

REALIZACIÓN DE CONSULTAS MULTITABLA CON COMPOSICIÓN EXTERNA

1. Base de datos Tienda Informática (2 tablas).

Muestra un listado de los fabricantes con los productos asociados, los tengan o no.

```
SELECT fabricante.*, producto.*
FROM fabricante LEFT JOIN producto ON fabricante.codigo = producto.codigo_fabricante;
```

Muestra un listado de los fabricantes que no tengan ningún producto asociado.

```
SELECT fabricante.*
FROM fabricante
LEFT JOIN producto ON fabricante.codigo = producto.codigo_fabricante
WHERE producto.codigo IS NULL;
```

2. Base de datos Gestión de Empleados (2 tablas).

Muestra los empleados con los datos del departamento en que trabajan, así como también información de los empleados que no trabajen en ningún departamento.

```
select * from empleado left outer join departamento
on empleado.codigo_departamento = departamento.codigo;
select * from departamento right outer join empleado
on empleado.codigo_departamento = departamento.codigo;
```

Muestra un listado de los empleados sin departamento asociado.

```
select * from empleado left outer join departamento
on empleado.codigo_departamento = departamento.codigo where departamento.codigo IS NULL;
```

Muestra los departamentos sin ningún empleado asociado.

```
select * from departamento left outer join empleado
on departamento.codigo = empleado.codigo_departamento where empleado.codigo IS NULL;
```

Muestra los empleados y los departamentos en los que trabajan, así como también los empleados que no trabajan en ningún departamento y los departamentos en los que no trabaja ningún empleado.

```
select * from empleado left outer join departamento on empleado.codigo_departamento = departamento.codigo
union
select * from empleado right outer join departamento on empleado.codigo_departamento = departamento.codigo;
```

Muestra los empleados que no estén asociados a ningún departamento y los departamentos sin empleados asociados.

```
select * from empleado left outer join departamento on empleado.codigo_departamento = departamento.codigo where departamento.codigo IS NULL
union
select * from empleado right outer join departamento on empleado.codigo_departamento = departamento.codigo where empleado.codigo IS NULL;
```

3. Base de datos Gestión de Ventas (3 tablas).

Muestra un listado de los TODOS clientes con los datos de sus pedidos realizados, también deben aparecer los clientes sin pedidos.

```
SELECT cliente.*, pedido.id AS id_pedido, pedido.total AS total_pedido, pedido.fecha AS fecha_pedido
FROM cliente LEFT JOIN pedido ON cliente.id = pedido.id_cliente;
```

Muestra un listado de TODOS los comerciales con los datos de los pedidos realizados, también deben aparecer los comerciales sin pedidos.

```
SELECT comercial.*, pedido.id AS id_pedido, pedido.total AS total_pedido, pedido.fecha AS fecha_pedido
FROM comercial LEFT JOIN pedido ON comercial.id = pedido.id_comercial;
```

Muestra un listado con solamente los clientes que no hayan realizado ningún pedido.

```
SELECT cliente.*
FROM cliente LEFT JOIN pedido ON cliente.id = pedido.id_cliente
WHERE pedido.id IS NULL;
```

Muestra un listado en el que aparezcan los comerciales que no hayan realizado ningún pedido.

```
SELECT comercial.*
FROM comercial LEFT JOIN pedido ON comercial.id = pedido.id_comercial
WHERE pedido.id IS NULL;
```

Muestra un listado con los clientes que no hayan solicitado un pedido y los comerciales que no hayan participado en ningún pedido.

```
SELECT c.id AS 'Cliente'
FROM cliente LEFT JOIN pedido ON cliente.id = pedido.id_cliente
WHERE p.id IS NULL

UNION

SELECT comercial.id AS 'Comercial'
FROM comercial LEFT JOIN pedido ON comercial.id = pedido.id_comercial
WHERE pedido.id IS NULL;
```

4. Base de datos NBA (4 tablas).

Muestra los nombres de los jugadores agrupados por conferencia y división.

```
select equipos.conferencia, equipos.division, jugadores.nombre from equipos left outer join jugadores
on equipos.nombre = jugadores.nombre_equipo;
```

Muestra los puntos por partido y los nombres de los jugadores ordenados descendientemente.

```
select puntos_por_partido, jugadores.nombre from estadisticas left outer join jugadores
on estadisticas.jugador = jugadores.codigo order by puntos_por_partido desc;
```

5. Base de datos Departamento Oficinas (5 tablas).

Une las tablas dpto_oficinas y oficina con LEFT OUTER JOIN, mostrando datos sólo de la ciudad de Pontevedra.

```
select * from dpto_oficinas left outer join oficina
on dpto_oficinas.oficina = oficina.oficina and ciudad = 'Pontevedra';
```

Une las tablas dpto_oficinas y oficina con RIGHT OUTER JOIN, mostrando datos sólo de la ciudad de Pontevedra.

```
select * from dpto_oficinas right outer join oficina
on dpto_oficinas.oficina = oficina.oficina and ciudad = 'Pontevedra';
```

6. Base de datos Jardinería (8 tablas).

Muestra un listado con los clientes que NO han realizado ningún pago.

```
SELECT cliente.*
FROM cliente LEFT JOIN pago ON cliente.codigo_cliente = pago.codigo_cliente
WHERE pago.codigo_cliente IS NULL;
```

Muestra un listado con los clientes que NO han realizado ningún pedido.

```
SELECT cliente.*
FROM cliente LEFT JOIN pedido ON cliente.codigo_cliente = pedido.codigo_cliente
WHERE pedido.codigo_cliente IS NULL;
```

Muestra un listado con los clientes que NI han realizado ningún pago NI ningún pedido.

```
SELECT cliente.*
FROM cliente LEFT JOIN pedido ON cliente.codigo_cliente = pedido.codigo_cliente
LEFT JOIN pago ON cliente.codigo_cliente = pago.codigo_cliente
WHERE pedido.codigo_cliente IS NULL AND pago.codigo_cliente IS NULL;
```

Muestra un listado con los empleados sin oficina asociada.

```
SELECT empleado.*
FROM empleado
LEFT JOIN oficina ON empleado.codigo_oficina = oficina.codigo_oficina
WHERE oficina.codigo_oficina IS NULL;
```

Muestra un listado con los empleados sin ningún cliente asociado.

```
SELECT empleado.*
FROM empleado LEFT JOIN cliente ON empleado.codigo_empleado = cliente.codigo_empleado_rep_ventas
WHERE cliente.codigo_empleado_rep_ventas IS NULL;
```

Muestra un listado de los empleados sin cliente asociado, junto con los datos de la oficina en la que trabajan.

```
SELECT empleado.*, oficina.*
FROM empleado LEFT JOIN cliente ON empleado.codigo_empleado = cliente.codigo_empleado_rep_ventas
LEFT JOIN oficina ON empleado.codigo_oficina = oficina.codigo_oficina
WHERE cliente.codigo_empleado_rep_ventas IS NULL;
```

Muestra un listado con los productos sin ningún pedido asociado.

```
SELECT producto.*
FROM producto LEFT JOIN detalle_pedido ON producto.codigo_producto = detalle_pedido.codigo_producto
WHERE detalle_pedido.codigo_producto IS NULL;
```

Muestra un listado con las oficinas en las que NO trabajen ningún empleado que haya sido representante de ventas de algún cliente que haya realizado la compra de algún producto de la gama Frutales.

```
SELECT oficina.*
FROM oficina LEFT JOIN empleado ON oficina.codigo_oficina = empleado.codigo_oficina
LEFT JOIN cliente ON empleado.codigo_empleado = cliente.codigo_empleado_rep_ventas
LEFT JOIN pedido ON cliente.codigo_cliente = pedido.codigo_cliente
LEFT JOIN detalle_pedido ON pedido.codigo_pedido = detalle_pedido.codigo_pedido
LEFT JOIN producto ON detalle_pedido.codigo_producto = producto.codigo_producto
LEFT JOIN gama_producto ON producto.gama = gama_producto.gama
WHERE gama_producto.gama = 'Frutales' AND empleado.codigo_empleado IS NULL;
```

Muestra un listado con empleados sin clientes asociados, junto con el nombre de su jefe.

```
SELECT empleado.*, jefe.nombre AS nombre_jefe
```

REALIZACIÓN DE CONSULTAS MULTITABLA CON COMPOSICIÓN EXTERNA

```
FROM empleado LEFT JOIN cliente ON empleado.codigo_empleado = cliente.codigo_empleado_rep_ventas  
LEFT JOIN empleado jefe ON empleado.codigo_jefe = jefe.codigo_empleado  
WHERE cliente.codigo_empleado_rep_ventas IS NULL;
```
