

תשובות לתרגיל מסכם SQL

1. POCO הוא מחלקה שבה יש את שדות מידע לפי העמודות בטבלה במאגר נתונים. חשוב לממש equal בגלל שאנחנו רוצים ששני אובייקטים יהיו שווים ע"פ התוכן שלהם ולא על פי הכתובת שלהם כלומר אם אנחנו מושכים פעמים את אותו מידע מהמאגר אנחנו רוצים ששני האובייקטים בשני הפעמים יהיו שווים למרות שמדובר באובייקטים שונים שנמצאים בכתובת שונה בזיכרון. אנחנו ממשים hashCode מסיבות דומות- שאנחנו משתמשים במילון ומשתמשים באובייקט עם מידע שנשלף מהמאגר נתונים אנחנו רוצים ששני אובייקטים יחשבו אותו דבר לפי התוכן שלהם גם אם מדובר בשני אובייקטים שונים.

2. א. PK נותן לכל שורה בטבלה ערך ייחודי. בכל טבלה מותר שיהיה PK אחד הוא חייב להיות ייחודי (UNIQUE) PK יכול להיות מורכב שערך אחד או מספר ערכים לדוגמה

```
CREATE TABLE Persons (  
  ID int NOT NULL,  
  LastName varchar(255) NOT NULL,  
  FirstName varchar(255),  
  Age int,  
  CONSTRAINT PK_Person PRIMARY KEY (ID,LastName)  
);
```

ב. Foreign key מפתח מקשר בין טבלה אחת לטבלה שנייה. כל ערך בטבלה אחת חייבת להיות משויך לערך מסוים וייחודי בטבלה השנייה (אחרת לא יהיה אפשר לקשר כמו שצריך בין הטבלאות). מספר ערכים באותה טבלה יכולים להיות מקושרים לאותו ערך בטבלה השנייה.

ג. יחסים של 1:1 הם יחסים שכל ערך בטבלה אחת מקושר לערך אחר בטבלה השנייה- אין שני ערכים שמקושרים לאותו ערך ולהפך. יחסים N:1 הם יחסים שבהם מספר ערכים בטבלה אחת יכולים להיות מקושרים לאותו ערך בטבלה השנייה אך לכל ערך בטבלה הראשונה יש ערך אחד ובלעדי שהוא משויך אליו. כמו עובדים ומעסיק אותו מעסיק יכול להיות מקשור למספר עובדים אבל כל עובד למעסיק אחד. יחסים M:N הם יחסים שבהם מספר ערכים מטבלה אחת יכולים להיות משויכים לאותו ערך בטבלה השנייה כמו מוצרים ולקוחות כל לקוח יכול לקנות מספר מוצרים וכל מוצא יכול להקנות ע"י מספר לקוחות (יוצרים יחסים כאלה ע"י טבלה שלישית)

ד. unique זה כאשר כל ירך בעמודה מסוימת חייבל היות ייחודי אנחנו נגיד unique בMSSQL ע"י כתיבת unique ליד שם העמודה כאשר אחנו יוצרים את הטבלה או שנשתמש או

```
CREATE TABLE Persons (  
  ID int NOT NULL,  
  LastName varchar(255) NOT NULL,  
  FirstName varchar(255),  
  Age int,
```

`CONSTRAINT UC_Person UNIQUE (ID,LastName)`
);

3. כדאי להשתמש ב store procedure כי הם נשמרות על השרת וכשאנחנו מבצעים שאילתותו בקוד בC# השאליות מתבצעות על השרת ולא צריך לקבל את המידע מהמאגר ואז לעשות אליו את השאליתה (כלומר זה מוריד את תעבורת המידע בין השרת לתוכנה)

4. תפקיד ה connection string הוא לא לקשר אותנו כשאנחנו כותבים קוד בC# אל מאגר הנתונים צריך לכתוב אותה בconfiguration file

5. Inner Join - בוחר שורות שבם יש ערכים לעמודות בשורה בשתי הטבלאות

Full Outer Join - מחזיר את כל הערכים מטבלה הימנית והשמאלית- גם אלו שמתאימים וגם אלו שלא

Left Join - את כל השורות מטבלה שמאל ואת הערכים המתאימים להם מהטבלה בימין איפה שאין ערך מתים שם null

Right Join - את כל השורות מטבלה ימין ואת הערכים המתאימים להם מטבלה שמאל איפה שאין ערך מתאים שם null

6. Language-Integrated Query (LINQ) זוהי דרך לבצע פעולות על רשימות בדרך קצרה שבדרך כלל גורמת לקוד להיות קצר פשוט יותר ומסודר. יש שתי סוגים של LINQ :
method syntax - כתיבה בעזרת פונקציות
query syntax - כתיבה בצורה שמזכירה SQL

7. Logger הוא כלי להוסיף לוגים לתוכנים. כדאי להתשמש ב Logger כשעובדים עם מאגרי מידע מכיוון שככה אנחנו יכולים לבדוק בצורה נוחה אם השאליות שעשינו התבצעו כמו שצריך.

8. DR- זה אירוע שגורם לאיבוד מידע מהמאגר או לשיבוש משמעותי בשרת. אפשר להתמודד עם מצב כזה לדעתי ע"י גיבוי המידע על יותר משרת אחד.

9. אנחנו משתמשים ב partition by כדי לחלק את המידע לחלקים לבצע פעולה (כמו למשל ממוצע) על כל אחד מססהחלקים כדאי להתשתמש בו כאשר אנחנו לבצע פעולה כמו ממוצע או סכום ולבחור עמודות ולא להכניס אותם ל group by בנוסף אנחנו יכולים להשתמש ב partition by בשילוב של RowNumber וכך לתת מספרים לשורות בתוך התת קבוצה כך למשל אנחנו יכולים לבחור בקלות את סרטים הראשונים בכל שנה מטבלת סרטים אם תאריך יציאה (עושים partition by לפי השנה בשילוב של RowNumber ובחורים רק את מי שמקבל את מספר אחד- כלומר הראשון בתת קבוצה)