

## CONSULTAS JARDINERÍA

### Consultas multitabla (introducción de: select, from, inner join y where)

1) Pedidos del 2009

```
select * from pedidos where fechapedido like "2009%"
```

```
select * from pedidos where fechapedido between "2009-01-01" and "2009-12-31"
```

```
select * from pedidos where fechapedido >= "2009-01-01" and fechapedido<="2009-12-31"
```

```
select * from pedidos where year(fechapedido)= 2009
```

2) Pedidos del 2009 o del 2006

```
select * from pedidos where fechapedido like "2009%" or fechapedido like "2006%"
```

3) Pedidos del 2009 o del 2006 del cliente de nombre "Tendo Garden"

```
select CodigoPedido, pedidos.CodigoCliente, FechaPedido
from pedidos inner join clientes on Pedidos.CodigoCliente=Clientes.CodigoCliente
where (FechaPedido like "2009%" or FechaPedido like "2006%") and NombreCliente like "Tendo Garden"
```

4) Pedidos de entre junio de 2006 y mayo de 2008 hechos por clientes de España.

```
select CodigoPedido, pedidos.CodigoCliente, FechaPedido
from pedidos inner join clientes on Pedidos.CodigoCliente=Clientes.CodigoCliente
where FechaPedido >= "2006/06/01" and FechaPedido <="2008/05/31" and Pais like "España"
```

6) Oficinas que no sean de EEUU ubicadas en una ciudad que empiece por A o acabe por A

```
select CodigoOficina
from Oficinas
where Pais not like "EEUU" and (Ciudad like "A%" or Ciudad like "%A")
```

7) Nombre de los productos para los que se haya pedido en algún pedido una cantidad entre 20 y 30.

```
select distinct Nombre
from Productos p inner join DetallePedidos dp on p.CodigoProducto=dp.CodigoProducto
where cantidad between 20 and 30
```

8) Precio de cada línea de detalle pedido señalando, precio unitario, cantidad, código de pedido y nombre del producto.

```
select CodigoPedido, nombre, PrecioUnidad, Cantidad, PrecioUnidad* Cantidad
```

**from Productos p inner join DetallePedidos dp on p.CodigoProducto=dp.CodigoProducto**

9) Pedidos que tengan algún producto de la gama “Frutales”

```
select distinct Pedidos.*  
from Pedidos inner join detallepedidos on Pedidos.CodigoPedido=DetallePedido.CodigoPedido  
inner join productos on detallepedidos.CodigoProducto= productos.CodigoProducto  
where Gama like "Frutales"
```

10) Empleados que tienen algún cliente de “España” que hayan hecho algún pedido en el 2008.

```
select distinct e.*  
from Empleados e inner join Clientes c on e.CodigoEmpleado= c.CodigoEmpleadoRepVentas  
inner join Pedidos p on p.CodigoCliente=c.CodigoCliente  
where c.Pais like "España" and FechaPedido like "2008%"
```

11) Nombre y apellidos de cada empleado en una misma columna y lo mismo para su jefe.

```
select concat(emp.Nombre," ", emp.Apellido1," ", emp.Apellido2) as NombreCompleto,  
concat (jefe.nombre," ", jefe.Apellido1," ", jefe.Apellido2) as NombreCompletoJ  
from empleados emp inner join Empleados jefe on emp.CodigoJefe= jefe.CodigoEmpleado
```

12) Empleados (nombre completo) que trabajen en la oficina 'SYD-AU' o que tengan el cargo de “Representante de Ventas” cuyo jefe se llame “Amy”

```
select concat(emp.Nombre," ", emp.Apellido1), emp.Puesto, emp.CodigoOficina,jefe.Nombre  
from Empleados emp inner join Empleados jefe on emp.CodigoJefe= jefe.CodigoEmpleado  
where emp.CodigoOficina like "SYD-AU" or emp.Puesto like "Representante Ventas" and  
jefe.Nombre like "Amy%"
```

13) Oficinas de España que no empiecen por B que tengan clientes de en la misma ciudad o en distinto país

```
select CodigoOficina  
from Oficinas inner join clientes on (clientes.Ciudad=Oficinas.ciudad) or (clientes.pais !=  
Oficinas.pais)  
where Oficinas.Pais like "España" and CodigoOficina like !"B%"
```

## Funciones de agregado

### Max, Min, Count, Sum(numérico), Avg(numérico)

14) Número de pedidos y número de pedidos entregados

```
select count(*), count(FechaEntrega)
from pedidos
```

15) Precio máximo, mínimo y medio de los productos para los que se haya realizado algún pedido en 2008

```
select max(PrecioVenta), min(PrecioVenta), avg(PrecioVenta)
from detallepedidos dp inner join productos pr on dp.CodigoProducto= pr.CodigoProducto
inner join pedidos p on p.CodigoPedido= dp.CodigoPedido
where FechaPedido like "2008%"
```

16) Fecha del primer y el último pedido.

```
select max(FechaPedido), min(FechaPedido)
from pedidos
```

17) Número de pedidos realizados en 2006 y fecha del primer pedido en ese año.

```
Select count(CodigoPedido), min(FechaPedido)
from pedidos
where FechaPedido like ("2006%")
```

18) Importe del pedido de CodigoPedido 1

```
select sum(PrecioUnidad*Cantidad)
from detallepedidos
where CodigoPedido = 1
```

19) Importe total de los pagos realizados por el cliente de nombre “Tendo Garden”.

```
select sum(pagos.Cantidad)
from clientes inner join pagos on clientes.CodigoCliente= pagos.CodigoCliente
where NombreCliente like "Tendo Garden"
```

20) Precio del producto más caro, del más barato y media del precio de los productos

```
select distinct max(PrecioVenta), min(PrecioVenta), avg(PrecioVenta)
from productos
```

21) Importe total del pedido de código 15

```
select sum(PrecioUnidad*Cantidad)
from detallepedidos
```

**where** *CodigoPedido* like ("15")

22) Número total de empleados

```
select count(*)  
from Empleados
```

23) Número total de empleados que tengan jefe

```
select count(*)  
from Empleados  
where CodigoJefe is not null
```

```
select count(CodigoJefe)  
from Empleados
```

25) Número de pedidos del 2007.

```
select count(CodigoPedido)  
from pedidos  
where FechaPedido like "2007%"
```

26) Número medio de días que tarda en servirse un pedido ( de aquellos que se han servido)

```
select avg(datediff(FechaEntrega,FechaPedido))  
from pedidos  
where fechaentrega is not null
```

## GROUP BY

27) Número de pedidos que ha hecho cada cliente (sólo CódigoCliente)

```
select CodigoCliente, count(CodigoPedido)  
from pedidos  
group by CodigoCliente
```

28) Número de pedidos que han hecho los clientes de España o Spain

```
select NombreCliente, count(*)  
from pedidos inner join clientes on pedidos.CodigoCliente= clientes.CodigoCliente  
where pais like "Spain" or pais like "España"  
group by Clientes.CodigoClientes ,NombreCliente
```

29) Número de pedidos que ha hecho cada cliente viendo el nombre y la dirección del cliente

```
select count(CodigoPedido), clientes.NombreCliente, clientes.LineaDireccion1  
from pedidos inner join clientes on pedidos.CodigoCliente= clientes.CodigoCliente  
group by NombreCliente, LineaDireccion1, clientes.CodigoCliente
```

30) Por cada categoría, precio máximo, mínimo y medio de sus productos

```
select gama, max(PrecioVenta), min(PrecioVenta), avg(PrecioVenta)
from productos
group by Gama
```

31) Fecha del primer pedido de cada cliente

```
select min(FechaPedido), CodigoCliente
from Pedidos
group by CodigoCliente
```

32) Importe total de cada pedido

```
select sum(cantidad*PrecioUnidad), CodigoPedido
from detallepedidos
group by CodigoPedido
```

33) Nombre completo del empleado y número de clientes que tiene a su cargo.

```
select concat(empleados.Nombre , empleados.Apellido1, empleados.Apellido2) nombreCompleto,
count(CodigoCliente)
from empleados inner join clientes on
empleados.CodigoEmpleado=clientes.CodigoEmpleadoRepVentas
group by concat(empleados.Nombre , empleados.Apellido1, empleados.Apellido2),
CodigoEmpleado
```

34) Nombre y dirección de cada oficina y número de empleados que tiene de las oficinas que no sean de España o Spain.

```
select oficinas.CodigoOficina, oficinas.LineaDireccion1, count(*)
from empleados inner join oficinas on empleados.CodigoOficina= oficinas.CodigoOficina
where Pais not like "Spain" and Pais not like "España"
group by oficinas.CodigoOficina, LineaDireccion1
```

35) Importe total de los pagos efectuados por cada cliente en cada año.

```
select clientes.CodigoCliente, NombreCliente, year(FechaPago), sum(Cantidad)
from pagos inner join clientes on clientes.CodigoCliente= pagos.CodigoCliente
group by clientes.CodigoCliente, NombreCliente, year(FechaPago)
```

36) Nombre del jefe y número de empleados que tiene a su cargo

```
select pl.CodigoEmpleado, concat(p1.nombre, " ", p1.apellido1), count(*)
from empleados em inner join empleados pl on em.CodigoJefe=pl.CodigoEmpleado
group by pl.CodigoEmpleado, concat(p1.nombre, " ", p1.apellido1)
```

37) Importe total de lo pedido cada año

```
select sum(PrecioUnidad*Cantidad), year(FechaPedido)
from detallepedidos inner join pedidos on detallepedidos.CodigoPedido= pedidos.CodigoPedido
group by year(FechaPedido)
```

38) Número de pedidos realizados por cada país ( por los clientes de un país).

```
select count(CodigoPedido), Pais
from pedidos inner join clientes on pedidos.CodigoCliente=clientes.CodigoCliente
group by Pais
```

39) Número de los pedidos realizados por cada país en cada año

```
select count(CodigoPedido), Pais, year(FechaPedido)
from pedidos inner join clientes on pedidos.CodigoCliente=clientes.CodigoCliente
group by Pais, year(FechaPedido)
```

## HAVING

40) Número de pedidos que ha hecho cada cliente viendo el nombre y la dirección del cliente, de los clientes que han hecho más de 3 pedidos.

```
select count(*), clientes.NombreCliente, clientes.LineaDireccion1
from pedidos inner join clientes on pedidos.CodigoCliente= pedidos.CodigoCliente
group by clientes.CodigoCliente, clientes.NombreCliente, clientes.LineaDireccion1
having count(*) > 3
```

41) Por cada categoría (gama), precio máximo, mínimo y medio de sus productos. De las categorías con menos de 10 productos

```
select gama, min(precioventa), max(precioventa), avg(precioventa), count(*)
from productos
group by gama
having count(*) < 10
```

42) Nombre completo del empleado y número de clientes que tiene a su cargo de los empleados que trabajen en oficinas de Madrid o Barcelona y tengan al menos tres clientes

```
select concat(Nombre, " ", Apellido1) NombreCompleto, count(*) "nº de clientes"
from empleados inner join clientes on empleados.CodigoEmpleado =
clientes.CodigoEmpleadoRepVentas
inner join oficinas on empleados.CodigoOficina = oficinas.codigoOficina
where oficinas.ciudad like "Madrid" or oficinas.ciudad like "Barcelona"
group by CodigoEmpleado, concat(Nombre, " ", Apellido1)
having count(*) > 2
```

43) Nombre y dirección de cada oficina y número de empleados que tiene de las oficinas que no sean de España o Spain y tengan menos de 10 empleados

```

select oficinas.CodigoOficina, LineaDireccion1, count(*)
from empleados inner join oficinas on empleados.CodigoOficina= oficinas.CodigoOficina
where Pais not like "Spain" and Pais not like "España"
group by oficinas.CodigoOficina, LineaDireccion1
having count(CodigoEmpleado)<10

```

44)Importe total de los pagos efectuados por cada cliente en cada año a partir del año 2007

```

select sum(Cantidad), CodigoCliente, year(FechaPago)
from Pagos
group by CodigoCliente, year(FechaPago)
having year(FechaPago)>=2007

```

(la restricción también podría haberse hecho en el WHERE)

45)Países en donde hay más de cinco clientes

```

select clientes.Pais
from clientes
group by Pais
having count(*)>5

```

46)Gamas con algún producto que valga más de 10 €.

```

select gama
from productos
group by gama
having max(PrecioVenta)>10

```

47)Fecha del primer pedido de cada cliente que haya hecho algún pedido en el 2008 o después  
(Mejor con having porque se evita la subconsulta)

```

select min(FechaPedido), NombreCliente, max(FechaPedido)
from pedidos inner join clientes on pedidos.CodigoCliente = clientes.CodigoCliente
group by clientes.CodigoCliente, NombreCliente
having max(year(FechaPedido))>=2008)

```

## SUBCONSULTAS

48)Fecha del primer pedido de cada cliente que haya hecho algún pedido en el 2008

```

select min(FechaPedido), NombreCliente, max(FechaPedido)
from pedidos inner join clientes on pedidos.CodigoCliente = clientes.CodigoCliente
where pedidos.CodigoCliente in (select CodigoCliente from pedidos
                                where year(fechapedido)>=2008)

```

**group by** clientes.CodigoCliente, NombreCliente

```
select min(FechaPedido), NombreCliente, max(FechaPedido),count(*)
from Pedidos p inner join Clientes on p.CodigoCliente = Clientes.CodigoCliente
where exists(select * from Pedidos pp
              where pp.FechaPedido like "2008%" and
p.CodigoCliente=pp.CodigoCliente)
group by Clientes.CodigoCliente, NombreCliente
```

49)Pedidos que se realizaron en la fecha en que se realizó algún pago

```
select *
from pedidos
where FechaPedido in(select fechapago from pagos)
```

NB: Sería equivalente a IN poner =ANY

```
select *
from Pedidos
where exists(select * from Pagos
              where FechaPedido=FechaPago)
```

50)Selecciona oficinas situadas en ciudades donde no está ubicado ningún cliente.

```
select *
from Oficinas
where Oficinas.Ciudad not in(select Clientes.Ciudad from clientes)
```

NB: Sería equivalente a NOT IN poner <>ALL

```
select *
from Oficinas
where NOT EXISTS (select * from clientes
                  where Clientes.Ciudad=Oficinas.Ciudad)
```

51)Nombre y precio del producto más caro

```
select Nombre, CodigoProducto, PrecioVenta
from productos
where PrecioVenta >=all(select PrecioVenta from productos where PrecioVenta is not null)
```

```
select Nombre, CodigoProducto, PrecioVenta
from productos
where PrecioVenta =(select max(PrecioVenta) from productos where PrecioVenta is not null)
```



NB: el filtro “**where PrecioVenta is not null**” solo es necesario si el campo PrecioVenta no tiene la restricción de VNN. Como está definido en el scrip Jardinería como: PrecioVenta numeric(15,2) NOT NULL ; NO ES NECESARIO PONERLO.

```
select Nombre, CodigoProducto, PrecioVenta
from productos p1
where NOT EXISTS(select * from productos p2
                  where p2.PrecioVenta> p1.PrecioVenta)
```

52)Precio de los productos que tienen un precio de venta mayor al del precio medio de cualquier gama.

```
select CodigoProducto, PrecioVenta
from productos
where PrecioVenta >all(select avg(PrecioVenta) from productos group by gama)
```

```
select CodigoProducto, PrecioVenta
from productos p1
where not exists(select avg(PrecioVenta) from productos p2
                  group by gama
                  having avg(p2.PrecioVenta)>p1.PrecioVenta)
```

53)Datos del empleado que tiene más clientes

```
select CodigoEmpleado, nombre, apellido1,count(*)
from empleados inner join clientes on
empleados.CodigoEmpleado=Clientes.CodigoEmpleadoRepVentas
group by CodigoEmpleado, nombre, apellido1
having count(*) >=all(select count(*)
                      from clientes group by CodigoEmpleadoRepVentas)
```

```
SELECT COUNT(*), e.Nombre, e.Apellido1, e.Apellido2
FROM clientes c1 INNER JOIN empleados e ON c1.CodigoEmpleadoRepVentas =
e.CodigoEmpleado
GROUP BY c1.CodigoEmpleadoRepVentas, e.Nombre, e.Apellido1, e.Apellido2
HAVING NOT EXISTS( SELECT COUNT(*)
                    FROM clientes AS c2
                    GROUP BY CodigoEmpleadoRepVentas
                    HAVING COUNT(c1.CodigoEmpleadoRepVentas) <
                    COUNT(c2.CodigoEmpleadoRepVentas))
```

54)Datos de la Gama que tiene menos productos

NB: pide datos de la GAMA. Sintácticamente hay que poner todos los campos proyectados en el select en el GROUP BY . Aunque en MySql hubiera bastado poner “gama”que es la clave primaria

```

select gamasproductos.*, count(*)
from productos inner join gamasproductos on productos.gama=gamasproductos.Gama
group by gamasproductos.gama, descripciontexto, descripcionhtml, imagen
having count(*) <= all(select count(*)from productos
                        group by gama)

```

```

SELECT Gama, COUNT(CodigoProducto)
FROM productos AS p1
GROUP BY Gama
HAVING NOT EXISTS ( SELECT COUNT(*) FROM productos AS p2
                    GROUP BY Gama
                    HAVING COUNT(p1.CodigoProducto) > COUNT(p2.CodigoProducto))

```

55) Datos del pedido más barato

NB: Aquí no he puesto todos los campos proyectados en el select en el group by y en otro gestor de BD os podría dar problemas.

```

select p.*
from detallepedidos d inner join pedidos p on d.Codigopedido=p.codigopedido
Group by p.CodigoPedido
Having sum(PrecioUnidad*Cantidad) <= all (select sum(PrecioUnidad*Cantidad) from
                                           detallepedidos
                                           group by Codigopedido)

```

```

select p.*
from detallepedidos d inner join pedidos p on d.Codigopedido=p.codigopedido
Group by p.CodigoPedido
Having not exists (select sum(d1.PrecioUnidad*d1.Cantidad) from detallepedidos d1
                    group by d1.Codigopedido
                    having sum(d1.PrecioUnidad*d1.Cantidad)<sum(d.PrecioUnidad*d.Cantidad))

```

56) Año(s) en que se realizaron más pedidos.

```

select year(FechaPedido), count(*)
from pedidos
group by year(fechapedido)
having count(*)>=all(select count(*)
                    from pedidos
                    group by year(fechapedido))

```

```

SELECT YEAR(FechaPedido), COUNT(CodigoPedido)
FROM Pedidos AS p1
GROUP BY YEAR(FechaPedido)
HAVING NOT EXISTS (SELECT COUNT(*) FROM Pedidos AS p2
                    GROUP BY YEAR(FechaPedido)
                    HAVING COUNT(p1.CodigoPedido) < COUNT(p2.CodigoPedido))

```

57) Datos del empleado que tiene más subordinados.

```
select jef.* , count(*)  
from empleados sub inner join empleados jef on sub.CodigoJefe=jef.CodigoEmpleado  
group by jef.CodigoEmpleado  
having count(*)>=all(select count(*) from empleados  
                      group by CodigoJefe)
```

```
SELECT COUNT(*), e2.*  
FROM empleados AS e1 INNER JOIN empleados AS e2 ON e1.CodigoJefe = e2.CodigoEmpleado  
GROUP BY (e1.CodigoJefe)  
HAVING NOT EXISTS ( SELECT COUNT(*) FROM empleados AS e3  
                    GROUP BY(CodigoJefe)  
                    HAVING COUNT(e1.CodigoJefe) < COUNT(e3.CodigoJefe));
```

58) Datos de los productos que tienen un precio superior al de cualquier producto de la gama Herramientas.

```
SELECT *  
FROM productos AS p1  
WHERE not EXISTS ( SELECT *  
                   FROM productos AS p2  
                   WHERE p2.Gama LIKE 'Herramientas' and  
                   p2.PrecioVenta > p1.PrecioVenta)
```

```
select * from productos  
where precioventa>(select max(precioventa) from productos  
                  where gama like "Herramientas")
```

59) Productos que tengan un precio único (distinto a cualquier producto)

```
SELECT * FROM productos AS p1 WHERE NOT EXISTS( SELECT * FROM productos AS p2 WHERE  
p1.CodigoProducto <> p2.CodigoProducto AND p1.PrecioProducto = p2.PrecioProducto)  
select *  
from productos p  
where p.PrecioVenta <> all(select p1.PrecioVenta from Productos p1  
                        where p.CodigoProducto<>p1.CodigoProducto)
```

60) Fecha del primer pedido de cada cliente que haya hecho algún pedido en el 2008 o después.

```
select CodigoCliente, min(fechapedido)  
from clientes  
where CodigoCliente in (select CodigoCliente from pedidos  
                        where year(fechapedido)>=2008)  
group by CodigoCliente
```

61) Número de pedidos que ha hecho cada cliente viendo el nombre y la dirección del cliente, de los clientes que han hecho pedidos por un importe mayor que 1000€.

```
select count(*), clientes.NombreCliente, clientes.LineaDireccion1
```

```
from clientes inner join pedidos on clientes.CodigoCliente=pedidos.CodigoCliente
where clientes.CodigoCliente IN (select CodigoCliente
                                from pedidos inner join detallepedidos on
                                detallepedidos.codigopedido= pedidos.CodigoPedido
                                group by CodigoCliente
                                having sum(Cantidad*PrecioUnidad)>1000 )
group by clientes.CodigoCliente, clientes.NombreCliente, clientes.LineaDireccion1
```

```
select count( distinct p.CodigoPedido),c.nombrecliente,lineadireccion1
from clientes c inner join pedidos p on c.CodigoCliente=p.Codigocliente inner join detallepedidos d
on p.CodigoPedido=d.CodigoPedido
group by c.nombrecliente,lineadireccion1, c.CodigoCliente
having sum(cantidad*preciounidad)>1000
```