**Зміст**

1. SELECT

2.DML (Data Manipulation Language)

* **INSERT**
* **UPDATE**
* **DELETE**

3. DDL (Data Definition Language)

* **CREATE TABLE**
* **ALTER TABLE**
* **DROP TABLE**

4. JOINS (Об'єднання)

* **INNER JOIN (ВНУТРІШНЄ об'єднання)**
* **LEFT JOIN (ЛІВЕ З'ЄДНАННЯ)**
* **RIGHT JOIN (ПРАВЕ З'ЄДНАННЯ)**
* **FULL OUTER JOIN (ПОВНЕ ЗОВНІШНЄ З'ЄДНАННЯ)**

5. IN та NOT IN

6. BETWEEN

7. ORDER BY Clause

8. GROUP BY

9. HAVING

10. Агрегатні функції - **Avg, Min, Max, Sum, Count**

11. =, >, AND, OR, NOT

12. Alias - AS

13. LIKE та NOT LIKE

14. Wildcard (символи)

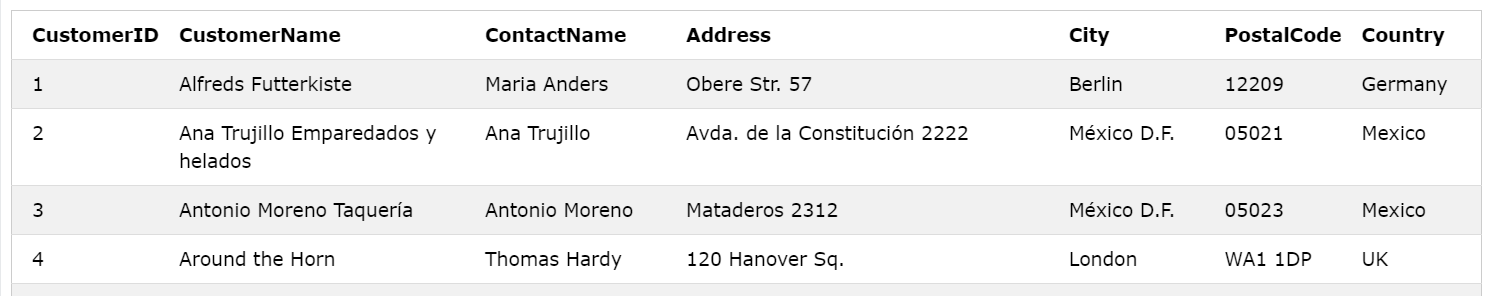
**SELECT**



**1. Вибрати всі стовпчики таблиці**

SELECT \*

FROM Customers;



**2. Вибрати стовпчики "CustomerName", "City", and "Country" із таблиці "Customer"**

SELECT CustomerName, City, Country

FROM Customers;



3. **Вибрати стовпчики "Country" із таблиці "Customer"**

SELECT Country

FROM Customers;



**4. Вибрати всі стовпчики з таблиці "Customer", де країна проживання Mexico**

SELECT \*

FROM Customers

WHERE Country='Mexico';



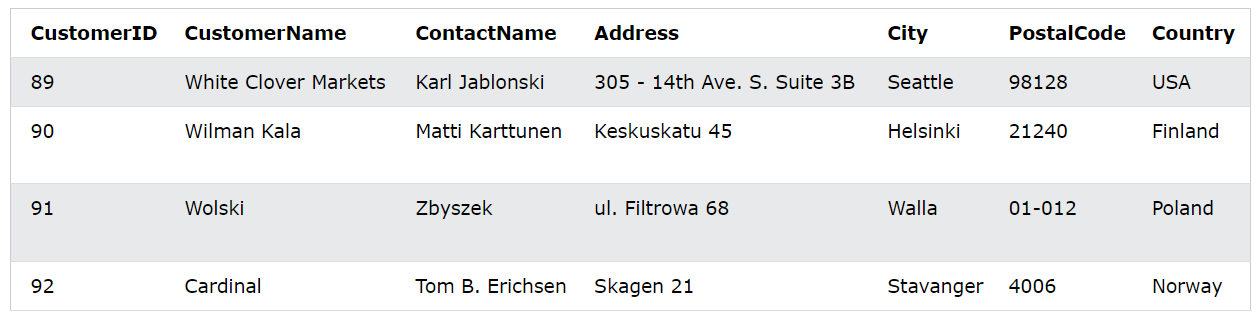
**DML (Data Manipulation Language)**

**INSERT, UPDATE, DELETE**

**INSERT**

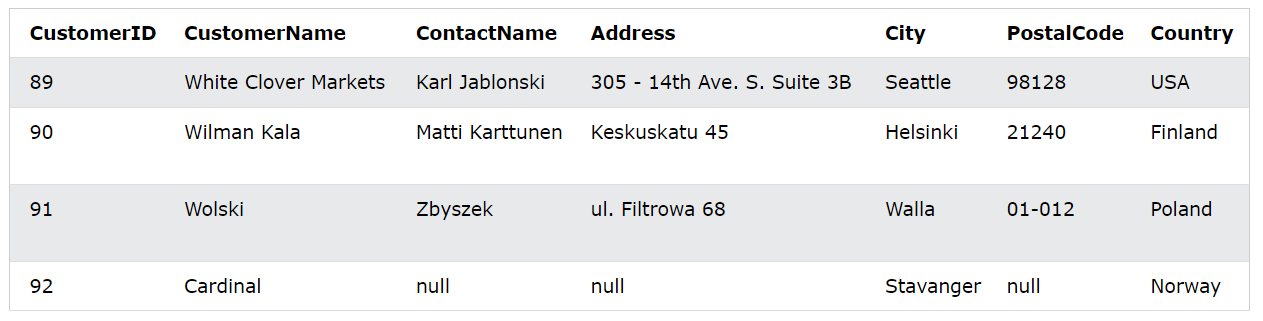
**1. Вставити новий запис в таблицю «Customers»**

INSERT INTO Customers (CustomerName, ContactName, Address, City, PostalCode, Country)  
 VALUES ('Cardinal', 'Tom B. Erichsen', 'Skagen 21', 'Stavanger', '4006', 'Norway');

****

**2. Вставити дані тільки в стовпці «CustomerName», «City» і «Country» в таблицю «Customers» (ідентифікатор клієнта буде оновлено автоматично)**

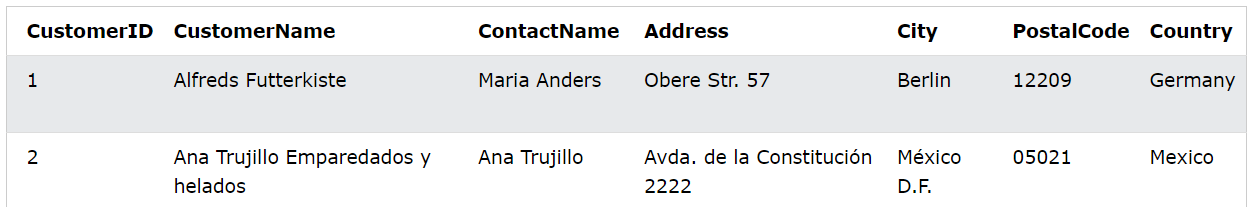
INSERT INTO Customers (CustomerName, City, Country)  
 VALUES ('Cardinal', 'Stavanger', 'Norway');

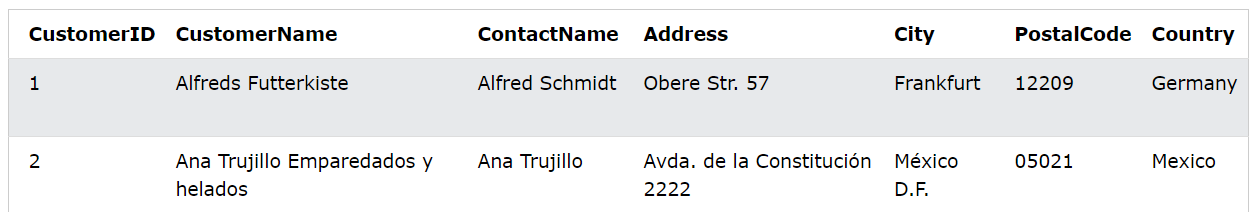
****

**UPDATE**

**1. Оновити дані клієнта з ID - 1, змінити ім'я і місто**

UPDATE Customers  
 SET ContactName = 'Alfred Schmidt', City = 'Frankfurt'  
 WHERE CustomerID = 1;

****

****

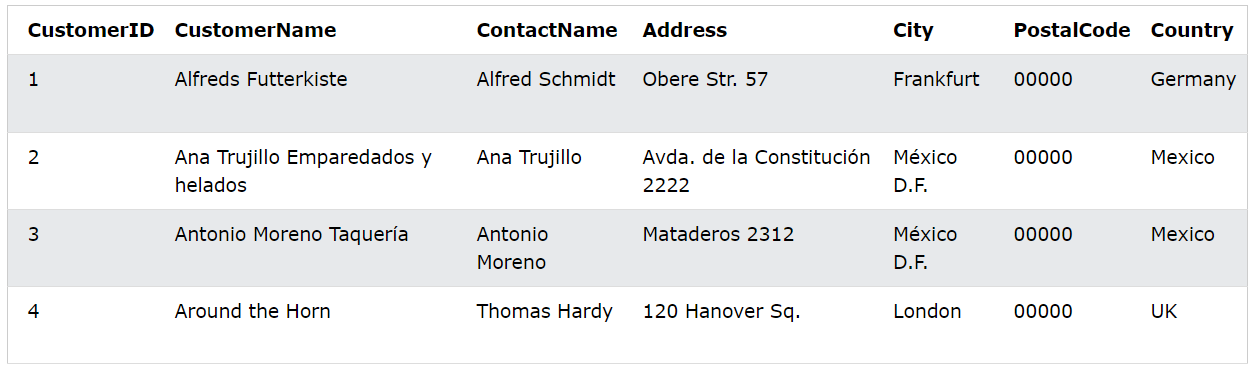
**2. Оновити PostalCode на 00000 для всіх записів, де країна «Мексика»**

UPDATE Customers  
 SET PostalCode = 00000  
 WHERE Country = 'Mexico';

****

**3. Оновити PostalCode на 00000 для всіх записів**

UPDATE Customers  
 SET PostalCode = 00000;

****

**DELETE**

**1. Видалити клієнта «Alfreds Futterkiste» з таблиці «Customers»**

DELETE FROM Customers

WHERE CustomerName='Alfreds Futterkiste';

**2. Видалити всі рядки в таблиці «Customers», не видаляючи саму таблицю**

DELETE FROM Customers;

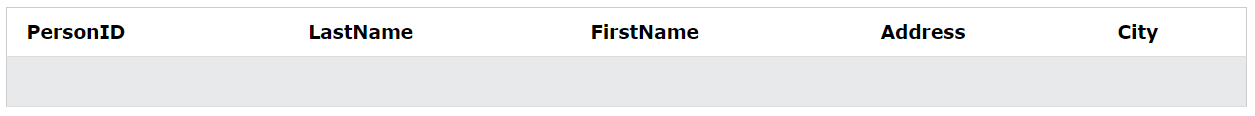
**DDL (Data Definition Language)**

CREATE TABLE, ALTER TABLE, DROP TABLE

**CREATE TABLE**

**1. Створити таблицю з назвою «Persons», яка містить п'ять стовпців: PersonID, Прізвище, Ім'я, Адреса та Місто:**

CREATE TABLE Persons (  
     PersonID int,  
    LastName varchar(255),  
     FirstName varchar(255),  
     Address varchar(255),  
    City varchar(255)  
 );

****

**ALTER TABLE**

**1. Додає стовпець «Email» до таблиці «Customers»**

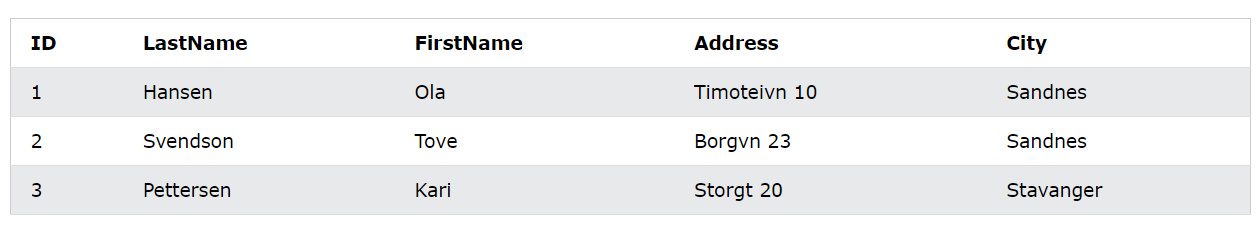
ALTER TABLE Customers  
 ADD Email varchar(255);

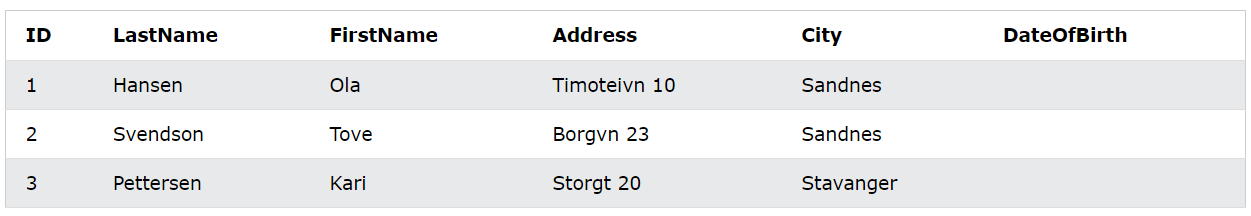
**2. видаляє стовпець «Email» з таблиці «Клієнти»**

ALTER TABLE Customers  
 DROP COLUMN Email;

**3. Додати стовпець з назвою «DateOfBirth» до таблиці «Persons»**

ALTER TABLE Persons  
 ADD DateOfBirth date;

****

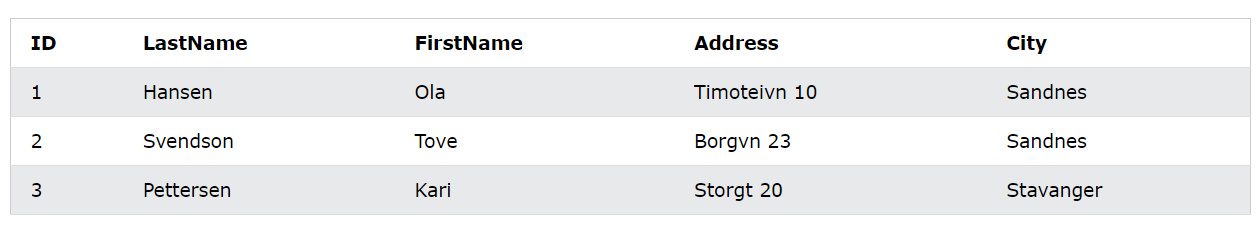
****

**4. Змінити тип даних стовпця з назвою «DateOfBirth» в таблиці «Persons»**

ALTER TABLE Persons  
 MODIFY COLUMN DateOfBirth year;

**5. Видалити стовпець з назвою «DateOfBirth» в таблиці «Persons»**

ALTER TABLE Persons  
 DROP COLUMN DateOfBirth;

****

**DROP TABLE**

**1. Видаляє існуючу таблицю «Persons»**

DROP TABLE Persons;

**JOINS**

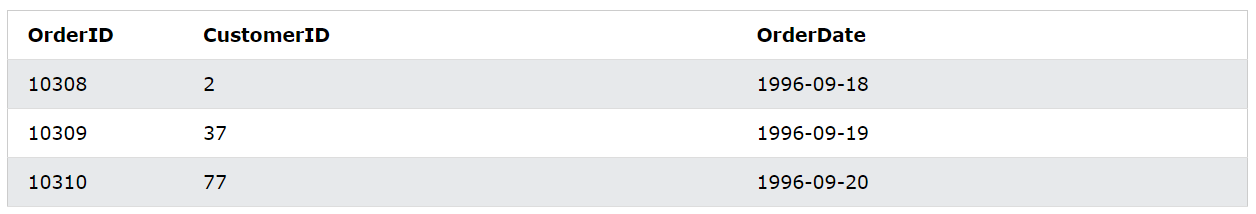
INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL OUTER JOIN

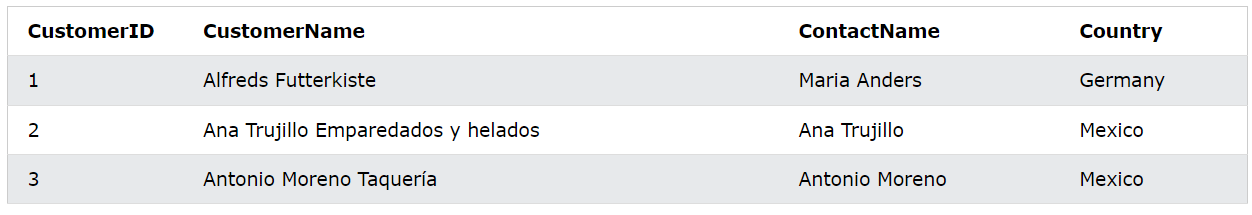
**INNER JOIN**

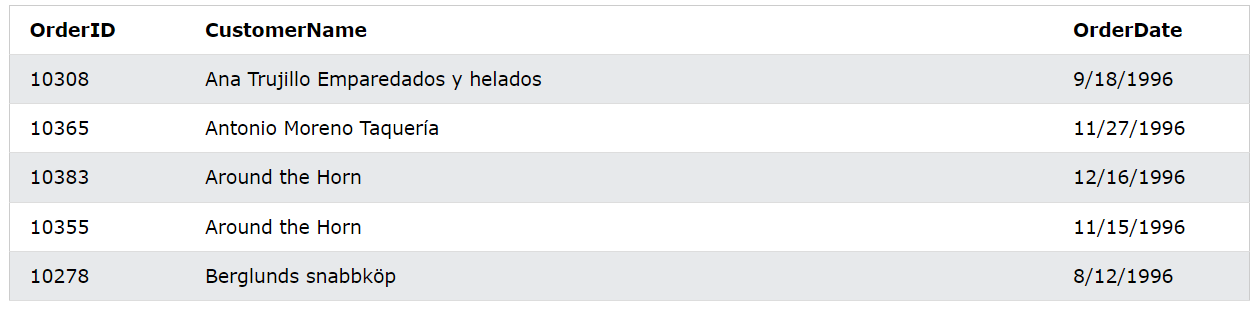
**1. Вибрати записи, що мають однакові значення в обох таблицях**

SELECT Orders.OrderID, Customers.CustomerName, Orders.OrderDate  
 FROM Orders  
 INNER JOIN Customers

ON Orders.CustomerID=Customers.CustomerID;

****

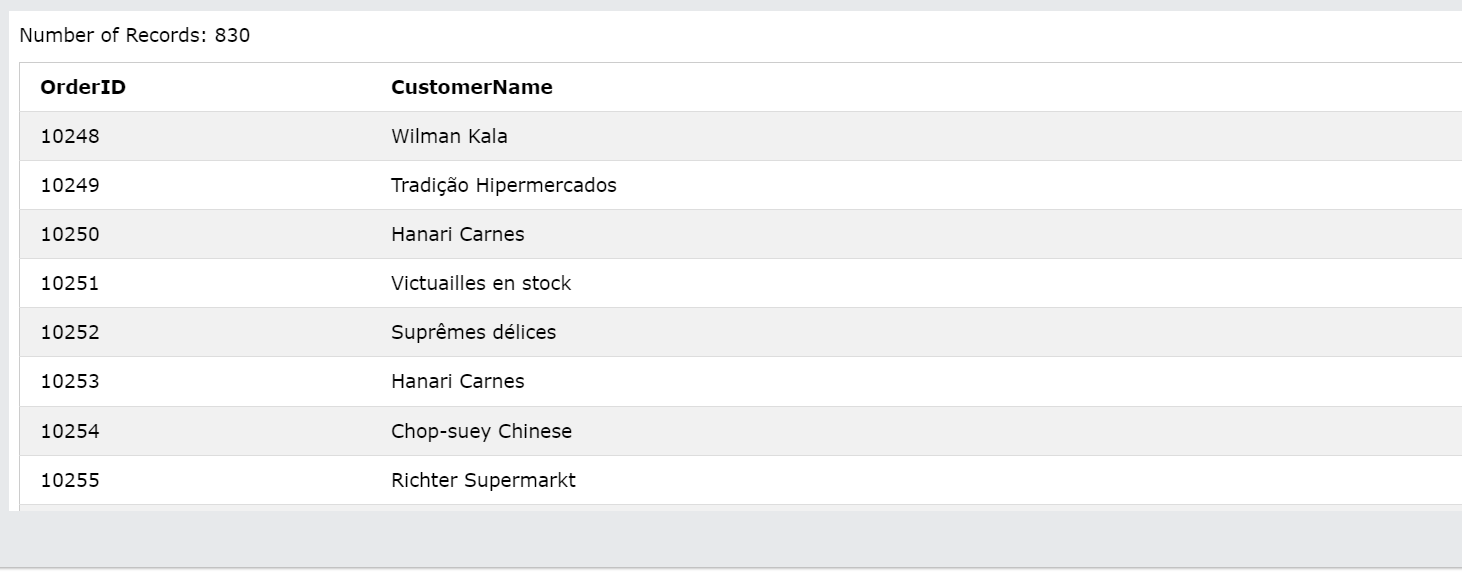
****

****

**2. Вибрати всі замовлення з інформацією про клієнта**

SELECT Orders.OrderID, Customers.CustomerName  
FROM Orders  
INNER JOIN Customers

ON Orders.CustomerID = Customers.CustomerID;

****

**3. Вибрати всі замовлення з інформацією про клієнта і вантажовідправника**

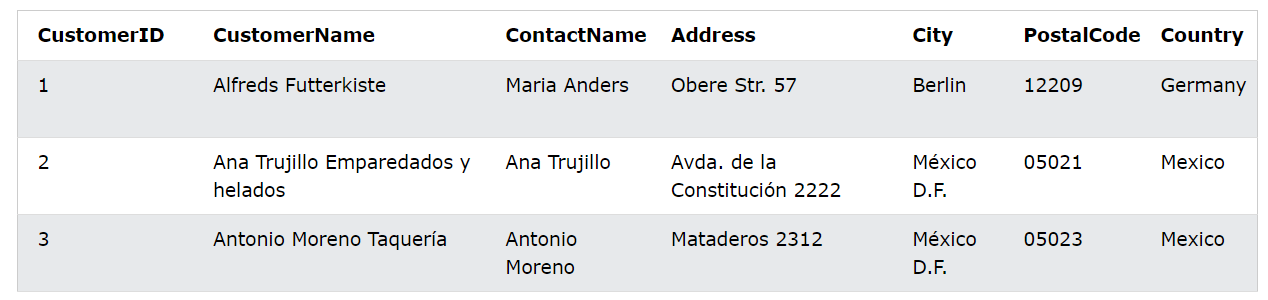
SELECT Orders.OrderID, Customers.CustomerName, Shippers.ShipperName  
 FROM ((Orders  
 INNER JOIN Customers ON Orders.CustomerID = Customers.CustomerID)  
 INNER JOIN Shippers ON Orders.ShipperID = Shippers.ShipperID);

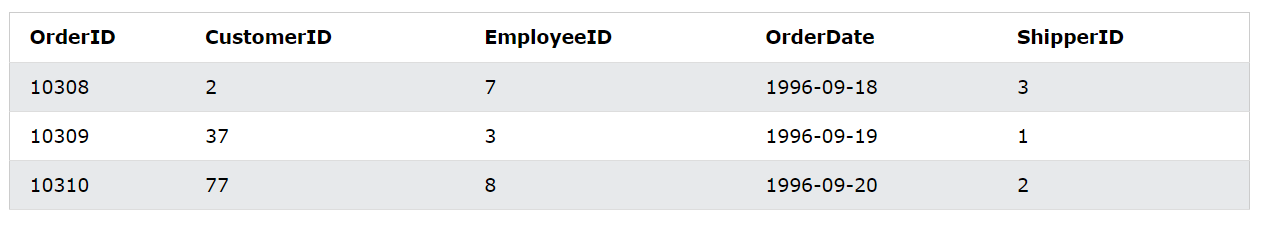
****

**LEFT JOIN**

**1. Вибра всіх клієнтів і всі замовлення, які вони можуть мати**

SELECT Customers.CustomerName, Orders.OrderID  
 FROM Customers  
 LEFT JOIN Orders ON Customers.CustomerID = Orders.CustomerID  
 ORDER BY Customers.CustomerName;

****

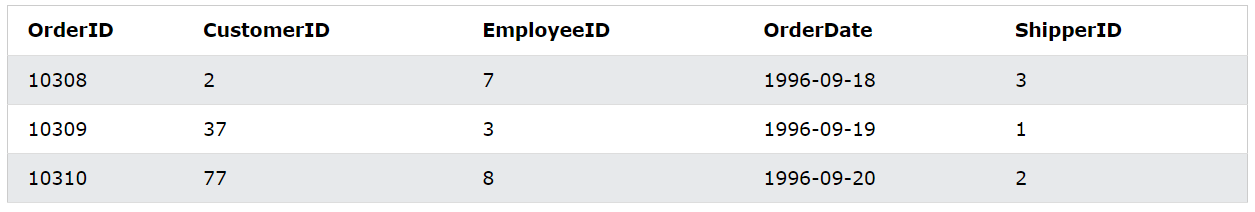
****

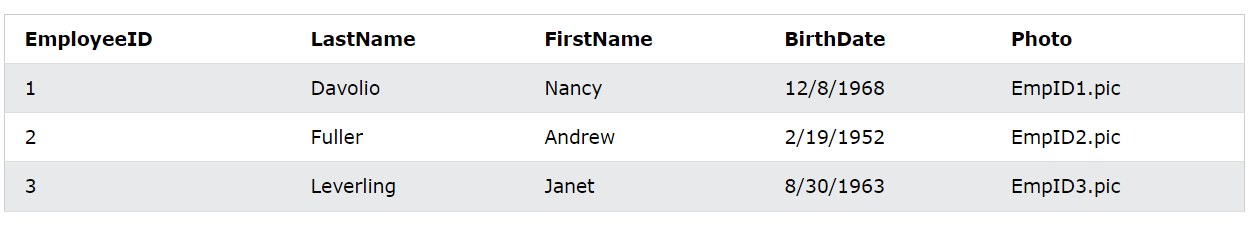
****

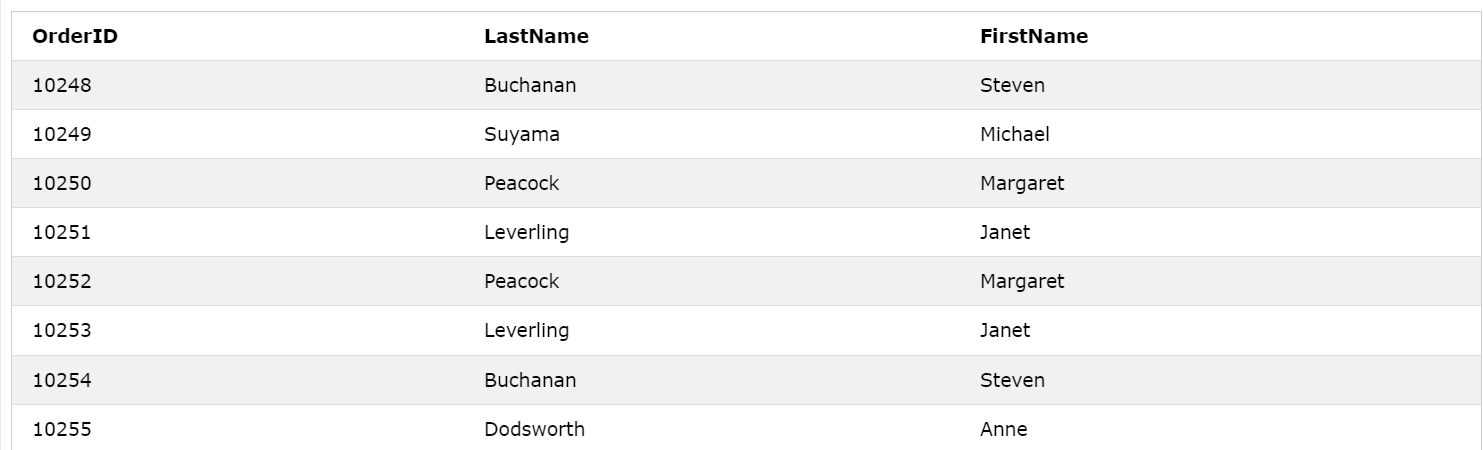
**RIGHT JOIN**

**1. Вибрати всіх співробітників і всі замовлення, які вони могли зробити**

SELECT Orders.OrderID, Employees.LastName, Employees.FirstName  
 FROM Orders  
 RIGHT JOIN Employees ON Orders.EmployeeID = Employees.EmployeeID  
 ORDER BY Orders.OrderID;

****

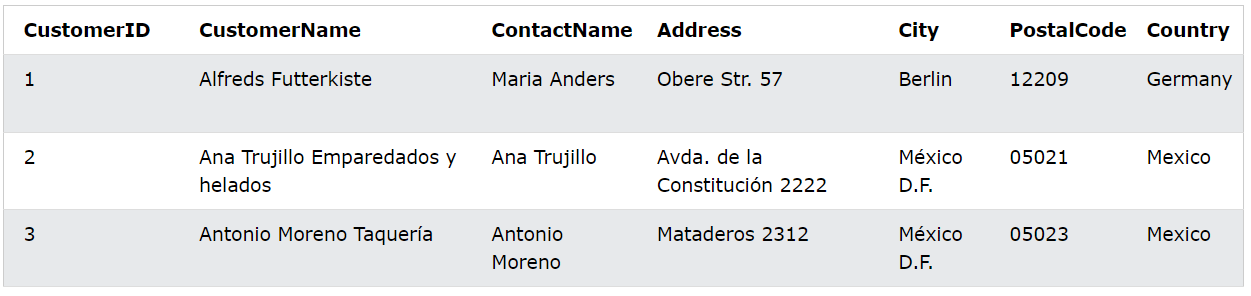
****

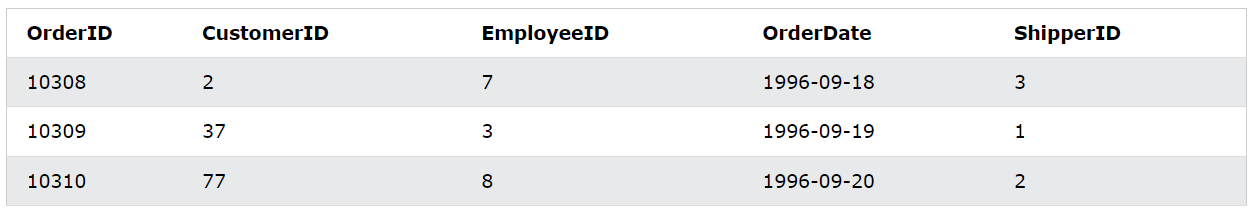
****

**FULL OUTER JOIN**

**1. Вибрати всіх клієнтів і всі замовлення**

SELECT Customers.CustomerName, Orders.OrderID  
 FROM Customers  
 CROSS JOIN Orders;







**2.** **Вибрати всіх клієнтів і всі замовлення**

SELECT Customers.CustomerName, Orders.OrderID  
 FROM Customers  
 CROSS JOIN Orders  
 WHERE Customers.CustomerID=Orders.CustomerID;



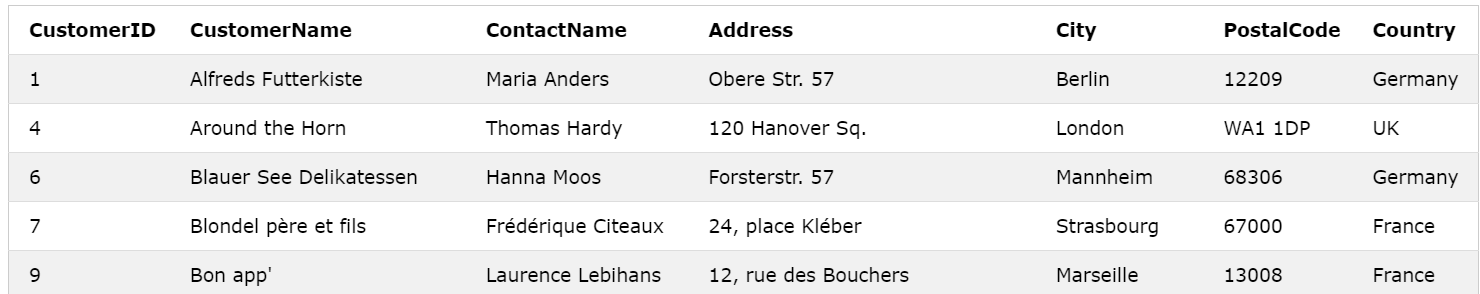
**IN та NOT IN**

**1. Вибрати всіх клієнтів, які знаходяться в «Germany», «France» або «UK»**

SELECT \*

FROM Customers  
 WHERE Country IN ('Germany', 'France', 'UK');

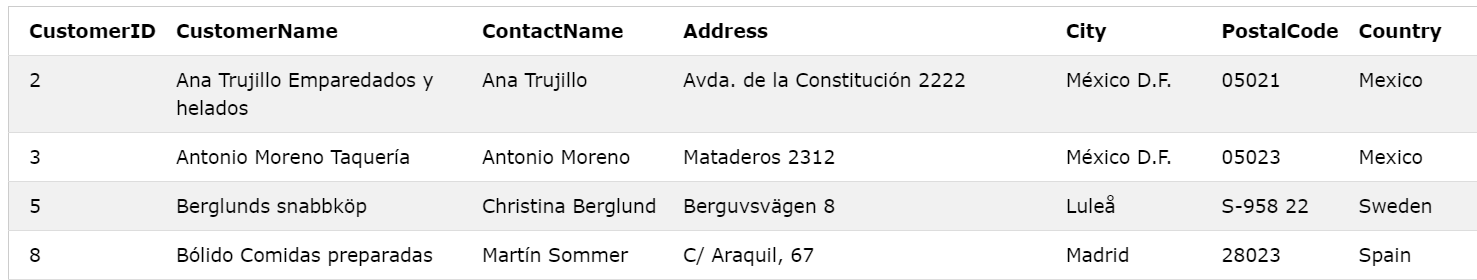




**2. Вибрати всіх клієнтів, які не знаходяться в «Germany», «France» або «UK»**

SELECT \*

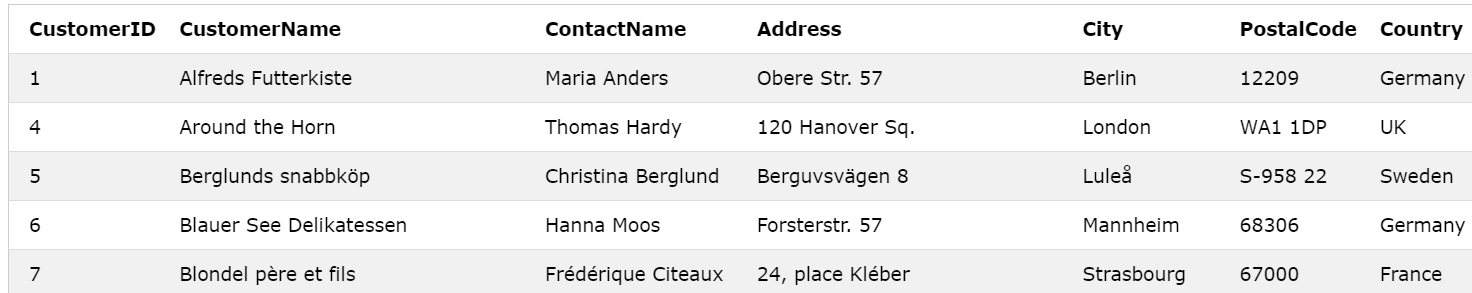
FROM Customers  
 WHERE Country NOT IN ('Germany', 'France', 'UK');



**3. Вибрати всіх клієнтів з тих самих країн, що й постачальники**

SELECT \*

FROM Customers  
 WHERE Country IN (SELECT Country FROM Suppliers);

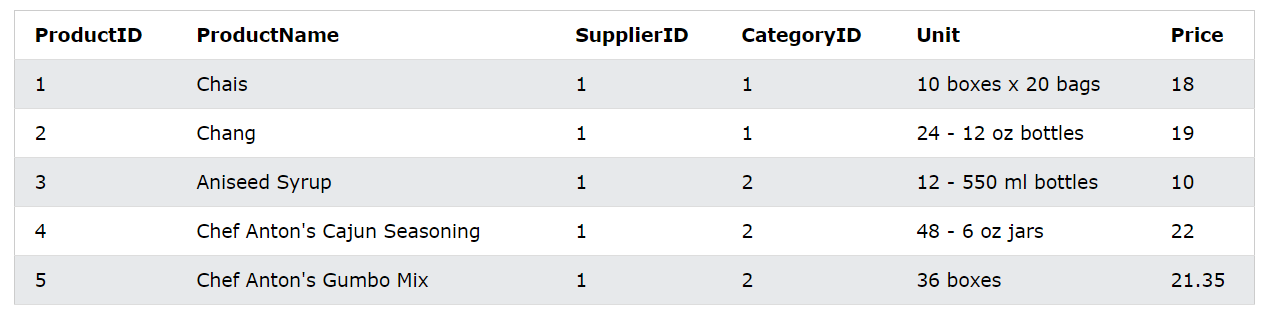


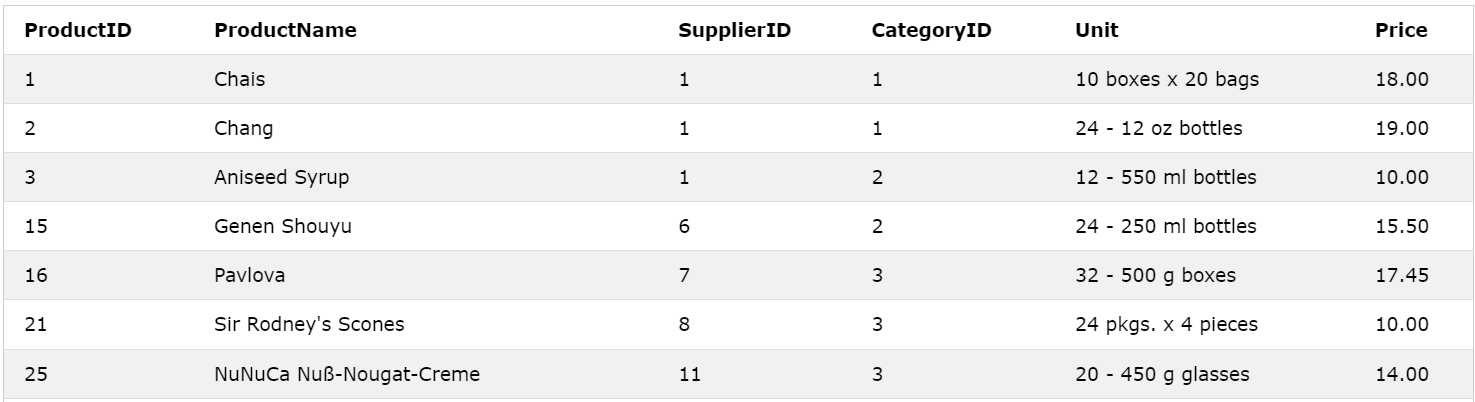
**BETWEEN**

**1. Вибрати всі товари з ціною від 10 до 20**

SELECT \*

FROM Products  
 WHERE Price BETWEEN 10 AND 20;

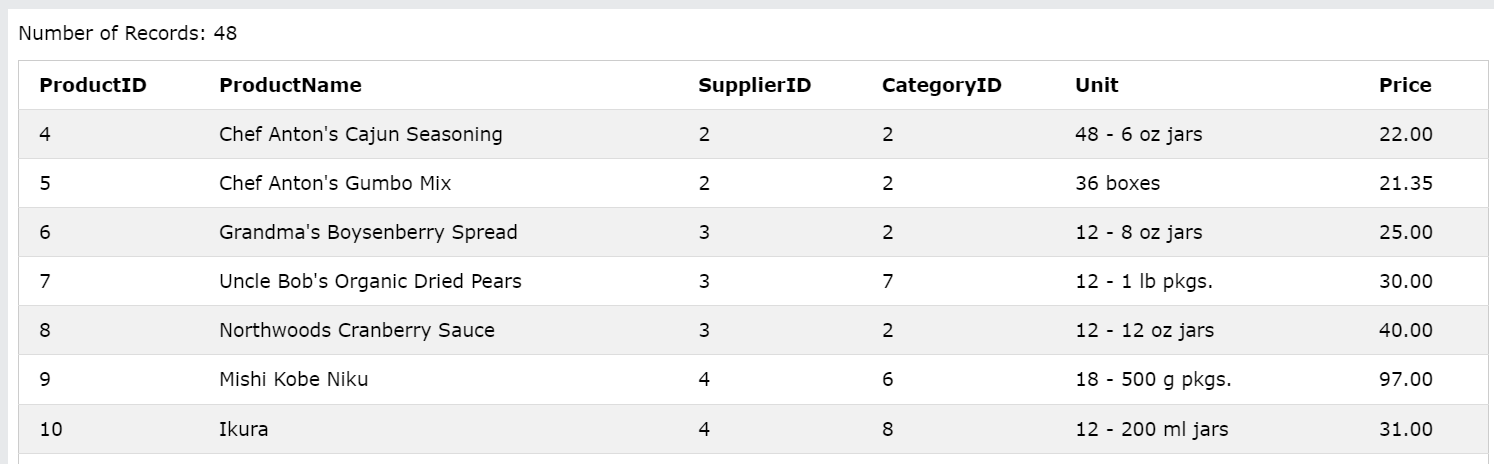




**2.** **Вибрати всі товари не з ціною від 10 до 20**

SELECT \*

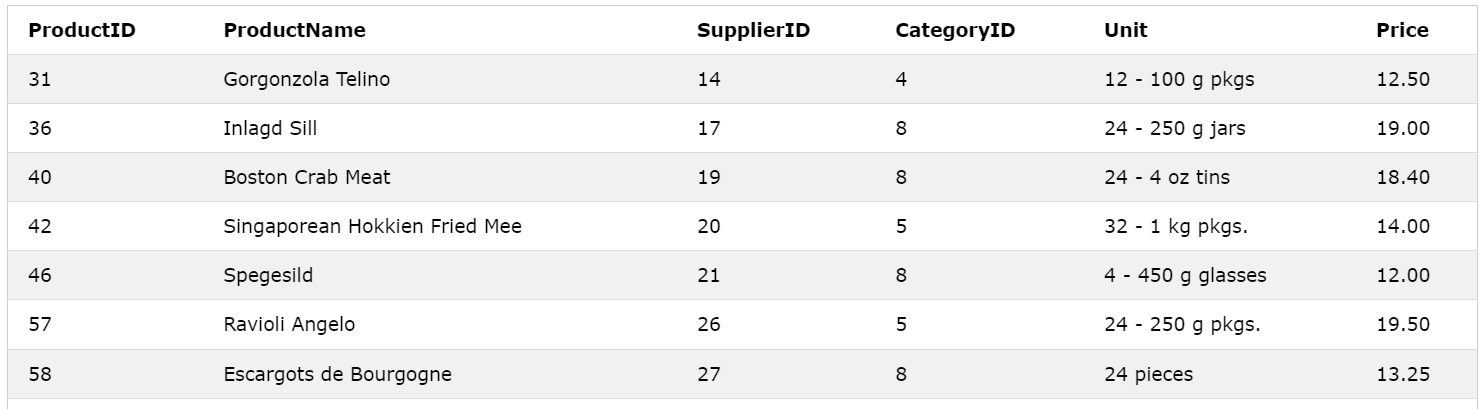
FROM Products  
 WHERE Price NOT BETWEEN 10 AND 20;



**3. Вибати всі товари з ціною від 10 до 20. Крім того, не показувати товари з CategoryID 1,2 або 3**

SELECT \*

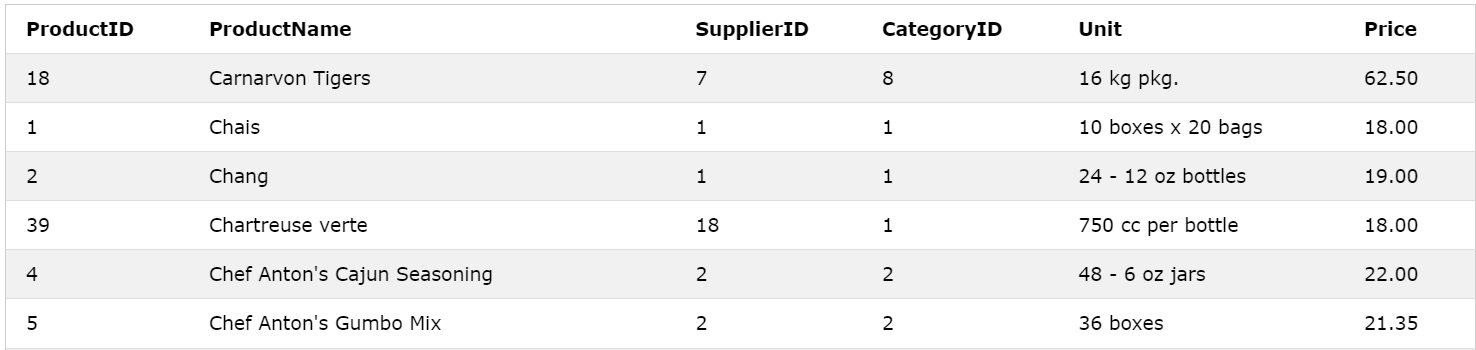
FROM Products  
 WHERE Price BETWEEN 10 AND 20  
 AND CategoryID NOT IN (1,2,3);



**4. Вибрати всі продукти з ProductName між «Carnarvon Tigers» та «Mozzarella di Giovanni»**

SELECT \*

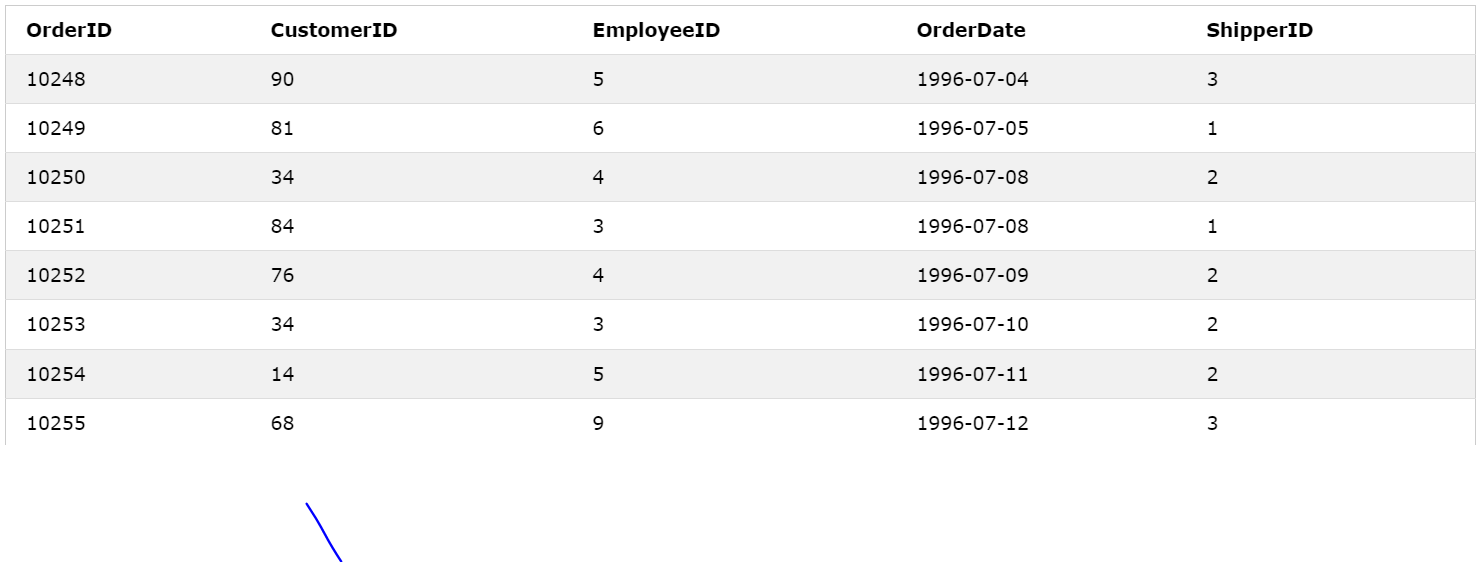
FROM Products  
 WHERE ProductName BETWEEN 'Carnarvon Tigers' AND 'Mozzarella di Giovanni'  
 ORDER BY ProductName;



**5. Вибрати всі замовлення з датою OrderDate між '01-July-1996' та '31-July-1996'**

SELECT \*

FROM Orders  
WHERE OrderDate BETWEEN '1996-07-01' AND '1996-07-31';



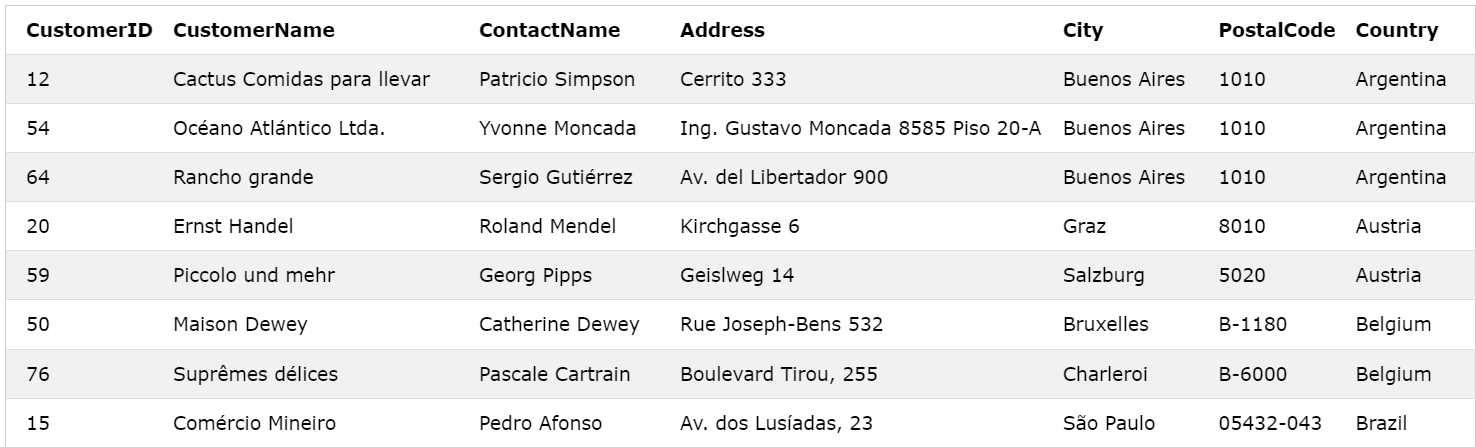
**ORDER BY Clause**

**1. Вибратиє всіх клієнтів з таблиці «Customers», відсортованих за стовпцем «Country»**

SELECT \*

FROM Customers  
 ORDER BY Country;

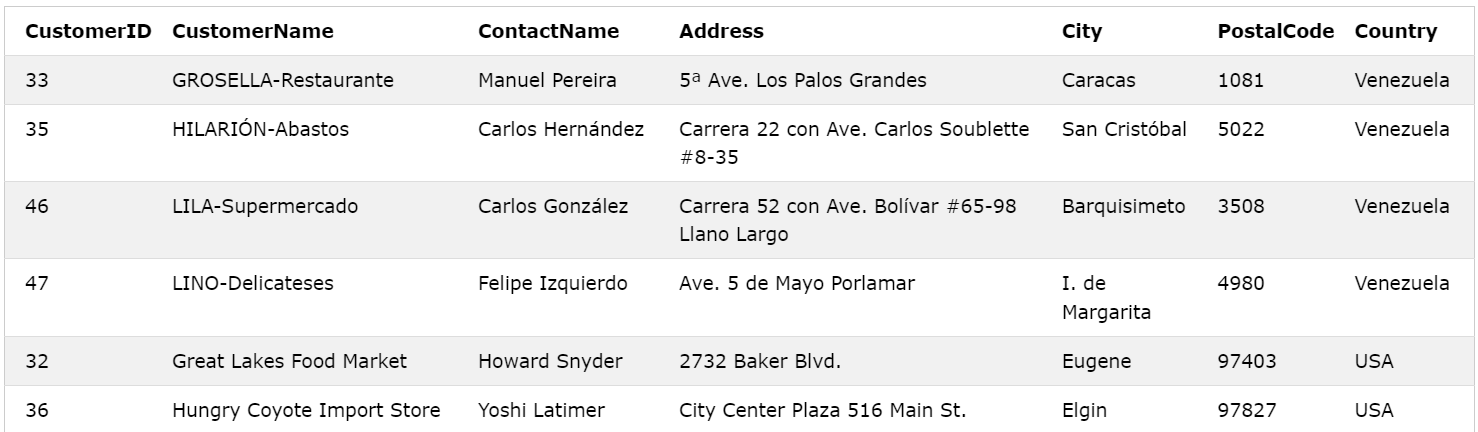




**2. Вибрати всіх клієнтів з таблиці «Customers», відсортованих за спаданням за стовпцем «Country»**

SELECT \*

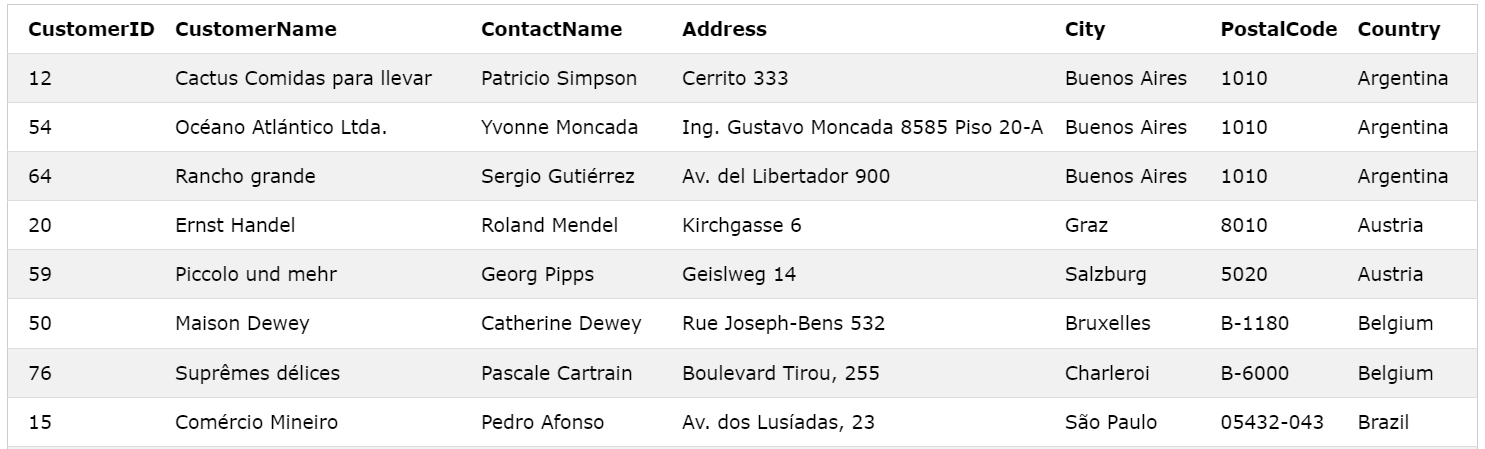
FROM Customers  
 ORDER BY Country DESC;



**3. Вибрати всіх клієнтів з таблиці «Customers», відсортованих за стовпцем «Country» та «CustomerName». Це означає, що він впорядковує за країною, але якщо деякі рядки мають однакову країну, він впорядковує їх за іменем клієнта**

SELECT \*

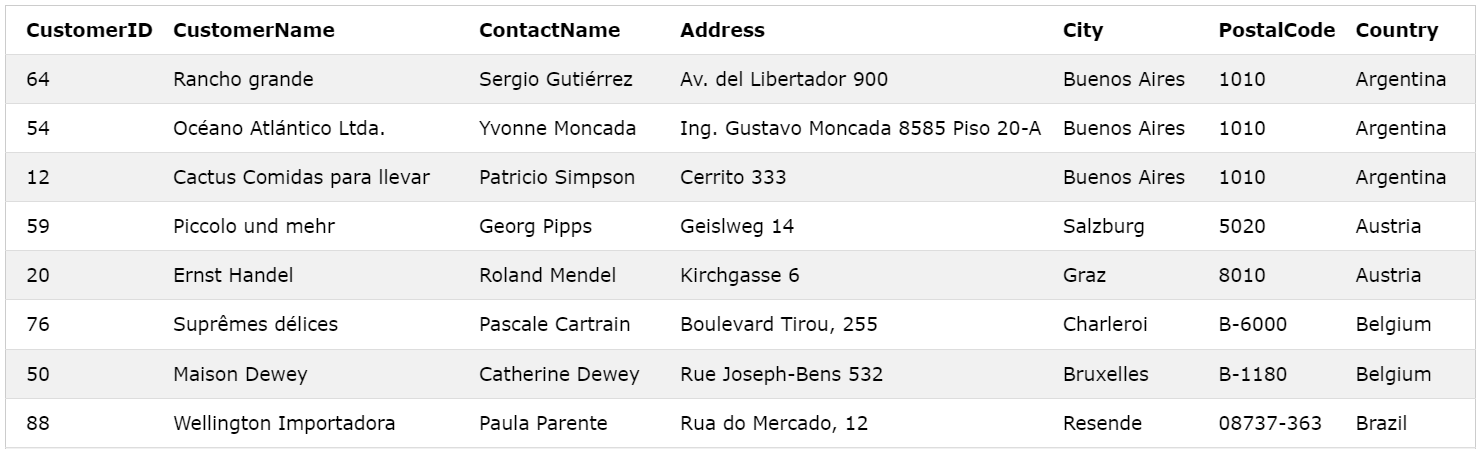
FROM Customers  
 ORDER BY Country, CustomerName;



**4. Вибрати всіх клієнтів з таблиці «Customers», відсортованих за зростанням за стовпцем «Country» і за спаданням за стовпцем «CustomerName»**

SELECT \*

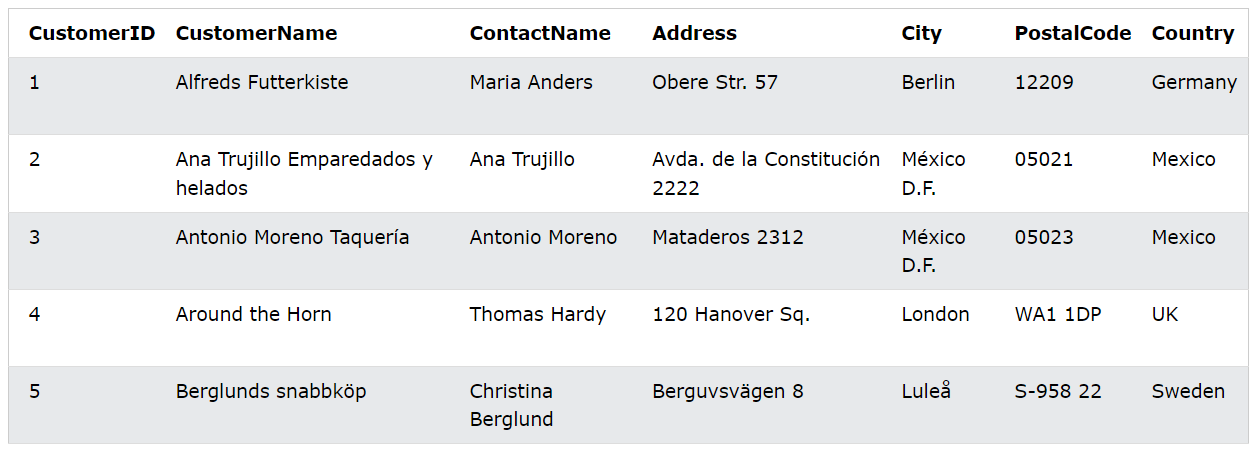
FROM Customers  
 ORDER BY Country ASC, CustomerName DESC;



**GROUP BY**

**1. Показати кількість клієнтів у кожній країні**

SELECT COUNT(CustomerID), Country  
 FROM Customers  
 GROUP BY Country;



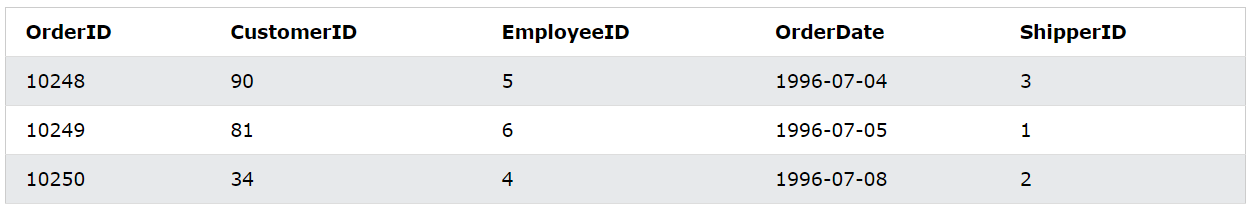
**2. Показати кількість клієнтів у кожній країні, відсортовану від більшої до меншої**

SELECT COUNT(CustomerID), Country  
 FROM Customers  
 GROUP BY Country  
 ORDER BY COUNT(CustomerID) DESC;

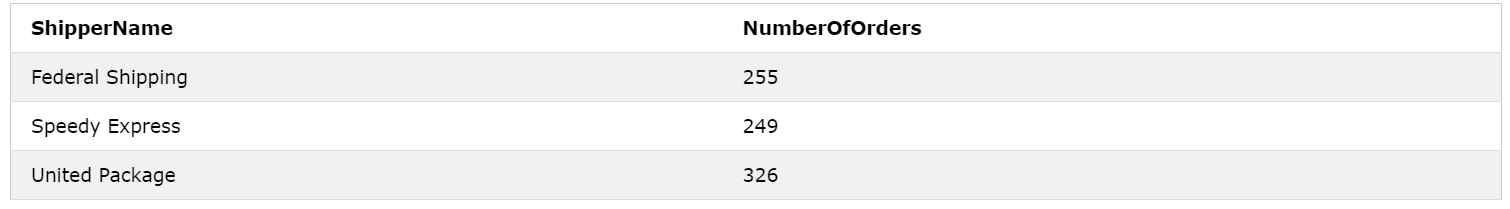


**3. Показати кількість замовлень, надісланих кожним відправником**

SELECT Shippers.ShipperName, COUNT(Orders.OrderID) AS NumberOfOrders FROM Orders  
LEFT JOIN Shippers ON Orders.ShipperID = Shippers.ShipperID  
GROUP BY ShipperName;







**HAVING**

**1. Перерахувати кількість клієнтів у кожній країні. Включити лише країни з більш ніж 5 клієнтами**

SELECT COUNT(CustomerID), Country  
 FROM Customers  
 GROUP BY Country  
 HAVING COUNT(CustomerID) > 5;





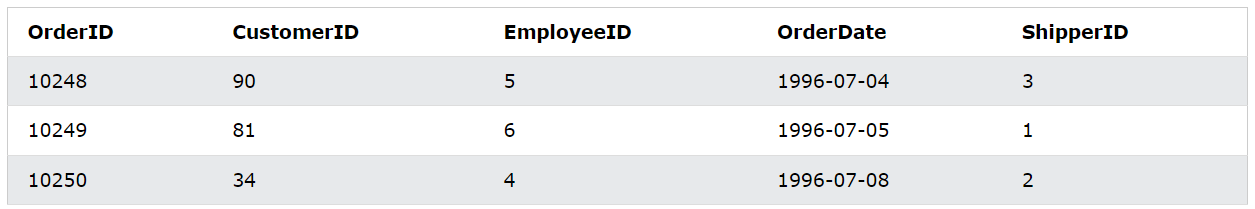
**2. Показати кількість клієнтів у кожній країні, відсортовану від більшої до меншої (враховуються тільки країни з більш ніж 5 клієнтами)**

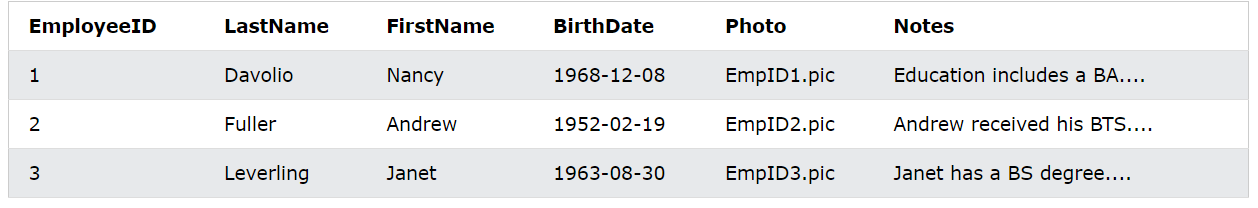
SELECT COUNT(CustomerID), Country  
 FROM Customers  
 GROUP BY Country  
 HAVING COUNT(CustomerID) > 5  
 ORDER BY COUNT(CustomerID) DESC;

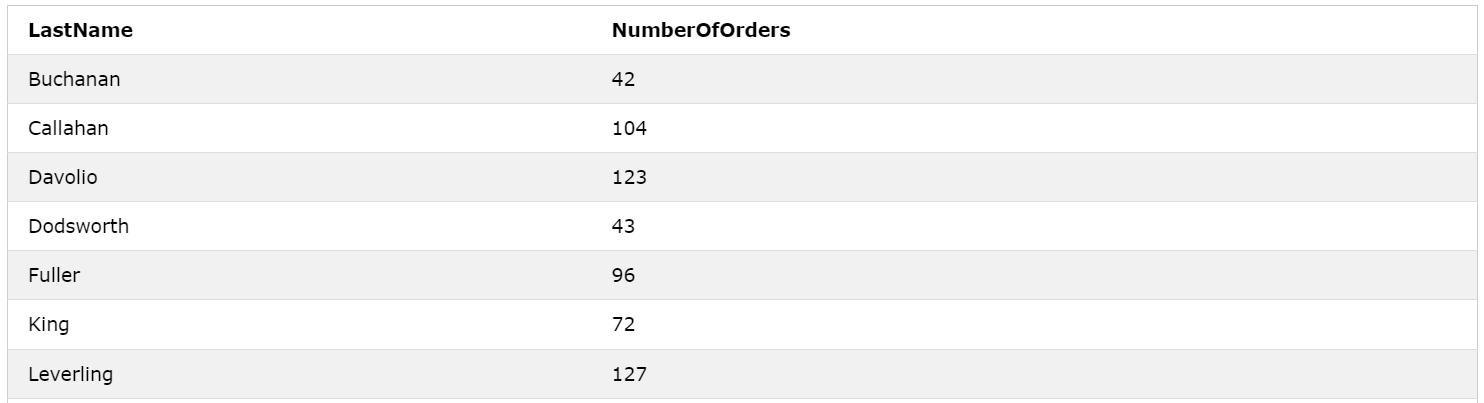


**3. Показати список співробітників, які зареєстрували більше 10 замовлень**

SELECT Employees.LastName, COUNT(Orders.OrderID) AS NumberOfOrders  
 FROM (Orders  
 INNER JOIN Employees ON Orders.EmployeeID = Employees.EmployeeID)  
 GROUP BY LastName  
 HAVING COUNT(Orders.OrderID) > 10;







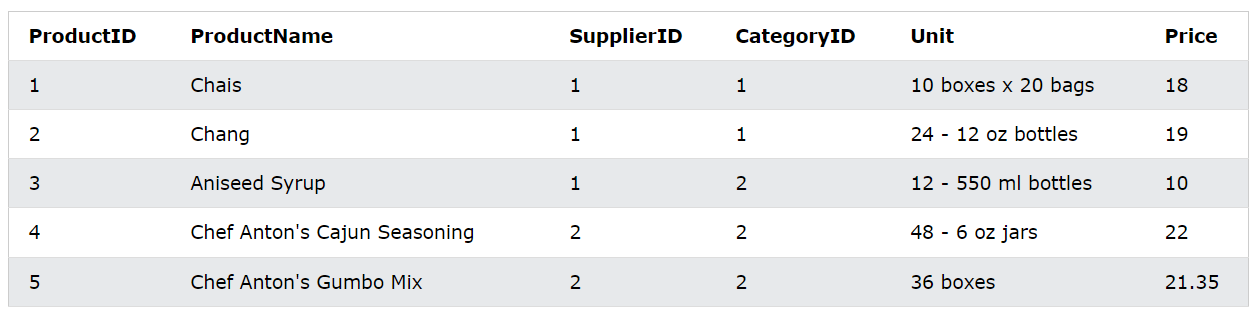
**Агрегатні функції**

Avg, Min, Max, Sum, Count

**MIN() та MAX()**

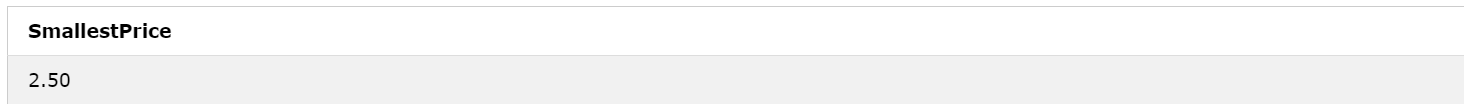
**MIN() -** повертає найменше значення вибраного стовпчика

**MAX() -** повертає найбільше значення вибраного стовпця

****

**1. Знайти ціну найдешевшого товару**

SELECT MIN(Price) AS SmallestPrice  
 FROM Products;

****

**2. Знайти ціну найдорожчого товару**

SELECT MAX(Price) AS LargestPrice  
 FROM Products;

****

**COUNT(), AVG() та SUM()**

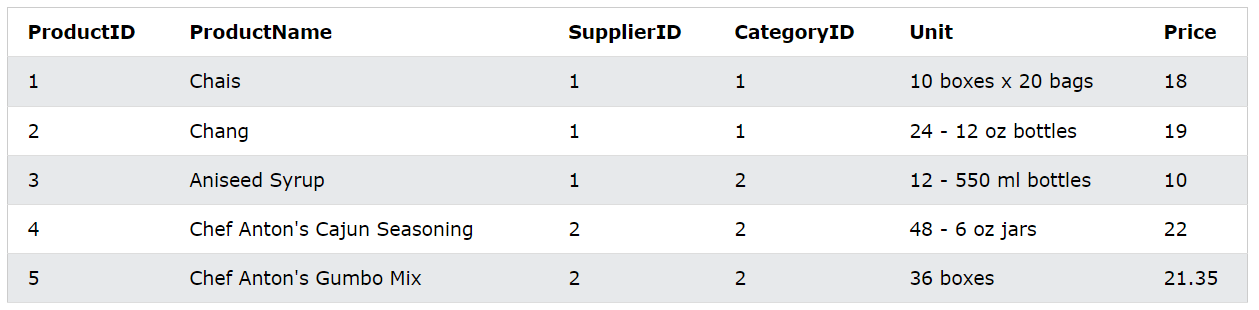
**COUNT()** повертає кількість рядків, що відповідають заданому критерію

**AVG()** повертає середнє значення числового стовпчика

**SUM()** повертає загальну суму числового стовпця

**1. Знайти кількість товару**

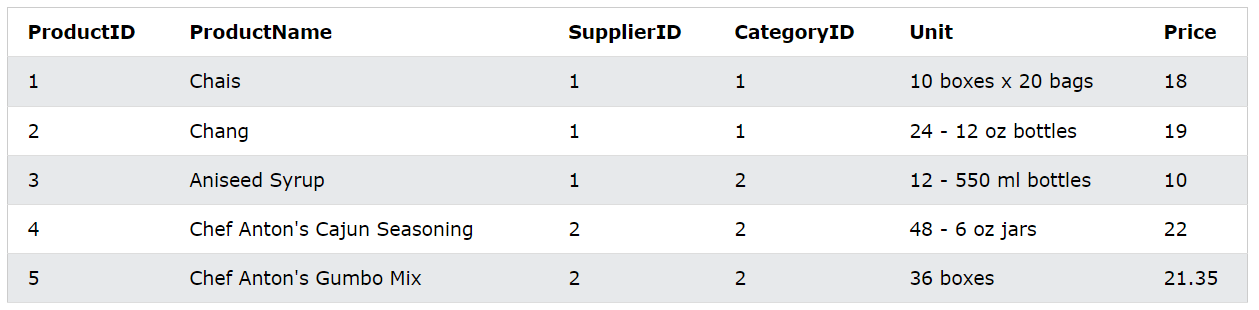
SELECT COUNT(ProductID)  
 FROM Products;





**2. Знайти середню ціну всіх товарів**

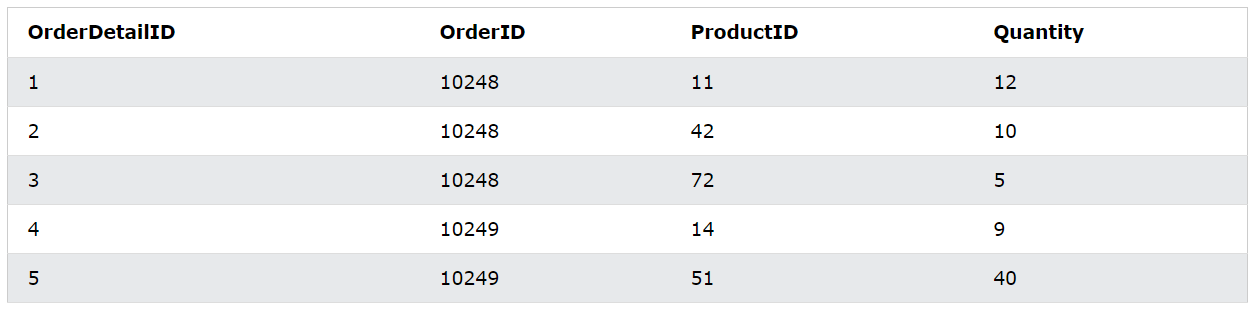
SELECT AVG(Price)  
 FROM Products;





**3. Знайти суму полів «Quantity» в таблиці «OrderDetails»**

SELECT SUM(Quantity)  
 FROM OrderDetails;





**=, >, AND, OR, NOT**

**=** Дорівнює

**>** Більше ніж

**<** Менше ніж

**>=** Більше або дорівнює

<= Менше або дорівнює

<> Не дорівнює. Примітка: У деяких версіях SQL цей оператор може бути записаний як !=

**AND та OR** - використовуються для фільтрації записів на основі більш ніж однієї умови:

**AND** - відображає запис, якщо всі умови, розділені символом AND, є істинними

**OR** - відображає запис, якщо будь-яка з умов, розділених символом OR, є ІСТИНА

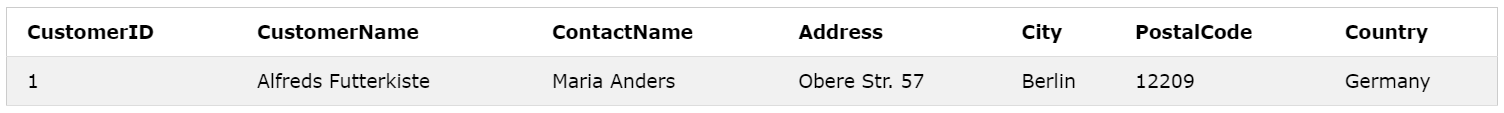
**NOT** - відображає запис, якщо умова(и) не є ІСТИНА



**1.** **Вибрати всі стовпчики з таблиці "Customer", де ID клієнта 1**

SELECT \*

FROM Customers  
 WHERE CustomerID = 1;

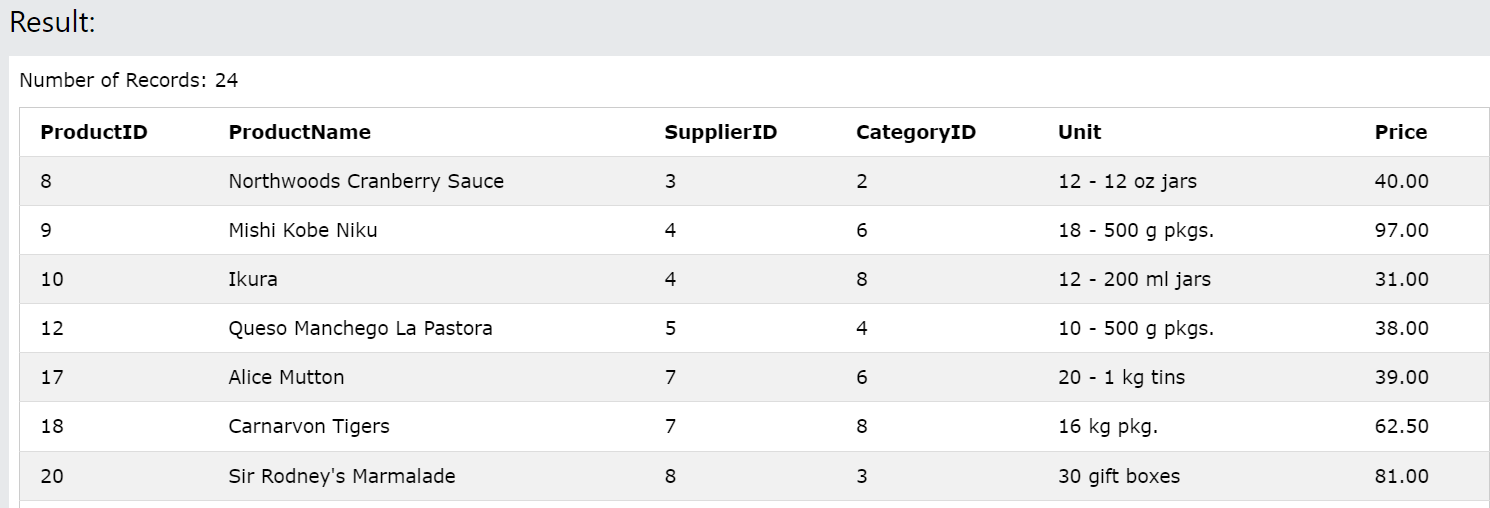


**2. Вибрати всі стовпчики з таблиці "Products", де ціна більше ніж 30**

SELECT \*

FROM Products

WHERE Price > 30;

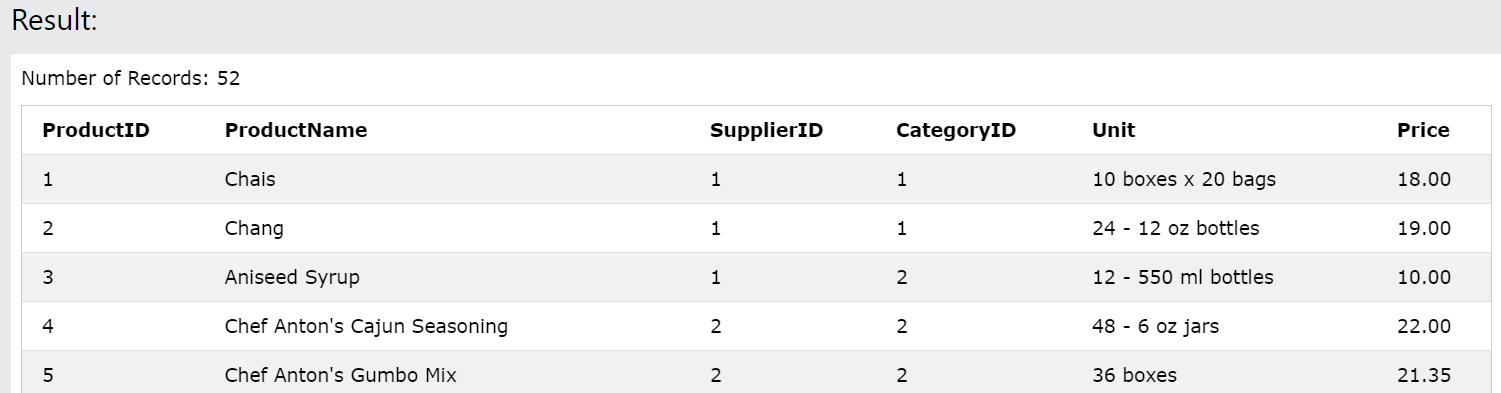


**3. Вибрати всі стовпчики з таблиці "Products", де ціна менше ніж 30**

SELECT \*

FROM Products

WHERE Price < 30;

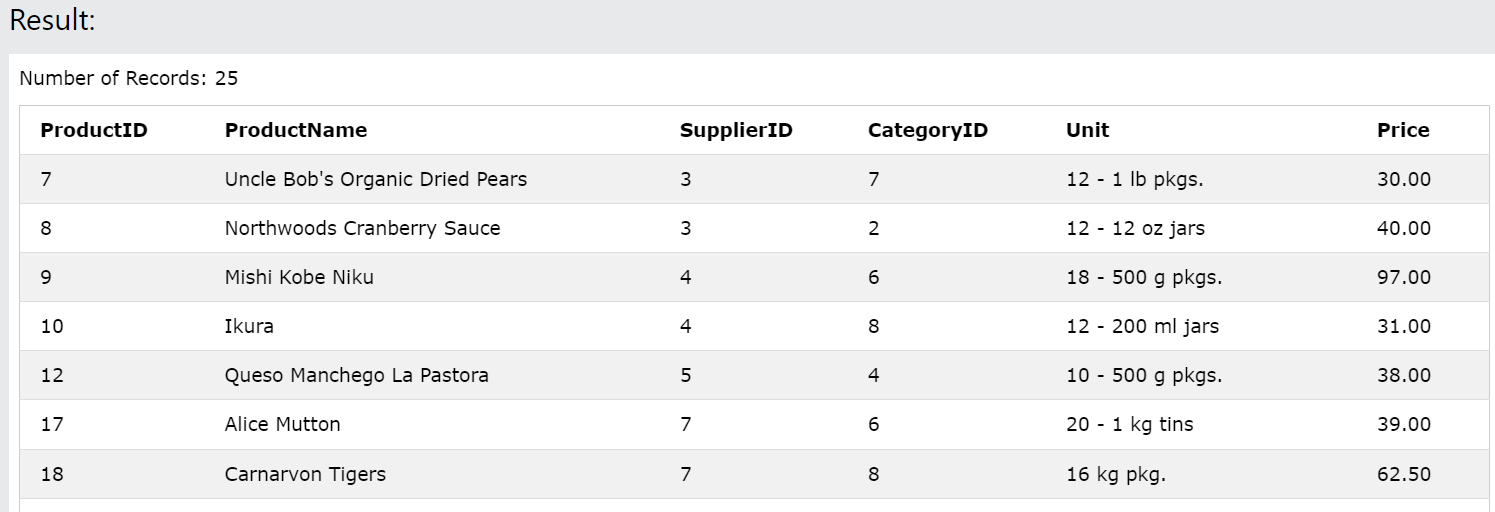


**4. Вибрати всі стовпчики з таблиці "Products", де ціна більше або дорівнює 30**

SELECT \*

FROM Products

WHERE Price >= 30;

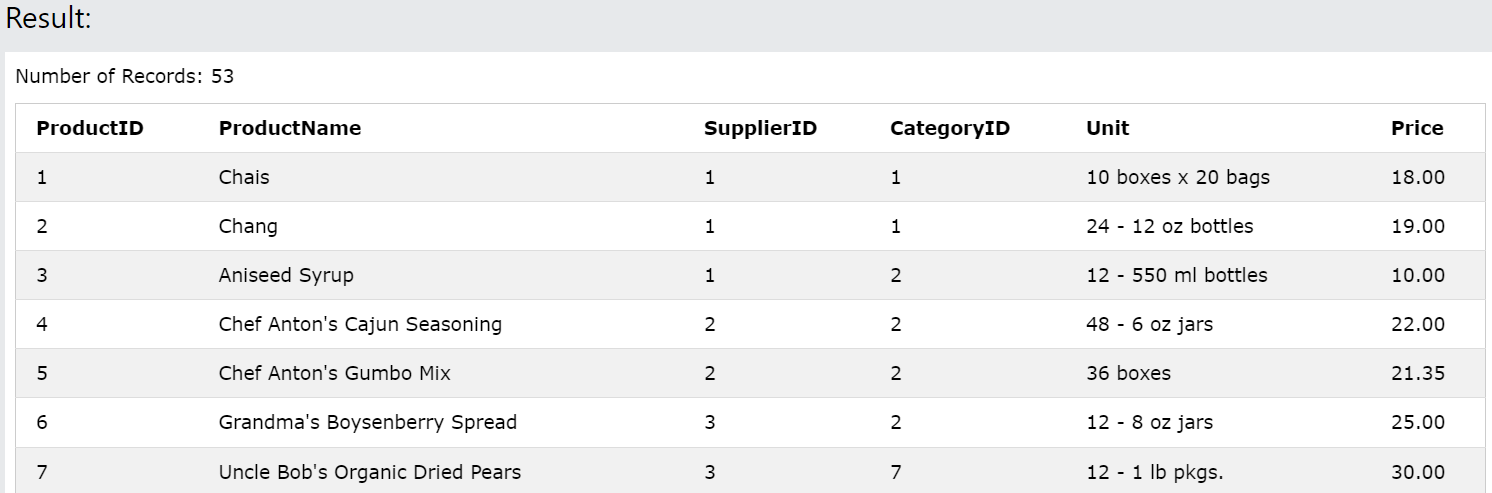
****

**5. Вибрати всі стовпчики з таблиці "Products", де ціна менше або дорівнює 30**

SELECT \*

FROM Products

WHERE Price <= 30;

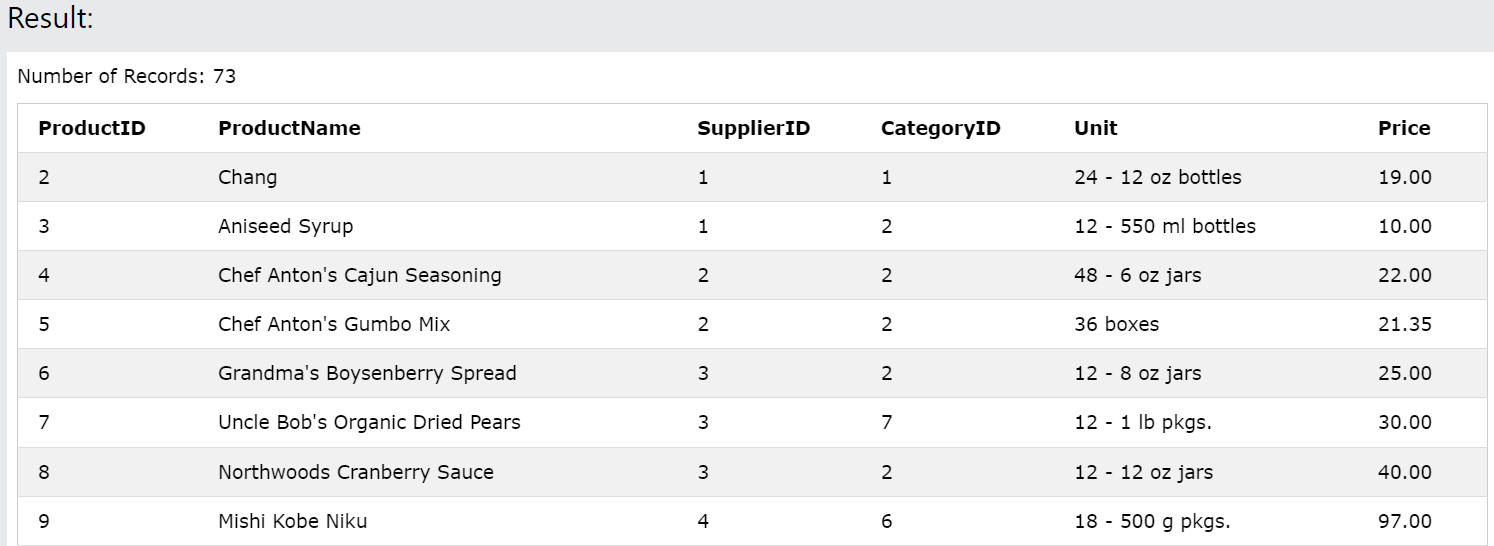


**6. Вибрати всі стовпчики з таблиці "Products", де ціна не дорівнює 30**

SELECT \*

FROM Products

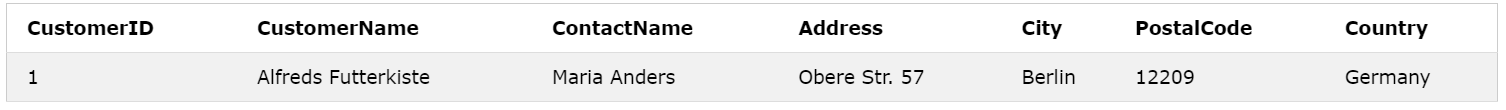
WHERE Price <> 18;



**7. Вибрати всі поля з таблиці «Customers», де країна – «Germany», а місто – «Berlin»**

SELECT \*

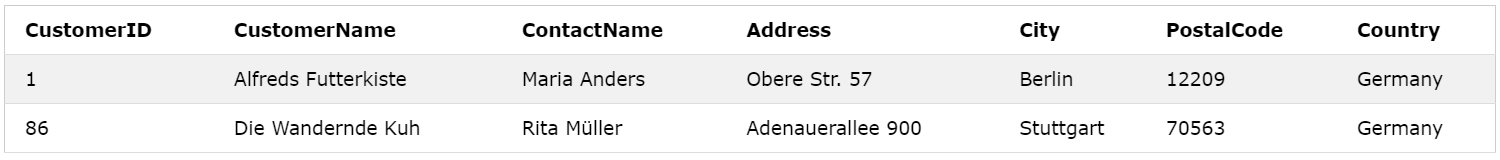
FROM Customers  
 WHERE Country = 'Germany' AND City = 'Berlin';



**8. Вибрати всі поля з таблиці «Customers», в яких місто є «Berlin» або «Stuttgart»**

SELECT \*

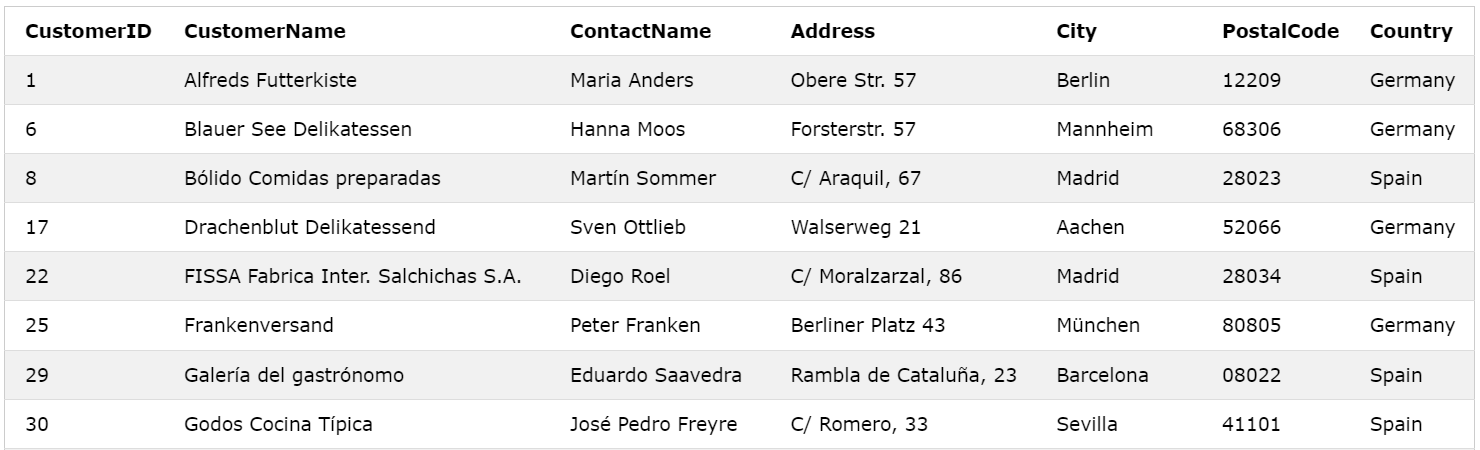
FROM Customers  
 WHERE City = 'Berlin' OR City = 'Stuttgart';



**9. Вибрати всі поля з поля «Customers», де країна «Germany» або «Spain»**

SELECT \*

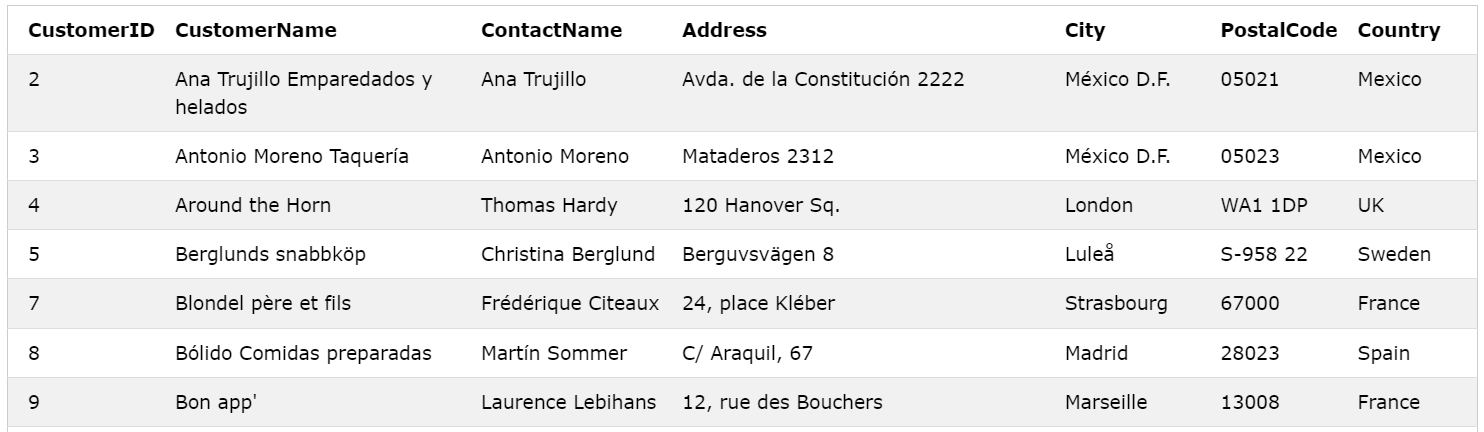
FROM Customers  
 WHERE Country = 'Germany' OR Country = 'Spain';



**10. Вибрати всі поля з поля « Customers», де країна НЕ «Germany»**

SELECT \*

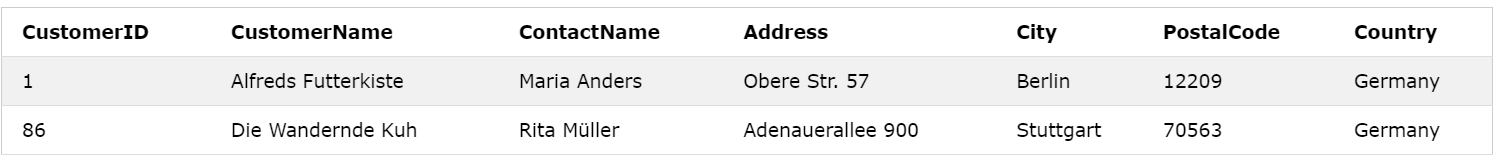
FROM Customers  
 WHERE NOT Country = 'Germany';



**11. Вибрати всі поля з таблиці «Customers», де країна «Germany», а місто повинно бути «Berlin» або «Stuttgart»**

SELECT \*

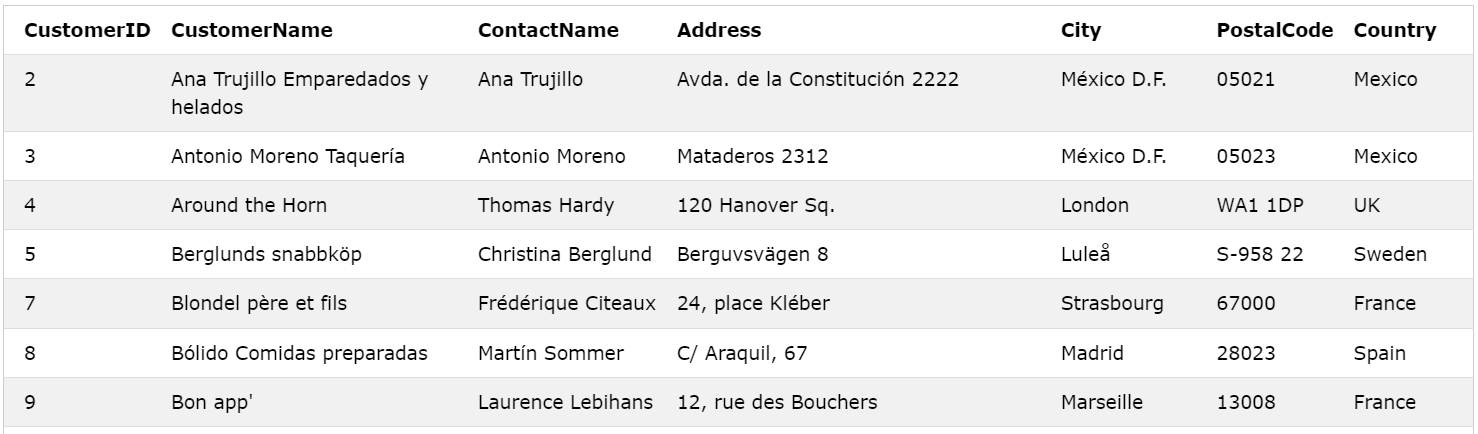
FROM Customers  
 WHERE Country = 'Germany' AND (City = 'Berlin' OR City = 'Stuttgart');



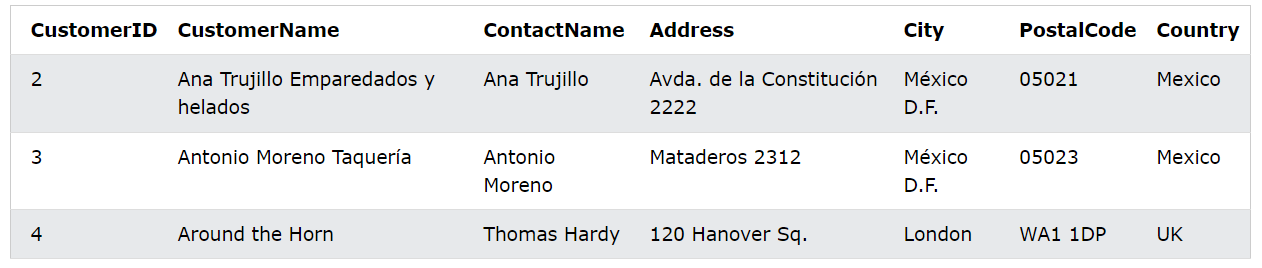
**12. Вибрати всі поля з поля «Customers», де країна НЕ «Germany» і НЕ «USA», де країна НЕ «Germany» і НЕ «USA»**

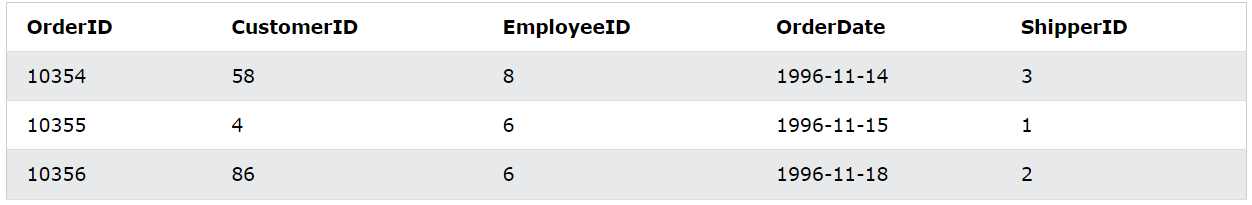
SELECT \*

FROM Customers  
 WHERE NOT Country = 'Germany' AND NOT Country = 'USA';



**Alias - AS**





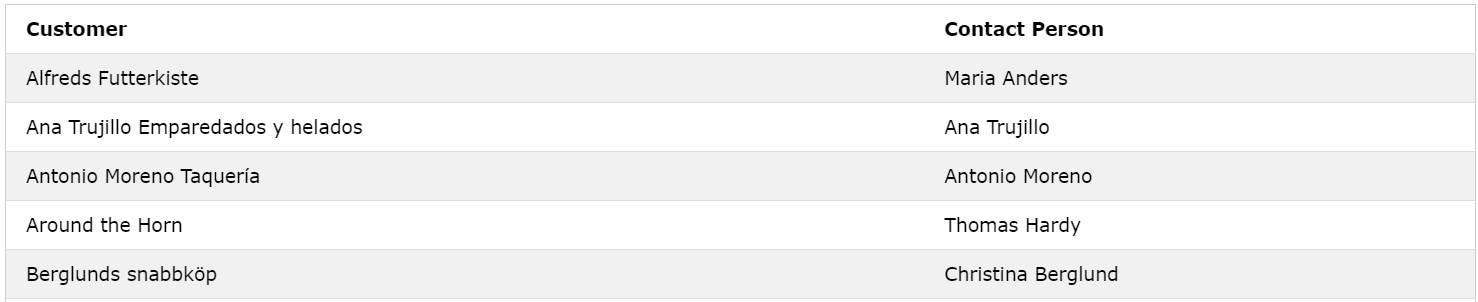
**1. Створити два псевдоніми, один для стовпця CustomerID і один для стовпця CustomerName**

SELECT CustomerID AS ID, CustomerName AS Customer  
 FROM Customers;



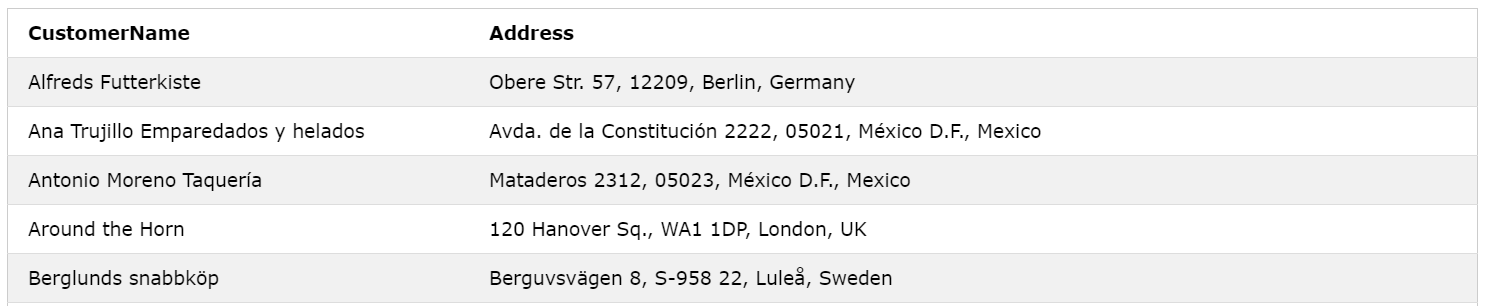
**2. Створитидва псевдоніми, один для стовпця CustomerName і один для стовпця ContactName**

SELECT CustomerName AS Customer, ContactName AS "Contact Person"  
 FROM Customers;



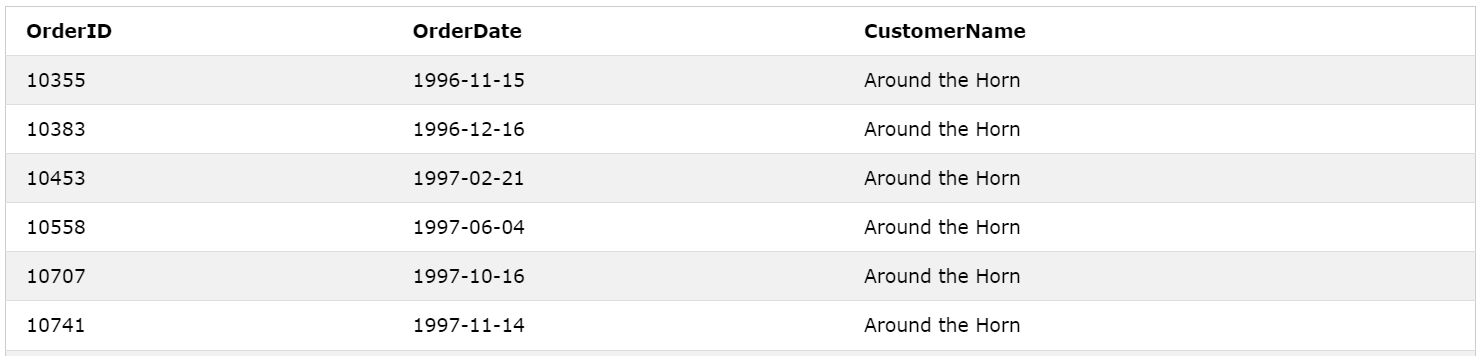
**3. Створити псевдонім «Address», який об'єднує чотири стовпці (Address, PostalCode, City та Country)**

SELECT CustomerName, CONCAT\_WS(', ', Address, PostalCode, City, Country) AS Address  
FROM Customers;

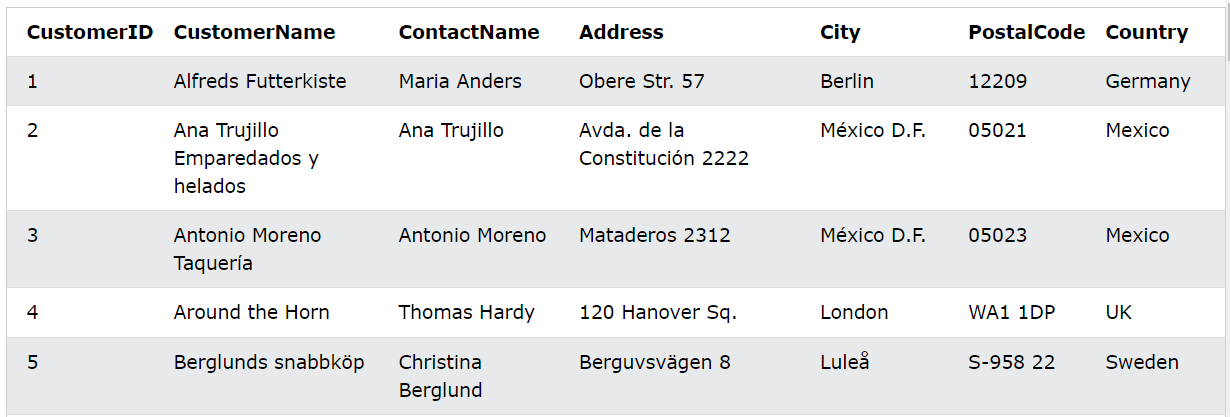


**4. Вибрати всі замовлення від клієнта з CustomerID=4 (Around the Horn) з таблиць «Customers» і «Orders», і даємо їм табличні псевдоніми «c» і «o» відповідно**

SELECT o.OrderID, o.OrderDate, c.CustomerName  
 FROM Customers AS c, Orders AS o  
 WHERE c.CustomerName='Around the Horn' AND c.CustomerID=o.CustomerID;



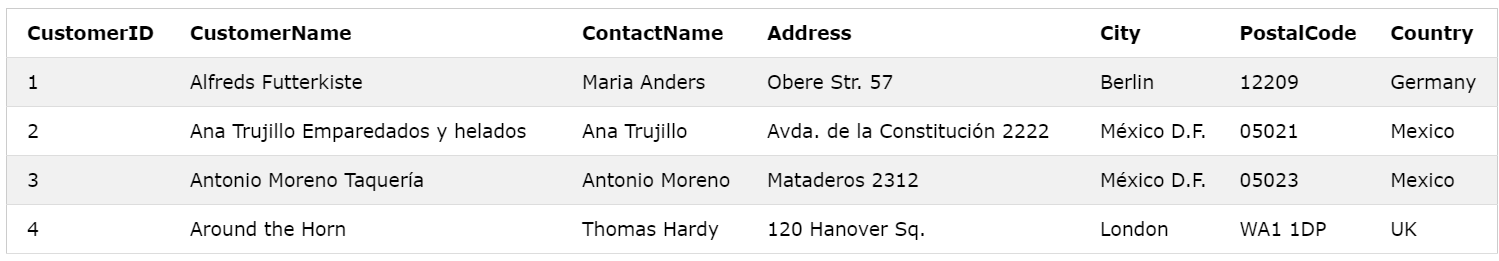
**LIKE та NOT LIKE**



**1. Вибрати всіх клієнтів з іменем CustomerName, що починається з «a»**

SELECT \*

FROM Customers  
 WHERE CustomerName LIKE 'a%';



**2. Вибрати всіх клієнтів з іменем CustomerName, що закінчується на «a»**

SELECT \*

FROM Customers  
 WHERE CustomerName LIKE '%a';



**3. Вибрати всіх клієнтів з іменем CustomerName, що містять «or» в будь-якій позиції**

SELECT \*

FROM Customers  
 WHERE CustomerName LIKE '%or%';



**4. Вибрати всіх клієнтів з іменем CustomerName, які мають «r» у другій позиції**

SELECT \*

FROM Customers  
 WHERE CustomerName LIKE '\_r%';



**5. Вибрати всіх клієнтів з іменами клієнтів, що починаються з «a» і мають довжину не менше 3 символів**

SELECT \*

FROM Customers  
 WHERE CustomerName LIKE 'a\_\_%';



**6. Вибрати всіх клієнтів з іменем ContactName, що починається на «a» і закінчується на «o»**

SELECT \*

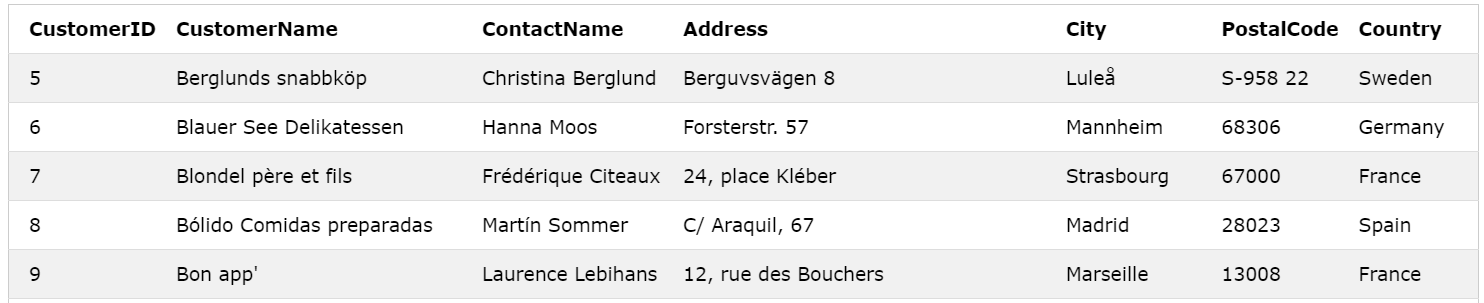
FROM Customers  
 WHERE ContactName LIKE 'a%o';



**7. Вибрати всіх клієнтів з іменем клієнта, яке НЕ починається з «a»**

SELECT \*

FROM Customers  
 WHERE CustomerName NOT LIKE 'a%';



**Wildcard** (символи)

|  |  |  |
| --- | --- | --- |
| **Символ** | **Опис** | **Приклад** |
| **%** | Представляє нуль або більше символів | **'a%'** ; **'%a'** ; **'%or%'** |
| **\_** | Представляє один символ | **'\_r%'** ; **'a\_%\_%'** |

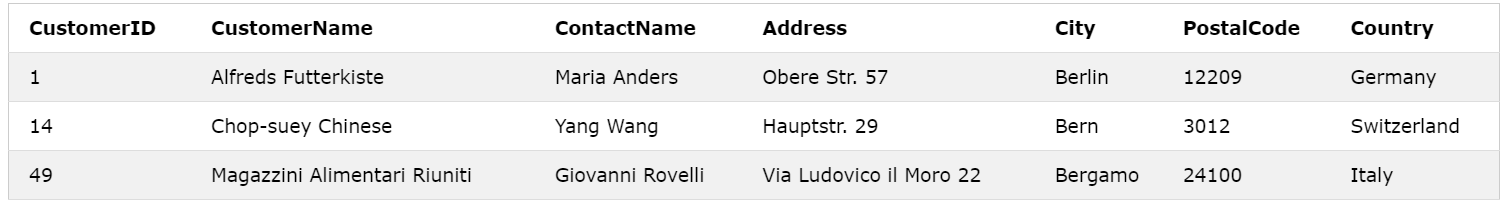
|  |  |
| --- | --- |
| **LIKE Operator** | **Опис** |
| WHERE CustomerName LIKE 'a%' | Знаходить всі значення, що починаються з «a» |
| WHERE CustomerName LIKE '%a' | Знаходить всі значення, що закінчуються на «a» |
| WHERE CustomerName LIKE '%or%' | Знаходить всі значення, що містять «or» в будь-якій позиції |
| WHERE CustomerName LIKE '\_r%' | Знаходить всі значення, що містять «r» на другій позиції |
| WHERE CustomerName LIKE 'a\_%\_%' | Знаходить всі значення, які починаються з «a» і мають довжину не менше 3 символів |
| WHERE ContactName LIKE 'a%o' | Знаходить всі значення, що починаються з «a» і закінчуються на «o» |



**1. Вибрати всіх клієнтів з містом, що починається на «ber»**

SELECT \*

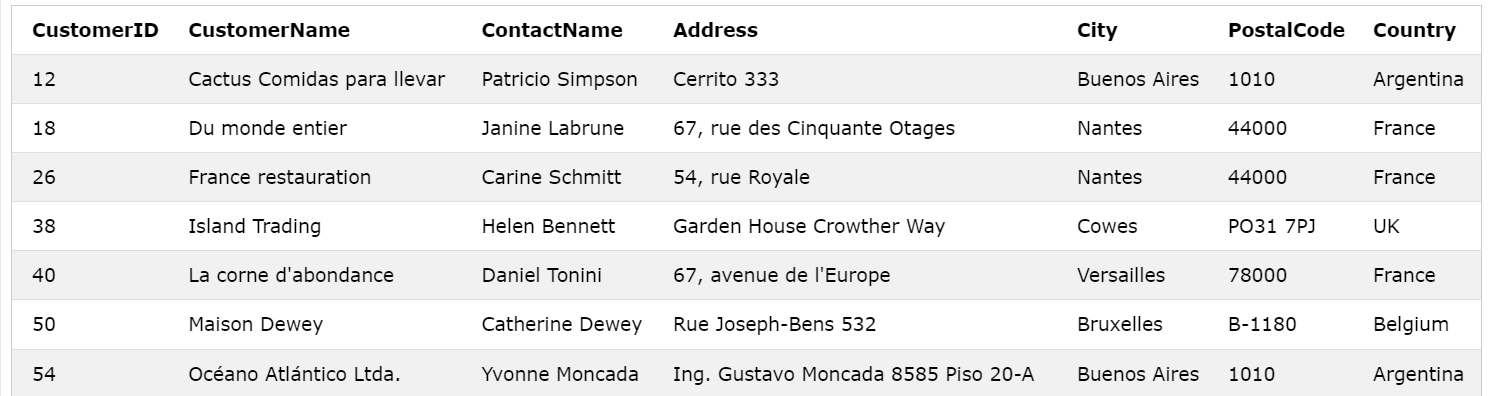
FROM Customers  
 WHERE City LIKE 'ber%';



**2. Вибрати всіх клієнтів з містом, що містить закінчення «es»**

SELECT \*

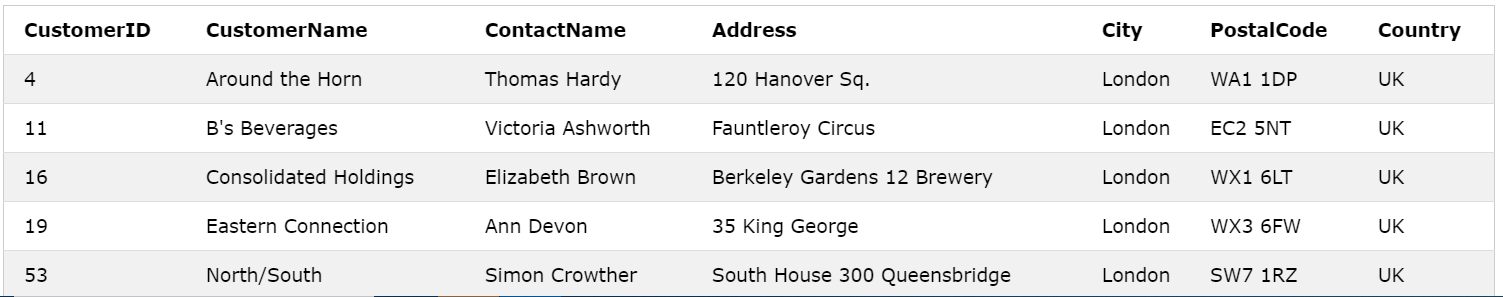
FROM Customers  
 WHERE City LIKE '%es%';



**3. Вибрати всіх клієнтів, назва міста яких починається з будь-якого символу, за яким слідує «ondon»**

SELECT \*

FROM Customers  
 WHERE City LIKE '\_ondon';



**4. Вибрати всіх клієнтів, місто яких починається з «L», далі йде будь-який символ, далі йде «n», далі йде будь-який символ, далі йде «on»**

SELECT \*

FROM Customers  
 WHERE City LIKE 'L\_n\_on';

