

# Subconsultas

---

## Autores

- Jose Alejandro Gonzalez Ortiz
- Isaac Damian Arrieta Mercado

## A. Con operadores básicos de comparación:

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

```
select nombre_cliente
from cliente
where limite_credito = (select max(limite_credito) from cliente);
```

```
mysql> select nombre_cliente
->      from cliente
->      where limite_credito = (select max(limite_credito) from cliente);
+-----+
| nombre_cliente |
+-----+
| Tendo Garden   |
+-----+
1 row in set (0,00 sec)
```

2. Devuelve el nombre del producto que tenga el precio de venta más caro.

```
select nombre
from producto
where precio_venta = (select max(precio_venta) from producto);
```

```
mysql> select nombre_cliente
->      from cliente
->      where limite_credito = (select max(limite_credito) from cliente);
+-----+
| nombre_cliente |
+-----+
| Tendo Garden   |
+-----+
1 row in set (0,00 sec)
```

3. Devuelve el nombre del producto del que se han vendido más unidades. (Tenga en cuenta que tendrá que calcular cuál es el número total de unidades que se han vendido de cada producto a partir de los datos de la tabla detalle\_pedido. Una vez que sepa cuál es el código del producto, puede obtener su nombre fácilmente.)

```

SELECT p.nombre, dp.cantidad
FROM producto p
INNER JOIN detalle_pedido as dp ON p.codigo_producto =
dp.codigo_producto
WHERE cantidad = (SELECT MAX(cantidad) from detalle_pedido);

```

```

mysql> SELECT p.nombre, dp.cantidad
-> FROM producto p
-> INNER JOIN detalle_pedido as dp ON p.codigo_producto = dp.codigo_producto
-> WHERE cantidad =(SELECT MAX(cantidad) from detalle_pedido);
+-----+-----+
| nombre                                | cantidad |
+-----+-----+
| Thymus Citriodra (Tomillo limón)     | 450      |
+-----+-----+
1 row in set (0,00 sec)

```

4. Los clientes cuyo límite de crédito sea mayor que los pagos que haya realizado. (Sin utilizar INNER JOIN).

```

SELECT c.codigo_cliente, c.nombre_cliente, c.limite_credito, t.total
FROM cliente c, (SELECT SUM(total) as total from pago) as t
WHERE limite_credito > t.total;

```

```

mysql> SELECT codigo_cliente, nombre_cliente, limite_credito
-> FROM cliente
-> WHERE limite_credito > (SELECT SUM(total) from pago);
+-----+-----+-----+
| codigo_cliente | nombre_cliente | limite_credito |
+-----+-----+-----+
| 5              | Tendo Garden  | 600000.00      |
| 14             | Dardena S.A.  | 321000.00      |
+-----+-----+-----+
2 rows in set (0,01 sec)

```

5. 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);

```

```
mysql> SELECT c.codigo_cliente, c.nombre_cliente, c.limite_credito, t.total
->      FROM cliente c, (SELECT SUM(total) as total from pago) as t
->      WHERE limite_credito > t.total;
+-----+-----+-----+-----+
| codigo_cliente | nombre_cliente | limite_credito | total |
+-----+-----+-----+-----+
|          5 | Tendo Garden  |    600000.00 | 197940.00 |
|          14 | Dardena S.A.  |    321000.00 | 197940.00 |
+-----+-----+-----+-----+
2 rows in set (0,00 sec)
```

6. Devuelve el producto que menos unidades tiene en stock.

```
SELECT nombre, cantidad_en_stock
FROM producto
WHERE cantidad_en_stock = (SELECT min(cantidad_en_stock) FROM
producto);
```

```
mysql> SELECT nombre, cantidad_en_stock
->      FROM producto
->      WHERE cantidad_en_stock = (SELECT min(cantidad_en_stock) FROM producto);
+-----+-----+
| nombre      | cantidad_en_stock |
+-----+-----+
| Brahea Armata |          0 |
+-----+-----+
1 row in set (0,00 sec)
```

7. Devuelve el nombre, los apellidos y el email de los empleados que están a cargo de Alberto Soria.

```
SELECT nombre, apellido1, apellido2, email
FROM empleado
WHERE nombre = 'Alberto' AND apellido1 = 'Soria';
```

```
mysql> SELECT nombre, apellido1, apellido2, email
->      FROM empleado
->      WHERE nombre = 'Alberto' AND apellido1 = 'Soria';
+-----+-----+-----+-----+
| nombre | apellido1 | apellido2 | email |
+-----+-----+-----+-----+
| Alberto | Soria    | Carrasco  | asoria@jardineria.es |
+-----+-----+-----+-----+
1 row in set (0,00 sec)
```

## B. Subconsultas con ALL y ANY

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

```
SELECT nombre_cliente
FROM cliente
```

```
WHERE limite_credito >= ALL(SELECT (limite_credito) FROM cliente );
```

```
mysql> SELECT nombre_cliente
->      FROM cliente
->      WHERE limite_credito >= ALL(SELECT (limite_credito) FROM cliente );
+-----+
| nombre_cliente |
+-----+
| Tendo Garden  |
+-----+
1 row in set (0,00 sec)
```

9. Devuelve el nombre del producto que tenga el precio de venta más caro.

```
SELECT nombre,precio_venta
FROM producto
WHERE precio_venta >= ALL (SELECT(precio_venta) FROM producto);
```

```
mysql> SELECT nombre,precio_venta
->      FROM producto
->      WHERE precio_venta >= ALL (SELECT(precio_venta) FROM producto);
+-----+-----+
| nombre          | precio_venta |
+-----+-----+
| Trachycarpus Fortunei |      462.00 |
+-----+-----+
1 row in set (0,00 sec)
```

10. Devuelve el producto que menos unidades tiene en stock.

```
SELECT nombre, cantidad_en_stock
FROM producto
WHERE cantidad_en_stock <= ALL (SELECT (cantidad_en_stock) FROM
producto);
```

```
mysql> SELECT nombre, cantidad_en_stock
->      FROM producto
->      WHERE cantidad_en_stock <= ALL (SELECT (cantidad_en_stock) FROM producto);
+-----+-----+
| nombre          | cantidad_en_stock |
+-----+-----+
| Brahea Armata   |          0        |
+-----+-----+
1 row in set (0,00 sec)
```

## C. Subconsultas con IN y NOT IN

11. Devuelve el nombre, apellido1 y cargo de los empleados que no representen a ningún cliente.

```
SELECT nombre, apellido1, puesto
FROM empleado
WHERE codigo_empleado NOT IN (SELECT codigo_empleado_rep_ventas FROM
cliente);
```

```
mysql> SELECT nombre, apellido1, puesto
-> FROM empleado
-> WHERE codigo_empleado NOT IN (SELECT codigo_empleado_rep_ventas FROM cliente);
```

nombre	apellido1	puesto
Marcos	Magaña	Director General
Ruben	López	Subdirector Marketing
Alberto	Soria	Subdirector Ventas
Maria	Solís	Secretaria
Juan Carlos	Ortiz	Representante Ventas
Carlos	Soria	Director Oficina
Hilario	Rodriguez	Representante Ventas
David	Palma	Representante Ventas
Oscar	Palma	Representante Ventas
Francois	Fignon	Director Oficina
Laurent	Serra	Representante Ventas
Hilary	Washington	Director Oficina
Marcus	Paxton	Representante Ventas
Nei	Nishikori	Director Oficina
Narumi	Riko	Representante Ventas
Takuma	Nomura	Representante Ventas
Amy	Johnson	Director Oficina
Larry	Westfalls	Representante Ventas
John	Walton	Representante Ventas
Kevin	Fallmer	Director Oficina

```
20 rows in set (0,00 sec)
```

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

```
select codigo_cliente, nombre_cliente, nombre_contacto,
apellido_contacto
from cliente
where cliente.codigo_cliente not in (select pago.codigo_cliente from
pago);
```

```
mysql> select codigo_cliente, nombre_cliente, nombre_contacto, apellido_contacto
-> from cliente
-> where cliente.codigo_cliente not in (select pago.codigo_cliente from pago);
```

codigo_cliente	nombre_cliente	nombre_contacto	apellido_contacto
6	Lasas S.A.	Antonio	Lasas
8	Club Golf Puerta del hierro	Paco	Lopez
10	DaraDistribuciones	David	Serrano
11	Madrileña de riegos	Jose	Tacaño
12	Lasas S.A.	Antonio	Lasas
17	Flowers, S.A	Beatriz	Fernandez
18	Naturajardin	Victoria	Cruz
20	Americh Golf Management SL	Mario	Suarez
21	Aloha	Cristian	Rodriguez
22	El Prat	Francisco	Camacho
24	Vivero Humanes	Federico	Gomez
25	Fuenla City	Tony	Muñoz Mena
29	Top Campo	Joseluis	Sanchez
31	Campohermoso	Luis	Jimenez
32	france telecom	Fraçois	Toulou
33	Musée du Louvre	Pierre	Delacroux
36	Flores S.L.	Antonio	Romero
37	The Magic Garden	Richard	Mcaín

18 rows in set (0,00 sec)

13. Devuelve un listado que muestre solamente los clientes que sí han realizado algún pago.

```
select codigo_cliente, nombre_cliente, nombre_contacto,
apellido_contacto
from cliente
where cliente.codigo_cliente in (select pago.codigo_cliente from pago);
```

```
mysql> select codigo_cliente, nombre_cliente, nombre_contacto, apellido_contacto
-> from cliente
-> where cliente.codigo_cliente in (select pago.codigo_cliente from pago);
```

codigo_cliente	nombre_cliente	nombre_contacto	apellido_contacto
1	GoldFish Garden	Daniel G	GoldFish
3	Gardening Associates	Anne	Wright
4	Gerudo Valley	Link	Flaute
5	Tendo Garden	Akane	Tendo
7	Beragua	Jose	Bermejo
9	Naturagua	Guillermo	Rengifo
13	Camunas Jardines S.L.	Pedro	Camunas
14	Dardena S.A.	Juan	Rodriguez
15	Jardin de Flores	Javier	Villar
16	Flores Marivi	Maria	Rodriguez
19	Golf S.A.	Luis	Martinez
23	Sotogrande	Maria	Santillana
26	Jardines y Mansiones Cactus SL	Eva María	Sánchez
27	Jardinerías Matías SL	Matías	San Martín
28	Agrojardin	Benito	Lopez
30	Jardineria Sara	Sara	Marquez
35	Tutifruti S.A	Jacob	Jones
38	El Jardin Viviente S.L	Justin	Smith

18 rows in set (0,01 sec)

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

```
select codigo_producto, nombre
from producto
where producto.codigo_producto not in (select
detalle_pedido.codigo_producto from detalle_pedido);
```

```
mysql> select codigo_producto, nombre
-> from producto
-> where producto.codigo_producto not in (select detalle_pedido.codigo_producto from detalle_pedido);
```

codigo_producto	nombre
FR-104	Olea-Olivos
FR-14	Calamondin Mini
FR-19	Camelia Blanco, Chrysler Rojo, Soraya Naranja,
FR-20	Landora Amarillo, Rose Gaujard bicolor blanco-rojo
FR-21	Kordes Perfect bicolor rojo-amarillo, Roundelay rojo fuerte
FR-24	Albaricoquero Corbato
FR-25	Albaricoquero Moniqui
FR-26	Albaricoquero Kurrot
FR-27	Cerezo Burlat
FR-28	Cerezo Picota
FR-30	Ciruelo R. Claudia Verde
FR-32	Ciruelo Golden Japan
FR-35	Ciruelo Claudia Negra
FR-38	Higuera Verdal
FR-39	Higuera Breva
FR-44	Melocotonero Spring Crest
FR-46	Melocotonero Federica
FR-49	Parra Uva de Mesa
FR-5	Mandarino -Plantón joven
FR-50	Peral Castell
FR-51	Peral Williams
FR-52	Peral Conference
FR-55	Olivo Cipresino
FR-59	Albaricoquero

15. Devuelve el nombre, apellidos, puesto y teléfono de la oficina de aquellos empleados que no sean representante de ventas de ningún cliente.

```
select nombre, apellido1, apellido2, puesto, oficina.telefono
from empleado, oficina
where empleado.codigo_empleado not in (select
cliente.codigo_empleado_rep_ventas from cliente)
and empleado.codigo_oficina = oficina.codigo_oficina
and empleado.puesto = 'Representante Ventas';
```



```
mysql> select codigo_oficina, telefono
->   from oficina
->   where oficina.codigo_oficina not in (select empleado.codigo_oficina from empleado
->   where empleado.codigo_empleado in (select cliente.codigo_empleado_rep_ventas from cliente
->   where cliente.codigo_cliente in (select pedido.codigo_cliente from pedido
->   where pedido.codigo_pedido in (select detalle_pedido.codigo_pedido from detalle_pedido
->   where detalle_pedido.codigo_producto in (select producto.codigo_producto from producto
->   where producto.nombre = 'Frutales'))));
```

codigo_oficina	telefono
BCN-ES	+34 93 3561182
BOS-USA	+1 215 837 0825
LON-UK	+44 20 78772041
MAD-ES	+34 91 7514487
PAR-FR	+33 14 723 4404
SFC-USA	+1 650 219 4782
SYD-AU	+61 2 9264 2451
TAL-ES	+34 925 867231
TOK-JP	+81 33 224 5000

9 rows in set (0,01 sec)

16. Devuelve las oficinas donde no trabajan ninguno de los empleados que hayan sido los representantes de ventas de algún cliente que haya realizado la compra de algún producto de la gama Frutales.

```
Select codigo_oficina
from oficina
where oficina.codigo_oficina not in (select empleado.codigo_oficina
from empleado
where empleado.puesto = 'Representante Ventas'
and empleado.codigo_empleado in (select
cliente.codigo_empleado_rep_ventas from cliente
where cliente.codigo_cliente in (select pedido.codigo_cliente from
pedido
where pedido.codigo_pedido in (select detalle_pedido.codigo_pedido from
detalle_pedido
where detalle_pedido.codigo_producto in (select
producto.codigo_producto from producto
where producto.gama = 'Frutales'))))));
```

```
mysql> Select codigo_oficina
->   from oficina
->   where oficina.codigo_oficina not in (select empleado.codigo_oficina from empleado
->   where empleado.puesto = 'Representante Ventas'
->   and empleado.codigo_empleado in (select cliente.codigo_empleado_rep_ventas from cliente
->   where cliente.codigo_cliente in (select pedido.codigo_cliente from pedido
->   where pedido.codigo_pedido in (select detalle_pedido.codigo_pedido from detalle_pedido
->   where detalle_pedido.codigo_producto in (select producto.codigo_producto from producto
->   where producto.gama = 'Frutales'))));
```

codigo_oficina
LON-UK
PAR-FR
TOK-JP

3 rows in set (0,01 sec)



17. Devuelve un listado con los clientes que han realizado algún pedido, pero no han realizado ningún pago.

```
select codigo_cliente, nombre_cliente, nombre_contacto,
apellido_contacto
from cliente
where cliente.codigo_cliente in (select pedido.codigo_cliente from
pedido)
and cliente.codigo_cliente not in (select pago.codigo_cliente from
pago);
```

```
mysql> select codigo_cliente, nombre_cliente, nombre_contacto, apellido_contacto
-> from cliente
-> where cliente.codigo_cliente in (select pedido.codigo_cliente from pedido)
-> and cliente.codigo_cliente not in (select pago.codigo_cliente from pago);
+-----+-----+-----+-----+
| codigo_cliente | nombre_cliente | nombre_contacto | apellido_contacto |
+-----+-----+-----+-----+
|          36   | Flores S.L.   | Antonio        | Romero            |
+-----+-----+-----+-----+
1 row in set (0,00 sec)
```

## D. Subconsultas con EXISTS y NOT EXISTS

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

```
select codigo_cliente, nombre_cliente, nombre_contacto,
apellido_contacto
from cliente
where exists (select pago.codigo_cliente from pago where
cliente.codigo_cliente = pago.codigo_cliente);
```

```
mysql> select codigo_cliente, nombre_cliente, nombre_contacto, apellido_contacto
-> from cliente
-> where exists (select pago.codigo_cliente from pago where cliente.codigo_cliente = pago.codigo_cliente);
+-----+-----+-----+-----+
| codigo_cliente | nombre_cliente | nombre_contacto | apellido_contacto |
+-----+-----+-----+-----+
|          1   | GoldFish Garden | Daniel G       | GoldFish          |
|          3   | Gardening Associates | Anne          | Wright            |
|          4   | Gerudo Valley   | Link          | Flaute            |
|          5   | Tendo Garden   | Akane         | Tendo             |
|          7   | Beragua        | Jose          | Bermejo           |
|          9   | Naturagua      | Guillermo     | Rengifo           |
|         13   | Camunas Jardines S.L. | Pedro        | Camunas           |
|         14   | Dardena S.A.   | Juan          | Rodriguez          |
|         15   | Jardin de Flores | Javier        | Villar            |
|         16   | Flores Marivi  | Maria         | Rodriguez          |
|         19   | Golf S.A.      | Luis          | Martinez           |
|         23   | Sotogrande     | Maria         | Santillana        |
|         26   | Jardines y Mansiones Cactus SL | Eva María    | Sánchez           |
|         27   | Jardinerías Matías SL | Matías       | San Martín        |
|         28   | Agrojardin     | Benito        | Lopez              |
|         30   | Jardineria Sara | Sara          | Marquez           |
|         35   | Tutifrutí S.A | Jacob         | Jones             |
|         38   | El Jardín Viviente S.L | Justin       | Smith             |
+-----+-----+-----+-----+
18 rows in set (0,00 sec)
```

19. Devuelve un listado que muestre solamente los clientes que sí han realizado algún pago.

```

select codigo_cliente, nombre_cliente, nombre_contacto,
apellido_contacto
from cliente
where not exists (select pago.codigo_cliente from pago where
cliente.codigo_cliente = pago.codigo_cliente);

```

```

mysql> select codigo_cliente, nombre_cliente, nombre_contacto, apellido_contacto
-> from cliente
-> where not exists (select pago.codigo_cliente from pago where cliente.codigo_cliente = pago.codigo_cliente);
+-----+-----+-----+-----+
| codigo_cliente | nombre_cliente | nombre_contacto | apellido_contacto |
+-----+-----+-----+-----+
| 6 | Lasas S.A. | Antonio | Lasas |
| 8 | Club Golf Puerta del hierro | Paco | Lopez |
| 10 | DaraDistribuciones | David | Serrano |
| 11 | Madrileña de riegos | Jose | Tacaño |
| 12 | Lasas S.A. | Antonio | Lasas |
| 17 | Flowers, S.A | Beatriz | Fernandez |
| 18 | Naturajardin | Victoria | Cruz |
| 20 | Americh Golf Management SL | Mario | Suarez |
| 21 | Aloha | Cristian | Rodrigez |
| 22 | El Prat | Francisco | Camacho |
| 24 | Vivero Humanes | Federico | Gomez |
| 25 | Fuenla City | Tony | Muñoz Mena |
| 29 | Top Campo | Joseluis | Sanchez |
| 31 | Campohermoso | Luis | Jimenez |
| 32 | france telecom | FraAgois | Toulou |
| 33 | Musée du Louvre | Pierre | Delacroux |
| 36 | Flores S.L. | Antonio | Romero |
| 37 | The Magic Garden | Richard | Mcain |
+-----+-----+-----+-----+
18 rows in set (0,00 sec)

```

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

```

select codigo_producto, nombre
from producto
where not exists (select detalle_pedido.codigo_producto from
detalle_pedido where detalle_pedido.codigo_producto =
producto.codigo_producto);

```

```

mysql> select codigo_producto, nombre
-> from producto
-> where not exists (select detalle_pedido.codigo_producto from detalle_pedido where detalle_pedido.codigo_producto = producto.codigo_producto);
+-----+-----+
| codigo_producto | nombre |
+-----+-----+
| FR-104 | Olea-Olivos |
| FR-14 | Calamondin Mini |
| FR-19 | Camelia Blanco, Chrysler Rojo, Soraya Naranja, |
| FR-20 | Landora Amarillo, Rose Gaujard bicolor blanco-rojo |
| FR-21 | Kordes Perfect bicolor rojo-amarillo, Roundelay rojo fuerte |
| FR-24 | Albaricoquero Corbato |
| FR-25 | Albaricoquero Moniqui |
| FR-26 | Albaricoquero Kurrot |
| FR-27 | Cerezo Burlat |
| FR-28 | Cerezo Picota |
| FR-30 | Ciruelo R. Claudia Verde |
| FR-32 | Ciruelo Golden Japan |
| FR-35 | Ciruelo Claudia Negra |
| FR-38 | Higuera Verdal |
| FR-39 | Higuera Breva |
| FR-44 | Melocotonero Spring Crest |
| FR-46 | Melocotonero Federica |
| FR-49 | Parra Uva de Mesa |
| FR-5 | Mandarino -Plantón joven |
| FR-50 | Peral Castell |
+-----+-----+

```

21. Devuelve un listado de los productos que han aparecido en un pedido alguna vez.

```

select codigo_producto, nombre
from producto

```

```
where exists (select detalle_pedido.codigo_producto from detalle_pedido
where detalle_pedido.codigo_producto = producto.codigo_producto);
```

```
mysql> select codigo_producto, nombre
-> from producto
-> where exists (select detalle_pedido.codigo_producto from detalle_pedido where detalle_pedido.codigo_producto = producto.codigo_producto);
```

codigo_producto	nombre
11679	Sierra de Poda 400MM
21636	Pala
22225	Rastrillo de Jardín
30310	Azadón
AR-001	Ajedrea
AR-002	Lavándula Dentata
AR-003	Mejorana
AR-004	Melissa
AR-005	Mentha Sativa
AR-006	Petrosillum Hortense (Peregil)
AR-007	Salvia Mix
AR-008	Thymus Citriodra (Tomillo limón)

## E. Subconsultas correlacionadas

### Consultas variadas:

22. Devuelve el listado de clientes indicando el nombre del cliente y cuántos pedidos ha realizado. Tenga en cuenta que pueden existir clientes que no han realizado ningún pedido.

```
Select c.nombre_cliente , p.total
from cliente c
left join (select codigo_cliente, count(codigo_cliente) as total from
pedido group by codigo_cliente) p on c.codigo_cliente = p.codigo_cliente;
```

```
mysql> Select c.nombre_cliente , p.total
-> from cliente c
-> left join (select codigo_cliente, count(codigo_cliente) as total from pedido group by codigo_cliente) p on c.codigo_cliente = p.codigo_cliente;
```

nombre_cliente	total
GoldFish Garden	11
Gardening Associates	9
Gerudo Valley	5
Tendo Garden	5
Lasas S.A.	NULL
Beragua	5
Club Golf Puerta del hierro	NULL
Naturagua	5
DaraDistribuciones	NULL
Madriña de riegos	NULL
Lasas S.A.	NULL
Camunas Jardines S.L.	5
Dardena S.A.	5
Jardin de Flores	5
Flores Marivi	10
Flowers, S.A	NULL
Naturajardin	NULL
Golf S.A.	5
Americh Golf Management SL	NULL
Aloha	NULL
El Prat	NULL
Sotogrande	5
Vivero Humanes	NULL
Fuenla City	NULL
Jardines y Mansiones Cactus SL	5
Jardineras Matias SL	5
Agrojardin	5
Top Campo	NULL
Jardineria Sara	10
Campohermoso	NULL

23. Devuelve un listado con los nombres de los clientes y el total pagado por cada uno de ellos. Tenga en cuenta que pueden existir clientes que no han realizado ningún pago.

```
Select c.codigo_cliente, p.total
from cliente c
left join (select codigo_cliente, sum(total) as total from pago group
```

```
by codigo_cliente) p on c.codigo_cliente = p.codigo_cliente
Order by c.codigo_cliente desc;
```

```
mysql> Select c.codigo_cliente, p.total
-> from cliente c
-> left join (select codigo_cliente, sum(total) as total from pago group by codigo_cliente) p on c.codigo_cliente = p.codigo_cliente
-> Order by c.codigo_cliente desc;
```

codigo_cliente	total
38	1171.00
37	NULL
36	NULL
35	3321.00
33	NULL
32	NULL
31	NULL
30	7863.00
29	NULL
28	8489.00
27	10972.00
26	18846.00
25	NULL
24	NULL
23	272.00
22	NULL
21	NULL
20	NULL
19	232.00
18	NULL

24. Devuelve el nombre de los clientes que hayan hecho pedidos en 2008 ordenados alfabéticamente de menor a mayor.

```
select nombre_cliente
from cliente
where exists (select pedido.codigo_cliente from pedido where
pedido.codigo_cliente = cliente.codigo_cliente and pedido.fecha_pedido like
'2008%')
order by nombre_cliente;
```

```
mysql> select nombre_cliente
-> from cliente
-> where exists (select pedido.codigo_cliente from pedido where pedido.codigo_cliente = cliente.codigo_cliente and pedido.fecha_pedido like '2008%')
-> order by nombre_cliente;
```

nombre_cliente
Camunas Jardines S.L.
Dardena S.A.
El Jardin Viviente S.L
Flores Marivi
Flores S.L.
Gerudo Valley
GoldFish Garden
Jardin de Flores
Jardines y Mansiones Cactus SL
Tendo Garden
Tutifruti S.A

11 rows in set (0.00 sec)

25. Devuelve el nombre del cliente, el nombre y primer apellido de su representante de ventas y el número de teléfono de la oficina del representante de ventas, de aquellos clientes que no hayan realizado ningún pago.

```
select c.nombre_cliente, e.nombre, e.apellido1, o.telefono
from cliente c
left join empleado e on c.codigo_empleado_rep_ventas =
e.codigo_empleado
left join oficina o on e.codigo_oficina = o.codigo_oficina
where not exists (select pago.codigo_cliente from pago where
c.codigo_cliente = pago.codigo_cliente);
```

```

mysql> select c.nombre_cliente, e.nombre, e.apellido1, o.telefono
-> from cliente c
-> left join empleado e on c.codigo_empleado_rep_ventas = e.codigo_empleado
-> left join oficina o on e.codigo_oficina = o.codigo_oficina
-> where not exists (select pago.codigo_cliente from pago where c.codigo_cliente = pago.codigo_cliente);
+-----+-----+-----+-----+
| nombre_cliente | nombre | apellido1 | telefono |
+-----+-----+-----+-----+
| Lasas S.A. | Mariano | López | +34 91 7514487 |
| Club Golf Puerta del hierro | Emmanuel | Magaña | +34 93 3561182 |
| DaraDistribuciones | Emmanuel | Magaña | +34 93 3561182 |
| Madrileña de riegos | Emmanuel | Magaña | +34 93 3561182 |
| Lasas S.A. | Mariano | López | +34 91 7514487 |
| Flowers, S.A | Felipe | Rosas | +34 925 867231 |
| Naturajardin | Julian | Bellinelli | +61 2 9264 2451 |
| Americh Golf Management SL | José Manuel | Martinez | +34 93 3561182 |
| Aloha | José Manuel | Martinez | +34 93 3561182 |
| El Prat | José Manuel | Martinez | +34 93 3561182 |
| Vivero Humanes | Julian | Bellinelli | +61 2 9264 2451 |
| Fuenla City | Felipe | Rosas | +34 925 867231 |
| Top Campo | Felipe | Rosas | +34 925 867231 |
| Campohermoso | Julian | Bellinelli | +61 2 9264 2451 |
| france telecom | Lionel | Narvaez | +33 14 723 4404 |
| Musée du Louvre | Lionel | Narvaez | +33 14 723 4404 |
| Flores S.L. | Michael | Bolton | +1 650 219 4782 |
| The Magic Garden | Michael | Bolton | +1 650 219 4782 |
+-----+-----+-----+-----+
18 rows in set (0,00 sec)

```

26. Devuelve el listado de clientes donde aparezca el nombre del cliente, el nombre y primer apellido de su representante de ventas y la ciudad donde está su oficina.

```

select c.nombre_cliente, e.nombre as empleado_nombre, e.apellido1 as
empleado_apellido1, o.ciudad
from cliente c
inner join empleado e on c.codigo_empleado_rep_ventas =
e.codigo_empleado
inner join (select codigo_oficina, ciudad from oficina ) o on
e.codigo_oficina = o.codigo_oficina;

```

```
mysql> select c.nombre_cliente, e.nombre as empleado_nombre, e.apellido1 as empleado_apellido1, o.ciudad
-> from cliente c
-> inner join empleado e on c.codigo_empleado_rep_ventas = e.codigo_empleado
-> inner join (select codigo_oficina, ciudad from oficina ) o on e.codigo_oficina = o.codigo_oficina;
```

nombre_cliente	empleado_nombre	empleado_apellido1	ciudad
GoldFish Garden	Walter Santiago	Sanchez	San Francisco
Gardening Associates	Walter Santiago	Sanchez	San Francisco
Gerudo Valley	Lorena	Paxton	Boston
Tendo Garden	Lorena	Paxton	Boston
Lasas S.A.	Mariano	López	Madrid
Beragua	Emmanuel	Magaña	Barcelona
Club Golf Puerta del hierro	Emmanuel	Magaña	Barcelona
Naturagua	Emmanuel	Magaña	Barcelona
DaraDistribuciones	Emmanuel	Magaña	Barcelona
Madrileña de riegos	Emmanuel	Magaña	Barcelona
Lasas S.A.	Mariano	López	Madrid
Camunas Jardines S.L.	Mariano	López	Madrid
Dardena S.A.	Mariano	López	Madrid
Jardin de Flores	Julian	Bellinelli	Sydney
Flores Marivi	Felipe	Rosas	Talavera de la Reina
Flowers, S.A	Felipe	Rosas	Talavera de la Reina
Naturajardin	Julian	Bellinelli	Sydney
Golf S.A.	José Manuel	Martinez	Barcelona
Americh Golf Management SL	José Manuel	Martinez	Barcelona
Aloha	José Manuel	Martinez	Barcelona
El Prat	José Manuel	Martinez	Barcelona
Sotogrande	José Manuel	Martinez	Barcelona
Vivero Humanes	Julian	Bellinelli	Sydney
Fuenla City	Felipe	Rosas	Talavera de la Reina
Jardines y Mansiones Cactus SL	Lucio	Campoamor	Madrid
Jardineras Matías SL	Lucio	Campoamor	Madrid
Agrojardin	Julian	Bellinelli	Sydney
Top Campo	Felipe	Rosas	Talavera de la Reina
Jardineria Sara	Felipe	Rosas	Talavera de la Reina
Campohermoso	Julian	Bellinelli	Sydney
france telecom	Lionel	Narvaez	Paris

27. Devuelve el nombre, apellidos, puesto y teléfono de la oficina de aquellos empleados que no sean representante de ventas de ningún cliente.

```
select e.nombre, e.apellido1, e.apellido2, e.puesto, o.telefono
from empleado e
inner join (select codigo_oficina, telefono from oficina ) o on
e.codigo_oficina = o.codigo_oficina
where not exists (select codigo_empleado from cliente where
codigo_empleado_rep_ventas = e.codigo_empleado);
```

```
mysql> select e.nombre, e.apellido1, e.apellido2, e.puesto, o.telefono
->   from empleado e
->   inner join (select codigo_oficina, telefono from oficina ) o on e.codigo_oficina = o.codigo_oficina
->   where not exists (select codigo_empleado from cliente where codigo_empleado_rep_ventas = e.codigo_empleado);
```

nombre	apellido1	apellido2	puesto	telefono
David	Palma	Aceituno	Representante Ventas	+34 93 3561182
Oscar	Palma	Aceituno	Representante Ventas	+34 93 3561182
Hilary	Washington		Director Oficina	+1 215 837 0825
Marcus	Paxton		Representante Ventas	+1 215 837 0825
Amy	Johnson		Director Oficina	+44 20 78772041
Larry	Westfalls		Representante Ventas	+44 20 78772041
John	Walton		Representante Ventas	+44 20 78772041
Carlos	Soria	Jimenez	Director Oficina	+34 91 7514487
Hilario	Rodriguez	Huertas	Representante Ventas	+34 91 7514487
Francois	Fignon		Director Oficina	+33 14 723 4404
Laurent	Serra		Representante Ventas	+33 14 723 4404
Kevin	Fallmer		Director Oficina	+61 2 9264 2451
Marcos	Magaña	Perez	Director General	+34 925 867231
Ruben	López	Martinez	Subdirector Marketing	+34 925 867231
Alberto	Soria	Carrasco	Subdirector Ventas	+34 925 867231
Maria	Solis	Jerez	Secretaria	+34 925 867231
Juan Carlos	Ortiz	Serrano	Representante Ventas	+34 925 867231
Nei	Nishikori		Director Oficina	+81 33 224 5000
Narumi	Riko		Representante Ventas	+81 33 224 5000
Takuma	Nomura		Representante Ventas	+81 33 224 5000

20 rows in set (0,00 sec)

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

```
Select o.ciudad, e.numero_empleados
from oficina o
left join (select codigo_oficina , count(codigo_empleado) as
numero_empleados from empleado group by codigo_oficina) e
on e.codigo_oficina = o.codigo_oficina;
```

```
mysql> Select o.ciudad, e.numero_empleados
->   from oficina o
->   left join (select codigo_oficina , count(codigo_empleado) as numero_empleados from empleado group by codigo_oficina) e
->   on e.codigo_oficina = o.codigo_oficina;
```

ciudad	numero_empleados
Barcelona	4
Boston	3
Londres	3
Madrid	4
Paris	3
San Francisco	2
Sydney	3
Talavera de la Reina	6
Tokyo	3

9 rows in set (0,00 sec)