
PRÁCTICA 12: CONSULTA Y ACTUALIZACIÓN

1. Obtener el nombre de los conductores con categoría 15.

```
SELECT nombre
FROM conductores
WHERE categ = 15
```

2. Obtener la descripción de los proyectos en los que se haya realizado trabajos durante los días 11 al 15 de septiembre de 2012.

Solución con subconsultas

```
SELECT descrip
FROM proyectos
WHERE codP IN (SELECT codP
FROM trabajos
WHERE fecha BETWEEN '11/sep/02' AND '15/sep/02');
```

Solución SQL-86

```
SELECT descrip
FROM proyectos, trabajos
WHERE proyectos.codP = trabajos.codP
AND fecha BETWEEN '11/sep/02' AND '15/sep/02';
```

Solución con Inner Join

```
SELECT descrip
FROM proyectos INNER JOIN trabajos ON proyectos.codP = trabajos.codP
WHERE fecha BETWEEN '11/sep/02' AND '15/sep/02';
```

3. Obtener el nombre de los conductores que hayan trabajado con una Hormigonera, ordenados descendientemente.

```
SELECT nombre
FROM conductores
WHERE codC IN (SELECT codC
FROM trabajos
WHERE codM IN (SELECT codM
FROM maquinas
WHERE nombre = 'Hormigonera'))
ORDER BY nombre DESC;
```

Con Inner Join:

```
SELECT conductores.nombre
FROM conductores INNER JOIN trabajos
on conductores.codC = trabajos.codC
INNER JOIN maquinas
on trabajos.codM = maquinas.codM
WHERE maquinas.nombre like 'Hormigonera'
ORDER BY nombre DESC;
```

4. Obtener el nombre de los conductores que hayan trabajado con una Hormigonera en proyectos de Arganda.

Solución con subconsultas

```
SELECT nombre
FROM conductores
WHERE codC IN (SELECT codC
                FROM trabajos
                WHERE codM IN (SELECT codM
                              FROM maquinas
                              WHERE nombre = 'Hormigonera')
                AND codP IN (SELECT codP
                              FROM proyectos
                              WHERE localidad = 'Arganda'));
```

Solución con SQL-86

```
SELECT conductores.nombre
FROM conductores, trabajos, maquinas, proyectos
WHERE proyectos.localidad = 'Arganda'
AND maquinas.nombre = 'Hormigonera'
AND proyectos.codP = trabajos.codP
AND trabajos.codM = maquinas.codM
AND trabajos.codC = conductores.codC;
```

Solución con Inner Join

```
SELECT conductores.nombre
FROM conductores INNER JOIN trabajos ON conductores.codC = trabajos.codC
INNER JOIN maquinas ON trabajos.codM = maquinas.codM
INNER JOIN proyectos ON proyectos.codP = trabajos.codP
WHERE proyectos.localidad like 'Arganda' AND maquinas.nombre like 'Hormigonera';
```

5. Obtener el nombre de los conductores y descripción del proyecto, para aquellos conductores que hayan trabajado con una Hormigonera en proyectos de Arganda durante los días 12 al 17 de Septiembre.

Solución con SQL-86

```
SELECT conductores.nombre, proyectos.descrip
FROM conductores, trabajos, maquinas, proyectos
WHERE proyectos.localidad = 'Arganda' AND maquinas.nombre = 'Hormigonera'
AND trabajos.fecha BETWEEN '10/sep/02' AND '18/sep/02'
AND proyectos.codP = trabajos.codP AND trabajos.codM = maquinas.codM
AND trabajos.codC = conductores.codC;
```

Solución con Inner Join

```
SELECT conductores.nombre, proyectos.descrip
FROM conductores INNER JOIN trabajos ON conductores.codC = trabajos.codC
INNER JOIN maquinas ON trabajos.codM = maquinas.codM
INNER JOIN proyectos ON proyectos.codP = trabajos.codP
WHERE proyectos.localidad = 'Arganda' AND maquinas.nombre = 'Hormigonera'
AND trabajos.fecha BETWEEN '2012/09/12' AND '2012/09/18'
```

Solución con Inner Join y subconsultas

```
SELECT conductores.nombre, proyectos.descrip
FROM conductores INNER JOIN trabajos ON conductores.codC = trabajos.codC
INNER JOIN proyectos ON proyectos.codP = trabajos.codP
WHERE trabajos.codM in (SELECT codM FROM maquinas where nombre like 'Hormigonera')
AND proyectos.localidad = 'Arganda'
AND trabajos.fecha BETWEEN '2012/09/12' AND '2012/09/18'
```

6. Obtener los conductores que trabajan en los proyectos de José Pérez.

Solución con SQL-86

```
SELECT DISTINCT conductores.nombre
FROM conductores, trabajos, proyectos
WHERE proyectos.cliente = 'José Pérez' AND proyectos.codP = trabajos.codP
AND trabajos.codC = conductores.codC;
```

Solución con Inner Join

```
SELECT DISTINCT conductores.nombre
FROM conductores INNER JOIN trabajos ON conductores.codC = trabajos.codC
INNER JOIN proyectos ON proyectos.codP = trabajos.codP
WHERE proyectos.cliente = 'José Pérez'
```

```
SELECT DISTINCT conductores.nombre
FROM conductores
Where codC in (select codC from trabajos
               Where codP in (select codP from proyectos
                              WHERE proyectos.cliente like 'José Pérez'));
```

7. Obtener el nombre y localidad de los conductores que NO trabajan en los proyectos de José Pérez

Solución errónea. Selecciona conductores que no hayan trabajado en alguna ocasión en proyectos de José Pérez, ya que establece la condición por cada fila. Puede haber conductores que aparezcan en una fila en proyectos que no sean de José Pérez y en otras filas en proyectos de otros clientes.

```
SELECT DISTINCT conductores.nombre, conductores.localidad
FROM conductores INNER JOIN trabajos ON conductores.codC = trabajos.codC
INNER JOIN proyectos ON proyectos.codP = trabajos.codP
WHERE proyectos.cliente <> 'José Pérez'
```

Solución correcta

```
SELECT conductores.nombre, conductores.localidad
FROM conductores
WHERE codC NOT IN (SELECT codC
                   FROM trabajos
                   WHERE codP IN (SELECT codP
                                  FROM proyectos
                                  WHERE cliente = 'José Pérez'));
```

8. Obtener todos los datos de los proyectos realizados en Rivas o que sean de un cliente llamado José.

Solución SQL-92

La evaluación de la condición lógica no es la correcta. Evalúa todas las condiciones hasta el OR como primer operando y deja como segundo operando la última condición.

```
SELECT DISTINCT proyectos.*
FROM proyectos
WHERE proyectos.localidad like 'Rivas'
OR proyectos.cliente like '%José%';
```

9. Obtener los conductores que habiendo trabajado en algún proyecto, figuren sin horas trabajadas.

```
SELECT DISTINCT conductores.*
FROM conductores inner join trabajos ON trabajos.codC = conductores.codC
WHERE tiempo IS NULL;
```

```
SELECT DISTINCT conductores.*
FROM conductores
Where codC in (select codc from trabajos WHERE tiempo IS NULL);
```

10. Obtener los empleados que tengan como apellido Pérez y hayan trabajado en proyectos de localidades diferentes a las suyas

```
SELECT conductores.*
FROM conductores INNER JOIN trabajos ON conductores.codC = trabajos.codC
INNER JOIN proyectos ON proyectos.codP = trabajos.codP
WHERE conductores.nombre like '%Pérez%'
AND proyectos.localidad <> conductores.localidad;
```

```
Select nombre
From conductores
where nombre like '%Perez'
and code not in (select code from trabajos
                 where codP in (select codP from proyectos where
                               conductores.localidad like proyectos.localidad)
and code in (select code from trabajos);
```

11. Obtener el nombre de los conductores y la localidad del proyecto, para aquellos conductores que hayan trabajado con máquinas con precio hora comprendido entre 10000 y 15000 ptas.

```
SELECT DISTINCT C.nombre As "Conductor", P.localidad AS "Localidad del Proyecto"
FROM conductores C INNER JOIN trabajos T ON C.codC = T.codC
INNER JOIN maquinas M ON M.codM = T.codM
INNER JOIN proyectos P ON P.codP = T.codP
WHERE M.preciohora BETWEEN 10000 AND 15000;
```

12. Obtener el nombre y localidad de los conductores, y la localidad del proyecto para aquellos proyectos que sean de Rivas y en los que no se haya utilizado una máquina de tipo Excavadora o una máquina de tipo Hormigonera.

Solución correcta con SQL-86. Cuidado con la expresión lógica de la subconsulta. Es necesario poner los paréntesis para indicar el orden de evaluación de la expresión lógica.

```
SELECT C.nombre As Conductor, C.localidad As Localidad_Conductor, P.localidad As Localidad_del
_Proyecto, M.nombre As Maquina
FROM conductores C, trabajos T, proyectos P, maquinas M
WHERE P.localidad = 'Rivas'
```

```

AND P.CodP NOT IN (SELECT codP
                    FROM trabajos T1, maquinas M1
                    WHERE (M1.nombre = 'Excavadora' OR M1.nombre = 'Hormigonera')
                    AND M1.codM = T1.codM )
AND M.codM = T.codM AND P.codP = T.codP AND T.codC = C.codC;

```

Solución SQL-92

```

SELECT C.nombre As Conductor, C.localidad As Localidad_Conductor, P.localidad As Localidad _del
_Proyecto, M.nombre As Maquina
FROM conductores C INNER JOIN trabajos T ON C.codC = T.codC
INNER JOIN maquinas M ON M.codM = T.codM
INNER JOIN proyectos P ON P.codP = T.codP
WHERE P.localidad like 'Rivas'
AND P.CodP NOT IN (SELECT codP
                    FROM trabajos T1 inner join maquinas M1 ON M1.codM = T1.codM
                    WHERE (M1.nombre = 'Excavadora' OR M1.nombre = 'Hormigonera'));

```

13. Obtener todos los datos de los proyectos, y para aquellos proyectos realizados el día 15 de Septiembre, además incluir el nombre y localidad de los conductores que hayan trabajado en dicho proyecto.

En Interbase, ponemos en la segunda consulta cadenas de caracteres con el mismo dominio que las correspondientes columnas de la otra consulta.

```

SELECT DISTINCT p.*, c.nombre, c.localidad
FROM conductores C INNER JOIN trabajos T ON C.codC = T.codC
INNER JOIN proyectos P ON P.codP = T.codP
WHERE T.fecha = '2002-09-12'
UNION
SELECT DISTINCT proyectos.*, ''
FROM proyectos
WHERE codP NOT IN (SELECT codP FROM trabajos WHERE fecha = '2002-09-12');

```

14. Obtener el nombre de los conductores y el nombre y localidad de los clientes, en los que se haya utilizado la máquina con precio hora más elevado.

```

SELECT DISTINCT conductores.nombre, proyectos.cliente, proyectos.localidad
FROM conductores C INNER JOIN trabajos T ON C.codC = T.codC
INNER JOIN proyectos P ON P.codP = T.codP
WHERE T.codM IN (SELECT codM
                  FROM maquinas
                  WHERE preciohora IN (SELECT MAX(preciohora)
                                       FROM maquinas) );

```

15. Obtener todos los datos de los proyectos que *siempre* han utilizado la máquina de precio más bajo.

Solución con subconsultas

```

SELECT DISTINCT proyectos.*
FROM proyectos
WHERE proyectos.codP NOT IN (SELECT codP
                              FROM trabajos
                              WHERE codM IN (SELECT codM
                                              FROM maquinas
                                              WHERE preciohora > (SELECT MIN(preciohora)
                                                                    FROM maquinas)))
AND proyectos.codP in (select codP from trabajos);

```

Solución con inner join

```

SELECT DISTINCT proyectos.*
FROM trabajos inner join proyectos on AND proyectos.codP = trabajos.codP
WHERE trabajos.codP NOT IN (SELECT codP
                             FROM trabajos
                             WHERE codM IN (SELECT codM
                                             FROM maquinas
                                             WHERE preciohora <> (SELECT MIN(preciohora)
                                                                    FROM maquinas) ))

```

16. Obtener los proyectos en los que haya trabajado el conductor de categoría más alta menos dos puntos, con la máquina de precio/ hora más bajo.

Solución con inner join

```

SELECT DISTINCT proyectos.*
FROM trabajos inner join proyectos ON proyectos.codP = trabajos.codP
WHERE codM IN (SELECT codM
               FROM maquinas
               WHERE preciohora IN (SELECT MIN(preciohora) FROM
                                     maquinas) )
AND codC IN (SELECT codC FROM conductores
             WHERE categoria IN (SELECT MAX(categoria) - 2 FROM
                                   conductores) ) ;

```

Solución con subconsultas sólo

```

SELECT DISTINCT CodP
FROM trabajos
WHERE codM IN (SELECT codM
               FROM maquinas
               WHERE preciohora IN (SELECT MIN(preciohora) FROM maquinas) )
AND codC IN (SELECT codC FROM conductores
             WHERE categoria IN (SELECT MAX(categoria) - 2 FROM conductores) );

```

17. Obtener por cada uno de los clientes el tiempo total empleado en sus proyectos.

```

SELECT cliente, SUM(Tiempo)
FROM proyectos LEFT JOIN trabajos ON proyectos.codP = trabajos.codP
GROUP BY cliente;

```

18. Obtener por cada uno de los proyectos existentes en la BD, la descripción del proyecto, el cliente y el total a facturar en ptas y en euros. Ordenar el resultado por uno de los totales y por cliente.

```

SELECT cliente, descrip, SUM(Tiempo*preciohora) As TotalPtas,
SUM(Tiempo*preciohora) / 166.366 AS TotalEuros
FROM proyectos LEFT JOIN trabajos ON proyectos.codP = trabajos.codP
LEFT JOIN maquinas ON trabajos.codM = maquinas.codM
GROUP BY cliente, descrip

```

ORDER BY 3,1;

19. Obtener para el proyecto que más se vaya a facturar la descripción del proyecto, el cliente y el total a facturar en Ptas. y en euros

```
SELECT cliente, descrip, SUM(Tiempo*preciohora) As TotalPtas,  
SUM(Tiempo*preciohora) / 166.386 AS TotalEuros  
FROM proyectos INNER JOIN trabajos ON proyectos.codP = trabajos.codP  
INNER JOIN maquinas ON trabajos.codM = maquinas.codM  
GROUP BY descrip, cliente  
HAVING SUM(Tiempo*preciohora) = (SELECT SUM(Tiempo*preciohora)  
FROM trabajos INNER JOIN maquinas  
ON trabajos.codM = maquinas.codM  
GROUP BY CodP  
Order by 1 desc  
Limit 1);
```

20. Obtener los conductores que hayan trabajado en todos los proyectos de la localidad de Arganda.

Solución con inner join

```
SELECT trabajos.codc, nombre, COUNT(distinct trabajos.codp)  
FROM conductores INNER JOIN trabajos ON conductores.codC = trabajos.codC  
INNER JOIN proyectos ON trabajos.codp=proyectos.codp  
WHERE proyectos.localidad like 'Arganda'  
GROUP BY trabajos.codc, nombre  
HAVING COUNT(DISTINCT trabajos.codp) = (SELECT COUNT(*) FROM proyectos  
WHERE localidad like 'Arganda');
```

Solución con subconsultas sólo

```
SELECT trabajos.codc, COUNT(distinct trabajos.codp)  
FROM trabajos  
WHERE codp in (select codp from proyectos where localidad like 'Arganda')  
GROUP BY trabajos.codc  
HAVING COUNT(DISTINCT trabajos.codp) = (SELECT COUNT(*) FROM proyectos  
WHERE localidad like 'Arganda');
```

21. Obtener el tiempo máximo dedicado a cada proyecto para aquellos proyectos en los que haya participado más de un conductor diferente.

Solución errónea. Se pueden seleccionar proyectos que hayan tenido dos veces el mismo conductor en trabajos.

```
SELECT CodP, MAX(tiempo)  
FROM trabajos  
GROUP BY CodP  
HAVING COUNT(CodC) > 1;
```

Solución correcta.

```
SELECT CodP, MAX(tiempo) As MaxTiempo  
FROM trabajos
```

```
GROUP BY CodP
HAVING COUNT(DISTINCT CodC) > 1;
```

22. Obtener el número de partes de trabajo, código del proyecto, descripción y cliente para aquel proyecto que figure con más partes de trabajo.

```
SELECT proyectos.CodP, descrip, COUNT(*) As "Número de Trabajos"
FROM proyectos inner join trabajos ON proyectos.codP = trabajos.codP
GROUP BY proyectos.CodP, cliente, descrip
HAVING COUNT(*) >= ALL (SELECT COUNT(*)
                        FROM trabajos
                        GROUP BY CodP);
```

23. Obtener la localidad cuyos conductores (al menos uno) haya participado en más de dos proyectos diferentes.

```
SELECT localidad
FROM conductores
WHERE CodC IN (SELECT CodC FROM trabajos
              GROUP BY CodC
              HAVING COUNT(DISTINCT CodP) > 2);
```