**SQL - 2023-05-18**

**SQL**

Structured Query Language

שפת שאילתות, המאפשרת לנו לבצע פעולות על המידע שבתוך מסד הנתונים – להביא את המידע, ליצור מידע (רשומות )חדש, להקים טבלאות, וכו.

**Export** – exporting your DB as an external sql file

**Import** – the same action, but the opposite

* כאשר אתם מגישים מבחן או פרויקט (או צריכים לבצע מטלה / מבחן בית) – אתם צריכים לשלוח לפעמים את קובץ ה DB.
* אני אעשה IMPORT לקובץ שהעליתם לי.
* את פקודות ואת מילות המפתח השמורות של SQL, נהוג לכתוב באותיות גדולות.
* SQL , הוא לא case sensitive

פקודת SELECT

* מחזירה מידע ממסד הנתונים
* היא לא משנה את המידע שבמסד נתונים

**SELECT** \* **FROM** {$tableName}

בחירת עמודות ספציפיות:

SELECT ${columnName} FROM ${tableName}

**SELECT** ProductName, ProductID, UnitPrice **FROM** products

**החזרת עמודות ספציפיות וכל הרשומות, אך מתן כינויים לעמודות.**

**SELECT** ProductID **AS** id, ProductName **AS** product\_name, UnitPrice **AS** price **FROM** products

**החזרת חלק מהרשומות – רק אלו שעונות על תנאי מסוים:**

**SELECT** \* **FROM** products **WHERE** UnitPrice > 50

AND

**SELECT** \* **FROM** products **WHERE** UnitPrice >= 10 **AND** UnitPrice <= 20;

OR

**SELECT** \* FROM products **WHERE** UnitPrice = 10 **OR** UnitPrice = 20;

NOT

**SELECT** \* **FROM** products **WHERE** **NOT** UnitPrice > 20;

מיון – סדר עולה

**SELECT** \* **FROM** products **ORDER** **BY** UnitPrice

מיון – סדר יורד

**SELECT** \* **FROM** products **ORDER** **BY** UnitPrice **DESC**

במקום:

**SELECT** \* **FROM** products **WHERE** UnitPrice >= 10 **AND** UnitPrice <= 20;

ניתן לעשות:

**SELECT** \* **FROM** products **WHERE** UnitPrice **BETWEEN** 10 **AND** 20;

IN

**SELECT** \* **FROM** products **WHERE** UnitPrice **IN** (10,20,30);

LIKE

החזרת ערך מדויק שעונה על דרישת החיפוש שלי

SELECT \* FROM products WHERE ProductName LIKE 'Chai'

החזרת ערך שמתחיל בדרישת החיפוש שלי

SELECT \* FROM products WHERE ProductName LIKE 'Ch%'

החזרת ערך שמסתיים בדרישת החיפוש שלי

SELECT \* FROM products WHERE ProductName LIKE '%Ch'

החזרת ערך שכולל (לא משנה באיזה מיקום בתוך המחרוזת)

SELECT \* FROM products WHERE ProductName LIKE '%Ch%'

­­

פונקציות סקלריות מתמטיות ( scalar - ערך בודד ) . כלומר – מכל ערך, היא מחזירה לי ערך בודד

* Floor – round to nearest lowest number
* CEIL – round to nearest highest number and remove decimals
* ROUND – by math lows

**SELECT** ProductName as product\_name , UnitPrice as price, floor(UnitPrice) as floorPrice, ceil(UnitPrice) as ceilPrice, round(UnitPrice) as roundPrice **FROM** products;

פונקציות סקלריות מחרוזתיות

SELECT ProductName, Lower(ProductName) AS lowerPName, upper(ProductName) AS upperPName, length(ProductName) AS productNameLength FROM products;

פונקציות סקלריות על תאריכים

SELECT FirstName, LastName, BirthDate, year(BirthDate) as year, date\_format(BirthDate,'%d/%m/%y') as birthDate from employees;

טיפול בערכי null

SELECT CompanyName, Fax FROM customers WHERE Fax IS NULL;

SELECT CompanyName, Fax FROM customers WHERE Fax IS NOT NULL

SELECT CompanyName , coalesce(Fax,'No Fax') from customers

DISTINCT

SELECT DISTINCT Country FROM customers

פונקציות Grouping

SELECT max(UnitPrice) FROM products;

SELECT min(UnitPrice) AS lowestPrice, Max(UnitPrice) AS highestPrice, AVG(UnitPrice) AS avaragePrice, count(UnitPrice), sum(UnitPrice) FROM products;

פונקציות grouping שניתן לבצע לפי קבוצות ע"י עמודה נוספת

SELECT CategoryID, avg(UnitPrice) from products GROUP BY CategoryID

SELECT Country, count(country) FROM customers GROUP BY Country

JOIN - פקודה המאפשרת להחזיר מידע ממספר טבלאות – כאשר קיים קשר בין הטבלאות .

**INNER JOIN -** החזרת רק הרשומות המקושרות בין שתי טבלאות .

SELECT ProductName, UnitPrice, CategoryName

FROM products JOIN categories

ON products.CategoryID = categories.CategoryID

LEFT JOIN

החזרת כל הרשומות מהטבלה שמשמאל ורק את התואמות להן מהטבלה שמימין.

SELECT ProductName, UnitPrice, CategoryName

FROM products LEFT JOIN categories

ON products.CategoryID = categories.CategoryID

RIGHT JOIN

החזרת כל הרשומות מטבלה שמימין ורק את התואמות להן מהטבלה שבמשאל

SELECT ProductName, UnitPrice, CategoryName

FROM products RIGHT JOIN categories

ON products.CategoryID = categories.CategoryID

ביצוע JOIN בין שלוש טבלאות

SELECT ProductName, UnitPrice, CategoryName, CompanyName

FROM products

JOIN categories ON products.CategoryID = categories.CategoryID

JOIN suppliers ON products.SupplierID = suppliers.SupplierID

מתן כינויים לטבלאות לצורך שימוש ב JOIN

SELECT ProductName , UnitPrice, CategoryName, P.CategoryID

FROM products AS P

JOIN categories AS C

ON P.CategoryID = C.CategoryID