

מעבדה מס' 4

אפשרויות נוספות בשאילתה

מחקו את הטבלאות Account ו-Customer2 והריצו את הסקריפט insert.txt.

השוואת מחרוזות

ניתן גם לבצע השוואת מחרוזות בעזרת הסימנים % ו- (קו תחתון), כאשר % מייצג 0 תווים או יותר ו- _ מייצג תו בודד אחד. דוגמאות:

SELECT * FROM Customer2 WHERE Name LIKE 'Moshe _____'; רווח ו-5 קווים תחתונים אחרי משה;

הגרש מייצג תחילת מחרוזת והקומפיילר מחכה לגרש סוגר

SELECT * FROM Customer2 WHERE Name LIKE '%Azrieli';

SELECT * FROM Customer2 WHERE Name LIKE '%"%' ; **/// בשביל לחפש גרש אני צריך לרשום "**

כדי לייצג את הסימן % (או _) נכתוב:

SELECT * FROM Customer2 WHERE Name LIKE 'x%%x%' ESCAPE 'x';

נקבל את כל השמות שמתחילים ומסתיימים ב-%.

סידור פלט השאילתה:

SELECT * FROM Account WHERE AccNo<=100 ORDER BY Balance;

SELECT * FROM Account WHERE AccNo<=100 ORDER BY Balance, Type;

SELECT * FROM Account WHERE AccNo<=100 ORDER BY Balance DESC;

דרכים שונות להגדרת מפתח ראשי

נחזור לטבלת חשבונות בנק ונראה 4 דרכים (ישנן יותר) להגדרת מפתח ראשי:

```
CREATE TABLE Account (Accno INTEGER PRIMARY KEY NOT NULL,
                        Balance FLOAT,
                        Type CHAR(10));
```

```
CREATE TABLE Account (Accno INTEGER NOT NULL,
                        Balance FLOAT,
                        Type CHAR(10),
                        PRIMARY KEY (Accno, Balance));
```

```
CREATE TABLE Account (Accno INTEGER NOT NULL,
                        Balance FLOAT,
                        Type CHAR(10),
                        CONSTRAINT cc_num PRIMARY KEY (Accno, Balance));
```

אפשר להגדיר מפתח משולב

```
CREATE TABLE Account (Accno INTEGER NOT NULL,
                        Balance FLOAT,
                        Type CHAR(10));
```

ALTER TABLE Account ADD PRIMARY KEY (Accno); **// עבודה תוך כדי**

נשים לב כי הדרך הראשונה היא הפשוטה ביותר אך אינה מאפשרת מפתח של יותר משדה אחד. הדרך השנייה כן מאפשרת זאת. הדרך השלישית מאפשרת מפתח של יותר משדה אחד ונותנת שם לאילוץ המפתח הראשי, על מנת שנוכל לבטלו בהמשך. הדרך הרביעית מאפשרת הגדרת המפתח הראשי לאחר הגדרת הטבלה, וכמובן מאפשרת מפתח של יותר משדה אחד.

במהלך עבודתכם, בחרו לכם את השיטה המתאימה להגדרת המפתחות הראשיים.

שאלות של יותר מטבלה אחת

```
SELECT * FROM Emp;  
SELECT * FROM Dept;  
SELECT * FROM Emp, Dept;
```

בשאלות הבאות נשים לב כי יש לציין את שם הטבלה של השדה, היות ואותו שדה קיים בשתי הטבלאות:

```
SELECT * FROM Emp, Dept WHERE Emp.DeptNo=Dept.DeptNo;  
SELECT * FROM Emp, Dept WHERE Emp.DeptNo=Dept.DeptNo AND Emp.DeptNo=20;  
SELECT EmpNo, Dept.DeptNo, DName, Loc FROM Emp, Dept WHERE Emp.DeptNo=Dept.DeptNo;  
SELECT EmpNo, Dept.DeptNo, DName, Loc FROM Emp, Dept WHERE Emp.DeptNo=Dept.DeptNo  
AND EmpNo>=7900;
```

איחוד, חיתוך והפרש של שאלות

נכניס לסקריפט חדש בשם Operations.txt את רצף השורות הבא:

```
CREATE TABLE Table1 (Num1 INTEGER,  
                      Num2 INTEGER,  
                      Num3 INTEGER,  
                      PRIMARY KEY (Num1, Num2));
```

```
CREATE TABLE Table2 (Num1 INTEGER,  
                      Num2 INTEGER,  
                      Num3 INTEGER,  
                      PRIMARY KEY (Num1, Num2));
```

הוסיפו לשתי הטבלאות את השורות הבאות (סה"כ יש להוסיף 6 שורות):

			Table2		
Num1	Num2	Num3	Num1	Num2	Num3
1	4	3	1	3	5
1	1	5	1	4	3
1	10	11	1	10	12

כעת נבנה מס' שאלות על שני הטבלאות יחד:

```
(SELECT Num1, Num3 FROM Table1) INTERSECT (SELECT Num1, Num3 FROM Table2);  
(SELECT Num1, Num3 FROM Table1 WHERE Num3>4) INTERSECT (SELECT Num1, Num3 FROM  
Table2);  
(SELECT Num1, Num3 FROM Table1 WHERE Num3>4) INTERSECT (SELECT Num1, Num3 FROM  
Table2 WHERE Num2<0);  
(SELECT Num1, Num3 FROM Table1 WHERE Num3>4) UNION (SELECT Num1, Num3 FROM  
Table2);  
(SELECT Num1, Num3 FROM Table1 WHERE Num3>4) EXCEPT (SELECT Num1, Num3 FROM  
Table2);  
(SELECT Num2 AS Num1, Num3 FROM Table1) INTERSECT (SELECT Num1, Num3 FROM Table2);
```

5.2.6 Exercises for Section 5.2

Exercise 5.2.1: Using the database schema of our running movie example

```

Movie(title, year, length, inColor, studioName, producerC#)
StarsIn(movieTitle, movieYear, starName)
MovieStar(name, address, gender, birthdate)
MovieExec(name, address, cert#, netWorth)
Studio(name, address, presC#)

```

write the following queries in SQL.

- * a) Who were the male stars in *Terms of Endearment*?
- b) Which stars appeared in movies produced by MGM in 1995?
- c) Who is the president of MGM studios?
- *! d) Which movies are longer than *Gone With the Wind*?
- ! e) Which executives are worth more than Merv Griffin?

Exercise 5.2.2: Write the following queries, based on the database schema

```

Product(maker, model, type)
PC(model, speed, ram, hd, cd, price)
Laptop(model, speed, ram, hd, screen, price)
Printer(model, color, type, price)

```

of Exercise 4.1.1, and evaluate your queries using the data of that exercise.

- * a) Give the manufacturer and speed of laptops with a hard disk of at least one gigabyte.
- * b) Find the model number and price of all products (of any type) made by manufacturer *B*.
- c) Find those manufacturers that sell Laptops, but not PC's.
- ! d) Find those hard-disk sizes that occur in two or more PC's.
- ! e) Find those pairs of PC models that have both the same speed and RAM. A pair should be listed only once; e.g., list (i, j) but not (j, i) .
- !! f) Find those manufacturers of at least two different computers (PC's or laptops) with speeds of at least 133.

Exercise 5.2.3: Write the following queries, based on the database schema