**PRUEBA TÉCNICA SQL**

1.- Seleccionar todos los clientes ordenados por nombre del cliente

SELECT \*  
FROM CLIENTES  
ORDER BY NOM\_CLIENTE;

2.- Seleccionar los clientes cuya localidad sea Madrid

SELECT \*

FROM CLIENTES

WHERE LOC\_CLIENTE ='Madrid';

3.- Seleccionar todos los clientes cuyo código postal pertenezca a la provincia de Sevilla (los que comienzan por 41)

SELECT \*  
FROM CLIENTES  
WHERE LEFT(CAST(CP\_CLIENTE AS varchar), 2) = '41';

4.- Seleccionar todos los clientes cuyo código sea inferior a 10 o cuyo nombre de cliente comience por la letra A

SELECT \*  
FROM CLIENTES  
WHERE COD\_CLIENTE<10 AND LEFT(NOM\_CLIENTE,1)= 'A';

5.- Contar cuántos clientes existen por cada localidad

SELECT LOC\_CLIENTE, COUNT(\*) AS CUANTOS

FROM CLIENTES

GROUP BY LOC\_CLIENTE;

6.- Borrar los clientes cuyo nombre comience por la letra Z y termine por la letra Z

DELETE FROM CLIENTES  
WHERE LEFT(NOM\_CLIENTE,1)= 'Z' AND RIGHT(NOM\_CLIENTE,1)= 'Z';

7.- Modificar el valor del código postal de los clientes cuya localidad sea Madrid con el valor 28001

UPDATE CLIENTES SET CP\_CLIENTE = 28001   
WHERE LOC\_CLIENTE='Madrid';

8.- Seleccionar los clientes cuyo importe de sus VENTAS sea distinto de cero.

SELECT \*

FROM CLIENTES, VENTAS

WHERE VENTAS.COD\_CLIENTE=CLIENTES.COD\_CLIENTE

AND VENTAS.IMPORTE <>0;

9.- Seleccionar todos los clientes junto con el importe de las ventas de cada uno.

SELECT CLIENTES.\*, VENTAS.IMPORTE

FROM CLIENTES LEFT JOIN VENTAS  
ON VENTAS.COD\_CLIENTE=CLIENTES.COD\_CLIENTE;

10.- Modificar el importe de las ventas de los clientes a cero para aquellos clientes cuyo nombre contenga al menos una letra A.

UPDATE VENTAS SET IMPORTE = 0

WHERE COD\_CLIENTE IN

(SELECT COD\_CLIENTE

FROM CLIENTES

WHERE NOM\_CLIENTE LIKE '%A%');

11. Subirlo a Github para compartirlo.