**JPA**

**תרגיל.**

**חלק א'- בסיס.**

**צרי 2 מחלקות שייוצרו עבורן טבלאות בDB:**

מחלקה בשם Address אשר מכילה נתונים של כתובת כמו: קוד רחוב, שם הרחוב, מס' בית, עיר וכו'. וכן שדה עבור קומה, עבורו יש ליצור enum של קומות.

הוסיפי פונקציות בסיסיות.

מחלקה בשם Person אשר מכילה שם, מספר זהות (String) , וכו', הוסיפי פונקציות בונות,get וset,

בצעי למחלקת address את המיפויים הבאים:

1. שם הטבלה יהיה addresses.
2. העמודה בטבלה של שם הרחוב יהיה: street\_name.
3. קוד הרחוב יהיה מפתח ראשי וייוצר באופן אוטומטי.
4. מספר בית יהיה שדה חובה.
5. בשדה מטיפוס enum יוכנסו הנתונים כמחרוזת
6. הוסיפי שדה שלא יישמר בטבלה

**חלק ב' – embedded.**

\* הוסיפי לפרויקט הנ"ל מחלקה נוספת בשם "UserCredentials" - משמעותה: פרטי זיהוי של הלקוח. המחלקה תכיל 2 שדות באמצעותן יזוהה הלקוח: שם וסיסמא.

\* הוסיפי במחלקת Person שדה נוסף מטיפוס מחלקת UserCredentials . דאגי שבעת שמירת הטבלה בDB הפרטים הללו יכנסו לטבלה.

\* הוסיפי שדה נוסף מטיפוס מחלקת UserCredentials שישמשו לזיהוי לקוח למטרה נוספת. (לדוגמא: פרטי זיהוי עבור גישה לחשבון הmail ופרטי זיהוי שונים עבור )

הוסיפי אנוטיישן. נכונים למניעת התנגשות שמות בשדות העמודות שבטבלה.

הגדירי אובייקטים מטיפוס Person מלאים בנתונים, בדקי את הטבלאות.

**חלק ג' – composite keys**

ברצוננו שהמפתח הראשי של טבלת Person יורכב משני שדות: 1.מספר זהות 2. קוד נוסף. בצעי את השנוי הנ"ל בשני הדרכים שלמדנו. (במחלקות נפרדות למניעת בלבול)

הגדירי אובייקטים מטיפוס Person מלאים בנתונים, בדקי את הטבלאות.

**חלק ד'- ירושה**

צרי 2 מחלקות שיורשות את מחלקת Person . הוסיפי למחלקה אחת שדה אחד נוסף ולמחלקה השניה 2 שדות נוספים. סמני את המחלקות כאנטיטי.

מעל מחלקת Person הוסיפי אנוט.. לקביעת אופן יצירת הטבלאות.

(@Inheritance(strategy= InheritanceType.TABLE\_PER\_CLASS))

צרי אובייקטים ועקבי אחר הטבלאות שנוצרות בכל אחת מ3 האפשרויות.

**חלק ה' – קשרים בין מחלקות**

**צרי מופע של address בתוך person קבעי ע"י אנוטישין את סוג הקשר בינהם.**

**צרי באחת המחלקות מופע של מחלקה אחרת כרצונך, קבעי סוג קשר.**

**בדקי את הטבלאות שנוצרו ואת הקשר בין הטבלאות.**

**חלק ו'- שאילתות**

**דאגי למלא טבלה מסוימת ב 5 רשומות לפחות. הפעילי על הטבלה מספר שאילתות**

**יש להשתמש בשאילתא find וב 4 שאילתות נוספות שונות. יש להפעיל חלק מהשאילתות עי הפונקציה createNamedQuary.**