

SUBCONSULTAS SOBRE BD. EMPLEADOS Y BD. VENTAS

SUBCONSULTAS BD EMPLEADOS

Modelo entidad/relación



Realizar las siguientes consultas:

Con operadores básicos de comparación

1. Devuelve un listado con todos los empleados que tiene el departamento de Sistemas.
(Sin utilizar INNER JOIN).

```
SELECT *  
FROM empleados  
WHERE departamento = (SELECT codigo FROM departamentos WHERE nombre =  
    'Sistemas');
```

2. Devuelve el nombre del departamento con mayor presupuesto y la cantidad que tiene asignada.

```
SELECT nombre, presupuesto  
FROM departamentos  
WHERE presupuesto = (SELECT MAX(presupuesto) FROM departamentos);
```

3. Devuelve el nombre del departamento con menor presupuesto y la cantidad que tiene asignada.

```
SELECT nombre, presupuesto  
FROM departamentos  
WHERE presupuesto = (SELECT MIN(presupuesto) FROM departamentos);
```

Subconsultas con ALL y ANY

1. Devuelve el nombre del departamento con mayor presupuesto y la cantidad que tiene asignada. Sin hacer uso de MAX, ORDER BY ni LIMIT.

```
SELECT nombre, presupuesto  
FROM departamentos  
WHERE presupuesto >= ALL (SELECT presupuesto FROM departamentos);
```

2. Devuelve el nombre del departamento con menor presupuesto y la cantidad que tiene asignada. Sin hacer uso de MIN, ORDER BY ni LIMIT.

```
SELECT nombre, presupuesto
```

```
FROM departamentos
```

```
WHERE presupuesto <= ALL (SELECT presupuesto FROM departamentos);
```

3. Devuelve los nombres de los departamentos que tienen empleados asociados.
(Utilizando ALL o ANY).

```
SELECT nombre
```

```
FROM departamentos
```

```
WHERE codigo = ANY (SELECT departamento FROM empleados);
```

4. Devuelve los nombres de los departamentos que no tienen empleados asociados.
(Utilizando ALL o ANY).

```
SELECT nombre
```

```
FROM departamentos
```

```
WHERE codigo <> ALL (SELECT departamento FROM empleados);
```

Subconsultas con IN y NOT IN

1. Devuelve los nombres de los departamentos que tienen empleados asociados.
(Utilizando IN o NOT IN).

```
SELECT nombre
```

```
FROM departamentos
```

```
WHERE codigo IN (SELECT departamento FROM empleados);
```

2. Devuelve los nombres de los departamentos que no tienen empleados asociados.
(Utilizando IN o NOT IN).

```
SELECT nombre
```

```
FROM departamentos
```

```
WHERE codigo NOT IN (SELECT departamento FROM empleados);
```

Subconsultas con EXISTS y NOT EXISTS

1. Devuelve los nombres de los departamentos que tienen empleados asociados.
(Utilizando EXISTS o NOT EXISTS).

```
SELECT nombre
```

```
FROM departamentos d
```

```
WHERE EXISTS (SELECT 1 FROM empleados e WHERE e.departamento = d.codigo);
```

2. Devuelve los nombres de los departamentos que tienen empleados asociados.
(Utilizando EXISTS o NOT EXISTS)

```
SELECT nombre
```

```
FROM departamentos d
```

```
WHERE NOT EXISTS (SELECT 1 FROM empleados e WHERE e.departamento =  
d.codigo);
```

SUBCONSULTAS BD VENTAS

Modelo entidad/relación



Realizar las siguientes consultas:

Con operadores básicos de comparación

1. Devuelve un listado con todos los pedidos que ha realizado Adela Salas Díaz. (Sin utilizar INNER JOIN).

```
SELECT *  
FROM pedidos  
WHERE cliente = (SELECT codigo FROM clientes WHERE nombre = 'Adela Salas  
Díaz');
```

2. Devuelve el número de pedidos en los que ha participado el comercial Daniel Sáez Vega. (Sin utilizar INNER JOIN)

```
SELECT COUNT(*)  
FROM pedidos  
WHERE comercial = (SELECT codigo FROM comerciales WHERE nombre = 'Daniel  
Sáez Vega');
```

3. Devuelve los datos del cliente que realizó el pedido más caro en el año 2019. (Sin utilizar INNER JOIN)

```
SELECT *  
FROM clientes  
WHERE codigo = (SELECT cliente FROM pedidos WHERE año = 2019 ORDER BY  
total DESC LIMIT 1);
```

4. Devuelve la fecha y la cantidad del pedido de menor valor realizado por el cliente Pepe Ruiz Santana.

```
SELECT fecha, total  
FROM pedidos  
WHERE cliente = (SELECT codigo FROM clientes WHERE nombre = 'Pepe Ruiz  
Santana')  
ORDER BY total ASC  
LIMIT 1;
```

5. Devuelve un listado con los datos de los clientes y los pedidos, de todos los clientes que han realizado un pedido durante el año 2017 con un valor mayor o igual al valor medio de los pedidos realizados durante ese mismo año.

```

SELECT *
FROM clientes
WHERE codigo IN (
    SELECT cliente
    FROM pedidos
    WHERE año = 2017 AND total >= (SELECT AVG(total) FROM pedidos WHERE año =
    2017)
);

```

Subconsultas con ALL y ANY

1. Devuelve el pedido más caro que existe en la tabla pedido sin hacer uso de MAX, ORDER BY ni LIMIT.

```

SELECT *
FROM pedidos
WHERE total >= ALL (SELECT total FROM pedidos);

```

2. Devuelve un listado de los clientes que no han realizado ningún pedido. (Utilizando ANY o ALL).

```

SELECT nombre
FROM clientes
WHERE codigo <> ALL (SELECT cliente FROM pedidos);

```

3. Devuelve un listado de los comerciales que no han realizado ningún pedido. (Utilizando ANY o ALL).

```

SELECT nombre
FROM comerciales
WHERE codigo <> ALL (SELECT comercial FROM pedidos);

```

Subconsultas con IN y NOT IN

1. Devuelve un listado de los clientes que no han realizado ningún pedido. (Utilizando IN o NOT IN).

```

SELECT nombre
FROM clientes
WHERE codigo NOT IN (SELECT cliente FROM pedidos);

```

2. Devuelve un listado de los comerciales que no han realizado ningún pedido. (Utilizando IN o NOT IN).

```

SELECT nombre
FROM comerciales
WHERE codigo NOT IN (SELECT comercial FROM pedidos);

```

Subconsultas con EXISTS y NOT EXISTS

1. Devuelve un listado de los clientes que no han realizado ningún pedido. (Utilizando EXISTS o NOT EXISTS).

```
SELECT nombre
```

```
FROM clientes c
```

```
WHERE NOT EXISTS (SELECT 1 FROM pedidos p WHERE p.cliente = c.codigo);
```

2. Devuelve un listado de los comerciales que no han realizado ningún pedido. (Utilizando EXISTS o NOT EXISTS).

```
SELECT nombre
```

```
FROM comerciales c
```

```
WHERE NOT EXISTS (SELECT 1 FROM pedidos p WHERE p.comercial = c.codigo);
```