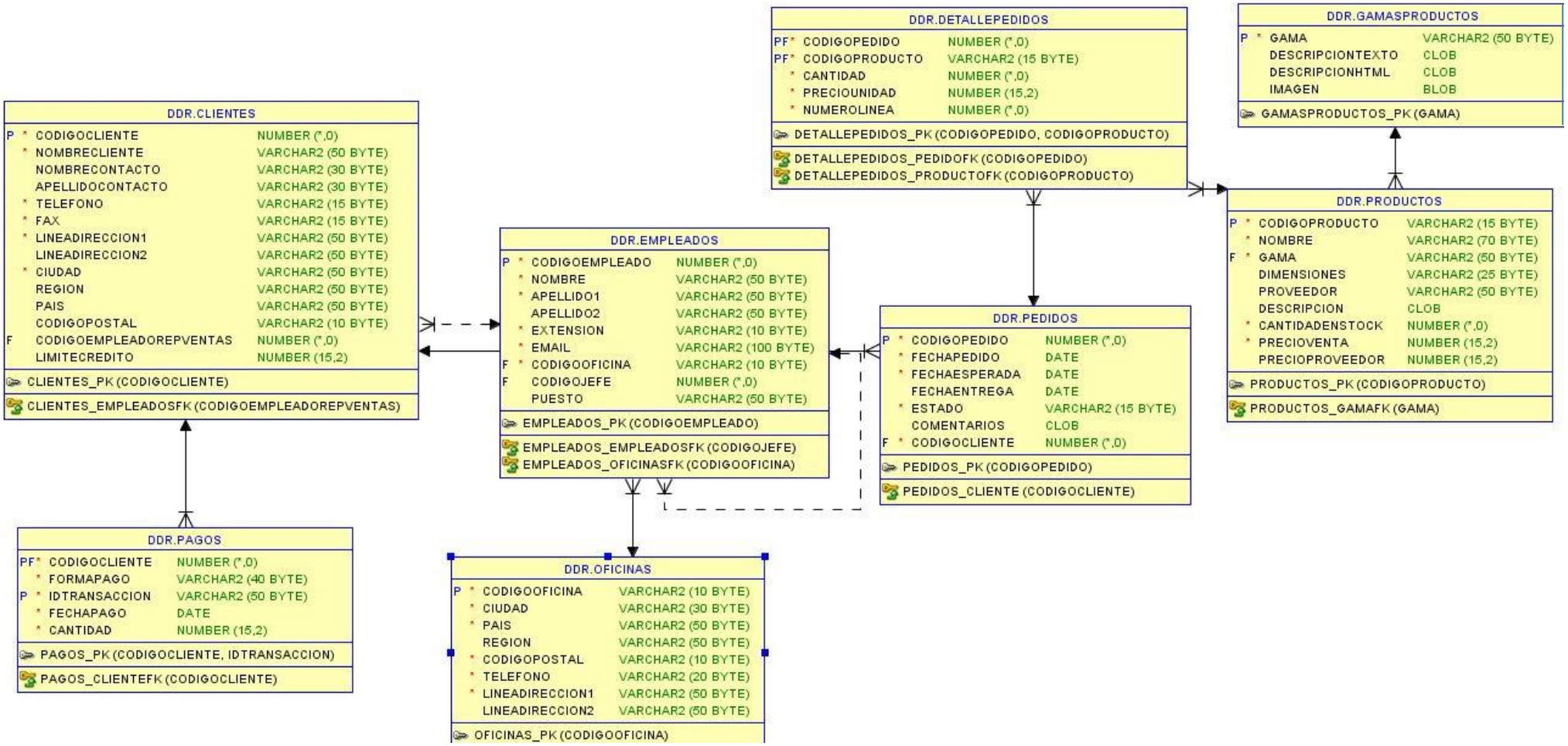
Ejercicios propuestos y resueltos de consultas Oracle (Jardinería)



1. Muestra la ciudad y el codigo postal de las oficinas de España.

```
select ciudad, codigopostal
from OFICINAS
where lower(pais) = 'españa';
```

```
--1. Muestra la ciudad y el codigo postal de las oficinas de España.

select ciudad, codigopostal
from OFICINAS
where lower(pais) = 'españa';
```

Salida de Script | Resultado de la Co...

SQL | Todas las Filas Recuperadas: 3 en 0,114 segundos

	CIUDAD	CODIGOPOSTAL
1	Barcelona	08019
2	Madrid	28032
3	Talavera de la Reina	45632

2. Obtener el nombre y apellidos del jefe de la empresa.

```
select nombre, apellido1, apellido2
from empleados
where codigojefe is null;
```

```
select nombre, apellido1, apellido2
from empleados
where codigojefe is null;
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 1 en 0,006 segundos

	NOMBRE	APELLIDO1	APELLIDO2
1	Marcos	Magaña	Perez

3. Mostrar a) el nombre y cargo de los empleados que sean directores de oficina. b) Y los que no sean directores de oficina.

Apartado a:

```
select nombre, puesto
from empleados
where lower(puesto) = 'director oficina';
```

```
select nombre, puesto
from empleados
where lower(puesto) = 'director oficina';
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 8 en 0,005 segundos

	NOMBRE	PUESTO
1	Carlos	Director Oficina
2	Emmanuel	Director Oficina
3	Francois	Director Oficina
4	Michael	Director Oficina
5	Hilary	Director Oficina
6	Nei	Director Oficina
7	Amy	Director Oficina
8	Kevin	Director Oficina

Nota: en algunas pantallas no se muestran todas las filas totales devueltas por cuestión de espacio

Apartado b:

```
select nombre, puesto
from empleados
where lower(puesto) <> 'director oficina';
```

```
select nombre, puesto
from empleados
where lower(puesto) <> 'director oficina';
```

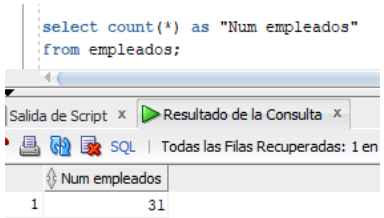
Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 23 en 0,008 se

	NOMBRE	PUESTO
1	Marcos	Director General
2	Ruben	Subdirector Marketing
3	Alberto	Subdirector Ventas
4	Maria	Secretaria
5	Felipe	Representante Ventas
6	Juan Carlos	Representante Ventas
7	Mariano	Representante Ventas
8	Lucio	Representante Ventas
9	Hilario	Representante Ventas
10	José Manuel	Representante Ventas
11	David	Representante Ventas
12	Oscar	Representante Ventas
13	Lionel	Representante Ventas
14	Laurent	Representante Ventas
15	Walter Santiago	Representante Ventas
16	Marcus	Representante Ventas
17	Lorena	Representante Ventas

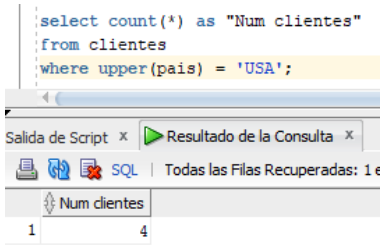
4. Muestra el número de empleados que hay en la empresa.

```
select count(*) as "Num empleados"
from empleados;
```



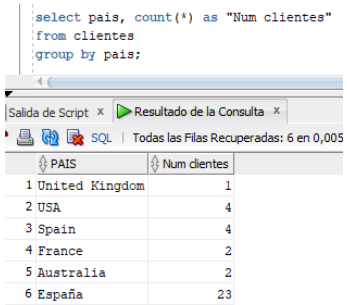
5. Muestra el número de clientes norteamericanos.

```
select count(*) as "Num clientes"
from clientes
where upper(pais) = 'USA';
```



6. Número de clientes de cada país.

```
select pais, count(*) as "Num clientes"
from clientes
group by pais;
```



7. Muestra el nombre del cliente y el nombre de su representante de ventas (si lo tiene).

```
select c.nombrecliente as "Nombre cliente", e.nombre as "Nombre representante"
from clientes c, empleados e
where c.codigoempleadorepventas=e.CODIGOEMPLEADO;
```

Hoja de Trabajo | Generador de Consultas

```
select c.nombrecliente as "Nombre cliente", e.nombre as "Nombre representante"
from clientes c, empleados e
where c.codigoempleadorepventas=e.CODIGOEMPLEADO;
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 36 en 0,011 segundos

	Nombre cliente	Nombre representante
1	Jardineria Sara	Felipe
2	Flowers, S.A	Felipe
3	Flores Marivi	Felipe
4	Top Campo	Felipe
5	Fuenla City	Felipe
6	Camunas Jardines S.L.	Mariano
7	Dardena S.A.	Mariano
8	Lasas S.A.	Mariano
9	Lasas S.A.	Mariano
10	Jardines y Mansiones CACTUS SL	Lucio
11	Jardinerías Matías SL	Lucio
12	Naturagua	Emmanuel
13	Club Golf Puerta del hierro	Emmanuel
14	Beragua	Emmanuel
15	DaraDistribuciones	Emmanuel
16	Madrileña de riegos	Emmanuel
17	Sotogrande	José Manuel
18	El Prat	José Manuel
19	Golf S.A.	José Manuel
20	AYMERICH GOLF MANAGEMENT, SL	José Manuel
21	Aloha	José Manuel
22	Musée du Louvre	Lionel

8. Nombre de los clientes que hayan hecho un pago en 2007

```
select distinct c.nombrecliente
from clientes c, pagos p
where c.CODIGOCLIENTE=p.CODIGOCLIENTE
and p.FECHAPAGO like '%/07';
```

```
select distinct c.nombrecliente
from clientes c, pagos p
where c.CODIGOCLIENTE=p.CODIGOCLIENTE
and p.FECHAPAGO like '%/07';
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 2 en 0,0

	NOMBRECLIENTE
1	Tutifruti S.A
2	Gerudo Valley

9. Los posibles estados de un pedido.

```
select distinct lower(estado)
from pedidos
```

```
select distinct lower(estado)
from pedidos
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperada

	LOWER(ESTADO)
1	rechazado
2	pediente
3	pendiente
4	entregado

10. Muestra el número de pedido, el nombre del cliente, la fecha de entrega y la fecha requerida de los pedidos que no han sido entregados a tiempo.

```
select p.codigopedido, c.nombrecliente, p.fechaentrega, p.fechaesperada
from clientes c, pedidos p
where c.codigocliente = p.CODIGOCLIENTE
and p.FECHAESPERADA < p.fechaentrega;
```

```
select p.codigopedido, c.nombrecliente, p.fechaentrega, p.fechaesperada
from clientes c, pedidos p
where c.codigocliente = p.CODIGOCLIENTE
and p.FECHAESPERADA < p.fechaentrega;
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 31 en 0,01 segundos

	CODIGOPEDIDO	NOMBRECLIENTE	FECHAENTREGA	FECHAESPERADA
1	9	DGPRODUCTIONS GARDEN	28/12/08	27/12/08
2	60	DGPRODUCTIONS GARDEN	28/12/08	27/12/08
3	28	Gardening Associates	20/02/09	17/02/09
4	68	Gardening Associates	20/02/09	17/02/09
5	32	Gerudo Valley	27/01/07	19/01/07
6	13	Beragua	15/01/09	14/01/09
7	16	Beragua	15/01/09	07/01/09
8	17	Beragua	11/01/09	09/01/09
9	22	Naturagua	13/01/09	11/01/09
10	18	Naturagua	07/01/09	06/01/09
11	31	Camunas Jardines S.L.	04/10/08	30/09/08
12	55	Dardena S.A.	11/01/09	10/01/09
13	40	Golf S.A.	13/03/09	10/03/09
14	42	Golf S.A.	27/03/09	23/03/09
15	38	Golf S.A.	07/03/09	06/03/09
16	39	Golf S.A.	09/03/09	07/03/09
17	46	Sotogrande	05/03/09	04/03/09
18	44	Sotogrande	30/03/09	27/03/09
19	43	Sotogrande	28/03/09	26/03/09
20	45	Sotogrande	07/03/09	04/03/09
21	92	Jardinerias Matias SL	03/05/09	30/04/09
22	123	Jardineria Sara	24/01/09	20/01/09
23	103	Jardineria Sara	24/01/09	20/01/09
24	106	Jardineria Sara	20/05/09	15/05/09

11. Muestra el código, nombre y gama de los productos que nunca se han pedido (detalle pedidos).

```
select p.codigoproducto, p.nombre, p.gama
from productos p, GAMASPRODUCTOS g
where p.gama=g.gama
and not exists (select CODIGOPRODUCTO from detallepedidos where codigoproducto=p.CODIGOPRODUCTO);
```

```
select p.codigoproducto, p.nombre, p.gama
from productos p, GAMASPRODUCTOS g
where p.gama=g.gama
and not exists (select CODIGOPRODUCTO from detallepedidos where codigoproducto=p.CODIGOPRODUCTO);
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 132 en 0,028 segundos

	CODIGOPRODUCTO	NOMBRE	GAMA
1	FR-44	Melocotonero Spring Crest	Frutales
2	OR-105	Mimosa Semilla Espectabilis	Ornamentales
3	OR-106	Mimosa Semilla Longifolia	Ornamentales
4	OR-107	Mimosa Semilla Floribunda 4 estaciones	Ornamentales
5	OR-108	Abelia Floribunda	Ornamentales
6	OR-109	Callistemom (Mix)	Ornamentales
7	OR-110	Callistemom (Mix)	Ornamentales
8	OR-111	Corylus Avellana \"Contorta\"	Ornamentales
9	OR-112	Escallonia (Mix)	Ornamentales
10	OR-113	Evonimus Emerald Gayeti	Ornamentales
11	OR-114	Evonimus Pulchellus	Ornamentales
12	OR-117	Hibiscus Syriacus \"Helene\" -Blanco-C.rojo	Ornamentales
13	OR-118	Hibiscus Syriacus \"Pink Giant\" Rosa	Ornamentales
14	OR-121	Lonicera Nitida \"Maigrum\"	Ornamentales
15	OR-124	Prunus pisardii	Ornamentales
16	OR-126	Weigelia \"Bristol Ruby\"	Ornamentales
17	OR-131	Leptospermum formado PIRAMIDE	Ornamentales
18	OR-132	Leptospermum COPA	Ornamentales
19	OR-133	Nerium oleander-CALIDAD \"GARDEN\"	Ornamentales
20	OR-134	Nerium Oleander Arbusto GRANDE	Ornamentales
21	OR-135	Nerium oleander COPA Calibre 6/8	Ornamentales
22	OR-137	ROSAL TREPADOR	Ornamentales
23	OR-138	Camelia Blanco, Chrysler Rojo, Soraya Naranja,	Ornamentales
24	OR-142	Solanum Jazminoide	Ornamentales
25	OR-143	Wisteria Sinensis azul, rosa, blanca	Ornamentales
26	OR-144	Wisteria Sinensis INJERTADAS DECÃ"	Ornamentales
27	OR-145	Bougamvillea Sanderiana Tutor	Ornamentales
28	OR-148	Bougamvillea Sanderiana Espaldera	Ornamentales
29	OR-149	Bougamvillea Sanderiana Espaldera	Ornamentales
30	OR-151	Bougamvillea Sanderiana, 3 tut. piramide	Ornamentales
31	OR-153	Expositor Árboles clima mediterráneo	Ornamentales

12. Muestra el nombre y apellidos de los empleados que trabajan en Barcelona.

```
select e.nombre, e.apellido1 || ' ' || e.apellido2
from empleados e, oficinas o
where e.codigooficina = o.codigooficina
and trim(lower(o.ciudad)) = 'barcelona';
```

```
select e.nombre, e.apellido1 || ' ' || e.apellido2
from empleados e, oficinas o
where e.codigooficina = o.codigooficina
and trim(lower(o.ciudad)) = 'barcelona';
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 4 en 0,009 segundos

	NOMBRE	E.APELLIDO1||"||E.APELLIDO2
1	José Manuel	Martinez De la Osa
2	Oscar	Palma Aceituno
3	Emmanuel	Magaña Perez
4	David	Palma Aceituno

13. Muestra el código y la cantidad de veces que se ha pedido un producto al menos una vez.

```
select p.codigoproducto, sum(dp.cantidad) as "cantidad pedida"
from productos p, detallepedidos dp
where p.codigoproducto = dp.codigoproducto
group by p.codigoproducto;
```

```
select p.codigoproducto, sum(dp.cantidad) as "cantidad pedida"
from productos p, detallepedidos dp
where p.codigoproducto = dp.codigoproducto
group by p.codigoproducto;
```

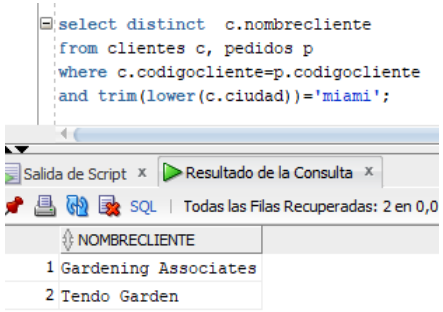
Salida de Script | Resultado de la Consulta

SQL | Se han recuperado 50 filas en 0,007 segundos

	CODIGOPRODUCTO	cantidad pedida
1	FR-4	88
2	OR-218	38
3	FR-69	21
4	OR-234	55
5	FR-13	13
6	FR-84	11
7	FR-23	8
8	FR-9	8
9	OR-186	40
10	OR-129	3
11	FR-12	32
12	FR-72	4
13	FR-7	12
14	OR-165	3
15	OR-225	84

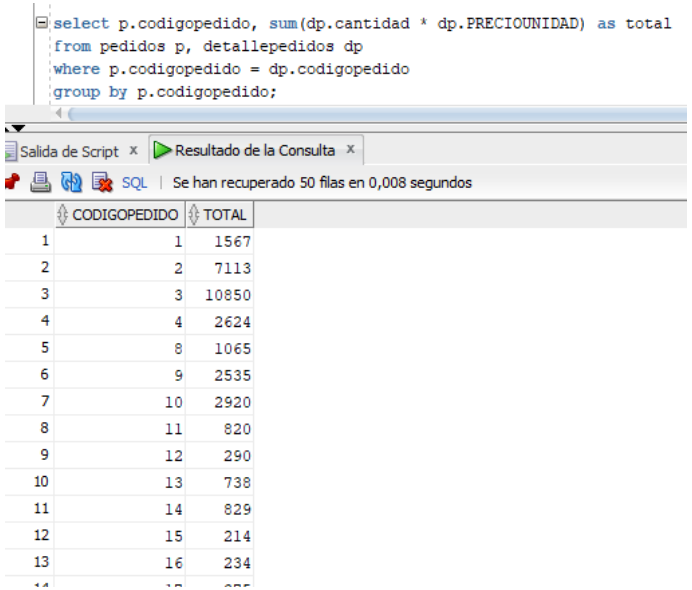
14. Muestra el nombre de los clientes de Miami que han realizado algún pedido.

```
select distinct c.nombrecliente
from clientes c, pedidos p
where c.codigocliente=p.codigocliente
and trim(lower(c.ciudad))='miami';
```



15. Mostrar el precio final de cada pedido.

```
select p.codigopedido, sum(dp.cantidad * dp.PRECIOUNIDAD) as total
from pedidos p, detallepedidos dp
where p.codigopedido = dp.codigopedido
group by p.codigopedido;
```



	CODIGOPEDIDO	TOTAL
1	1	1567
2	2	7113
3	3	10850
4	4	2624
5	8	1065
6	9	2535
7	10	2920
8	11	820
9	12	290
10	13	738
11	14	829
12	15	214
13	16	234

16. Mostrar lo que ha pagado cada cliente.

```
select nombrecliente, sum(p.cantidad) as total_pagado
from clientes c, pagos p
where c.codigocliente=p.codigocliente
group by c.nombrecliente;
```

```
select nombrecliente, sum(p.cantidad) as total_pagado
from clientes c, pagos p
where c.codigocliente=p.codigocliente
group by c.nombrecliente;
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 18 en 0,019 segundos

	NOMBRECLIENTE	TOTAL_PAGADO
1	DGPRODUCTIONS GARDEN	4000
2	Camunas Jardines S.L.	2246
3	Naturagua	929
4	Jardines y Mansiones CACTUS SL	18846
5	Flores Marivi	4399
6	Gardening Associates	10926
7	Tendo Garden	23794
8	Jardin de Flores	12081
9	Sotogrande	272
10	Agrojardin	8489

17. Mostrar el número de productos de cada gama.

```
select p.gama, count(*) as numero_productos
from productos p, gamasproductos gp
where p.gama = gp.gama
group by p.gama;
```

```
select p.gama, count(*) as numero_productos
from productos p, gamasproductos gp
where p.gama = gp.gama
group by p.gama;
```

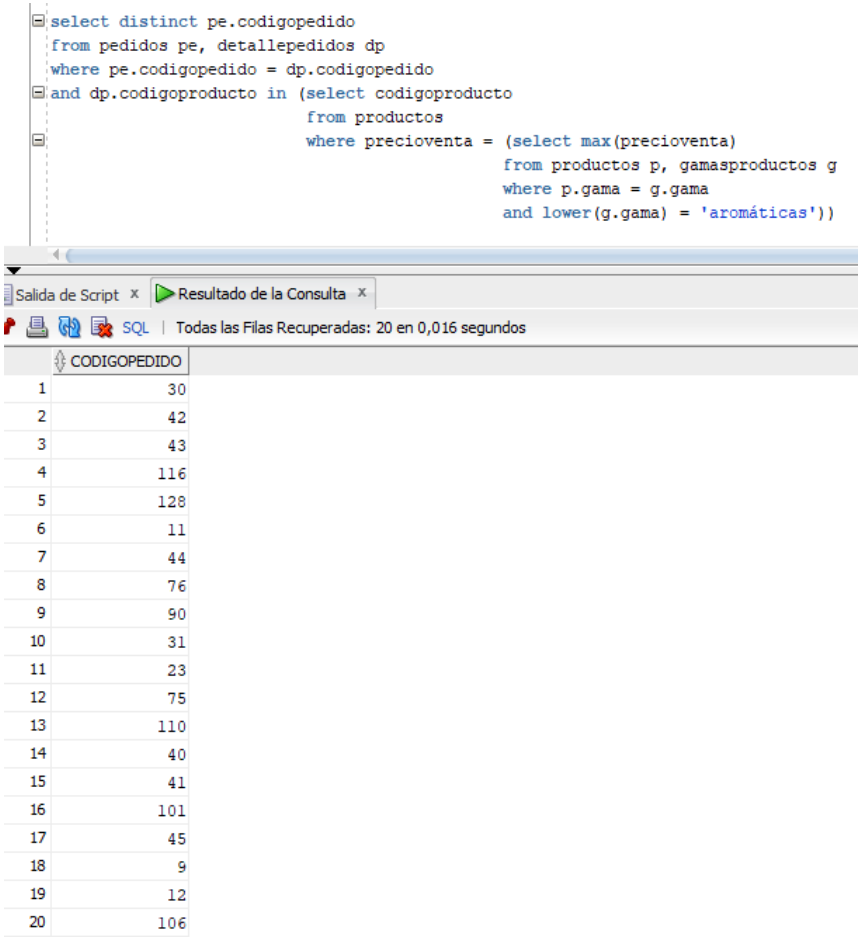
Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 4 en 0,005 segundo

	GAMA	NUMERO_PRODUCTOS
1	Aromáticas	10
2	Ornamentales	154
3	Frutales	108
4	Herramientas	4

18. Mostrar el codigo de los pedidos donde se haya vendido el producto de la gama ‘Aromáticas’ más caro.

```
select distinct pe.codigopedido
from pedidos pe, detallepedidos dp
where pe.codigopedido = dp.codigopedido
and dp.codigoproducto in (select codigoproducto
                          from productos
                          where precioventa = (select max(precioventa)
                                               from productos p, gamasproductos g
                                               where p.gama = g.gama
                                               and lower(g.gama) = 'aromáticas'))
```



19. Mostrar el codigo de los pedidos donde se hayan vendido más de 6 productos.

```
select pe.codigopedido
from pedidos pe, detallepedidos dp
where pe.codigopedido = dp.codigopedido
group by pe.codigopedido
having count(*)>=6;
```

```
select pe.codigopedido
from pedidos pe, detallepedidos dp
where pe.codigopedido = dp.codigopedido
group by pe.codigopedido
having count(*)>=6;
```

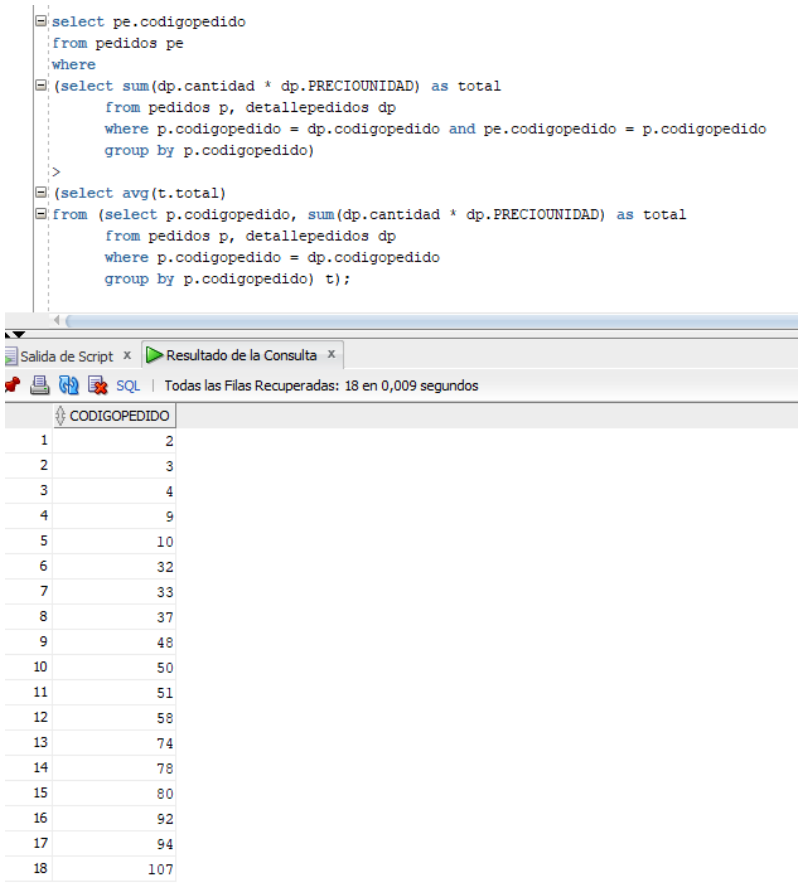
Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 8 en 0,004

	CODIGOPEDIDO
1	2
2	3
3	4
4	30
5	54
6	56
7	89
8	109

20. Mostrar el codigo de los pedidos donde el precio del pedido sea superior a la media de todos los pedidos.

```
select pe.codigopedido
from pedidos pe
where
(select sum(dp.cantidad * dp.PRECIOUNIDAD) as total
      from pedidos p, detallepedidos dp
      where p.codigopedido = dp.codigopedido and pe.codigopedido = p.codigopedido
      group by p.codigopedido)
>
(select avg(t.total)
from (select p.codigopedido, sum(dp.cantidad * dp.PRECIOUNIDAD) as total
      from pedidos p, detallepedidos dp
      where p.codigopedido = dp.codigopedido
      group by p.codigopedido) t);
```



	CODIGOPEDIDO
1	2
2	3
3	4
4	9
5	10
6	32
7	33
8	37
9	48
10	50
11	51
12	58
13	74
14	78
15	80
16	92
17	94
18	107

21. Realiza una vista que muestre los datos de un empleado (nombre, apellidos, ciudad de la oficina) y lo mismo para su jefe (en la misma fila).

```
create or replace view empleados_jefes as
select e_subor.nombre as nombre_subor,
        e_subor.apellido1 || ' ' || e_subor.apellido2 as apellidos_subor,
        o_subor.ciudad as ciudad_subor,
        e_jefe.nombre as nombre_jefe,
        e_jefe.apellido1 || ' ' || e_jefe.apellido2 as apellidos_jefe,
        o_subor.ciudad as ciudad_jefe
from empleados e_subor, empleados e_jefe, oficinas o_subor, oficinas o_jefe
where e_subor.codigojefe = e_jefe.codigoempleado
and o_subor.codigooficina = e_subor.codigooficina
and e_jefe.codigooficina = o_jefe.CODIGOOFICINA
```

```
select * from empleados_jefes;
```

```
create or replace view empleados_jefes as
select e_subor.nombre as nombre_subor,
        e_subor.apellido1 || ' ' || e_subor.apellido2 as apellidos_subor,
        o_subor.ciudad as ciudad_subor,
        e_jefe.nombre as nombre_jefe,
        e_jefe.apellido1 || ' ' || e_jefe.apellido2 as apellidos_jefe,
        o_subor.ciudad as ciudad_jefe
from empleados e_subor, empleados e_jefe, oficinas o_subor, oficinas o_jefe
where e_subor.codigojefe = e_jefe.codigoempleado
and o_subor.codigooficina = e_subor.codigooficina
and e_jefe.codigooficina = o_jefe.CODIGOOFICINA

select * from empleados_jefes;
```

Salida de Script | Resultado de la Consulta

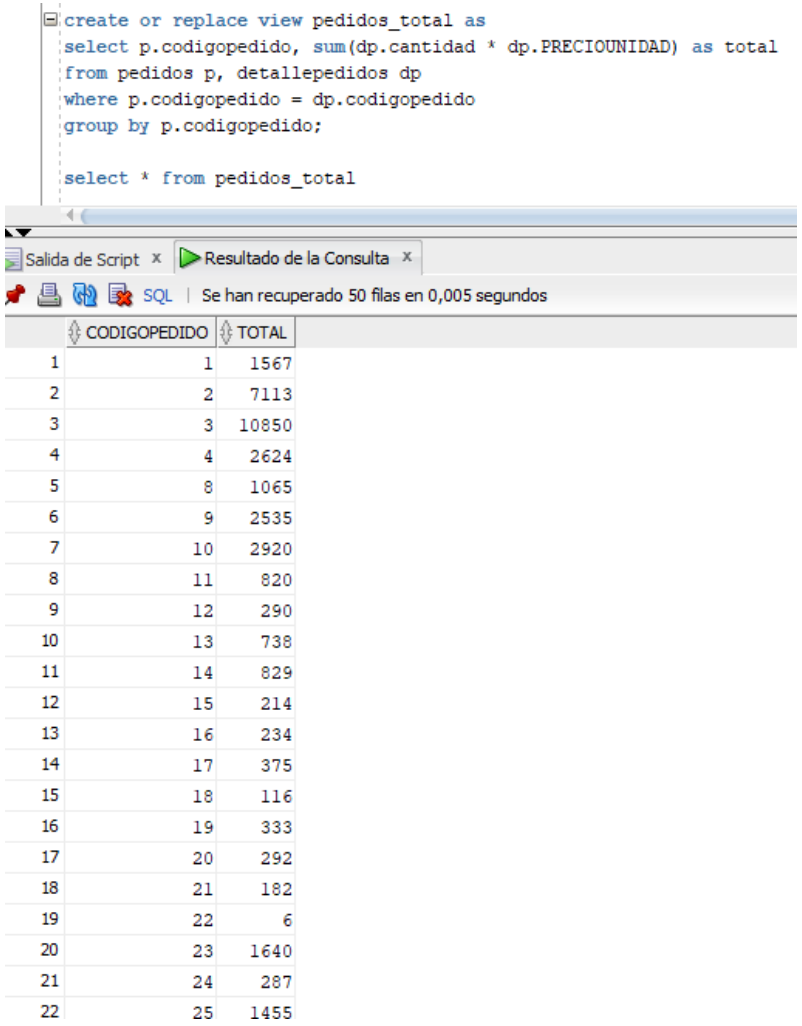
SQL | Todas las Filas Recuperadas: 30 en 0,004 segundos

	NOMBRE_SUBOR	APELLIDOS_SUBOR	CIUDAD_SUBOR	NOMBRE_JEFE	APELLIDOS_JEFE	CIUDAD_JEFE
1	Ruben	López Martinez	Talavera de la Reina	Marcos	Magaña Perez	Talavera de la Reina
2	Alberto	Soria Carrasco	Talavera de la Reina	Ruben	López Martinez	Talavera de la Reina
3	Maria	Solís Jerez	Talavera de la Reina	Ruben	López Martinez	Talavera de la Reina
4	Nei	Nishikori	Tokyo	Alberto	Soria Carrasco	Tokyo
5	Juan Carlos	Ortiz Serrano	Talavera de la Reina	Alberto	Soria Carrasco	Talavera de la Reina
6	Felipe	Rosas Marquez	Talavera de la Reina	Alberto	Soria Carrasco	Talavera de la Reina
7	Kevin	Fallmer	Sydney	Alberto	Soria Carrasco	Sydney
8	Michael	Bolton	San Francisco	Alberto	Soria Carrasco	San Francisco
9	Francois	Fignon	Paris	Alberto	Soria Carrasco	Paris
10	Carlos	Soria Jimenez	Madrid	Alberto	Soria Carrasco	Madrid
11	Amy	Johnson	Londres	Alberto	Soria Carrasco	Londres
12	Hilary	Washington	Boston	Alberto	Soria Carrasco	Boston
13	Emmanuel	Magaña Perez	Barcelona	Alberto	Soria Carrasco	Barcelona
14	Hilario	Rodriguez Huertas	Madrid	Carlos	Soria Jimenez	Madrid
15	Mariano	López Murcia	Madrid	Carlos	Soria Jimenez	Madrid
16	Lucio	Campoamor Martín	Madrid	Carlos	Soria Jimenez	Madrid
17	David	Palma Aceituno	Barcelona	Emmanuel	Magaña Perez	Barcelona
18	Oscar	Palma Aceituno	Barcelona	Emmanuel	Magaña Perez	Barcelona
19	José Manuel	Martinez De la Osa	Barcelona	Emmanuel	Magaña Perez	Barcelona
20	Laurent	Serra	Paris	Francois	Fignon	Paris
21	Lionel	Narvaez	Paris	Francois	Fignon	Paris
22	Walter Santiago	Sanchez Lopez	San Francisco	Michael	Bolton	San Francisco
23	Marcus	Paxton	Boston	Hilary	Washington	Boston
24	Lorena	Paxton	Boston	Hilary	Washington	Boston
25	Narumi	Riko	Tokyo	Nei	Nishikori	Tokyo

22. Realiza una vista que muestre el codigo de pedido y su total en euros.

```
create or replace view pedidos_total as
select p.codigopedido, sum(dp.cantidad * dp.PRECIOUNIDAD) as total
from pedidos p, detallepedidos dp
where p.codigopedido = dp.codigopedido
group by p.codigopedido;

select * from pedidos_total
```



	CODIGOPEDIDO	TOTAL
1	1	1567
2	2	7113
3	3	10850
4	4	2624
5	8	1065
6	9	2535
7	10	2920
8	11	820
9	12	290
10	13	738
11	14	829
12	15	214
13	16	234
14	17	375
15	18	116
16	19	333
17	20	292
18	21	182
19	22	6
20	23	1640
21	24	287
22	25	1455

23. Realiza una vista con la información del pedido (codigo, fechapedido, fechaesperada, fechaentrega, nombre cliente y total en euros) ordenado por total de forma descendente.

```
create or replace view info_pedido as
select p.codigopedido,
        p.fechapedido,
        p.fechaesperada,
        p.fechaentrega,
        c.nombrecliente,
        pt.total
from pedidos p, clientes c, PEDIDOS_TOTAL pt
where p.CODIGOCLIENTE = c.CODIGOCLIENTE
and pt.CODIGOPEDIDO = p.CODIGOPEDIDO
order by pt.total desc;

select * from info_pedido
```

```
create or replace view info_pedido as
select  p.codigopedido,
        p.fechapedido,
        p.fechaesperada,
        p.fechaentrega,
        c.nombrecliente,
        pt.total
from pedidos p, clientes c, PEDIDOS_TOTAL pt
where p.CODIGOCLIENTE = c.CODIGOCLIENTE
and pt.CODIGOPEDIDO = p.CODIGOPEDIDO
order by pt.total desc;

select * from info_pedido
```

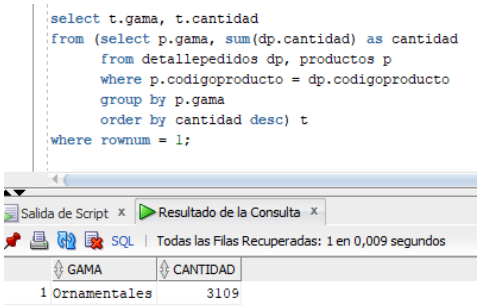
Salida de Script | Resultado de la Consulta

SQL | Se han recuperado 50 filas en 0,017 segundos

	CODIGOPEDIDO	FECHAPEDIDO	FECHAESPERADA	FECHAENTREGA	NOMBRECLIENTE	TOTAL
1	33	20/05/07	28/05/07	(null)	Gerudo Valley	73226
2	3	20/06/08	25/06/08	(null)	Tendo Garden	10850
3	51	01/10/08	14/10/08	14/10/08	Jardines y Mansiones CACTUS SL	7750
4	2	23/10/07	28/10/07	26/10/07	Tendo Garden	7113
5	48	17/03/08	30/03/08	29/03/08	Jardines y Mansiones CACTUS SL	6398
6	80	25/01/09	26/01/09	(null)	Agrojardin	5773
7	94	18/10/09	01/11/09	(null)	Jardinerias Matias SL	5759
8	58	24/01/09	31/01/09	30/01/09	Gardening Associates	4775
9	78	15/12/08	17/12/08	17/12/08	Jardin de Flores	4660
10	74	14/01/09	22/01/09	(null)	Jardin de Flores	3562
11	50	17/03/08	09/08/08	(null)	Jardines y Mansiones CACTUS SL	3506
12	107	06/04/09	10/04/09	10/04/09	Jardineria Sara	3216
13	32	07/01/07	19/01/07	27/01/07	Gerudo Valley	3089
14	10	15/01/09	20/01/09	(null)	Gardening Associates	2920
15	92	19/04/09	30/04/09	03/05/09	Jardinerias Matias SL	2906
16	4	20/01/09	26/01/09	(null)	Tendo Garden	2624
17	9	22/12/08	27/12/08	28/12/08	DGPRODUCTIONS GARDEN	2535
18	37	03/11/08	13/11/09	(null)	Gerudo Valley	2284
19	76	15/11/08	23/11/08	23/11/08	Jardin de Flores	2223
20	82	20/01/09	29/01/09	29/01/09	Agrojardin	2176
21	99	15/02/09	27/02/09	(null)	Flores Marivi	2070
22	28	10/02/09	17/02/09	20/02/09	Gardening Associates	2052

24. Devolver la gama de productos más vendida. Sin vistas

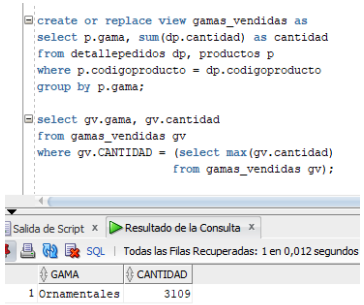
```
select t.gama, t.cantidad
from (select p.gama, sum(dp.cantidad) as cantidad
      from detallepedidos dp, productos p
      where p.codigoproducto = dp.codigoproducto
      group by p.gama order by cantidad desc) t where rownum = 1;
```

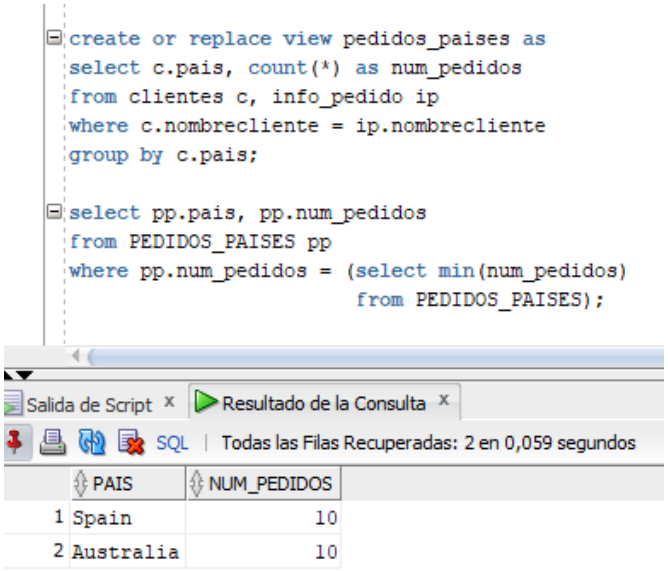


25. Devolver la gama de productos más vendida. Usa vistas

```
create or replace view gamas_vendidas as
select p.gama, sum(dp.cantidad) as cantidad
from detallepedidos dp, productos p
where p.codigoproducto = dp.codigoproducto
group by p.gama;

select gv.gama, gv.cantidad
from gamas_vendidas gv
where gv.CANTIDAD = (select max(gv.cantidad) from gamas_vendidas gv);
```



26. Muestra el pais(cliente) donde menos pedidos se hacen.

```
create or replace view pedidos_paises as
select c.pais, count(*) as num_pedidos
from clientes c, info_pedido ip
where c.nombrecliente = ip.nombrecliente
group by c.pais;

select pp.pais, pp.num_pedidos
from PEDIDOS_PAISES pp
where pp.num_pedidos = (select min(num_pedidos)
                        from PEDIDOS_PAISES);
```