

## Consultas con el predicado IN, GROUP BY, producto cartesiano y la cláusulas RIGTH, LEFT.

/\*1. Hallar por departamentos la edad en años cumplidos del empleado más mayor, así como la edad media en el mismo (solo para los empleados con comisión). Ordenar el resultado por edades.\*/

-- Vemos la tabla

```
SELECT NUMDE, NUMEM, TIMESTAMPDIFF(YEAR,FECNA,SYSDATE())
FROM temple
WHERE COMIS IS NOT NULL
ORDER BY 1;
```

```
SELECT NUMDE, MAX(TIMESTAMPDIFF(YEAR,FECNA,SYSDATE())) AS 'EDAD EMPLEADO
MAYOR',ROUND(AVG(TIMESTAMPDIFF(YEAR,FECNA,SYSDATE())),2) AS 'MEDIA DE EDAD'
FROM temple
WHERE COMIS IS NOT NULL
GROUP BY NUMDE
ORDER BY 2;
```

/\*2. Agrupando por departamento y número de hijos, hallar cuantos empleados hay en cada grupo.\*/

```
SELECT NUMDE,NUMHI,COUNT(*)
FROM temple
GROUP BY NUMDE, NUMHI
ORDER BY 1,2;
```

/\*3. Para cada extensión telefónica, hallar cuantos empleados la usan y salario medio de estos.\*/

```
SELECT EXTEL, COUNT(*) , ROUND(AVG(SALAR),2)
FROM temple
GROUP BY EXTEL
ORDER BY 1;
```

/\*4. Para los departamentos cuyo salario medio supera al de la empresa, hallar cuántas extensiones telefónicas tienen. Se debe mostrar el número de departamento (numde) y el número de extensiones telefónicas distintas que tiene cada uno de ellos\*/

```
SELECT NUMDE, COUNT(DISTINCT EXTEL)
FROM temple
GROUP BY numde
having AVG(SALAR)> (SELECT AVG(SALAR)
FROM temple);
```

/\*5. Hallar el máximo valor de la suma de los salarios de los departamentos. Queremos obtener número de departamento (numde) y la suma de sus salarios, pero del departamento cuya suma de salarios es la mayor de todas.\*/

/\*Primero vemos en cada departamento cuanto suman sus salario, y ya sabemos que en la solución debe salirnos el departamento 112 con suma de salario 6700, puesto que es la suma con valor más alto.\*/

---

## Relación de Ejercicios 4

---

```
SELECT NUMDE,SUM(SALAR)
FROM temple
GROUP BY NUMDE
ORDER BY NUMDE;
```

/\*max(sum(salar)) no se puede poner porque  
"NO es posible usar una función de agregado con una expresión que contiene  
un agregado o una subconsulta".\*/

```
SELECT NUMDE,SUM(SALAR)
FROM temple
GROUP BY NUMDE
HAVING SUM(SALAR) >= ALL ( SELECT SUM(SALAR)
                           FROM temple
                           GROUP BY NUMDE)
ORDER BY NUMDE;
```

-- Otra solución:  
-- Veamos primero el resultado de esta primera consulta:

```
SELECT SUM(SALAR)
FROM temple
GROUP BY NUMDE
ORDER BY 1 DESC
LIMIT 1;
```

-- Quedaría así:

```
SELECT NUMDE,SUM(SALAR)
FROM temple
GROUP BY NUMDE
HAVING SUM(SALAR) = ( SELECT SUM(SALAR)
                     FROM temple
                     GROUP BY NUMDE
                     ORDER BY 1 DESC
                     LIMIT 1)
ORDER BY NUMDE;
```

/\* 6. Para cada departamento con presupuesto inferior a 10000 euros  
hallar el nombre del centro donde está ubicado y el máximo salario  
de sus empleados, si éste excede de 1500 euros. Clasificar alfabéticamente  
por nombre de departamento. Hacer el ejercicio de dos maneras:  
con producto cartesiano y con join.\*/

```
SELECT NOMDE,NOMCE,MAX(SALAR)
FROM tcentr c, tdepto d, temple e
WHERE (c.NUMCE=d.NUMCE) AND (d.NUMDE=e.NUMDE) AND PRESU<10000
GROUP BY d.NUMDE
HAVING MAX(SALAR)>1500
ORDER BY 1;
```

Nota: En **SQL Server**, hubieramos tenido que agrupar también por NOMCE para poder  
ponerlo como expresión del **SELECT**

/\*Pretendemos que si algún departamento no tiene asignado centro  
y el máximo salario de sus empleados excede los 1500 euros,

---

## Relación de Ejercicios 4

---

obtenemos: el nombre del departamento, NULL y el máximo salario de sus empleados. Si el departamento no tiene empleados no tendría que salir, pues nunca ocurrirá que el máximo salario excederá a los 1500 euros.\*/

-- Para comprobar la diferencia entre producto cartesiano, join y right join, pon  
-- el numce del departamento 110 a nulo.

```
SELECT NOMDE,IFNULL(NOMCE,'Sin Departamento'),MAX(SALAR)
FROM (tcentr c RIGHT JOIN tdepto d ON (c.NUMCE=d.NUMCE)) JOIN temple e ON
(d.NUMDE=e.NUMDE)
WHERE PRESU<10000
GROUP BY d.NUMDE
HAVING MAX(SALAR)>1500
ORDER BY 1;
```

/\*7. Hallar por orden alfabético los nombres de los departamentos  
que dependen de los que tienen un presupuesto inferior a 10000 euros.  
Realizar la consulta de cuatro formas distintas: con predicado IN,  
con predicado ANY, con producto cartesiano y con join.\*/

```
SELECT NOMDE
FROM tdepto
WHERE DEPDE IN (SELECT NUMDE
                FROM tdepto
                WHERE PRESU<10000)
ORDER BY 1;
```

```
SELECT NOMDE
FROM tdepto
WHERE DEPDE =ANY (SELECT NUMDE
                  FROM tdepto
                  WHERE PRESU<10000)
ORDER BY 1;
```

```
SELECT T1.NOMDE
FROM tdepto T1, tdepto T2
WHERE T1.DEPDE=T2.NUMDE AND T2.PRESU<10000
ORDER BY 1;
```

```
SELECT T1.NOMDE
FROM tdepto T1 JOIN tdepto T2 ON (T1.DEPDE=T2.NUMDE)
WHERE T2.PRESU<10000
ORDER BY 1;
```

/\* 8. Obtener por orden alfabético los nombres de los departamentos  
cuyo presupuesto es inferior al 10 % de la suma de los salarios anuales  
(14 pagas) de sus empleados.\*/

```
SELECT *
FROM tdepto;
```

```
SELECT NOMDE
FROM tdepto D
where PRESU < (SELECT SUM(SALAR*14)* 0.10
               FROM temple
```

---

## Relación de Ejercicios 4

---

```
WHERE NUMDE=D.NUMDE)
ORDER BY 1;

-- O bien concatenando las tablas y haciendo grupos
SELECT NOMDE
FROM tdepto d JOIN temple e on (d.NUMDE=e.NUMDE)
GROUP BY NOMDE,PRESU
HAVING PRESU < SUM(SALAR*14)* 0.1
ORDER BY 1;

-- Para poder usar PRESU en el HAVING debemos agrupar por él

/*9.Obtener el nombre de cada departamento, junto con el nombre
del centro al que pertenece.*/

SELECT NOMDE,NOMCE
FROM TDEPTO D,TCENTR C
WHERE D.NUMCE=C.NUMCE;
-- O bien,
SELECT NOMDE,NOMCE
FROM TDEPTO D JOIN TCENTR C ON(D.NUMCE=C.NUMCE);
/* Esta primera forma de hacerlo podría omitir los nombres
de los departamento que no tuvieran asignado centro,
puesto que numce en departamento admite nulos. Para
comprobarlo pon numce del departamento de 110 a null*/

-- Así si sale:
SELECT NOMDE, NOMCE
FROM TCENTR C RIGHT JOIN TDEPTO D ON (C.NUMCE=D.NUMCE);

/*10. Obtener el nombre de cada empleado, el número
y el nombre del departamento al que pertenece y
el nombre del correspondiente centro.*/

SELECT e.numem, E.NOMEM, E.NUMDE,D.NOMDE,C.NOMCE
FROM TEMPLE E,TDEPTO D,TCENTR C
WHERE (E.NUMDE=D.NUMDE) AND (D.NUMCE=C.NUMCE);
-- O bien,
SELECT e.numem, E.NOMEM, E.NUMDE,D.NOMDE,C.NOMCE
FROM (TEMPLE E JOIN TDEPTO D ON (E.NUMDE=D.NUMDE)) JOIN TCENTR C
ON (D.NUMCE=C.NUMCE);

/*Esta primera forma de hacerlo prodría omitir los empleados que no tienen
asignado departamento (numde en EMPLEADO admite nulos)
y también los empleados que están en un departamento que no tiene asignado
centro (numce en DEPARTAMENTO admite nulos)
Probamos a poner:
El empleado 130 con numde a NULL.
El departamento 110 con numce a NULL. El empleado 180 y 190 están en
el departamento 110*/
SELECT * FROM temple;
SELECT E.NUMEM, E.NOMEM,D.NUMDE,D.NOMDE,C.NOMCE
FROM TCENTR C RIGHT JOIN (TDEPTO D RIGHT JOIN TEMPLE E ON (D.NUMDE=E.NUMDE)) ON
(C.NUMCE=D.NUMCE);
-- Así salen todos los empleados aunque no tengan departamento y
-- todos los departamentos aunque no tengan centro.
```

---

## Relación de Ejercicios 4

---

/\*11. Obtener para el departamento 110, su nombre, nombre del centro y la dirección, junto con el nombre del empleado que es el director. \*/

```
SELECT D.NOMDE,C.NOMCE,C.SEÑAS,E.NOMEM
FROM TEMPLE E, TDEPTO D, TCENTR C
WHERE D.NUMCE=C.NUMCE AND D.DIREC=E.NUMEM AND D.NUMDE=110;
```

-- O bien

```
SELECT D.NOMDE, C.NOMCE, C.SEÑAS, E.NOMEM
FROM (TCENTR C JOIN TDEPTO D ON (C.NUMCE=D.NUMCE)) JOIN TEMPLE E ON
(E.NUMEM=D.DIREC)
WHERE D.NUMDE=110;
/*Pero de esta manera no me sale ningún resultado puesto que el
departamento 110 NOMINAS no tiene asignado centro
(numce en DEPARTAMENTO está a NULL)*/
```

-- O bien,

```
SELECT D.NOMDE,C.NOMCE, C.SEÑAS, E.NOMEM
FROM TCENTR C RIGHT JOIN (TDEPTO D JOIN TEMPLE E ON (D.DIREC=E.NUMEM)) ON
(C.NUMCE=D.NUMCE)
WHERE D.NUMDE=110;
/* De estas dos últimas maneras si el departamento no tiene asignado director,
tampoco saldría el departamento, pon el DIREC del departamento 110 a null y
compruébalo*/
```

-- Así sí sale:

```
SELECT D.NOMDE, C.NOMCE, C.SEÑAS, E.NOMEM
FROM (TCENTR C RIGHT JOIN TDEPTO D ON (C.NUMCE=D.NUMCE)) LEFT JOIN TEMPLE E ON
(E.NUMEM=D.DIREC)
WHERE D.NUMDE=110 ;
```

-- El resultado final podríamos ponerlo así:

```
SELECT D.NOMDE AS 'Departamento',
       IFNULL(C.NOMCE,'Centro no asignado') AS 'Centro',
       IFNULL(C.SEÑAS,'No asignada') AS 'Dirección',
       IFNULL(E.NOMEM,'Sin director') AS 'Director'
FROM (TCENTR C RIGHT JOIN TDEPTO D ON (C.NUMCE=D.NUMCE)) LEFT JOIN TEMPLE E ON
(E.NUMEM=D.DIREC)
WHERE D.NUMDE=110 ;
```

/\*12. Utilizando las tablas TCENTR y TDEPTO plantear un enunciado, de modo que para obtener la correspondiente consulta sea necesario utilizar la cláusula RIGHT JOIN. Añadir en modo visual las filas necesarias a las tablas para así poder comprobar el resultado. La solución del ejercicio consta de enunciado y consulta\*/

/\*ENUNCIADO:

Obtener para cada departamento su nombre y el nombre del centro en el que se encuentra. Debe aparecer el nombre de todos los departamentos aunque no tenga centro asignado.\*/

-- CONSULTA:

```
SELECT nomde,nomce
FROM tcentr c right join tdepto d on c.numce=d.numce
ORDER BY 1;
```

---

