

## תרגיל להגשה - Spring Core + JPA

### הוראות:

1. לצורך ביצוע המשימה יש ברשותכם 3.5 שעות, לאחר סיום המשימה, הבדיקה תתבצע בצורה פרונטאלית עבור כל תלמיד בנפרד.
2. חומר עזר: אינטרנט
3. **חל איסור מוחלט לדבר ולהיעזר בחברכם לכיתה או בכל "חבר טלפוני".**
4. **חל איסור מוחלט ליצור קשר דרך האינטרנט בין חבריכם לכיתה או מול כל גורם אחר.**
5. המשימה מהווה מבחן!
6. התייחסו למשימה כחלק מראיון עבודה.
7. זכרו, לא רק התוצאה משקפת את היכולת והציון שלכם אלא גם הדרך, צורת הכתיבה, Best Practice סדר וארגון בקוד וכדומה.

במשימה זו נגדיר רכיב "Company" אשר מכיל עובדים (Employees). לכל עובד מספר משימות (Jobs) המטרה היא להגדיר רכיב Singleton בשם "Company" אשר משתמש ב-Repositories על מנת לנהל נתוני עובדים ומשימות ב-DB.

לשם כך, יש לבצע את השלבים הבאים:

#### 1. בניית פרוייקט

- הורדת פרוייקט Spring Boot עם ההרחבה Spring-JPA
- בנוסף יש להוריד Driver מתאים ל-DB שבשימוש או לעדכן מאוחר יותר בקובץ pom.xml
- יש לייבא את הפרוייקט ל-Eclipse
- יש לעדכן את application.properties על מנת להתחבר ל-DB הרצוי

#### 2. בניית Entity Beans ליצירת הטבלאות

- Job
  - id – long (PK)
  - description – String
  - endDate – java.sql.Date
- Employee
  - id – long (PK)
  - name – String
  - salary – double
  - jobs – List<Job> - OneToMany,
    - Cascade type: PERSIST
    - FetchType : EAGER

3. הגדרת Repositories

- ממשק EmployeeRepository
  - להוסיף מתודה לשליפת רשימת עובדים לפי name
- ממשק JobRepository
  - להוסיף מתודה לשליפת משימות לפי תאריך סיום
  - להוסיף מתודה לשליפת משימות בין 2 תאריכים

4. הגדרת רכיב Company (Singleton)

- רכיב זה משתמש ב-JobRepository & EmployeeRepository
- מאפשר את הפעולות הבאות:
  - הוספת עובד (עם משימות)
  - שליפת עובד לפי id
  - שליפת רשימת עובדים לפי name
  - שליפת כל העובדים
  - שליפת כל המשימות
  - שליפת משימות לפי תאריך סיום
  - שליפת משימות בין 2 תאריכים

5. יש לממש ב-Main יצירה של 2 עובדים עם 2 משימות וליישם עבודה עם מתודות של Company