

## Lab 7

אבישי עוז – 326489069

ליאון גורין - 214511214

שאלה 1:

(א) תפקיד ה-ORM (Object Relational Mapping) הוא למפות נתונים בין שפת תכנות מונחה עצמים לבין מסד נתונים יחסי. התפקיד העיקרי של ORM הוא לגשר על הפער בין המודל מונחה העצמים המשמש בקוד יישומים לבין המודל ההתייחסותי המשמש בבסיסי נתונים.

(ב) כמה מקרים שאנחנו יכולים לחשוב עליהם שרצוי להשתמש ב-SQL ולא ב-ORM הם:

- **פעולות מתקדמות במסד הנתונים:** ORM יכול להגביל את הגישה לפעולות מתקדמות במסד הנתונים או לאפשר אותן בצורה פחות אפקטיבית. אם יש צורך לבצע פעולות מורכבות, כמו טריגרים, פרוצדורות מאוחסנות, צפייה בתוצאות ישירות מהמסד ועוד, ייתכן שיהיה רצוי להשתמש בשפת SQL המקורית על מנת לנהל את הפעולות המתקדמות בצורה מיטבית.
- **ביצועים מיטביים:** במקרים שבהם ביצועים מיטביים הם גורם קריטי, ייתכן שכתובת שאילתות SQL ימצאו ביצועים טובים יותר מאשר ORM. בחלק מהמקרים, היישום של שאילתות SQL ישירות עשוי לאפשר למפתחים לבצע אופטימיזציות מתקדמות, ליצור שאילתות מותאמות במיוחד לדרישות המסוימות ולשדרג את ביצועי המערכת.
- **תצורה ותחזוקה מתקדמות של מסד הנתונים:** במקרים שבהם נדרשת תצורה מתקדמת של מסד הנתונים, כגון שינוי סכמה, יצירת אינדקסים מתקדמים, כתיבה ישירה לטבלאות מסוימות ועוד, SQL יכול להיות האמצעי הטוב ביותר לניהול ותחזוקת מסד הנתונים.

(ג) דוגמא לאנוטציה שאיננה מובנת ב-java היא @Table כמו שיש בקוד הנתון במעבדה, האנוטציה מגדירה את השם השורות בטבלה שבה אנחנו שומרים את המידע

## שאלה 2:

יחיד ליחיד (one to one) – קשר מסוג זה משמעו ששדה אחד בטבלה אחת, קשור לשדה אחד בלבד בטבלה האחרת, ולהיפך.

**לדוגמא:** לחברה יש בדיוק מנהל אחד, מנהל מנהל חברה אחת.

יחיד לרבים – קשר מסוג זה משמעו ששדה אחד בטבלה אחת קשור לשדות רבים בטבלה אחרת אך השדה מהטבלה האחרת קשור רק לשדה אחד מהטבלה הראשונה

**לדוגמא:** לחברה עובדים רבים, עובד מועסק בחברה אחת.

רבים לרבים – קשר מסוג זה משמעו ששדה אחד בטבלה אחת קשור לשדות רבים בטבלה אחרת ולהפך (או שלא יהיה לו קשר בכלל).

**דוגמא:** עובד יכול לעבוד עבור מספר מנהלים, למנהל יכולים להיות מספר עובדים. ייתכן שלמנהל לא יהיו עובדים כלל.