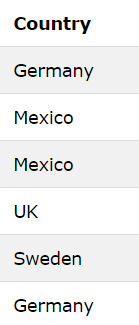
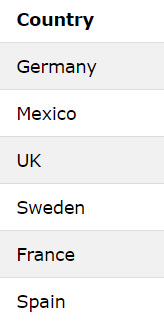
**SELECT** Country **FROM** Customers;

*Muestra todos los países (no importa si estan repetidos) que tenga registrados en la tabla customers.*



**SELECT DISTINCT** Country **FROM** Customers;

*Muestra todos los países (sin repetir ninguno) que tenga registrados en la tabla customers.*

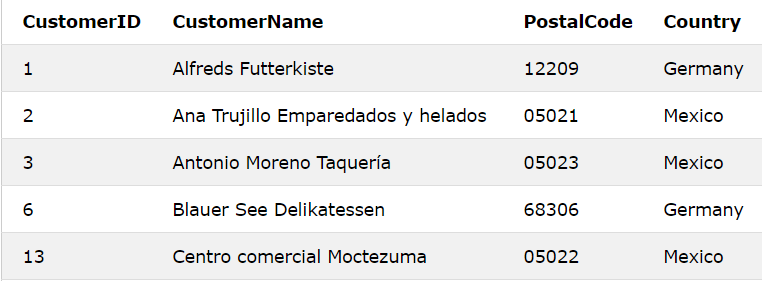


**SELECT \* FROM** Customers **WHERE** Country='Mexico';

Muestra todos los registros de la tabla customers que el país sea Mexico.

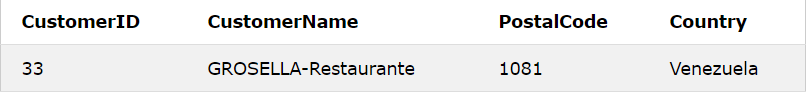


**SELECT \* FROM** Customers **WHERE** Country='Mexico' **OR** Country='Germany';

Muestra todos los registros que tengan como país México o Alemania

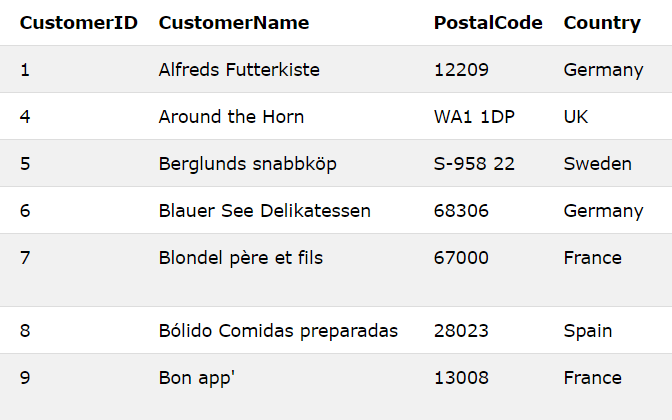
**SELECT \* FROM** Customers **WHERE** Country='Venezuela' **AND** City='Caracas';

Muestra todos los registros que cumplan con esas dos condiciones



**SELECT \* FROM** Customers **WHERE NOT** Country='Mexico';

Muestra todos los registros que no tengan como país Alemania



**SELECT \* FROM** Customers **ORDER BY** Country **ASC**, CustomerName **DESC**;

Selecciona todos los registros ordenándolos primero por país de forma ascendente y si hay varios registros con el mismo país se ordenara esos registros por CustomerName de forma descendente.

**SELECT \*FROM** Customers **WHERE** Address **IS NULL;**

Selecciona todos los registros de customers donde el campo address sea null

**SELECT \*FROM** Customers **WHERE** Address **IS NOT NULL;**

Selecciona todos los registros de customers donde el campo address **no** sea null.

**SELECT TOP** 3 \* **FROM** Customers;

**SELECT \* FROM** Customers **LIMIT** 3;

**SELECT \* FROM** Customers **WHERE ROWNUM** <= 3;

Selecciona los 3 primeros registros de la tabla.

**SELECT MIN**(Price) **AS** SmallestPrice **FROM** Products;

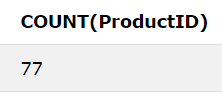
Selecciona el registro con el precio más bajo (barato)

**SELECT MAX**(Price) **AS** LargestPrice **FROM** Products;

Selecciona el registro con el precio más alto (caro)

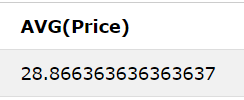
**SELECT COUNT**(ProductID)**FROM** Products;

Muestra cuantos productos están registrados



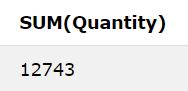
**SELECT AVG**(Price) **FROM** Products;

Muestra el precio promedio de los productos



**SELECT SUM**(Quantity) **FROM** OrderDetails;

Muestra la suma de la cantidad de productos



**SELECT \* FROM** Customers **WHERE** CustomerName **LIKE** 'a%';

Selecciona todos los registros de Customers donde el nombre empiece con a

**SELECT \* FROM** Customers **WHERE** CustomerName **LIKE** '%a';

Selecciona todos los registros de Customers donde el nombre termine con a

**SELECT \* FROM** Customers **WHERE** CustomerName **LIKE** '%or%';

Selecciona todos los registros de Customers donde el nombre contenga or en cualquier posición

**SELECT \* FROM** Customers **WHERE** CustomerName **LIKE** '\_r%';

Selecciona todos los registros de Customers donde el nombre tenga r en la segunda posición

El operador IN le permite especificar múltiples valores en una cláusula WHERE; como también es una abreviatura para múltiples condiciones OR

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

Selecciona todos los registros que tengan esos países

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

Selecciona todos los registros que **NO** tengan esos países

El operador **BETWEEN** selecciona valores dentro de un rango dado. Los valores pueden ser números, texto o fechas.

**SELECT \* FROM** Products **WHERE** Price **BETWEEN** 10 AND 20;

Selecciona todos los productos con un precio ENTRE 10 y 20

**SELECT \* FROM** Products **WHERE** Price **NOT BETWEEN** 10 AND 20;

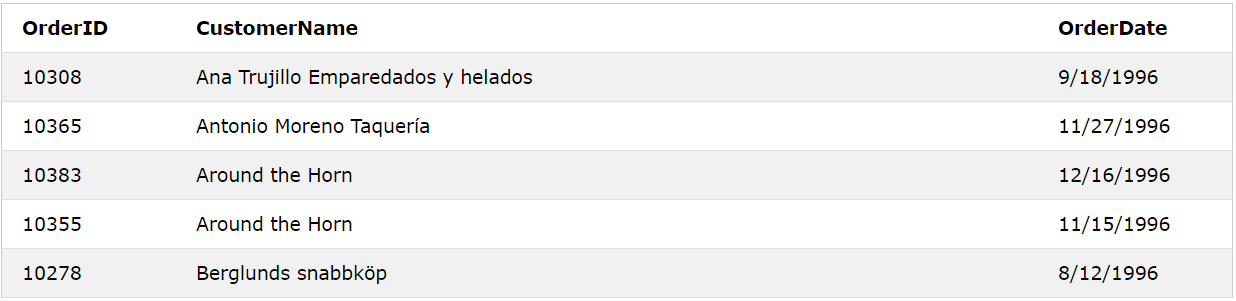
Selecciona todos los productos con un precio que NO este ENTRE 10 y 20

Una cláusula **JOIN** se usa para combinar filas de dos o más tablas, en función de una columna relacionada entre ellas.

La palabra clave **INNER JOIN** selecciona registros que tienen valores coincidentes en ambas tablas.



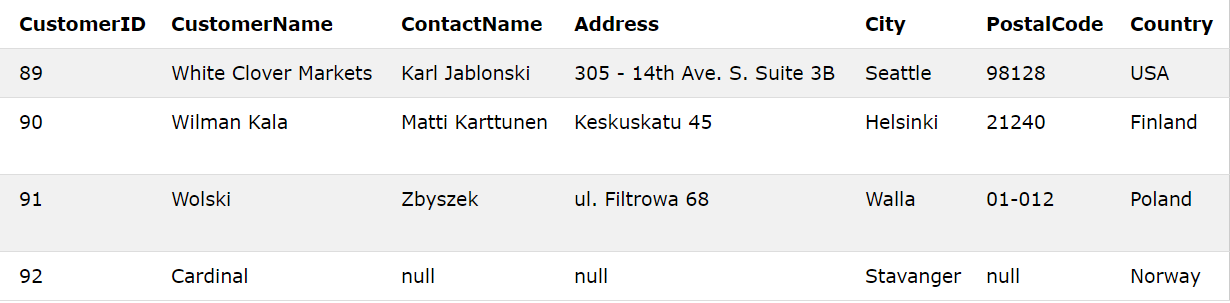
**SELECT** Orders.OrderID, Customers.CustomerName, Orders.OrderDate **FROM** Orders **INNER JOIN** Customers **ON** Orders.CustomerID=Customers.CustomerID;



**INSERT INTO** Customers (CustomerName, City, Country)

**VALUES** ('Cardinal', 'Stavanger', 'Norway');

Para insertar un registro en una tabla, si algún campo puede tomar el valor null, se puede omitir



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

Actualiza el registro donde CustomerID sea 1 los campos contactName a Alfred chmidt y city a Frankfurt

**DELETE FROM** Customers **WHERE** CustomerName='Alfreds Futterkiste';

Borra los registros que tengan como CustomerName a Alfreds Futterkiste'