

---

# Soluciones al examen de consultas sobre la base de datos Jardinería

---

# 01.- Obtén un listado de los nombres de los distintos ciruelos (de cualquier tipo). Ordena el resultado en orden alfabético.

```
select distinct nombre
from producto
where nombre like '%Ciruelo%'
order by nombre;
```

```
+-----+
| nombre                |
+-----+
| Ciruelo                |
| Ciruelo Claudia Negra |
| Ciruelo Friar          |
| Ciruelo Golden Japan   |
| Ciruelo R. Claudia Verde |
| Ciruelo Reina C. De Ollins |
| Ciruelo Santa Rosa     |
+-----+
```

# 02.- Obtén un listado del nombre, y la cantidad en stock de los distintos productos de gama Ornamentales cuyo stock sea inferior a 50. Ordena el resultado primero por stock de forma descendente y luego por orden alfabético de nombre.

```
select distinct nombre, cantidad_en_stock
from producto
where gama = 'Ornamentales'
and cantidad_en_stock < 50
order by cantidad_en_stock desc, nombre;
```

```
+-----+-----+
| nombre                | cantidad_en_stock |
+-----+-----+
| Arbustos Mix Maceta   | 25                |
| Washingtonia Robusta | 15                |
| Yucca Jewel           | 15                |
| Zamia Furfuracaea     | 15                |
| Beucarnea Recurvata   | 5                 |
| Bismarckia Nobilis    | 4                 |
+-----+-----+
```

Beucarnea Recurvata	2	
Dracaena Drago	2	
Trachycarpus Fortunei	2	
Dracaena Drago	1	
Brahea Armata	0	

+-----+

# 03.- Obtén el nombre, apellido y teléfono del contacto del cliente cuyo nombre es Jardin de Flores. Se debe mostrar en una única columna llamada 'contacto' y todas las letras deben mostrarse en mayúsculas.

```
select upper(concat_ws(' ', nombre_contacto, apellido_contacto,
telefono)) as contacto
from cliente
where nombre_cliente = 'Jardin de Flores';
```

contacto	
----------	--

+-----+

JAVIER VILLAR 654865643	
-------------------------	--

+-----+

# 04.- Muestra los códigos y la fecha de los pedidos realizados entre enero y junio (ambos meses incluidos) de 2008, ordenados por fecha de más a menos reciente.

```
select codigo_pedido, fecha_pedido
from pedido
where fecha_pedido between '2008-01-01' and '2008-06-30'
order by fecha_pedido desc;
```

codigo_pedido	fecha_pedido	
---------------	--------------	--

+-----+

116	2008-06-28	
3	2008-06-20	
96	2008-03-20	
48	2008-03-17	
50	2008-03-17	
35	2008-03-10	
111	2008-03-05	
95	2008-01-04	

# 05.- Muestra el nombre y el total pagado de los cinco clientes que hayan pagado más en total.

```
select c.nombre_cliente, sum(pa.total)
from cliente as c join pago as pa
on c.codigo_cliente = pa.codigo_cliente
group by c.nombre_cliente
order by sum(pa.total) desc
limit 5;
```

```
+-----+-----+
|nombre_cliente          |sum(pa.total)|
+-----+-----+
|Gerudo Valley           |81849.00     |
|Tendo Garden            |23794.00     |
|Jardines y Mansiones Cactus SL|18846.00     |
|Jardin de Flores        |12081.00     |
|Jardinerías Matías SL   |10972.00     |
+-----+-----+
```

# 06.- Muestra los distintos nombres de productos que tengan siete caracteres.

```
select distinct nombre
from producto
where nombre like '_____';
```

```
+-----+
|nombre |
+-----+
|Ajedrea|
|Ciruelo|
|Granado|
|Higuera|
|Manzano|
|Nispero|
+-----+
```

# 07.- Muestra los distintos países de los clientes cuya región no se indique.

```
select distinct pais
from cliente
where region is null;
```

```
+-----+
|pais  |
+-----+
```

```
| USA      |
| France |
+-----+
```

# 08.- Haz un listado del nombre de los clientes y la forma de pago de los clientes que no hayan pagado con PayPal.

```
select c.nombre_cliente, pa.forma_pago
from cliente as c join pago as pa
on c.codigo_cliente = pa.codigo_cliente
where pa.forma_pago <> 'PayPal';
```

```
+-----+-----+
| nombre_cliente | forma_pago |
+-----+-----+
| Tendo Garden  | Transferencia |
| Beragua       | Cheque       |
+-----+-----+
```

# 09.- Muestra los distintos nombres de los productos de la gama 'Aromáticas' que hayan aparecido en los pedidos realizados y entregados en 2008.

```
select distinct pr.nombre
from producto as pr
  join detalle_pedido as dp
  on pr.codigo_producto = dp.codigo_producto
  join pedido as pe
  on dp.codigo_pedido = pe.codigo_pedido
where pr.gama = 'Aromáticas'
and pe.estado = 'Entregado'
and year(pe.fecha_pedido) = 2008;
```

```
+-----+
| nombre |
+-----+
| Ajedrea |
| Melissa |
| Mentha Sativa |
| Thymus Citriodra (Tomillo limón) |
| Thymus Vulgaris |
+-----+
```

# 10.- Muestra el nombre de los clientes cuyo nombre comience por G y que han rechazado pedidos, junto con los distintos comentarios que aparecen en el pedido. También deben

aparecer los clientes que no han hecho comentarios, en cuyo caso se deberá mostrar "Sin comentarios". Ordena los resultados alfabéticamente por nombre de cliente.

```
select distinct c.nombre_cliente, ifnull(pe.comentarios, 'Sin
comentarios')
from cliente as c
left join pedido as pe
on c.codigo_cliente = pe.codigo_cliente
where pe.estado = 'Rechazado'
and c.nombre_cliente like 'G%'
order by c.nombre_cliente;
```

```
+-----+-----+
|nombre_cliente      |ifnull(pe.comentarios, 'Sin comentarios')      |
+-----+-----+
|Gardening Associates|El cliente anula la operacion para adquirir mas producto|
|Gerudo Valley       |El pedido fue anulado por el cliente            |
|Gerudo Valley       |Limite de credito superado                      |
|GoldFish Garden     |El cliente carece de saldo en la cuenta asociada  |
|Golf S.A.           |Sin comentarios                                |
+-----+-----+
```

# 11.- Muestra los nombres de los clientes de la región de Madrid que no han realizado ningún pedido. Ordena el resultado por nombre de cliente.

```
select c.nombre_cliente
from cliente as c
left join pedido as pe
on c.codigo_cliente = pe.codigo_cliente
where pe.codigo_cliente is null
and c.region = 'Madrid'
order by c.nombre_cliente;
```

```
+-----+
|nombre_cliente      |
+-----+
|Campohermoso        |
|Club Golf Puerta del hierro|
|DaraDistribuciones  |
|Fuenla City         |
|Lasas S.A.          |
|Lasas S.A.          |
|Madrileña de riegos  |
|Naturajardin        |
|Top Campo           |
|Vivero Humanes      |
+-----+
```

# 12.- Calcula el total del importe del el pedido de código 90. Muéstralo en una columna llamada total\_pedido.

```
select sum(dp.precio_unidad * dp.cantidad) as total_pedido
from detalle_pedido as dp
where codigo_pedido = 90;
```

```
+-----+
|total_pedido|
+-----+
| 41.00      |
+-----+
```

# 13.- Muestra el nombre y el número de pedidos del cliente que más pedidos ha realizado.

```
select c.nombre_cliente, count(pe.codigo_cliente) as pedidos_cliente
from cliente as c
join pedido as pe
on c.codigo_cliente = pe.codigo_cliente
group by c.nombre_cliente
order by pedidos_cliente desc
limit 1;
```

```
+-----+-----+
|nombre_cliente |pedidos_cliente|
+-----+-----+
|GoldFish Garden|11            |
+-----+-----+
```

# 14.- Muestra los nombres de los clientes que han realizado algún pedido pero no han realizado ningún pago.

```
select distinct c.nombre_cliente
from cliente as c
join pedido as pe
on c.codigo_cliente = pe.codigo_cliente
left join pago as pa
on c.codigo_cliente = pa.codigo_cliente
where pa.codigo_cliente is null;
```

```
+-----+
|nombre_cliente|
+-----+
|Flores S.L.   |
+-----+
```

# 15.- Calcula el precio medio de todos los pedidos. Muestra el resultado con dos decimales y en una columna llamada precio\_medio.

```
select round(avg(dp.precio_unidad * dp.cantidad), 2) as precio_medio
from detalle_pedido as dp;
```

```

+-----+
|precio_medio|
+-----+
| 684.71      |
+-----+

```

# 16.- Muestra cuántos productos hay de cada gama.

```

select gama, count(gama)
from producto
group by gama;

```

```

+-----+-----+
|gama          |count(gama)|
+-----+-----+
|Aromáticas    |10          |
|Frutales      |108         |
|Herramientas  |4           |
|Ornamentales  |154         |
+-----+-----+

```

# 17.- Muestra el precio medio por unidad (con todos sus decimales) de los productos de la gama 'Frutales'.

```

select avg(dp.precio_unidad)
from producto as pr
join detalle_pedido as dp
on pr.codigo_producto = dp.codigo_producto
where pr.gama = 'Frutales';

```

```

+-----+
|avg(dp.precio_unidad)|
+-----+
|35.993243             |
+-----+

```

# 18.- Muestra el nombre del cliente que han realizado la mayor suma de pagos en total, junto con el número de pedidos que ha realizado. Se deben mostrar nombre, cuánto ha pagado en total y número de pedidos de este cliente.

```

select c.nombre_cliente, sum(pa.total), count(pe.codigo_cliente)
from cliente as c

```

```

join pago as pa
on c.codigo_cliente = pa.codigo_cliente
join pedido as pe
on c.codigo_cliente = pe.codigo_cliente
group by c.nombre_cliente
order by sum(pa.total) desc
limit 1;

```

```

+-----+-----+-----+
|nombre_cliente|sum(pa.total)|count(pe.codigo_cliente)|
+-----+-----+-----+
|Gerudo Valley |409245.00    |25                        |
+-----+-----+-----+

```

# 19.- Muestra el nombre de las gamas de productos de los que se han vendido en promedio al menos 50 unidades.

```

select pr.gama
from producto as pr
      join detalle_pedido as dp
on pr.codigo_producto = dp.codigo_producto
group by pr.gama
having avg(dp.cantidad) >= 50;

```

```

+-----+
|gama      |
+-----+
|Aromáticas|
+-----+

```

# 20.- Muestra el nombre de los clientes que han realizado pagos superiores a la media de todos los pagos. No se mostrarán nombres de clientes repetidos.

```

select distinct c.nombre_cliente, sum(pa.total)
from cliente as c
      join pago as pa
on c.codigo_cliente = pa.codigo_cliente
group by c.nombre_cliente
having sum(pa.total) > (select avg(pago.total)
                        from pago);

```

```

+-----+-----+-----+
|nombre_cliente          |sum(pa.total)|
+-----+-----+-----+
|Gardening Associates    |10926.00     |
|Gerudo Valley           |81849.00     |
|Tendo Garden           |23794.00     |
|Jardin de Flores       |12081.00     |
|Jardines y Mansiones Cactus SL|18846.00    |

```



Jardinerías Matías SL	10972.00	
Agrojardin	8489.00	
Jardineria Sara	7863.00	
+-----+-----+		

NOTA: la columna de la suma no es necesaria. Se considera válida mostrar únicamente la columna nombre\_cliente.

# 21.- Muestra el número de clientes que no tienen una región asignada.

```
select nombre_cliente
from cliente
where region is null;
```

+-----+	
nombre_cliente	
+-----+	
GoldFish Garden	
Gerudo Valley	
Tendo Garden	
france telecom	
Musée du Louvre	
+-----+	

# 22.- Calcula cuántos clientes tiene cada representante de ventas. Muestra el nombre de los representantes y a continuación el número de clientes que representa. Ordena el resultado por orden alfabético de representante.

```
select e.nombre, count(c.codigo_cliente)
from empleado as e join cliente as c
on e.codigo_empleado = c.codigo_empleado_rep_ventas
where puesto = "Representante ventas"
group by e.nombre
order by e.nombre;
```

+-----+-----+	
nombre	count(c.codigo_cliente)
+-----+-----+	
Felipe	5
José Manuel	5
Julian	5
Lionel	2
Lorena	2
Lucio	2
Mariano	4
Mariko	2
Walter Santiago	2
+-----+-----+	

# 23- Devuelve el nombre del cliente de aquellos clientes españoles que no hayan realizado ningún pago cuyo representante de ventas se llame Felipe.

```
select c.nombre_cliente
from cliente as c
left join pago as pa
on c.codigo_cliente = pa.codigo_cliente
join empleado as e
on c.codigo_empleado_rep_ventas = e.codigo_empleado
where pa.codigo_cliente is null
and c.pais = 'Spain'
and e.nombre = 'Felipe';
```

```
+-----+
|nombre_cliente|
+-----+
|Flowers, S.A  |
|Fuenla City   |
|Top Campo     |
+-----+
```

# 24.- Muestra un listado indicando todas las ciudades donde hay oficinas.

```
select distinct ciudad
from oficina;
```

```
+-----+
|ciudad          |
+-----+
|Barcelona        |
|Boston           |
|Londres          |
|Madrid           |
|Paris            |
|San Francisco    |
|Sydney           |
|Talavera de la Reina|
|Tokyo            |
+-----+
```

# 25.- Muestra cuantos pedidos tienen 1 línea, cuántos tienen 2 líneas, cuántos tienen 3 líneas y así sucesivamente.

```
select max(numero_linea), count(numero_linea)
from detalle_pedido
group by numero_linea
order by max(numero_linea);
```

```
+-----+-----+
|max(numero_linea)|count(numero_linea)|
```

1	116
2	82
3	58
4	30
5	19
6	8
7	4
8	1

# 26.- Muestra el promedio del límite de crédito que tienen los países en que residen los clientes que han realizado pagos en total por un importe inferior a la media del total de los pagos. Agrupa el resultado por países. Las columnas que debes mostrar son: país, promedio del límite de crédito por cada país y total de los pagos por cada país. Los datos numéricos deben mostrarse sin decimales.

```
select cliente.pais, round(avg(cliente.limite_credito)) ,
round(sum(pago.total))
from cliente join pago
on cliente.codigo_cliente = pago.codigo_cliente
group by cliente.pais
having sum(pago.total) < (select avg(total)
from pago);
```

pais	round(avg(cliente.limite_credito))	round(sum(pago.total))
Australia	9000	4492

# 27.- Muestra el nombre completo del director de la oficina de Sidney.

```
select concat_ws(' ', nombre, apellido1, apellido2)
from empleado
where codigo_oficina like 'SYD%'
and puesto = 'Director Oficina';
```

concat_ws(' ', nombre, apellido1, apellido2)
Kevin Fallmer

# 28.- Muestra nombre y apellidos (en una columna llamada empleados) con el email de los empleados que tienen un email de jardineria.es

```
select concat_ws(' ', nombre, apellido1, apellido2), email
from empleado
where email like '%jardineria.es';
```

<code>concat_ws(' ', nombre, apellido1, apellido2)</code>	<code>email</code>
Marcos Magaña Perez	marcos@jardineria.es
Ruben López Martinez	rlopez@jardineria.es
Alberto Soria Carrasco	asoria@jardineria.es
Maria Solís Jerez	msolis@jardineria.es
Felipe Rosas Marquez	frosas@jardineria.es
Juan Carlos Ortiz Serrano	cortiz@jardineria.es
Carlos Soria Jimenez	csoria@jardineria.es
Mariano López Murcia	mlopez@jardineria.es
Lucio Campoamor Martín	lcampoamor@jardineria.es
Hilario Rodriguez Huertas	hrodriguez@jardineria.es
Emmanuel Magaña Perez	manu@jardineria.es
David Palma Aceituno	dpalma@jardineria.es
Oscar Palma Aceituno	opalma@jardineria.es

# 29.- Muestra cuántos pedidos rechazados hay cada año. Ordena el resultado por año.

```
select year(fecha_pedido), count(estado)
from pedido
where estado = 'Rechazado'
group by year(fecha_pedido)
order by year(fecha_pedido);
```

<code>year(fecha_pedido)</code>	<code>count(estado)</code>
2007	1
2008	9
2009	14

# 30.- Muestra los proveedores que ofrecen productos que puedan describirse como limoneros o naranjos. Y, para cada uno de estos productos, muestra su nombre, el precio del proveedor, el precio de venta y el beneficio que obtiene el vendedor para cada producto.

```
select proveedor, nombre, precio_proveedor, precio_venta,
(precio_venta - precio_proveedor) as beneficio
```

```

from producto
where descripcion like '%limonero%' or descripcion like '%naranja%';

```

proveedor	nombre	precio_proveedor	precio_venta	beneficio
NaranjasValencianas.com	Limonero 2 años injerto	5.00	7.00	2.00
NaranjasValencianas.com	Limonero 30/40	80.00	100.00	20.00
NaranjasValencianas.com	Naranja -Plantón joven 1 año injerto	4.00	6.00	2.00
NaranjasValencianas.com	Naranja 2 años injerto	5.00	7.00	2.00
NaranjasValencianas.com	Naranja calibre 8/10	23.00	29.00	6.00
NaranjasValencianas.com	Limonero -Plantón joven	4.00	6.00	2.00
NaranjasValencianas.com	Limonero calibre 8/10	23.00	29.00	6.00