**תרגיל לימוד בנושא SQL**

**Primary Key, Foreign Key, Auto Increment**

שפת SQL משמשת עבור מסדי נתונים שבהם הנתונים מאורגנים בטבלאות. לכל טבלה יש שם והיא בנויה מעמודות (columns), ושורות (rows) כאשר לכל עמודה יש שם וסוג.

**מינוח והגדרות:**

Table – טבלה

Column – עמודה

Record – רשומה – שורה של נתונים בטבלה

Field – שדה – נתון אחד (מתוך record כלומר מתוך שורה)

**מספור אוטומטי:**

בעמודה מטיפוס מספרי ניתן להגדיר מספור אוטומטי על-ידי התוספת:

GENERATED ALWAYS AS IDENTITY (START WITH 1, INCREMENT BY 1)

**אילוצים:**

Constraint – המילה constraint פירושה אילוץ. אילוצים בשפת SQL מציינים כללים עבור הנתונים בטבלה.

ניתן להחיל אילוצים שונים כפי שמוסבר להלן:

**מילים שמורות ב SQL המשמשות להגדרת אילוץ:**

NOT NULL – אילוץ זה קובע שהנתון בעמודה חייב להכיל ערך

UNIQUE – אילוץ זה קובע שהנתון יהיה ייחודי לעמודה (כלומר לא יופיע יותר מפעם אחת בעמודה)

PRIMARY KEY – אילוץ זה קובע שהנתון יהיה גם NOT NULL וגם UNIQUE. אילוץ זה משמש לזיהוי רשומה (שורה) בטבלה. טבלה יכולה להכיל רק primary key אחד שיכול להיות מורכב משדה אחד או משילוב של מספר שדות.

FOREIGN KEY – אילוץ זה קובע שהנתון משמש כ foreign key. Foreign key משמש לקשר בין שתי טבלאות. foreign key הוא שדה או אוסף שדות בטבלה אחת שמתייחס ל primary key בטבלה אחרת. הטבלה שמכילה foreign key מכונה child table ואילו הטבלה שמכילה את ה primary key אליו מתייחסים נקראת referenced table או parent table.

REFERENCES – מילה שמורה שמשמשת לציין לאיזה primary key לייחס את ה foreign key.

דוגמה מעבר לדף...

-- create a parent table with a primary key as auto increment

CREATE TABLE customers(

customer\_id INTEGER GENERATED ALWAYS AS IDENTITY (START WITH 1, INCREMENT BY 1),

customer\_name VARCHAR(20) UNIQUE NOT NULL,

customer\_age INTEGER NOT NULL,

PRIMARY KEY(customer\_id)

)

-- to insert records do not specify the auto generated field

INSERT INTO customers(customer\_name, customer\_ age) VALUES('Dan', 36)

/\* create a child table with a primary key as auto increment and foreign key which references the primary key in the parent table (customers) \*/

CREATE TABLE orders(

order\_id INTEGER GENERATED ALWAYS AS IDENTITY (START WITH 1, INCREMENT BY 1),

order\_number INTEGER NOT NULL,

customer\_id INTEGER,

PRIMARY KEY(order\_id),

FOREIGN KEY(customer\_id) REFERENCES customers(customer\_id)

)

-- to insert records do not specify the auto generated field

INSERT INTO orders (order\_number, customer\_id) VALUES(111, 2)

**קריאת רשומות ממספר טבלאות – משפט JOIN**

משפט JOIN משמש לשילוב רשומות (שורות) משתי טבלאות או יותר, בהתבסס על עמודה משותפת בינהן.

**INNER JOIN**

לבחירת רשומות שיש להן ערכים תואמים בשתי הטבלאות.

תחביר:

**SELECT column\_name(s)**

**FROM table1**

**INNER JOIN table2**

**ON table1.column\_name = table2.column\_name**

**=== for example ===**

**SELECT customers.customer\_id, customers.customer\_name, orders.\* FROM customers**

**INNER JOIN orders**

**ON customers.customer\_id=orders.customer\_id**

**LEFT JOIN**

לבחירת כל הרשומות מטבלה שמאל (1) וכן את הרשומות התואמות מטבלה ימין (2). התוצאה של טבלה ימין תהיה NULL אם אין ערכים תואמים.

תחביר:

**SELECT column\_name(s)**

**FROM table1**

**LEFT JOIN table2**

**ON table1.column\_name = table2.column\_name;**

**RIGHT JOIN**

לבחירת כל הרשומות מטבלה ימין (2) וכן את הרשומות התואמות מטבלה שמאל (1). התוצאה של טבלה שמאל תהיה NULL אם אין ערכים תואמים.

תחביר:

**SELECT column\_name(s)**

**FROM table1**

**RIGHT JOIN table2**

**ON table1.column\_name = table2.column\_name**