[lecturer2024@outlook.com](mailto:lecturer2024@outlook.com)

מזה דבופס

סט פרקטיקות – אנחנו רוצים ש

מטרות?

כל הזמן להגביר את הדחיפות שמוציאים מוצר או פיצ'ר

הדחיפות שמשחררים את המוצר הרבה יותר מהר

יכולת לפתח עוד חלקים לאחר שמזרזים פיתוח יחידי ככה הכל נהיה יותר מהר

תהילך שחרור דפלוימנט

שחרור חלקים בצורה מהירה יותר, האיכות לאו דווקא נפגעת

האיכות השחרור תהיה גבוהה

איך אדם חדש נכנס לפוריקט גדול ולאחר כמה ימים מתחיל לרוץ על הפרויקט?

עקרונות:

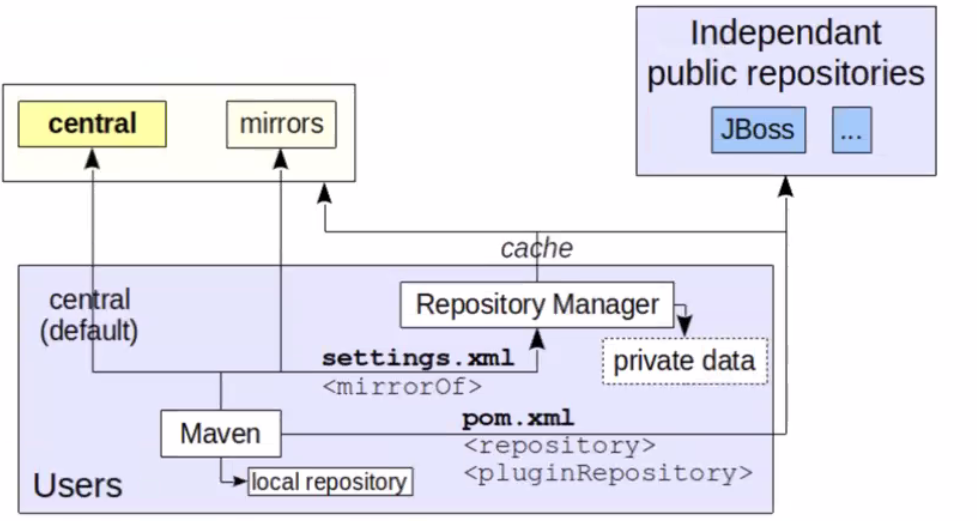
1. אוטומציה – פעולות בצורה אוטומטית:  
   כותבים קוד בתוכנה – הקוד הוא יישות חסרת חיים- לא נטען לשום מקום(אין משאבים)  
   כאשר טוענים קוד לזיכרון היא מקבלת משאבים ממערכת ההפעלה  
   יהיה לה מרחב זכרון משלה  
   מרגע כתיבת הקוד – ואז הקוד מקבל זמן ריצה  
   build - כאשר מריצים קוד, מה קורה?  
   JAVA – פיתוח בשפה לא מותאמת באופן ישיר למערכת הפעלה  
   היא רצה מעל מכונה וירטואלית  
   כותבים קוד בשפה עם מחלקות מובנות בשפה או שאנחנו יצרנו ומה שקורה בשלב הראשון הקוד שלנו עובר תהליך קומפילציה – הופך להיות קוד מותאם למערכת ההפעלה שלי,  
   קוד מקומפל למחשב וארכיטקטורה ספציפית.  
   בJAVA לאחר קומפילר עובר לביתקוד   
   אוטומציה לתהליך הבילד משלב יצירת הקוד לריצה שלה  
   סיומת ביטקוד .class ביקוד וריפייר בודק שהקוד תקין ובטוח  
   class loader מחפש את הקלאס וטוען לזיכרון  
   כל המחלקות שצריך יהיה קישור לפרויקט   
   המחלקה הראשית נעטנה עם כל המחלקות שקשורות אליה.  
   לבסוף הקוד רץ שורה אחר שורה  
   CD-CI שלב אוטומטי לעשות אינטגרציה ושינויים בקוד  
   לקחת את הקוד לאחר בדיקה ועכשיו אנחנו רוצים להנגיש אותו ללקוחות  
   GIT עוזר לנהל גרסאות  
   IFC   
     
   שחרור חלקים מהר יעיל ובטוח – נרצה כל הזמן שהכל תקין ובטוח  
   בין הצוותים פיתוח IT כולם אחראים למוצר יש תקשורת פתוחה, יש רצון ללמוד כל הזמן ללמוד ולהתנסות.
2. תכנון -  
   staging ?

**Maven**

לעשות תהליך הבילד בצורה אוטומטית, מאפשר באופן מובנה לבנות פרויקטים בJAVA ומגדיר את זמן חיים של התוכנה

קומפילציה – מעבירים לשלב ביטקוד שיכול להיטען לזכרון ולעלות את הקוד כניתן להרצה ומנונגש ללקוחות ואפשרי להשתמש בו

נותן לנו אפשרות לבצע מעקב אחרי גרסאות



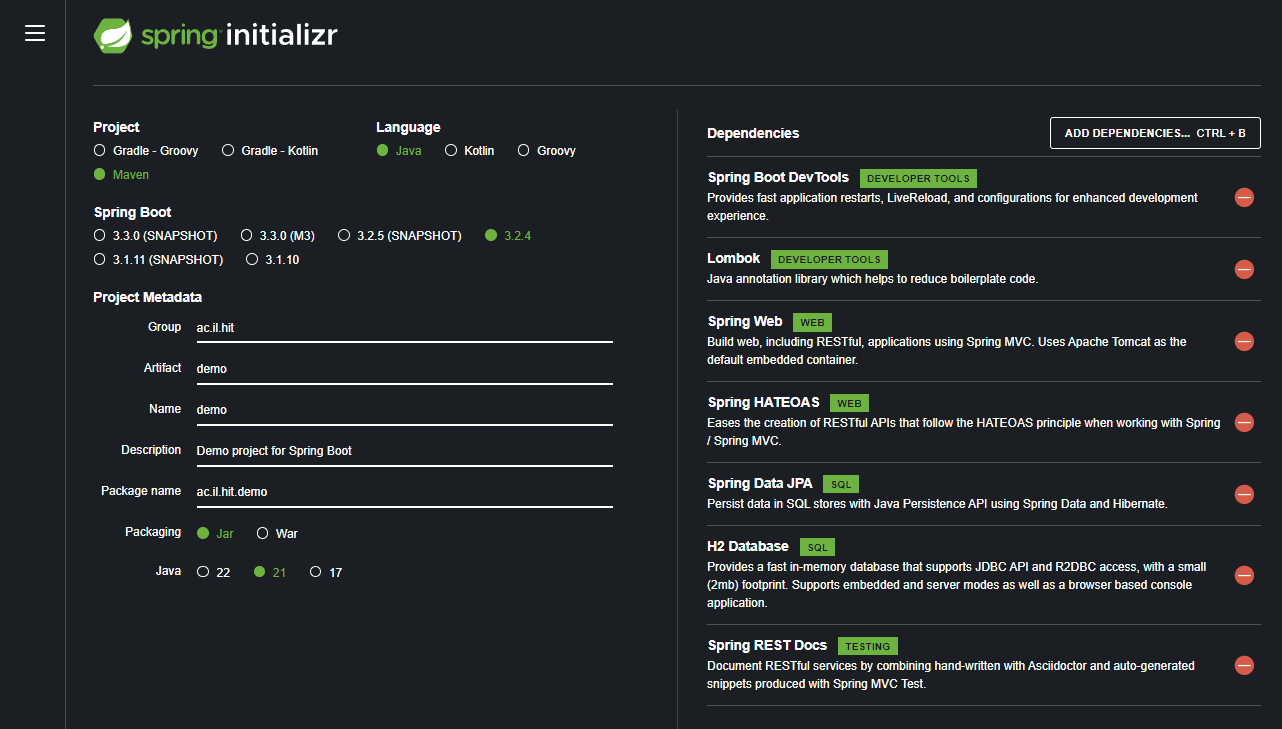
פום - הגדרת את כל התלויות

כלי תוכנה לנהל פרוקטים גדולים בJAVA

הקמת פרוקט מורכב יש אוסף סופי של מבנה עם התלויות שלו ואם איך שאנחנו עושים לו בילד

שבתכלס קורה בכל תוכנה ואפשר לקחת ולהתאים לצרכים

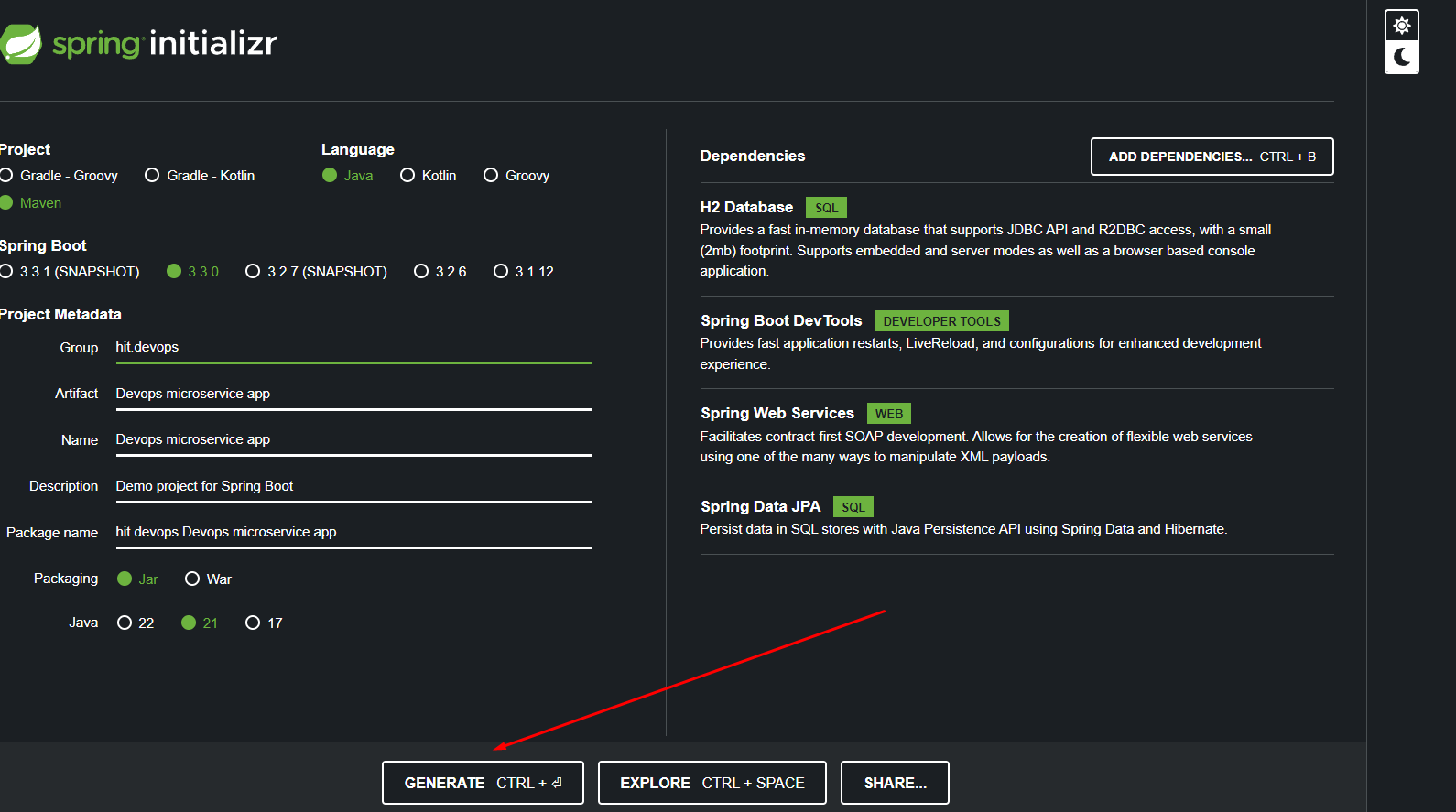
 מוצר מרכזי



[lecturer2024@outlook.com](mailto:lecturer2024@outlook.com)

<https://www.postman.com>

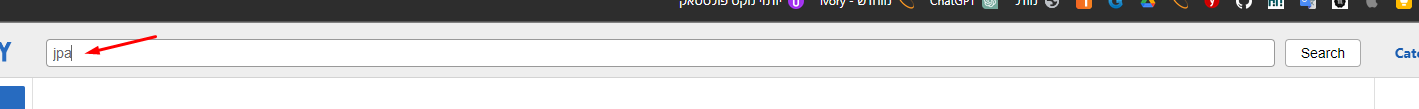
**פרויקט**



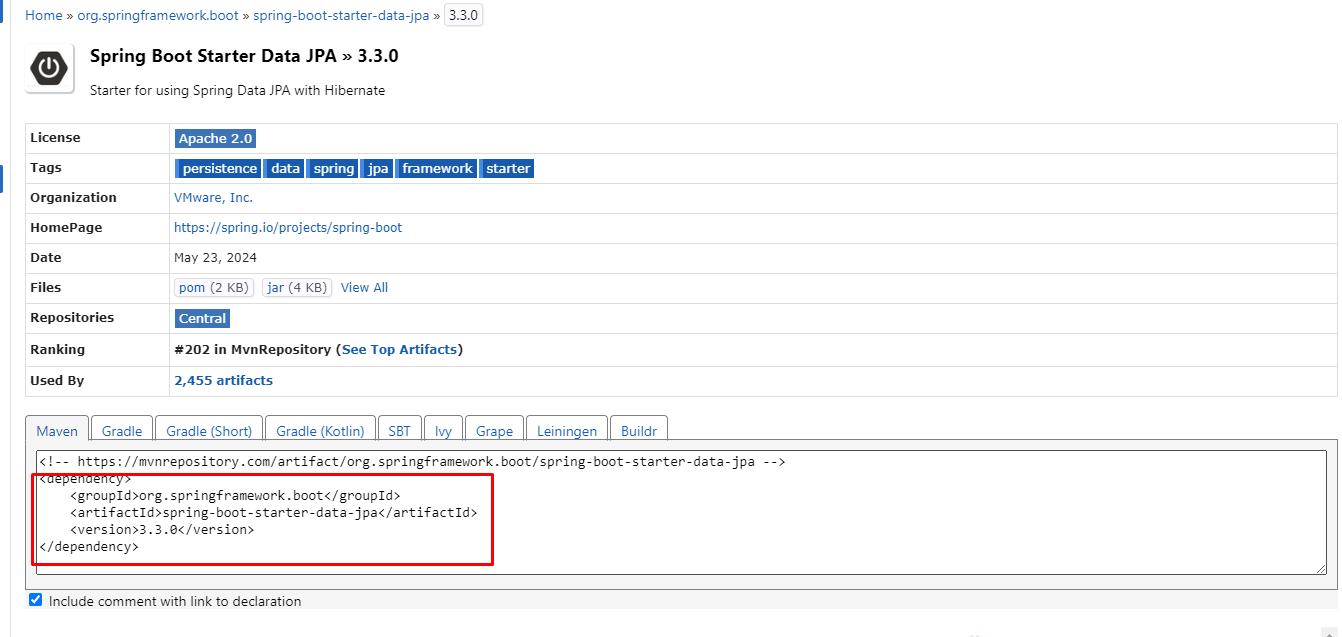
פותחים אינטלגיי – open ואז פותחים את הקובץ POM

במידה ואני רוצה להוסיף או תלות

נכנסים [לקישור](https://mvnrepository.com/repos/central)



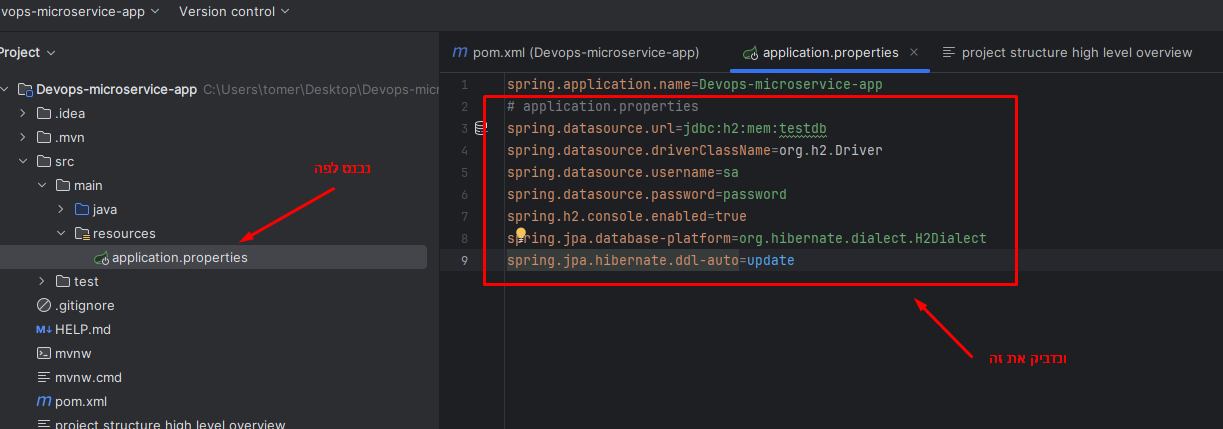
נכנס לגרסה שנרצה



מעתיקים ומכניסים לקובץ POM

זהו ייצוג טקסטואלי של הפרויקט – זה בהיי לבל מה שנרצה לעשות בפרויקט

זה האבני בניין לפרויקט המסכם.



# application.properties  
spring.datasource.url=jdbc:h2:mem:testdb  
spring.datasource.driverClassName=org.h2.Driver  
spring.datasource.username=sa  
spring.datasource.password=password  
spring.h2.console.enabled=true  
spring.jpa.database-platform=org.hibernate.dialect.H2Dialect  
spring.jpa.hibernate.ddl-auto=update

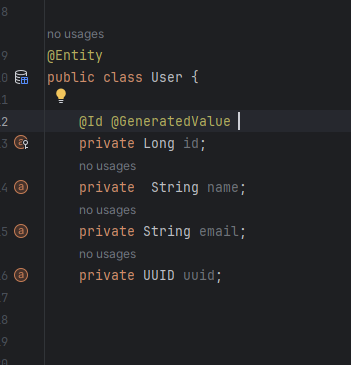
תמונה שמכילה טקסט, צילום מסך, תוכנה, תכונות מולטימדיה

התיאור נוצר באופן אוטומטי

מיועד לייצוג בעתיד כטבלה

השגיאה בUSER כי צריך להוסיף מפתח ראשי

נסמן את השדה שאותו נרצה לסמן ככזה



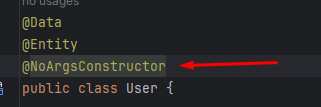
הג'נרט כדי לקדם את הID דרך ה-DB עצמו

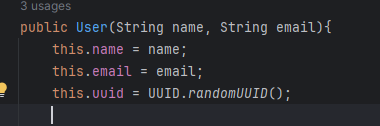
נוסיף לקובץ POME את התלות הזו

<dependency>  
  <groupId>org.projectlombok</groupId>  
  <artifactId>lombok</artifactId>  
  <version>1.18.30</version>  
  <scope>provided</scope>  
 </dependency>

ונעשה רענן – בזה בעצם הוספנו את לומבוק

וכך נוכל להוסיף את - > 

 מייצר קונסטרקטור אוטומטי



ה – UUID יוצר מזהה שנותן לי בעצם אפשרות מזהה חד חד ערכי של 128BIT

הוא בעצם התעודה הפנימית שתלך איתי לכל מקום לא משנה איזה DB נבחר.

שיעור 07/06/2024

הכנס לפרויקט – פתוחים את הספרינג ויוצרים קובץ לפי הנתונים מטה

