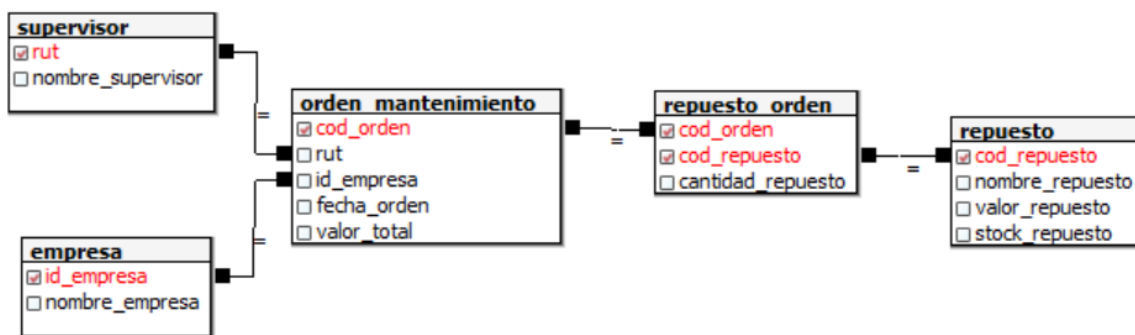


**Guía IN1078C****Prof. Mariella Gutiérrez****Objetivo: conocer la sintaxis y aplicar subconsultas en la sentencia SELECT del lenguaje SQL.**

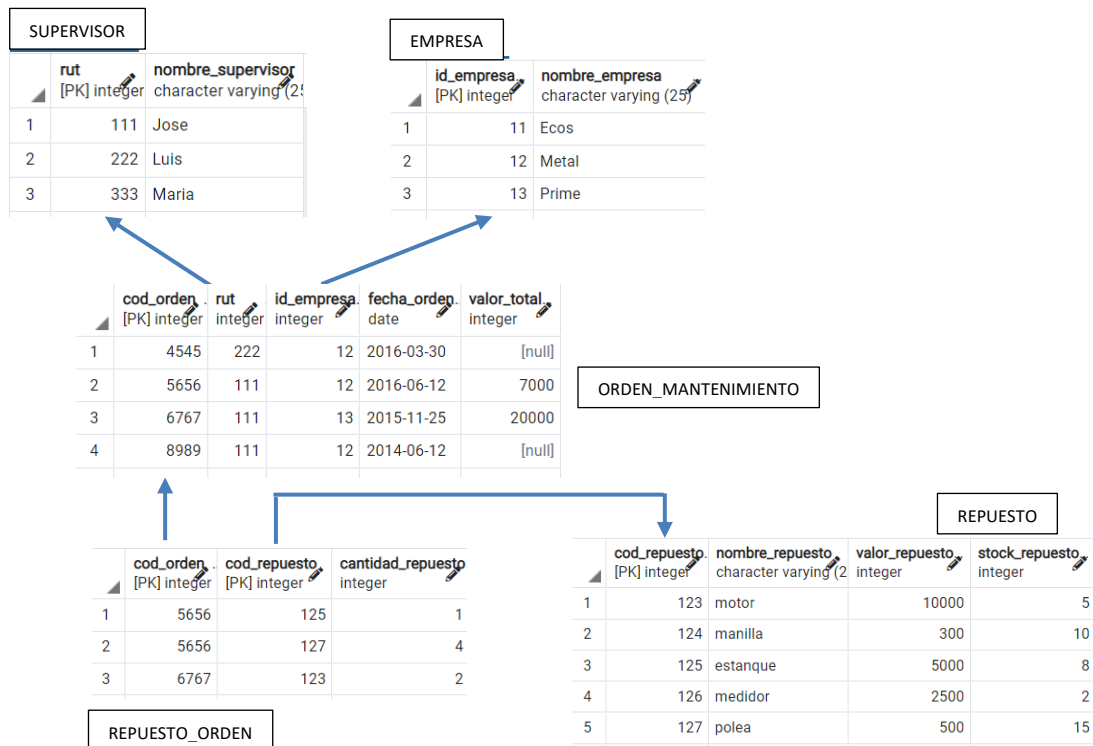
Se usará la misma base de datos utilizada en la primera guía. La base de datos de mantenimiento corresponde al siguiente modelo.



Las relaciones entre las tablas son las siguientes:

	Source Column	Join Type	Destination Column
1	empresa.id_empresa	+ =	orden_mantenimiento.id_empresa +
2	supervisor.rut	+ =	orden_mantenimiento.rut +
3	orden_mantenimiento.cod_orden	+ =	repuesto_orden.cod_orden +
4	repuesto.cod_repuesto	+ =	repuesto_orden.cod_repuesto +

El modelo en extensión es el siguiente:



1. Sintaxis de subconsultas en una sentencia SELECT

Una subconsulta es una sentencia SELECT que aparece dentro de otra sentencia SELECT que llamaremos consulta principal.

SELECT [DISTINCT | ALL] { * | <lista de columnas> | **(subconsulta)** }

FROM {tabla | **(subconsulta)** as alias }

WHERE condition operador **(subconsulta)**;

- ▶ Se puede encontrar en la lista de selección, en la cláusula WHERE o en la cláusula HAVING de la consulta principal.
- ▶ También puede aparecer en el FROM de una sentencia SELECT. En ese caso la subconsulta se considera como una tabla temporal y DEBE tener un alias para darle un nombre a esa tabla.

2. Subconsultas en la lista de selección

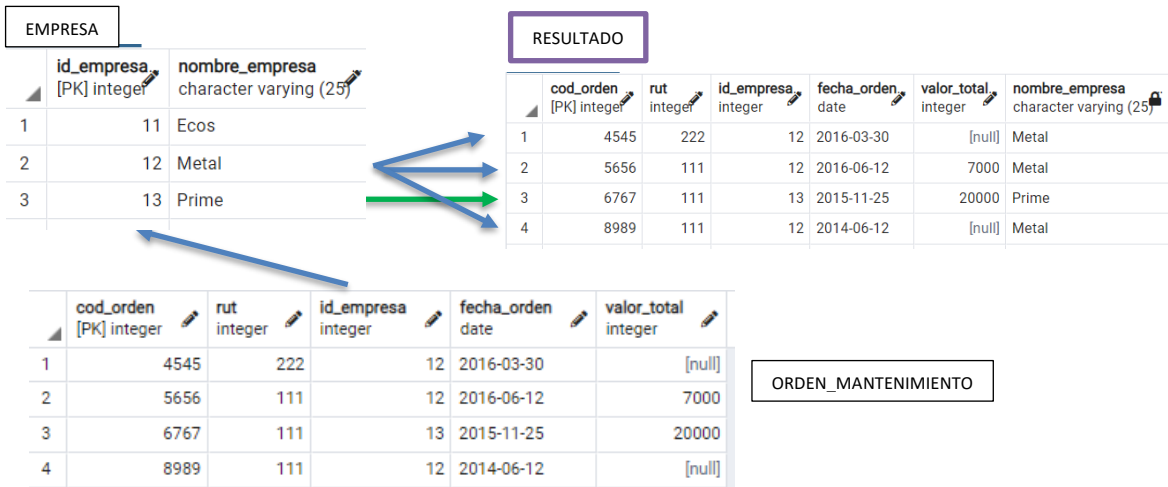
- ▶ Cuando la subconsulta va en la lista de selección de un SELECT, no puede devolver varias columnas, ni varias filas.
- ▶ La subconsulta para retornar una columna en la lista de selección se ejecuta para cada fila de la sentencia select principal, por lo tanto, debe estar enlazada a la consulta principal a través de un campo en común).

a) Seleccionar las órdenes de mantenimiento con el nombre de empresa que le corresponde.



```
SELECT orden_mantenimiento.* ,
       (SELECT nombre_empresa
        FROM empresa
        WHERE id_empresa =
        orden_mantenimiento.id_empresa)
FROM empresa;
```

Subconsulta: devuelve para cada id_empresa en orden de mantenimiento el nombre_empresa recuperado de la tabla empresa



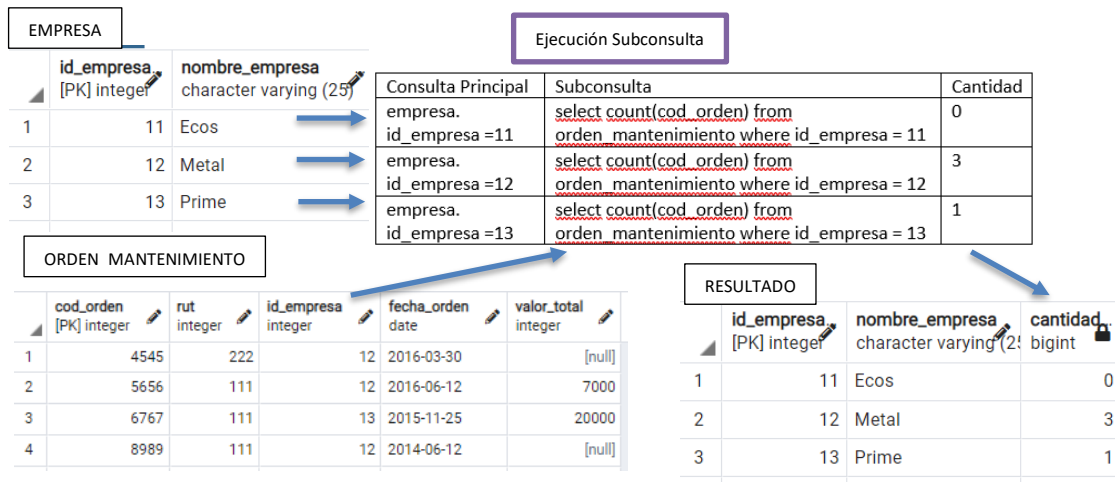
- b) Seleccionar los datos de todas las empresas con el número de órdenes de mantenimiento que le corresponden, ordenados por identificador de la empresa de forma ascendente.

Sin Subconsulta:

```
SELECT empresa.*, count(cod_orden) AS cantidad
FROM empresa left join orden_mantenimiento using(id_empresa)
GROUP BY id_empresa
Order by id_empresa;
```

Con subconsulta:

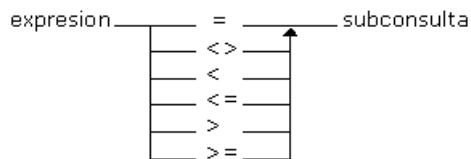
```
SELECT empresa.*, (select count(cod_orden) from orden_mantenimiento where id_empresa =
empresa.id_empresa) AS Cantidad
FROM empresa
Order by id_empresa;
```



3. Subconsultas en la cláusula Where

Tiene diferentes comportamientos dependiendo del operador que se utilice.

a) Operadores de comparación



En este caso la subconsulta debe retornar un único valor.

- Seleccionar las órdenes de mantenimiento cuyo valor total es mayor al promedio del valor total de todas las órdenes.

```
SELECT *
FROM orden_mantenimiento
WHERE valor_total > (select avg(valor_total) from orden_mantenimiento);
```



ORDEN MANTENIMIENTO

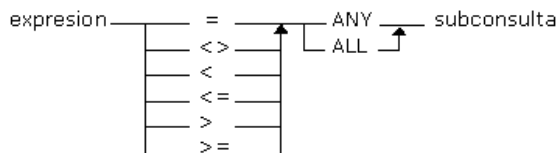
	cod_orden [PK] integer	rut integer	id_empresa integer	fecha_orden date	valor_total integer
1	4545	222	12	2016-03-30	[null]
2	5656	111	12	2016-06-12	7000
3	6767	111	13	2015-11-25	20000
4	8989	111	12	2014-06-12	[null]

Ejecución Subconsulta

Consulta Principal		Operador	Subconsulta	Resultado subconsulta
cod_orden=4545	valor_total=NULL	>	select avg(valor_total) from orden_mantenimiento	13.500
cod_orden=5656	valor_total=7000			
cod_orden=6767	valor_total=20000			
cod_orden=8989	valor_total=NULL			

RESULTADO

	cod_orden [PK] integer	rut integer	id_empresa integer	fecha_orden date	valor_total integer
1	6767	111	13	2015-11-25	20000

b) Operadores de comparación cuantificada

Ejemplo: Seleccionar todas las empresas que tienen alguna orden de mantenimiento.

Primero: Se realiza una selección para conocer el conjunto de empresas que tienen órdenes de mantenimiento.

select id_empresa from orden_mantenimiento

	id_empresa integer
1	13
2	12

Segundo: Se seleccionan los datos de las empresas cuyo identificador es igual a algunos de los valores recuperados en el select anterior.

SELECT empresa.*

FROM empresa

WHERE id_empresa = ANY (select id_empresa from orden_mantenimiento)

	id_empresa [PK] integer	nombre_empresa character varying(255)
1	12	Metal
2	13	Prime

c) Operadores de pertenencia a un conjunto (IN – NOT IN)

expresion IN subconsulta

Ejemplo: Seleccionar todos los supervisores que han supervisado órdenes de mantenimiento.



Primero: Se realiza una selección para conocer qué supervisores han supervisado órdenes de mantenimiento

```
SELECT distinct.rut FROM orden_mantenimiento;
```

	rut integer
1	111
2	222

Segundo: Se usa el select anterior para obtener los datos de los supervisores que están dentro del listado de rut que se obtienen de la subconsulta anterior.

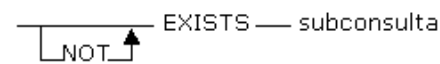
```
SELECT supervisor.*
```

```
FROM supervisor
```

```
WHERE rut IN (select distinct rut from orden_mantenimiento);
```

	rut [PK] integ	nombre_supervisor character varying (25)
1	111	Jose
2	222	Luis

d) Comparación de Existencia



EXISTS devuelve un valor booleano (Verdadero o Falso).

Ejemplo: Seleccionar todos los supervisores que han supervisado órdenes de mantenimiento.

Busca cada rut de supervisor en la tabla de órdenes de mantenimiento y si existe, selecciona todos los datos del supervisor.

```
SELECT supervisor.*
```

```
FROM supervisor
```

```
WHERE EXISTS (select distinct om.rut
               from orden_mantenimiento om
               where om.rut =supervisor.rut)
```

	rut [PK] integ	nombre_supervisor character varying (25)
1	111	Jose
2	222	Luis

4. Subconsultas en la cláusula From

- Seleccionar los datos de la(s) órdenes de mantenimiento más recientes de cada supervisor.

Primero: Se crea una consulta para conocer cuál es la fecha más reciente de orden de mantenimiento de cada supervisor



```
select rut, max(fecha_orden) fecha_orden from orden_mantenimiento group by rut
```

	rut integer	fecha_orden date
1	111	2016-06-12
2	222	2016-03-30

Segundo: Se usa el select anterior para obtener los datos de las órdenes de mantenimiento que corresponden a los rut y fechas que se obtienen de la tabla auxiliar.

```
select om.*
from orden_mantenimiento om, (select rut, max(fecha_orden) fecha_orden from
orden_mantenimiento group by rut) as tabla
where om.rut = tabla.rut
and om.fecha_orden = tabla.fecha_orden;
```

	cod_orden [PK] integer	rut integer	id_empresa integer	fecha_orden date	valor_total integer
1	4545	222	12	2016-03-30	[null]
2	5656	111	12	2016-06-12	7000

5. Ejercicios

- Listar todos los repuestos (cod_repuesto, nombre_repuesto, valor_repuesto, stock_repuesto) que tienen un stock menor al stock promedio de todos los repuestos.
- Listar todos los repuestos (cod_repuesto, nombre_repuesto, valor_repuesto, stock_repuesto) que tienen un valor mayor al valor promedio de todos los repuestos.
- Listar todas las órdenes de mantenimiento (cod_orden, fecha_orden, id_empresa, rut, valor_total) que tienen un valor total menor al valor promedio de todas las órdenes.
- Seleccione los datos de la o las empresas (id_empresa, nombre_empresa, cantidad de órdenes) que han solicitado la mayor cantidad de órdenes de mantenimiento.
- Seleccione los datos de la o las órdenes de mantenimiento (cod_orden, fecha_orden, cantidad de repuestos) que han utilizado una mayor cantidad de repuestos.
- Seleccione los datos del o los supervisores (rut, nombre_supervisor, cantidad de órdenes) que han supervisado la menor cantidad de órdenes de mantenimiento.