Las subconsultas son la estructura más complejas de trabajar, muy a menudo una solución mediante subconsultas se puede resolver mediante relaciones normales con JOIN, LEFT JOIN, RIGHT JOIN O INNER JOIN. Para ello debes de dividir el problema en varias partes más pequeñas e ir siguiendo la relación.  
Te paso los enlaces de la página oficial de MySQL para que puedas profundizar con subconsultas. [**https://dev.mysql.com/doc/refman/8.0/en/subqueries.html**](https://dev.mysql.com/doc/refman/8.0/en/subqueries.html) y [**https://www.mysqltutorial.org/mysql-subquery/**](https://www.mysqltutorial.org/mysql-subquery/)

Te paso algunos ejercicios complementarios que tratan sobre la misma actividad de la unidad 2.  
Relaciones y agrupaciones  
1. Obtén el nombre de los clientes que han realizado pedidos después del 01/01/2023.

SELECT p.idCliente, c.nombre, c.apellido,

COUNT(\*) as total\_pedidos

FROM pedidos p

JOIN clientes c ON p.idCliente = c.idCliente

WHERE fecha >= "2023-01-01"

GROUP BY idCliente;

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

2. Encuentra el nombre y el precio promedio de los productos en cada categoría.

SELECT c.nombre,

AVG(precio) as precio\_medio

FROM productos p

JOIN categorias c ON p.idCategoria = c.idCategoria

GROUP BY nombre

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

3. Obtén el total de ventas (suma de los totales de los pedidos) para cada cliente.

SELECT c.idCliente, c.nombre as nombre\_cliente, c.apellido as apellido,

SUM(p.total) as total\_ventas

FROM clientes c

JOIN pedidos p ON c.idCliente = p.idPedido

GROUP BY c.idCliente, c.nombre

ORDER BY total\_ventas DESC;

Tabla

Descripción generada automáticamente con confianza media

4. Muestra la cantidad de productos vendidos en cada pedido.

SELECT idPedido,

SUM(cantidad) as cantidad\_total

FROM lineas\_pedidos

group by idPedido

ORDER BY cantidad\_total DESC

Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente

5. Obtén el nombre del cliente y el número de pedidos realizados por cada cliente.

SELECT c.idCliente, c.nombre, c.apellido,

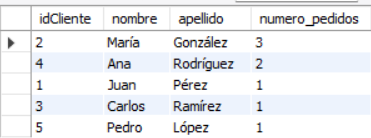
COUNT(p.idPedido) as numero\_pedidos

FROM clientes c

RIGHT JOIN pedidos p ON c.idCliente = p.idCliente

group by idCliente, nombre, apellido

ORDER BY numero\_pedidos DESC



Subconsultas  
1. Obtén el nombre y el correo electrónico de todos los clientes cuyos correos electrónicos no aparezcan en ningún pedido.

select nombre, email

from clientes

where idCliente not in (select c.idCliente

from clientes c

join pedidos p on c.idCliente = p.idCliente)

Interfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamente

2. Encuentra el nombre y el precio de los productos que se han vendido en pedidos con un descuento superior al 10%.

select distinct nombre, precio

from productos

where precio in (select precioUnitario

from lineas\_pedidos

where descuento > (precioUnitario - (precioUnitario \* 10 / 100)));

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

3. Obtén el nombre de las categorías que no tienen ningún producto asociado.

select nombre as categorias\_sin\_productos

from categorias

where idCategoria not in (select idCategoria

from productos);

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

4. Encuentra el nombre de los clientes que han realizado más de 3 pedidos.

select c.idCliente as id, c.nombre as nombre, c.apellido as apellido,

sum(cantidad) as total\_pedidos

from clientes c

join pedidos p on c.idCliente = p.idCliente

join lineas\_pedidos l on p.idPedido = l.idPedido

where l.idPedido in (select idPedido

from lineas\_pedidos

where cantidad > 3)

group by id, nombre, apellido

Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

5. Obtén el nombre de las categorías de productos que tienen un precio promedio superior a 75.

select nombre

from categorias

where idCategoria in (select idCategoria

from productos

group by idCategoria

having avg(precio) > avg(precio) \* 75 / 100);

Tabla

Descripción generada automáticamente

6. Obtén la lista de clientes que han realizado pedidos, mostrando sus nombres y correos

select idCliente, nombre, apellido, email

from clientes

where idCliente in (select idCliente

from pedidos);

Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente

7. Encuentra el nombre del cliente que ha gastado más dinero en total.

SELECT c.idCliente as id, c.nombre as nombre, c.apellido as apellido, p.total as total\_gastado

from clientes as c

join pedidos as p on c.idCliente = p.idCliente

where p.total in (select max(total)

from pedidos);



8. Encuentra el nombre de la categoría con la menor cantidad de productos.

-- 8. Encuentra el nombre de la categoría con la menor cantidad de productos.

USE pedidos;

SELECT

id, nombreCategoria, cantidadProductos

FROM

(SELECT

c.idCategoria AS id,

c.nombre AS nombreCategoria,

(SELECT

COUNT(\*)

FROM

productos p

WHERE

p.idCategoria = c.idCategoria) AS cantidadProductos

FROM

categorias c) subconsulta

WHERE

cantidadProductos = (SELECT

MIN(totalProductos)

FROM

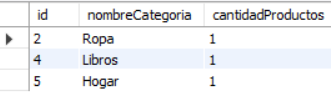
(SELECT

COUNT(\*) AS totalProductos

FROM

productos

GROUP BY idCategoria) subconsulta2);

  
  
Para la subconsultas, los ejercicios 7 y 8 son los más complejos. los restantes mira siempre de sacar las dos consultas apartes y luego mira de unirlas.  
Por ejemplo, del ejercicio 1 de subconsultas, primero mira como se obtiene el nombre y email de todos los clientes, y luego como se obtiene el email del cliente que ha realizado pedidos. Ya solo tienes que unir las dos consultas para buscar aquellos que NO aparecen en la segunda consulta.