

## Consultas jardinería

*Ángel Leonardo Cáceres Quintero 02210131007*

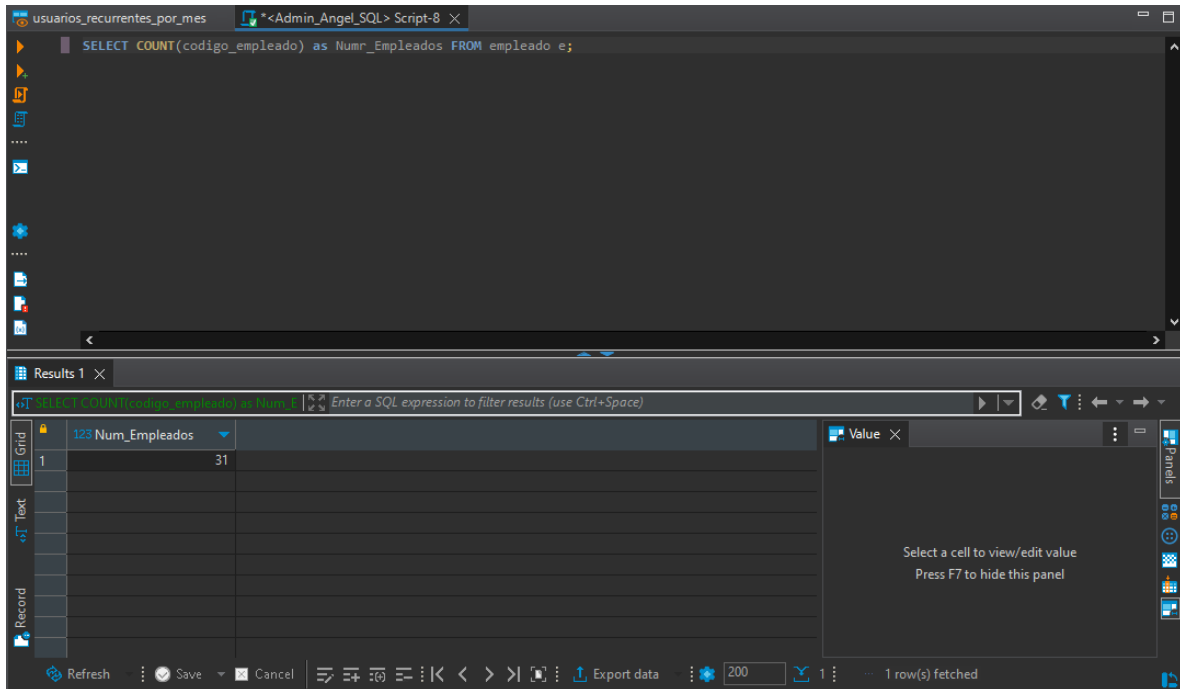
Sistema gestión de base de datos.

Facultad Ing. De sistemas

**Universidad De Santander**

¿Cuántos empleados hay en la compañía?

```
SELECT COUNT(codigo_empleado) as Num_Empleados FROM empleado e;
```



¿Cuántos clientes tienen cada país?

```
SELECT COUNT(codigo_cliente) as Num_Clientes, pais
```

```
FROM cliente c
```

```
group by pais
```

```
order by Num_Clientes ASC;
```

The screenshot shows a SQL IDE interface. The top panel displays a SQL query:

```
SELECT COUNT(codigo_cliente) as Num_Clientes, pais
FROM cliente c
group by pais
order by Num_Clientes ASC;
```

The bottom panel shows the results in a grid view. The grid has two columns: 'Num\_Clientes' and 'pais'. The data is as follows:

Num_Clientes	pais
1	United Kingdom
2	Australia
2	France
4	USA
27	Spain

The status bar at the bottom indicates '5 row(s) fetched - 123ms, on 2023-05-03 at 18:17:17'.

¿Cuál fue el pago medio en 2009?

SELECT AVG(total) as pago\_medio\_2009

FROM pago p

WHERE YEAR(fecha\_pago) = '2009';

The screenshot shows a SQL IDE interface. The top panel displays a SQL query:

```
SELECT AVG(total) as pago_medio_2009
FROM pago p
WHERE YEAR(fecha_pago) = '2009';
```

The bottom panel shows the results in a grid view. The grid has two columns: 'pago\_medio\_2009'. The data is as follows:

pago_medio_2009
4,504.076923

The status bar at the bottom indicates '1 row(s) fetched - 123ms, on 2023-05-03 at 18:17:17'.

### ¿Cuántos pedidos hay en cada estado?

```
SELECT estado, COUNT(codigo_pedido) as num_pedidos
```

```
FROM pedido p
```

```
GROUP BY estado
```

```
ORDER BY num_pedidos DESC;
```

The screenshot shows a SQL IDE window titled "usuarios\_recurrentes\_por\_mes" with a sub-window "Admin\_Angel\_SQL> Script-8". The SQL query entered is:

```
SELECT estado, COUNT(codigo_pedido) as num_pedidos
FROM pedido p
GROUP BY estado
ORDER BY num_pedidos DESC;
```

The output panel, titled "pedido 1", shows the results of the query in a table format:

Grid	estado	num_pedidos
1	Entregado	61
2	Pendiente	30
3	Rechazado	24

The interface also includes a "Value" panel on the right with the text "Select a cell to view/edit value. Press F7 to hide this panel." and a status bar at the bottom indicating "3 row(s) fetched".

### Calcula el precio del producto más caro y más barato en una consulta

```
SELECT
```

```
CONCAT('$', MAX(precio_venta)) AS producto_caro,
```

```
CONCAT('$', MIN(precio_venta)) AS producto_barato
```

```
FROM producto p;
```

The screenshot shows a SQL script editor with the following query:

```
SELECT
  CONCAT('$', MAX(precio_venta)) AS producto_caro,
  CONCAT('$', MIN(precio_venta)) AS producto_barato
FROM producto p;
```

The results panel shows the following data:

	producto_caro	producto_barato
1	\$462.00	\$1.00

The status bar indicates: 1 row(s) fetched - 123ms, on 2023-05-03 at 18:28:55.

**Devuelve el nombre del cliente con mayor límite de crédito.**

```
SELECT nombre_cliente
```

```
FROM cliente
```

```
WHERE limite_credito = (SELECT MAX(limite_credito) FROM cliente);
```

The screenshot shows a SQL script editor with the following query:

```
SELECT nombre_cliente
FROM cliente
WHERE limite_credito = (SELECT MAX(limite_credito) FROM cliente);
```

The results panel shows the following data:

	nombre_cliente
1	Tendo Garden

The status bar indicates: 1 row(s) fetched - 123ms, on 2023-05-03 at 18:33:34.

Devuelve el producto que más unidades tiene en stock.

```
SELECT nombre, cantidad_en_stock
```

```
FROM producto
```

```
WHERE cantidad_en_stock = (SELECT MAX(cantidad_en_stock) FROM producto);
```

The screenshot shows a database management tool interface. The top panel displays a SQL query: `SELECT nombre, cantidad_en_stock FROM producto WHERE cantidad_en_stock = (SELECT MAX(cantidad_en_stock) FROM producto);`. The bottom panel shows the results of the query in a table view. The table has two columns: 'nombre' and 'cantidad\_en\_stock'. The results show 10 rows, all with a quantity of 400. A 'Value' dialog box is open on the right, prompting the user to 'Select a cell to view/edit value' and 'Press F7 to hide this panel'. The bottom status bar indicates '35 row(s) fetched'.

	nombre	cantidad_en_stock
1	Rosal copa	400
2	Albaricoquero Corbato	400
3	Albaricoquero Moniqui	400
4	Albaricoquero Kurrot	400
5	Cerezo Burlat	400
6	Cerezo Picota	400
7	Cerezo Napoleón	400
8	Ciruelo R. Claudia Verde	400
9	Ciruelo Santa Rosa	400
10	Ciruelo Golden Japan	400

Devuelve un listado que muestre solamente los clientes que no han realizado ningún pago.

```
SELECT nombre_cliente
```

```
FROM cliente c
```

```
WHERE codigo_cliente NOT IN (SELECT codigo_cliente FROM pago);
```

The screenshot shows a SQL client window with the following components:

- Query Editor:** Contains the SQL query: 

```
SELECT nombre_cliente  
FROM cliente c  
WHERE codigo_cliente NOT IN (SELECT codigo_cliente FROM pago)
```
- Output Panel:** Displays the results of the query in a grid format. The grid has two columns: `nombre_cliente` and an empty column. The data rows are:

	nombre_cliente	
1	Lasas S.A.	
2	Club Golf Puerta del hierro	
3	DaraDistribuciones	
4	Madriñeña de riegos	
5	Lasas S.A.	
6	Flowers, S.A	
7	Naturajardin	
8	Americh Golf Management SL	
9	Alonha	
- Value Panel:** A panel on the right side of the output grid, currently displaying the text: "Select a cell to view/edit value. Press F7 to hide this panel".
- Status Bar:** At the bottom, it shows "18 row(s) fetched - 124ms (1ms fetch), on 2023-05-0".

Devuelve un listado de productos que nunca han aparecido en un pedido.

```
SELECT DISTINCT(nombre)
FROM producto p
WHERE codigo_producto NOT IN
( SELECT DISTINCT codigo_producto
FROM detalle_pedido
);
```

The screenshot shows a SQL IDE interface with the following components:

- Top Editor:** Contains the SQL query:
 

```
SELECT DISTINCT(nombre)
FROM producto p
WHERE codigo_producto NOT IN
( SELECT DISTINCT codigo_producto
FROM detalle_pedido
);
```
- Bottom Panel:** Displays the results of the query in a table view. The table has two columns: 'nombre' and an empty column. The results are:
 

	nombre	
64	Cassia Corimbosa	
65	Chitalpa Summer Bells	
66	Erythra Kafra	
67	Eucalyptus Citriodora	
68	Eucalyptus Ficifolia	
69	Hibiscus Syriacus Var. Injertadas 1 Tallo	
70	Lagunaria Patersonii	
71	Morus Alba	
72	Platanus Acerifolia	
- Status Bar:** Shows '103 row(s) fetched - 125ms, on 2023-05-03 at 18:48:34'.



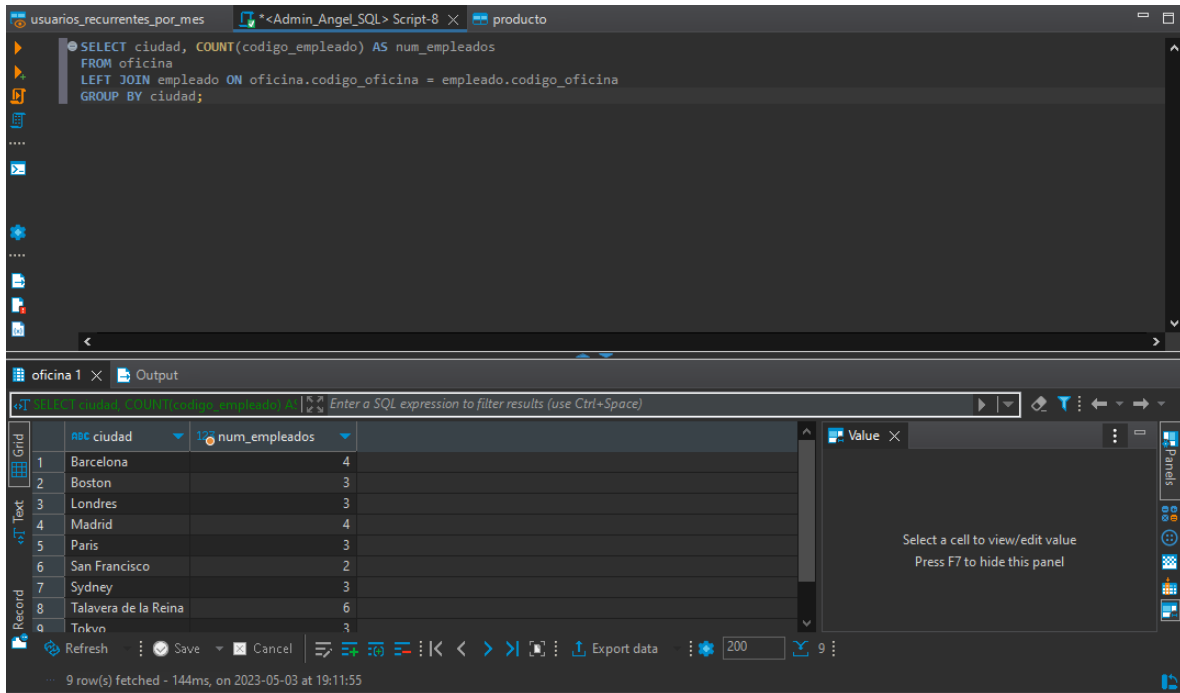
Devuelve un listado indicando todas ciudades donde hay oficinas y el número de empleados que tiene.

```
SELECT ciudad, COUNT(codigo_empleado) AS num_empleados
```

```
FROM oficina
```

```
LEFT JOIN empleado ON oficina.codigo_oficina = empleado.codigo_oficina
```

```
GROUP BY ciudad;
```



The screenshot shows a SQL IDE window with a query editor at the top and a results grid at the bottom. The query editor contains the following SQL code:

```
SELECT ciudad, COUNT(codigo_empleado) AS num_empleados
FROM oficina
LEFT JOIN empleado ON oficina.codigo_oficina = empleado.codigo_oficina
GROUP BY ciudad;
```

The results grid displays the following data:

	ciudad	num_empleados
1	Barcelona	4
2	Boston	3
3	Londres	3
4	Madrid	4
5	Paris	3
6	San Francisco	2
7	Sydney	3
8	Talavera de la Reina	6
9	Tokyo	3

The IDE interface includes a toolbar with various icons for navigation and execution, and a status bar at the bottom indicating "9 row(s) fetched - 144ms, on 2023-05-03 at 19:11:55".