

Consultas con los predicados básicos, NULL, IN, ANY, ALL y LIKE, predicados compuestos, cláusulas GROUP BY y HAVING, producto cartesiano, JOIN, RIGHT JOIN, LEFT JOIN y FULL JOIN, funciones escalares y de agregado.

Relación de Ejercicios 4

```
USE empresa;
```

```
/*1. Para cada extensión telefónica, hallar  
cuántos empleados la usan  
y salario medio de estos.*/
```

```
SELECT extel, COUNT(*), AVG(salar)  
FROM temple  
GROUP BY extel  
ORDER BY 1;
```

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

```
SELECT numde, numhi, COUNT(*)  
FROM temple  
GROUP BY numde, numhi  
ORDER BY 1,2;
```

/*3.Hallar por departamentos la edad en años cumplidos del empleado más mayor, así como la edad media del mismo (el empleado debe tener comisión). Ordenar el resultado por edades.*/

--Vemos la tabla. Nota que las dos últimas columnas son iguales.

```
SELECT numde,
       numem,
       YEAR(CURRENT_TIMESTAMP) - YEAR(fecna),
       DATEDIFF(YEAR,fecna,GETDATE())
FROM temple
WHERE comis IS NOT NULL
ORDER BY 1;
```

/*Para obtener la edad de forma precisa, debemos poner como primer parámetro de la función DATEDIFF el valor DAY. Así nos da la diferencia en días.*/

/*Observa la diferencia entre la tercera y la cuarta columna.*/

```
SELECT numde,
       numem,
       DATEDIFF(DAY,FECNA,GETDATE())/365,
       YEAR(CURRENT_TIMESTAMP) - YEAR(fecna)
FROM temple
WHERE comis IS NOT NULL
ORDER BY 1;
```

```
/*Con esta consulta vemos qué resultado nos deberá salir*/
```

```
SELECT numde,  
       numem,  
       DATEDIFF(DAY,FECNA,GETDATE())/365  
FROM temple  
WHERE comis IS NOT NULL  
ORDER BY 1;
```

```
/*Si todas las medias de edad dan un valor entero,  
vamos a modificar la edad de algún empleado,por  
ejemplo el 130 para que tenga 27 años, así la  
media de edad del departamento 112 es 7,66666666*/
```

```
UPDATE temple  
SET FECNA='01/01/1998'  
WHERE NUMEM=130;
```

```
/*Observa en la siguiente consulta que la edad  
media del departamento 112 sale 37. El problema es  
que la media de números enteros, obtiene un  
entero.*/
```

```
SELECT numde,  
       MAX(DATEDIFF(DAY,FECNA,GETDATE())/365),  
       AVG(DATEDIFF(DAY,FECNA,GETDATE())/365)  
FROM temple  
WHERE comis IS NOT NULL  
GROUP BY numde  
ORDER BY 1;
```

/*Si queremos que en la edad media se tenga en cuenta los decimales,debemos convertir los números enteros que representan los años, por ejemplo, a float*/

```
SELECT numde,  
       MAX(DATEDIFF(DAY,FECNA,GETDATE())/365),  
       AVG( CONVERT(FLOAT,  
                   DATEDIFF(DAY,FECNA,GETDATE() )/365) )  
FROM temple  
WHERE comis IS NOT NULL  
GROUP BY numde  
ORDER BY 1;
```

/* Con la función ROUND conseguimos truncar para que nos queden dos decimales.*/

--SOLUCIÓN:

```
SELECT numde,  
       MAX(DATEDIFF(DAY,FECNA,GETDATE())/365),  
       ROUND( AVG( CONVERT(FLOAT,  
                   DATEDIFF( DAY,FECNA,GETDATE() )/365) ),2,1)  
FROM temple  
WHERE comis IS NOT NULL  
GROUP BY numde  
ORDER BY 2;
```

/*El tercer parámetro de ROUND es opcional:

If 0, it rounds the result to the number of decimal (saldría 37,67).

If another value than 0, it truncates the result to the number of decimals (saldría 37,66).

Default value is 0*/

/*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), AVG(salar)
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 el 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 cuánto 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.*/

```
SELECT numde, SUM(salar)
FROM temple
GROUP BY numde
ORDER BY 2 DESC;
```

/*Ahora nos quedamos solo con los grupos que cumplan lo que nos piden.*/

--SOLUCIÓN:

```
SELECT numde, SUM(salar)
FROM temple
GROUP BY numde
HAVING SUM(salar) >= ALL (SELECT SUM(salar)
                        FROM temple
                        GROUP 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".*/

--Otra solución:

```
SELECT numde, SUM(salar)
FROM temple
GROUP BY numde
HAVING SUM(salar) = (SELECT TOP(1) SUM(salar)
                    FROM temple
                    GROUP BY numde
                    ORDER BY 1 DESC);
```

/* 6. Para cada departamento con presupuesto inferior a 10000 euros obtener el nombre, 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.*/

--SOLUCIÓN con producto cartesiano:

```
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 nomde,nomce
HAVING MAX(salar)>1500
ORDER BY 1;
```

--SOLUCIÓN con JOIN:

```
SELECT nomde, nomce, MAX (salar)
FROM (tcentr c JOIN tdepto d ON (c.numce=d.numce))
     JOIN temple e ON (d.numde=e.numde)
WHERE presu < 10000
GROUP BY nomde,nomce
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. Mostrar el nombre del departamento y el nombre del departamento del que dependen. Realizar la consulta de cuatro formas distintas: con predicado IN, con predicado ANY, con producto cartesiano y con JOIN.*/

--SOLUCIÓN con predicado IN:

```
SELECT nomde, depde
FROM tdepto
WHERE depde IN (SELECT numde
                FROM tdepto
                WHERE presu < 10000)
ORDER BY 1;
```

--SOLUCIÓN con predicado ANY:

```
SELECT nomde, depde
FROM tdepto
WHERE depde = ANY (SELECT numde
                  FROM tdepto
                  WHERE presu < 10000)
ORDER BY 1;
```

--SOLUCIÓN con producto cartesiano:

```
SELECT t1.nomde, t2.nomde
FROM tdepto t1, tdepto t2
WHERE t1.depde = t2.numde AND t2.presu < 10000
ORDER BY 1;
```

--SOLUCIÓN con JOIN:

```
SELECT t1.nomde, t2.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 de sus empleados (sin tener en cuenta la comisión y son 14 pagas).Hacer el ejercicio con predicado básico y con agrupamiento.*/

--SOLUCIÓN con predicado básico:

```
SELECT nomde  
FROM tdepto d  
WHERE presu < (SELECT SUM(SALAR)*14  
FROM temple  
WHERE numde=d.numde)*0.10  
ORDER BY 1;
```

--O bien:

```
SELECT nomde  
FROM tdepto  
WHERE presu < (SELECT SUM(salar*14)*0.10  
FROM temple  
WHERE numde=tdepto.numde)  
ORDER BY 1;
```

--SOLUCIÓN con agrupamiento:

/*Observa que también debes agrupar por presu para poder usarlo como expresión en la cláusula HAVING. Podemos hacerlo porque a igual nombre de departamento tenemos el mismo presupuesto.*/

```
SELECT nomde
FROM tdepto d JOIN temple e ON (d.numde=e.numde)
GROUP BY nomde,presu
HAVING presu < SUM(salar*14)*0.10
ORDER BY 1;
```

/*9. Ejecutar las siguientes sentencias:*/

--Añadir los siguientes centros:

```
INSERT INTO tcentr (numce,nomce,señas)
VALUES (30,'PRODUCCIÓN','C. DEL ARTE, 13, MADRID'),
       (40,'INNOVACIÓN','AVDA. ANDALUCÍA, 20, MÁLAGA');
```

--Añadir los siguientes departamentos:

```
INSERT INTO
tdepto(numde,numce,direc,tidir,presu,depde,nomde)
VALUES (122,NULL,NULL,'F',10000,120,'MARKETING Y
PUBLICIDAD'),
       (123,20, NULL,'F',10000,120,'COMPRAS Y
LOGÍSTICA');
```

--Añadir los siguientes empleados:

```
INSERT INTO
temple(numem,numde,extel,fecna,fecin,salar,comis,numhi,nomem)
VALUES (381,122,350,'12/03/2000','8/1/2025',1800,100,0,'ROMERO,
MÍRIAM'),
       (382,122,350,'13/04/1998','8/1/2025',1800,100,1,'SÁNCHEZ,
LUCÍA'),
       (383,NULL,350,'14/05/1997','8/1/2025',1800,100,1,'LÓPEZ,
LAURA');
```

--Asignar el empleado 381 como director del departamento 122.

```
UPDATE tdepto
SET direc =381
WHERE numde=122;
```

/*Una vez ejecutadas estas sentencias, consultar las tablas tcentr, tdepto y temple por separado para comprobar que tenemos:

- Los centros 30 y 40 que aún no tienen departamentos ubicados en los mismos.
- El departamento 123 que aún no tiene empleados.
- El empleado 383 que aún no se le ha asignado departamento.
- El departamento 122 que aún no se le ha asignado centro.
- El departamento 120 (ya existía) y el 123 que aún no tienen directores.
- El departamento 100 (ya existía) que no depende de ningún otro.

*/

```
SELECT * FROM tcentr;
SELECT * FROM tdepto;
SELECT * FROM temple;
```

```
/*10. Para los centros de Madrid, obtener el
nombre de cada centro junto con el nombre de los
departamentos que tienen. Si un centro aún o tiene
departamentos que pertenezcan al mismo, el nombre
del centro debe salir igualmente y en el nombre
del departamento debe aparecer "Sin departamento
por el momento".*/
```

```
--Vemos que la información que sale es correcta.
```

```
SELECT *
FROM tcentr c LEFT JOIN tdepto d ON
(c.numce=d.numce)
WHERE señas LIKE '%MADRID';
```

```
--SOLUCIÓN:
```

```
SELECT nomce,
       ISNULL(nomde, 'Sin departamento por el
              momento')
FROM tcentr c LEFT JOIN tdepto d ON
      (c.numce=d.numce)
WHERE señas LIKE '%MADRID';
```

/*11. Para los departamentos con director obtener el nombre del centro en el que se encuentra y la dirección. Si un departamento aún no tiene asignado centro, el nombre del departamento debe salir igualmente y en el nombre del centro debe aparecer "Sin ubicar" y en la dirección "Desconocida."*/

--Vemos que la información que sale es correcta.

```
SELECT *  
FROM tcentr c RIGHT JOIN tdepto d ON  
(c.numce=d.numce)  
WHERE direc IS NOT NULL;
```

--SOLUCIÓN:

```
SELECT nomde,  
       ISNULL(nomce, 'Sin ubicar'),  
       ISNULL(nomce, 'Desconocida')  
FROM tcentr c RIGHT JOIN tdepto d ON  
      (c.numce=d.numce)  
WHERE direc IS NOT NULL;
```

```
/*12. Obtener para todos los departamentos su
nombre, junto con el nombre y fecha de ingreso en
la empresa en formato español de sus empleados. Si
un departamento aún no tiene empleados, el nombre
del departamento debe salir igualmente, en nombre
de empleado debe aparecer "Sin empleados" y en
fecha "Sin fecha".*/
```

```
--Vemos que la información que sale es correcta.
```

```
SELECT *
FROM tdepto d LEFT JOIN temple e ON
(d.numde=e.numde);
```

```
--SOLUCIÓN:
```

```
SELECT nomde AS 'Departamento',
       ISNULL(nomem,'Sin empleados') AS
       'Empleado',
       ISNULL(FORMAT(fecin,'d'),'Sin fecha') AS
       'Fecha ingreso'
FROM tdepto d LEFT JOIN temple e ON
(d.numde=e.numde);
```

/*13. Para los empleados que han ingresado en la empresa en el año actual, obtener su nombre y salario, así como el nombre y el presupuesto del departamento al que pertenece. Si al empleado aún no se le ha asignado departamento, el nombre del empleado y su salario deben salir igualmente y en nombre de departamento y en presupuesto debes poner el mensaje "Sin asignar". Debes poner alias para todas las columnas.*/

--Vemos que la información que sale es correcta.

```
SELECT *  
FROM tdepto d RIGHT JOIN temple e ON  
      (d.numde=e.numde)  
WHERE YEAR(GETDATE())=YEAR(FECIN);
```

--SOLUCIÓN:

```
SELECT nomem AS 'Empleado',  
       salar AS 'Salario',  
       ISNULL(nomde,'Sin asignar') AS  
         'Departamento',  
       ISNULL(CONVERT(VARCHAR(12),presu),'Sin  
         asignar') AS 'Presupuesto'  
FROM tdepto d RIGHT JOIN temple e ON  
      (d.numde=e.numde)  
WHERE YEAR(GETDATE())=YEAR(FECIN);
```

```

/*14.Para los departamentos de "Nóminas",
"Organización", "Personal" y "Compras y
Logística", obtener el número de departamento
(identificador único), así como el nombre y
extensión telefónica de los directores. Si alguno
de estos departamentos no tiene aún asignado un
director, el número del departamento debe salir
igualmente y en nombre del director y extensión
telefónica debes poner el mensaje.
Debes poner alias para todas las columnas.*/
--Vemos que la información que sale es correcta.
SELECT *
FROM tdepto d LEFT JOIN temple e ON
      (d.direc=e.numem)
WHERE nomde IN ('NOMINAS', 'ORGANIZACION',
                'PERSONAL', 'COMPRAS Y LOGÍSTICA');

--SOLUCIÓN:
/*Aunque que extel es de tipo VARCHAR(4), debo
convertirlo a un VARCHAR de más caracteres para
que no trunque la cadena 'Sin datos'*/
SELECT d.numde AS 'Número de departamento',
      ISNULL(nomem,'Pendiente de asignar') AS
      'Director',
      ISNULL(CONVERT(VARCHAR(10),extel),'Sin
      datos') AS 'Teléfono'
FROM tdepto d LEFT JOIN temple e ON
      (d.direc=e.numem)
WHERE nomde IN ('NOMINAS', 'ORGANIZACION',
                'PERSONAL', 'COMPRAS Y LOGÍSTICA');

```


/*15. Para cada departamento obtener el nombre y el nombre del departamento del que depende, si existe algún departamento que no depende de ningún otro, el nombre del departamento debe salir igualmente y en la columna depde debe aparecer "No depende de ningún departamento."*/

--Vemos que la información que sale es correcta.

```
SELECT *  
FROM tdepto d1 LEFT JOIN tdepto d2 ON  
(d1.depde=d2.numde);
```

--SOLUCIÓN:

```
SELECT d1.nomde,  
       ISNULL(d2.nomde, 'No depende de ningún  
               departamento')  
FROM tdepto d1 LEFT JOIN tdepto d2 ON  
(d1.depde=d2.numde);
```

```
/*16. Obtener para los empleados con hijos y con
comisión,su nombre, el nombre del departamento
para el que trabajan y el nombre del centro en el
que se encuentra su departamento. Si el empleado
no tiene departamento, este debe salir igualmente,
y si el departamento en el que está el empleado no
tiene centro también debe salir.*/
--Vemos que nos sale el empleado 383 aunque no
tiene departamento
```

```
SELECT *
FROM tdepto d RIGHT JOIN temple e ON
    (e.numde=d.numde)
WHERE numhi>0 AND comis IS NOT NULL;
```

```
/*Ahora concatenamos con la tabla de centros y
vemos que nos sale el departamento 122 aunque no
tiene centro*/
```

```
SELECT *
FROM (tdepto d RIGHT JOIN temple e ON
    (e.numde=d.numde)) LEFT JOIN
    tcentr c ON (c.numce=d.numce)
WHERE numhi>0 AND comis IS NOT NULL;
```

--SOLUCIÓN:

```
SELECT nomem, nomde, nomce
FROM (tdepto d RIGHT JOIN temple e ON
    (e.numde=d.numde)) LEFT JOIN
    tcentr c ON (c.numce=d.numce)
WHERE numhi>0 AND comis IS NOT NULL;
```

/*17. Obtener los nombres de todos los centros junto con los nombres de los departamentos que tiene. Si un centro aún no tiene departamentos, este debe salir igualmente. Además, debes obtener el nombre de todos los departamentos aunque no tengan asignado centro. En los centros sin departamento debes poner en la columna nombre de departamento el mensaje "Sin departamento" y en los departamentos sin centros debes poner en la columna centro "Sin centro". Debes poner alias para todas las columnas. Ordena por el nombre del centro ascendentemente y a igual nombre del centro por nombre de departamento descendentemente.

*/

--Vemos que la información que sale es correcta.

```
SELECT *  
FROM tcentr c FULL JOIN tdepto d ON  
(c.numce=d.numce);
```

--SOLUCIÓN:

```
SELECT ISNULL(nomce, 'Sin centro') AS 'Centro',  
       ISNULL(nomde, 'Sin departamento') AS  
       'Departamento'  
FROM tcentr c FULL JOIN tdepto d ON  
       (c.numce=d.numce)  
ORDER BY 1,2 DESC;
```

/*18. Obtener por cada centro, cuántos empleados hay que trabajen en departamentos que están ubicados en los mismos. Debe salir el nombre del centro y la frase "X empleados", donde X es el número de empleados que trabajan en departamentos ubicados en cada centro. Debes poner alias para todas las columnas.*/

--Vemos que la información que sale es correcta.

```
SELECT *  
FROM (tcentr c LEFT JOIN tdepto d ON  
      (c.numce=d.numce)) LEFT JOIN  
      temple e ON (d.numde=e.numde);
```

--SOLUCIÓN:

/*No podemos poner COUNT(*) porque sino nos cuenta la fila de los centros 30 y 40*/

```
SELECT nomce AS 'Centro',  
       CONCAT(CONVERT(VARCHAR(6),COUNT(d.numde)),  
              ' empleados') AS 'Numero de empleados'  
FROM (tcentr c LEFT JOIN tdepto d ON  
      (c.numce=d.numce)) LEFT JOIN  
      temple e ON (d.numde=e.numde)  
GROUP BY nomce;
```