

Consultas con JOIN, predicados compuestos, subselects y predicados: IN, ANY, ALL, BETWEEN, LIKE y EXISTS.

Relación de Ejercicios 3

```
USE EMPRESA;
```

```
/*1. Obtener por orden alfabético los nombres de los  
empleados cuyos salarios superen la mitad del salario del  
empleado 180.*/
```

```
SELECT nomem  
FROM temple  
WHERE salar > (SELECT salar * 0.5  
              FROM temple  
              WHERE numem=180)  
ORDER BY 1;
```

```
/*2. Obtener por orden alfabético los nombres de los  
empleados cuyos salarios superan dos veces al mínimo salario  
de los empleados del departamento 121.*/
```

```
SELECT nomem  
FROM temple  
WHERE salar > (SELECT MIN(salar)*2  
              FROM temple  
              WHERE numde=121)  
ORDER BY 1;
```

```
--Otra solución:
```

```
SELECT nomem  
FROM temple  
WHERE salar > ANY (SELECT salar*2  
                  FROM temple  
                  WHERE numde=121)  
ORDER BY 1;
```

/*3.Obtener por orden alfabético los nombres y los salarios de los empleados cuyo salario es inferior a tres veces la comisión más baja existente distinta de NULL.*/

```
SELECT nomem, salar
FROM temple
WHERE salar < (SELECT MIN(comis)*3
               FROM temple)
ORDER BY 1;
```

/*4. Obtener, utilizando el predicado BETWEEN, por orden alfabético los números (identificador único), nombres y los salarios de los empleados con hijos cuyo salario dividido por su número de hijos cumpla una, o ambas, de las dos condiciones siguientes:

- Que sea inferior a 1200 Euros.
- Que sea superior al doble de su comisión.

*/

--Vemos todos los datos que nos interesan.

```
SELECT numem, nomem, salar, comis, numhi, salar/numhi AS
'SALAR/NUMHI', comis*2 AS 'COMIS*2'
FROM temple
WHERE numhi>0
ORDER BY comis;
```

/*Las siguientes consultas no dan una solución correcta, puesto que los empleados 150 y 370 no salen y deberían salir, ya que su

(salar/numhi)>2*NULL

*/

```
SELECT numem, nomem, salar
FROM temple
WHERE (salar/numhi < 1200 OR salar/numhi > 2*comis) AND numhi
> 0
ORDER BY 1;
```

```

SELECT numem,nomem, salar
FROM temple
WHERE salar/numhi NOT BETWEEN 1200 AND 2*comis AND numhi > 0
ORDER BY 1

```

--SOLUCIÓN:

```

SELECT numem, nomem, salar
FROM temple
WHERE (salar/numhi NOT BETWEEN 1200 and 2*ISNULL(0,comis))
AND numhi > 0
ORDER BY 1;

```

--Otras soluciones:

```

SELECT numem, nomem, salar
FROM temple
WHERE (salar/numhi < 1200 or salar/numhi > 2*
ISNULL(0,comis)) AND numhi > 0
ORDER BY 1;

```

```

SELECT numem, nomem, salar
FROM temple
WHERE salar/numhi < 1200 AND numhi > 0
UNION
SELECT numem, nomem, salar
FROM temple
WHERE numhi>0 AND comis IS NULL
ORDER BY 1;

```

-- Si tienen hijos y no tiene comisión también lo cumplen.

```

SELECT numem, nomem, salar
FROM temple
WHERE ((salar/numhi NOT BETWEEN 1200 and 2* comis) OR comis
IS NULL ) AND numhi > 0

```

/*5. Obtener por orden alfabético los nombres de los empleados cuyo primer apellido es Mora o empieza por Mora.*/

```
SELECT nomem
FROM temple
WHERE nomem LIKE 'MORA%'
ORDER BY 1;
```

/*6. Obtener por orden alfabético los nombres de los empleados cuyo primer apellido termina en EZ y su nombre de pila termina en O y tiene al menos tres letras.*/

```
SELECT nomem
FROM temple
WHERE nomem LIKE '%EZ,%__O'
ORDER BY 1;
```

/*7. Obtener, utilizando el predicado IN, por orden alfabético

los nombres de los empleados del departamento 111 cuyo salario es igual a alguno de los salarios del departamento 112 ¿Cómo lo obtendrías con el predicado ANY?.*/

--Vemos todos los datos que nos interesan.

```
SELECT salar, numde
FROM temple
WHERE numde = 112;
```

```
SELECT nomem, numde, salar
FROM temple
WHERE numde = 111;
```

--SOLUCIÓN:

```
SELECT nomem
FROM temple
WHERE numde = 111 AND salar IN (SELECT salar
                                FROM temple
                                WHERE numde = 112)
ORDER BY 1;
```

```

SELECT nomem
FROM temple
WHERE numde = 111 AND salar = ANY (SELECT salar
                                   FROM temple
                                   WHERE numde = 112)

ORDER BY 1;

```

/*8. Obtener por orden alfabético los nombres y comisiones de los empleados del departamento 110 si hay en él algún empleado que tenga comisión.*/

/*Lo que dice el ejercicio es que si hay en el departamento 110 algún empleado que tenga comisión, entonces sacar los empleados*/

```

SELECT nomem, comis
FROM temple
WHERE numde = 110 AND EXISTS (SELECT *
                              FROM temple
                              WHERE numde=110 AND comis IS
                              NOT NULL);

```

/*Comprueba que si pones todas las comisiones de los empleados del departamento 110 a NULL, no saldría ningún empleado*/

```

SELECT *
FROM temple
WHERE numde = 110;

```

```

UPDATE temple
SET COMIS=NULL
WHERE numde = 110;

```

```

SELECT nomem, comis
FROM temple
WHERE numde = 110 AND EXISTS (SELECT *
                              FROM temple
                              WHERE numde=110 AND comis IS
                              NOT NULL);

```

--Volvemos a dejar la tabla como antes.

```
UPDATE temple  
SET comis=500  
WHERE numem=180;
```

/*9. Obtener por orden alfabético los nombres de los departamentos que tienen algún empleado sin comisión*/

--SOLUCIÓN con predicado EXISTS:

```
SELECT nomde  
FROM tdepto D  
WHERE EXISTS (SELECT *  
               FROM TEMPLE E  
               WHERE E.numde = D.numde AND comis IS NULL)  
ORDER BY 1;
```

--SOLUCIÓN con predicado ANY:

```
SELECT nomde  
FROM tdepto D  
WHERE numde = ANY (SELECT numde  
                   FROM TEMPLE  
                   WHERE comis IS NULL)  
ORDER BY 1;
```

--SOLUCIÓN con predicado IN:

```
SELECT nomde  
FROM tdepto D  
WHERE numde IN (SELECT numde  
                FROM TEMPLE  
                WHERE comis IS NULL)  
ORDER BY 1;
```

--SOLUCIÓN usando JOIN:

```
SELECT DISTINCT nomde
FROM tdepto D JOIN temple E ON (D.numde=E.numde)
WHERE comis IS NULL
ORDER BY 1;
```

/*10. Para los departamentos cuyo nombre empieza por las letras O o P, mostrar el nombre del departamento y el nombre del departamento del que depende.*/

--Vemos que la información es correcta.

```
SELECT *
FROM tdepto T1 JOIN tdepto T2 ON (T1.depde=T2.numde)
WHERE T1.nomde LIKE '[OP]%' ;
```

--SOLUCIÓN:

```
SELECT T1.nomde,T2.nomde
FROM tdepto T1 JOIN tdepto T2 ON (T1.depde=T2.numde)
WHERE T1.nomde LIKE '[OP]%' ;
```

/*11. Para los departamentos del centro 20 obtener el nombre del departamento y el nombre del director.*/

--Vemos que la información es correcta.

```
SELECT *
FROM tdepto D JOIN temple E ON (D.direc=E.numem)
WHERE numce=20;
```

--SOLUCIÓN:

```
SELECT nomde,nomem
FROM tdepto D JOIN temple E ON (D.direc=E.numem)
WHERE numce=20;
```

/*12. Obtener el nombre de los departamentos que no tienen empleados con menos de dos hijos. Realiza la consulta primero con un predicado ALL y después con un predicado EXISTS.*/
/*Nos pide los nombres de los departamentos que no tienen empleados con 0, ni 1 hijo. El departamento sale si todos sus empleados tienen 2 o más hijos*/
/*Al ejecutar las dos primeras consultas vemos que deben salir todos los departamentos menos el 112 y el 121.*/

```
SELECT *  
FROM tdepto  
ORDER BY numde;
```

```
SELECT numem,nomem, numhi, numde  
FROM temple  
ORDER BY 4;
```

--SOLUCIÓN con predicado ALL:

```
SELECT nomde  
FROM tdepto D  
WHERE 2 <= ALL (SELECT numhi  
                FROM temple E  
                WHERE D.numde=E.numde);
```

--Con predicado EXISTS:

```
SELECT *  
FROM tdepto D  
WHERE NOT EXISTS (SELECT *  
                  FROM temple E  
                  WHERE E.numde=D.numde AND numhi<2);
```


--En el subselect del EXISTS en vez de poner *, se suele poner 1 porque es más óptimo.

--SOLUCIÓN:

```
SELECT nomde
FROM tdepto D
WHERE NOT EXISTS (SELECT 1
                  FROM temple E
                  WHERE E.numde=D.numde AND numhi<2);
```