (תיקיית העבודה הנוכחית), או במילים אחרות התיקה שבה הקונסול ואתם נמצאים כעת. למשל, כאן למטה אפשר לראות שאנחנו נמצאים בתיקה שלי, בתוך תיקיית Users, בכונן C של המחשב שלי.

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.19042.928]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Hadas>
```

כדי לעבור בין תיקיות, נשתמש בפקודה cd, שהיא קיצור של change directory. (שימו לב, לעיתים רווחים בשמות של קבצים ותיקיות עושים בעיות. במקרה זה, שימו מרכאות סביב שם התיקה.)

```
C:\Users\Hadas>cd PycharmProjects\NLP
C:\Users\Hadas\PycharmProjects\NLP>
```

טיפ! אפשר להשתמש ב-Tab בשביל השלמה אוטומטית. למשל, אם במקום PycharmProjects הייתי מקלידה רק Py, לחיצה אחת על Tab הייתה מציעה לי את Pycharm2019.3. ולחיצה נוספת הייתה מציעה לי את PycharmProjects.

רוצים לחזור חזרה במעלה היררכיית התיקיות? כתבו .. אחרי cd.

```
C:\Users\Hadas>cd PycharmProjects\MLP
C:\Users\Hadas\PycharmProjects\MLP>cd ..
C:\Users\Hadas\PycharmProjects>cd ..
C:\Users\Hadas>
```

ואם תיקיית הג'אווה שלכם בכלל בכונן אחר, נרצה קודם לעבור כונן ואז לנווט בדיוק כפי שהוסבר לעיל. כיצד עוברים כונן? פשוט כתבו את שם הכונן באות קטנה, ואחריה נקודותיים ו-Enter:

```
C:\Users\Hadas>g:
G:\>
```

עבודה עם java ו-javac דרך הקונסול

כמה טיפים על חשבון הבית:

1. וודאו שג'אווה מותקנת כראוי ושהקונסול שלכם מכיר את הפקודות java ו-javac. אם לא, קפצו רגע לפרק 1.
2. נחכה לכם כאן, אל תדאגו.
2. כדי לקמפל יחד שני קבצי ג'אווה A.java, B.java, נווטו לתיקיה שבה הם נמצאים (בעזרת פקודת cd) וכתבו javac A.java B.java. (ולמי שמתעניין, javac הוא הקומפיילר שהגיע עם ה-JDK שהורדתם בפרק 1.)

הידעת? על אף שמפתה לומר "javak", שמו האמיתי של javac הוא "java-see". (וזאת הזדמנות לומר שזה "נייקי", לא "נייק", אם כבר התחלנו.)

3. כדי לקמפל יחדיו את כל קבצי ה-java. בתיקיה אחת, כתבו javac *.java.

הידעת? הכוכבית כאן היא מה שנקרא wildcard, שמתאים לכל צירוף אותיות שנגמר ב-java. במילים אחרות, זה כמו לומר "היי קומפיילר, קח לי את כל הקבצים עם שמות שנגמרים במחרוזת java". מי שתמשיך לעבוד עם שורת הפקודה תיתקל בכוכבית הזאת עוד הרבה.

4. כדי להריץ את פונקציית ה-main במחלקה A.java, כתבו A.java. (פקודה זו מניחה ש-A קומפלה בהצלחה ושהקובץ אכן נמצא בתיקיה.) שימו לב: java A.class לא יעבוד, אלא רק A.java.
5. ניתן לאחד בין טיפ 3 ו-4 כך: java & *.java *.java. פקודה זו משמעותה לקמפל ואז מיד להריץ.
6. נניח שקימפלתם, הרצתם, חזרתם לתקן ואתם רוצים כעת לחזור על כל התהליך. במקום להקליד מחדש, לחצו על החץ למעלה. זה יציע לכם את הפקודה האחרונה שהרצתם. רוצים אחת אחרת? המשיכו ללחוץ למעלה עד שתתקלו בזאת שברצונם להפעיל שוב.
7. הפקודה cls (בחלונות) או clear (במק ובלינוקס) תנקה לכם את המסך. ניתן לשלבה עם הנקודות הקודמות כך:

```
cls && javac *.java && java A
clear && javac *.java && java A
```

8. בכל שלב, ניתן לעצור ביצוע של פקודה על ידי לחיצה על ctrl-c. בתוכנות הקטנות שנכתוב, זאת פקודה בטוחה לחלוטין.

יצירת קובץ ריק

כיצד ניצור לנו את הקובץ Chat.java? אפשר דרך הקונסול, דרך עורך הקוד ואפילו דרך המקש הימני המוכר לנו:

דרך הקונסול: כן כן, ניתן ליצור קבצים דרך הקונסול! נווטו אל התיקיה שבה אתם רוצים ליצור את הקובץ בעזרת ההסברים למעלה. כעת כתבו:

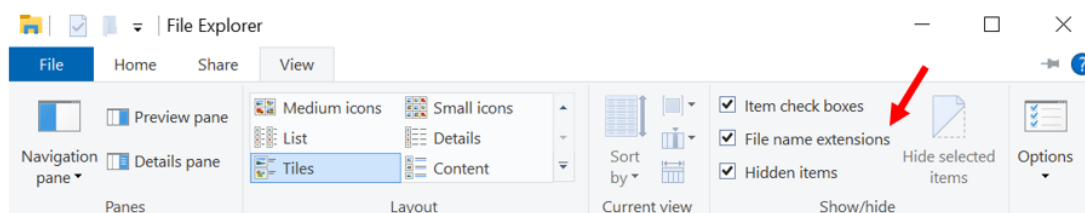
- בחלונות: type NUL > Chat.java
- במק/בלינוקס: touch Chat.java

דרך עורך הקוד: דרך נוספת ליצור קובץ היא פשוט לשמור קובץ חדש בשם הרצוי דרך עורך הקוד שלכם. (איזה עורך, תזכירי לי? קפצו רגע לפרק 2!)

דרך ה-context menu: ניתן ליצור קובץ java על ידי פתיחת ה-context menu (הדבר הזה שנפתח בלחיצת מקש ימני), יצירת קובץ טקסט ושינוי שמו לשם הרצוי. שימו לב: שיטה זו תעבוד רק אם המחשב שלכם מציג סיומות של קבצים. אחרת אתם תיצרו את הקובץ Chat.java.txt (שזה פשוט קובץ טקסט עם השם Chat.java) ולא Chat.java, וזה יהיה מאוד מבאס להבין שזה לא עובד.

בעצם, זה רגע טוב להגיד באופן כללי, שסיומות של קבצים הן כלי מאוד שימושי בפיתוח, וזה זמן מצוין להתחיל להתייחס אליהן. דרך טובה לראות אם למחשב שלכם מוגדר להראות סיומות היא על ידי ניסיון ליצור את הקובץ הנ"ל באחת מהשיטות. אם הקובץ מופיע בתור Chat, ולא Chat.java, הסיומות מוחבאות. איך לתקן?

- בחלונות: חפשו File Explorer במחשב ואז:



- במק: תחת Preferences > Finder לכו ל-Advanced וסמנו "Show all filename extensions".

פרק 4: הוראות הורדה והתקנה של IntelliJ IDEA

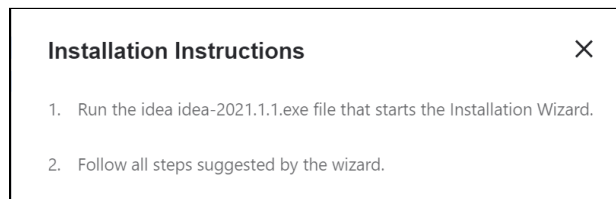
הורדה והתקנה

הגיע הזמן לעבור לעורך קוד של גדולים! ה-IDE שאיתו נעבוד נקרא [IntelliJ IDEA](#) מחברת JetBrains. ראשית, וודאו שיש לכם ג'אווה מותקנת על המחשב. אם לא, קפצו לפרק 1. אנחנו מחכים בינתיים.

הידעת? IDE (קיצור של Integrated Development Environment) או בעברית סביבת פיתוח משולבת, זה פשוט תוכנת עזר לפיתוח תוכנה. היא לרב כוללת עורך קוד, קומפיילר, דיבאגר ועוד הרבה כלים מגוונים שעוזרים לנו לעבוד בצורה יעילה יותר.

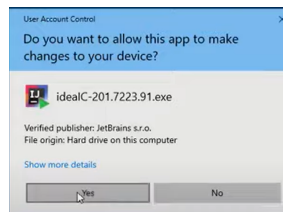
אנחנו הולכים להתקין את גרסת ה-community של IntelliJ, שהיא גרסת הקוד הפתוח החינמית, שכוללת את כל הפיצ'רים שנצטרך. (ולכל הסקרנים, מה היא לא כוללת? בעיקר תמיכה בשפות אחרות.)

באתר של JetBrains יש הוראות התקנה מפורטות מאוד:



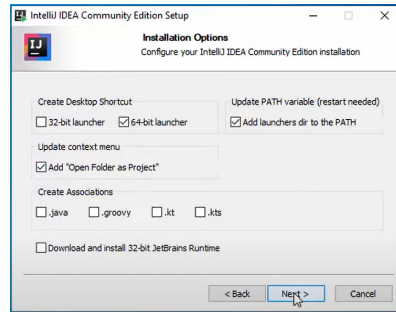
עם זאת, בואו בכל זאת נעבור על דברים יחידים.

1. לחצו על Download של גרסת ה-Community.
2. לאחר הורדת ה-exe, פתחו אותו.
3. בחלון הבא לחצו "כן":



4. כעת יפתח ה-Setup. מכאן עיקר התפקיד שלכם הוא ללחוץ Next, עם כמה הערות:

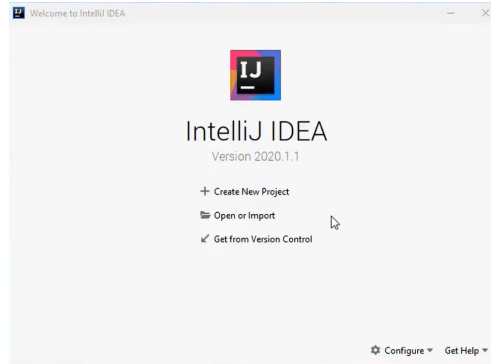
- החלון יציג לכם את תיקיית היעד שבה התוכנה תותקן. אין סיבה לשנות את זה. ואז Next.
- סמנו את הדברים הבאים. ואז Next.



- עוד חלון שלא צריך לגעת בו. ואז Next.
- 5. בשלב זה ההורדה צריכה להתחיל.
- 6. בסיום ההתקנה תוכלו לבחור Reboot now או Reboot manually later. אתם מוזמנים לבחור ב-Reboot manually later, להתרשם מהאייקון היפיפיה שנוצר לכם בשולחן העבודה, ואז לאתחל את המחשב בעצמכם.
- 7. לאחר האתחול, לחצו על האייקון בשולחן העבודה.
- 8. כרגיל, הסכימו לכל התנאים של JetBrains בלי לקרוא אף מילה.
- 9. חלון Data Sharing: לחצו על Don't Send. על אף ההחלטה הפזיזה שקיבלנו בסעיף הקודם, אולי בכל זאת כדאי לשמור קצת על פרטיות.
- 10. חלון בחירת UI Theme: בתור רופאי עיניים, אנחנו ממליצים לכם לבחור ב-Darcula, או מה שמכונה בקהילה "דארק מוד". זה גם יקל לכם על העיניים וגם יפחית את כמות האנשים שצוחקים עליכם שאתם לא עובדים בדארק מוד. ואז לחצו על Next.
- 11. חלון Tune IntelliJ IDEA to your tasks: לחצו על Next.
- 12. חלון Download featured plugins: לחצו על Start using IntelliJ IDEA. (תוכלו להתקין פלאגינים בעתיד, אם וכאשר תרצו.)

יצירת פרויקט

בסיום החלק הקודם של הפרק הייתם אמורים להגיע לכאן.



1. לחצו על Create New Project. בצד שמאל וודאו שהשפה המסומנת היא Java. תוכלו לראות ליד המילים Project SDK את גרסת הג'אווה שהתקנתם. נזכיר שוב **שיש להוריד לכל הפחות גרסה 10**, אחרת חלק מהקוד שנספק לכם פשוט לא יעבוד. לחצו על Next.
2. סמנו ב- ☒ את Create project from template. לחצו על Next.
3. תנו שם לפרויקט. שימו לב שבאופן דיפולטיבי, המיקום של הפרויקט שלכם יהיה בתיקייה IdeaProjects, בתוך תיקייה בשם הפרויקט. לחצו על Finish.
4. יפתח פרויקט עם קובץ Main שבו יש מחלקה Main עם פונקציית main ריקה. תוכלו לראות בצד את היררכיית התיקיות והקבצים שלכם.



זהו! צאו לדרך! יאללה, תתחילו לכתוב משהו כבר!

הגדרות נראות של IntelliJ

ניתן לשנות את הגדרות הפונט והנראות של IntelliJ כאן:

File → Settings → Appearance & Behaviour → Appearance

פרק 5: קיצורי מקשים של ה-IDE

אחד מהיתרונות הרבים של ה-IDE שנעבוד איתו הוא האפשרות של קיצורי מקשים שיזרזו וייעלו את תהליך כתיבת הקוד שלכם. (וגם חברים שלכם יחשבו שאתם קולים. עוד אומרים את זה בימינו?) למסמך קיצורי המקשים לחצו [כאן](#).

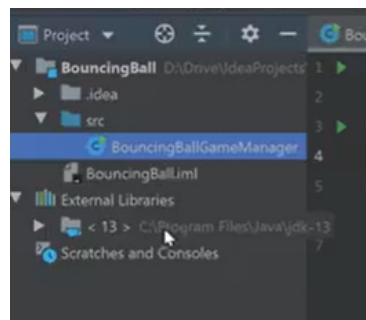
פרק 6: הוראות הורדה והתקנה של DanoGameLab

במסגרת תרגילי הקורס תצטרכו להשתמש בספריית צד שלישי בשם DanoGameLab, אותה ניתן להוריד [כאן](#). את סרטון ההתקנה המלא ניתן למצוא בתת פרק 4.4 (JavaX: Bricker), אבל בואו נעבור עליו בקצרה:

1. הורידו את קובץ ה-zip. הוא אמור להכיל קובץ jar, תיקיית src ותיקיית javadoc.
 2. פתחו תיקיה בשם DanoGameLab בתיקיית הפרויקטים (IdeaProjects) והכניסו אליה את תוכן ה-zip.
- כעת, ניצור פרויקט ונראה כיצד ניתן להגדיר בו תלות ב-DanoGameLab, פעם אחת בצורה טובה ופעם אחת בצורה לא טובה.

הדרך הנכונה: כיצד מגדירים תלות בספריה חיצונית?

1. כנסו ל-IntelliJ וצרו פרויקט חדש. שימו לב שב-External Libraries מופיעה רק גרסת הג'אווה שלכם.



2. הגיעו אל המקום שבו מגדירים תלויות: File → Project Structure → Modules → Dependencies.
3. לחצו על + ואז על JARs or directories.
4. נווטו אל התיקיה שיצרתם ל-DanoGameLab וסמנו גם את קובץ ה-jar וגם את תיקיית ה-src. (אפשר לסמן את שניהם על ידי לחיצה על Ctrl בעת הסימון.)

5. לחצו OK ואז OK שוב. עכשיו תחת External Libraries תוכלו לראות גם את הספרייה החדשה!

כדי לוודא שהתקנתם הכל כמו שצריך, ייבאו את התיקיה והריצו ב-main את השורה הבאה:

```
import danogl.GameManager;  
new GameManager().run();
```

השורה הזאת אמורה לפתוח חלון במסך מלא שאפשר לצאת ממנו עם Esc. (שימו לב שזאת שורה שונה מזו המוצגת בסרטון של דן במסגרת הקורס.)

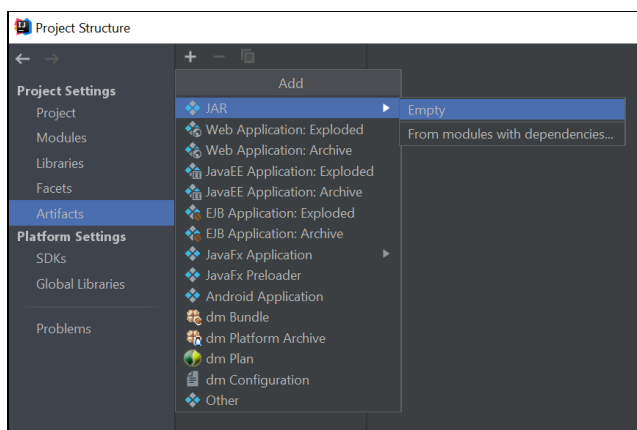
הדרך הלא נכונה: מה עושים אם הצורה הטובה לא עבדה? יש דרך נוספת. היא לא נכונה, אך טכנית היא תעבוד. העתיקו את התיקיה שבתוך src של DanoGameLab לתוך הפרויקט שלכם. במצב הזה אין פה בעצם תלות, אלא שזה כאילו שכתבתם את הקוד של DanoGameLab בעצמכם. הכל ירוץ כהלכה, אבל לא כך נרצה לעבוד.

פרק 7: הוראות ייצוא ל-jar

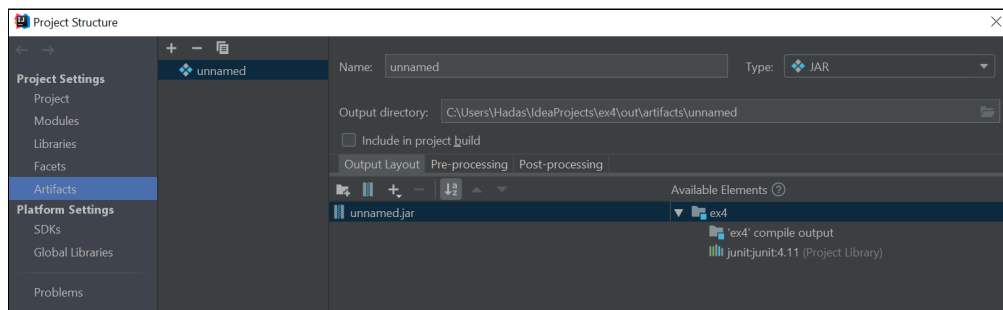
נניח שסיימנו לכתוב תכנית, או במקרה יותר ספציפי משחק במסגרת הקורס! איך נוכל לשלוח אותו לאמא שתגיד לנו איזה יפה יצא לנו? איך אמא תצליח להריץ אותו אם אין לה IDE? טוב ששאלתם.

נרצה לייצא את המשחק ל-jar משלנו. האפשרות הכי נקיה היא לכלול בתוך ה-jar גם את נכסי המשחק (assets). להלן ההוראות!

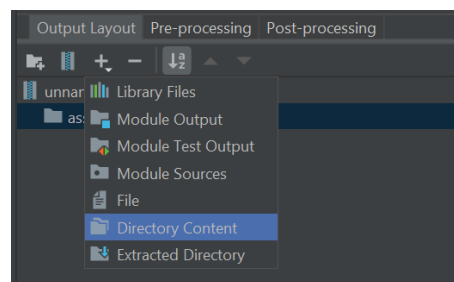
- File → Project Structure → Artifacts → +JAR from modules with dependencies → Choose main method → OK



- לאחר מכן, איפה שקודם לכן לא הופיע דבר (ולכן לחצתם על +) תופיע אפשרות נוספת. (בתמונה פה למטה הכל ריק כי אין תרגיל ברקע, אצלכם זה לא יהיה המצב.)



- שימו לב לתיקיית ה-Output. שם ייוצר קובץ ה-jar שלכם. סמנו ב- ✓ את Include in project build.
- לאחר מכן לחצו על האייקון ליצירת תקייה חדשה (תיקייה קטנה עם +). קראו לה assets. כעת, סמנו את assets ולחצו על ה-++ ואז על Directory Content, כמו בתמונה למטה. בחרו בתיקיית assets. לבסוף, לחצו OK.



לסיים:

- Build → Build Project

ה-jar ייוצר בתיקיית היעד שהוגדרה. בשביל להריץ אותו פשוט הפעילו אותו, או אם מסיבה כלשהי זה לא עובד, ניתן להריץ אותו מהטרמינל:

[navigate to the jar's directory using cd]

```
>>> java -jar myJar.jar
```

למקרה שלא הצלחתם לכלול את ה-assets בתוך ה-jar, ניתן ליצור jar גם בלעדיהם:

- File → Project Structure → Artifacts → +JAR from modules with dependencies → Choose main method → OK
- Note the output directory.
- Tick "include in project build"
- Click OK.
- Build → Build Project.

שימו לב: בשיטה הזאת ה-jar לא עומד בפני עצמו. הוא יחפש בסביבתו את נכסי המשחק, ולכן כשהוא מורץ, חייבת להיות לצדו התיקיה assets (ה-jar והתיקיה assets יושבים יחד באותה תיקיית אם).

ככל שזה נוגע לחברים (או לקוחות) שרוצים להריץ את המשחק שלכם: הם לא צריכים IntelliJ IDEA, אבל הם צריכים ג'אוה מותקנת. גרסת הג'אוה שלהם צריכה להיות לפחות הגרסה שבה יצרתם את ה-jar.

מה לגבי מחשבים שבהם ג'אוה לא מותקנת? עקרונית, ניתן לקמפל את הקוד שלנו גם לקוד מכונה באמצעות כלים חיצוניים, אבל כמובן שקוד המכונה הנ"ל ירוץ רק על מכונות מסוימות. זו לא שיטה נפוצה להפצה; הדרישה להתקין ג'אוה היא דרישה לגיטימית ונפוצה. (ומהי אמא ללא ג'אוה מותקנת על המחשב שלה, בעצם?)