

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 e.nombre
from empleado e
WHERE e.id_departamento = (
    SELECT d.id
    FROM departamento d
    WHERE d.nombre = "sistemas"
);
```

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

```
SELECT nombre, presupuesto
from departamento
WHERE presupuesto = (
    SELECT MAX(presupuesto) from departamento
)
```

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

```

SELECT nombre, presupuesto
from departamento
WHERE presupuesto = (
    SELECT MIN(presupuesto) from departamento
);

```

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 departamento
WHERE presupuesto > ALL(
    SELECT presupuesto
    FROM departamento
    WHERE nombre != departamento.nombre
)

```

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 departamento
WHERE presupuesto < ALL(
    SELECT presupuesto
    FROM departamento
    WHERE nombre != departamento.nombre
);

```

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

```

SELECT nombre
from departamento
WHERE id = ANY(
    SELECT id_departamento
    from empleado
);

```

4. Devuelve los nombres de los departamentos que no tienen empleados asociados.

(Utilizando ALL o ANY).

```
SELECT nombre
from departamento
WHERE id != all(
    SELECT id_departamento
    from empleado
);
```

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 departamento
WHERE id in (
    SELECT id_departamento
    from empleado
)
```

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

```
SELECT nombre
from departamento
WHERE id not in (
    SELECT id_departamento
    from empleado
);
```

Subconsultas con EXISTS y NOT EXISTS

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

```
SELECT d.nombre
from departamento d
WHERE EXISTS (
    SELECT 1
    from empleado e
```

```
WHERE e.id_departamento = d.id
);
```

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

```
SELECT d.nombre
from departamento d
WHERE not EXISTS (
    SELECT 1
    from empleado e
    WHERE e.id_departamento = d.id
);
```

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 pedido
WHERE id_cliente = (
    SELECT id
    from cliente
    WHERE nombre = "Adela" AND apellido1 = "Salas" AND apellido2 = "Díaz"
);
```

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

```
SELECT COUNT(*)
```

```

from pedido
WHERE id_comercial = (
    SELECT id
    from comercial
    WHERE nombre = "Daniel" and apellido1 = "Sáez" AND apellido2 = "Díaz"
);

```

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

```

select *
from cliente
WHERE id = (
    SELECT id_cliente
    from pedido
    WHERE Year(fecha) = 2019 AND total = (
        SELECT max(total)
        from pedido
        WHERE year(fecha) = 2019
    )
);

```

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

```

    SELECT fecha, total
from pedido
WHERE id_cliente = (
    SELECT id
    from cliente
    WHERE nombre = "Pepe" and apellido1 = "Ruiz" AND apellido2 = "Santana"
);

```

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 cliente
WHERE id IN(
    SELECT id_cliente
    from pedido
    WHERE year(fecha) = 2017 AND total >= (
        SELECT AVG(total)

```

```

        from pedido
        WHERE year(fecha) = 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 pedido
WHERE total > ALL (
    SELECT total
    from pedido
where total is not null
);

```

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

```

SELECT *
FROM cliente
WHERE id != ALL (
    SELECT id_cliente
    FROM pedido
)

```

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

```

SELECT *
from comercial
WHERE id != ALL(
    SELECT id_comercial
    from pedido
);

```

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 *  
FROM cliente  
WHERE id not in (  
    SELECT id_cliente  
    from pedido  
);
```

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

```
SELECT *  
FROM comercial  
WHERE id not in (  
    SELECT id_comercial  
    from pedido  
);
```

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 *  
FROM cliente c  
WHERE not EXISTS (  
    SELECT p.id_cliente  
    FROM pedido p  
    where p.id_cliente = c.id  
);
```

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

```
select *  
from comercial c  
WHERE not EXISTS (  
    SELECT p.id_comercial
```

```
from pedido p
WHERE p.id_comercial = c.id
)
```