

## USO DEL SUM(CASE.....)

### Resolución del Ejercicio 7, ejercitario de Funciones de Grupo y Subconsultas:

Para tener una idea de movimientos, se desea conocer el volumen (cantidad) de ventas por mes que se hicieron por cada artículo durante el año 2012. Debe listar también los artículos que no tuvieron movimiento.

#### Análisis y Resolución

- La cantidad de artículos vendidos se encuentra en el campo cantidad, de la tabla de detalles. Tomemos como ejemplo el artículo 634817, "Almohada Plumi 50x70 Altenburg".
- Para entender mejor como funciona esta consulta, realicemos la inserción de dos registros más con ventas de ese artículo:

```
insert into b_detalle_ventas values (1,634817,3,6,63000);
insert into b_detalle_ventas values (2,634817,4,11,62727);
commit;
```

- Veamos entonces las ventas de este artículo:

```
select
  extract (month from ve.fecha) mes_venta,
  dv.*
from b_detalle_ventas dv
join b_ventas ve
  on ve.id=dv.id_venta
  and extract (year from ve.fecha)='2011'
where id_articulo=634817;
```

MES_VENTA	ID_VENTA	ID_ARTICULO	CANTIDAD	NUMERO_ITEM	PRECIO
1	1	634817	3	6	6300
2	2	634817	4	11	62727
2	2	634817	10	4	62727

En el mes 1 (enero), se vendieron 3 unidades, en febrero, 14.

- Para esta consulta, vamos a utilizar un SUM, realizando un análisis del mes de las ventas y dependiendo del mes, devolvemos la cantidad o bien, 0.

#### RECORDATORIO:

Como el ejercicio pide traer todos los artículos aunque no tengan ventas, debemos usar una reunión externa (OUTER JOIN). Si queremos que la consulta nos devuelva todos los registros a la IZQUIERDA del join, utilizamos el LEFT JOIN:

... FROM B\_ARTICULOS LEFT JOIN B\_DETALLE\_VENTAS ON...

También podemos obtener el mismo resultado pero usando el RIGHT JOIN. Esta reunión nos traerá todos los registros de la tabla que está a la DERECHA del join, aunque no tengan correspondencia en la tabla de la izquierda:

... FROM B\_DETALLE\_VENTAS RIGHT JOIN B\_ARTICULOS ON...

- Analicemos ahora cada línea del detalle. Si el mes de la venta es Enero, entonces devolvemos la cantidad vendida, y si no, devolvemos 0 y así con todos los meses.

Podemos utilizar tanto CASE como DECODE:

```
--con CASE
Select ar.id,
       ar.nombre Nombre_articulo,
       case when(extract (month from ve.fecha))=1 then dv.cantidad else 0 end enero,
       case when(extract (month from ve.fecha))=2 then dv.cantidad else 0 end febrero
from b_articulos ar
left join b_detalle_ventas dv on dv.id_articulo = ar.id
join b_ventas ve on ve.id=dv.id_venta and extract (year from ve.fecha)='2011'
where id_articulo=634817
;
```

```
--con DECODE
Select ar.id,
       ar.nombre Nombre_articulo,
       decode(extract (month from ve.fecha),1,dv.cantidad,0) enero,
       decode(extract (month from ve.fecha),2,dv.cantidad,0) febrero
from b_articulos ar
left join b_detalle_ventas dv on dv.id_articulo = ar.id
join b_ventas ve on ve.id=dv.id_venta and extract (year from ve.fecha)='2011'
where id_articulo=634817
;
```

Con los datos que tenemos, vamos a ver que se devuelve una línea por cada venta realizada, con los datos en la columna del mes correspondiente:

ID	NOMBRE_ARTICULO	ENERO	FEBRERO
634817	Almohada Plumi 50x70 Altenburg	3	0
634817	Almohada Plumi 50x70 Altenburg	0	4
634817	Almohada Plumi 50x70 Altenburg	0	10

En Febrero, se vendieron 10 unidades del artículo en una línea del detalle y en otra, 4. Para agrupar todas las ventas de ese artículo por mes en una sola línea, agregaremos un SUM:

```
Select ar.id,
       ar.nombre Nombre_articulo,
       sum(case when(extract (month from ve.fecha))=1 then dv.cantidad else 0 end) enero,
       sum(case when(extract (month from ve.fecha))=2 then dv.cantidad else 0 end) febrero
from b_articulos ar
left join b_detalle_ventas dv on dv.id_articulo = ar.id
inner join b_ventas ve on ve.id=dv.id_venta
                        and extract (year from ve.fecha)='2011'
where id_articulo=634817
group by ar.id, ar.nombre ;
```

El resultado es el que estamos buscando:

ID	NOMBRE_ARTICULO	ENERO	FEBRERO
634817	Almohada Plumi 50x70 Altenburg	3	14