

1.1.2. Bloque - B**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 2002.

- Solución con subconsultas

```
SELECT descrip
  FROM proyectos
 WHERE codP IN (SELECT codP
                 FROM trabajos
                 WHERE fecha BETWEEN '2002-09-11' AND '2002-09-15');
```

- Solución con join

```
SELECT descrip
  FROM proyectos P, trabajos T
 WHERE P.codP = T.codP
        AND fecha BETWEEN '2002-09-11' AND '2002-09-15';
```

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;
```

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 condición de Join en el Where

```
SELECT C.nombre
FROM conductores C, trabajos T, maquinas M, proyectos P
WHERE
    T.codP = P.codP AND T.codM = M.codM AND T.codC = C.codC AND
    P.localidad = 'Arganda' AND M.nombre = 'Hormigonera' ;
```

- 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 = 'Arganda' AND maquinas.nombre = '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.**

```
SELECT C.nombre, P.descrip
FROM conductores C, trabajos T, maquinas M, proyectos P
WHERE codP = T.codP AND T.codM = M.codM AND T.codC = C.codC AND
    P.localidad = 'Arganda' AND M.nombre = 'Hormigonera' AND
    T.fecha BETWEEN '2002-09-12' AND '2002-09-17' ;
```

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

```
SELECT DISTINCT C.nombre
FROM conductores C, trabajos T, proyectos P
WHERE P.codP = T.codP AND T.codC = C.codC AND P.cliente = '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 C.nombre, C.localidad
FROM conductores C, trabajos T, proyectos P
WHERE P.codP = T.codP AND T.codC = C.codC AND
    P.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.

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

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

```
SELECT DISTINCT C.*
FROM conductores C, trabajos T
WHERE T.codC = C.codC AND tiempo IS NULL;
```

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

- Solución correcta en SqlServer e Internase.

```
SELECT C.*
FROM conductores C, trabajos T, proyectos P
WHERE P.codP = T.codP AND T.codC = C.codC AND
C.nombre like '%Pérez%' AND P.localidad <> C.localidad ;
```

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, trabajos T, proyectos P, maquinas M
WHERE M.preciohora BETWEEN 10000 AND 15000 AND M.codM = T.codM AND
P.codP = T.codP AND T.codC = C.codC;
```

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. 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;
```

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, trabajos T, proyectos P
WHERE P.codP = T.codP AND T.codC = C.codC AND T.fecha = '2002-09-15'
UNION
SELECT DISTINCT proyectos.*, ' ', ' ', ' '
FROM proyectos
WHERE codP NOT IN (SELECT codP FROM trabajos
WHERE fecha = '2002-09-15');
```

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 C.nombre, P.cliente, P.localidad
FROM conductores C, trabajos T, proyectos P
WHERE P.codP = T.codP AND T.codC = C.codC AND
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.

```
SELECT DISTINCT P.*
FROM trabajos T, proyectos P
WHERE P.codP = T.codP AND T.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.

```
SELECT DISTINCT P.*
FROM trabajos T, proyectos P
WHERE P.codP = T.codP AND
codM IN (SELECT codM
FROM maquinas
WHERE preciohora IN (SELECT MIN(preciohora)
FROM maquinas)) AND
codC IN (SELECT codC
FROM conductores
WHERE categ IN (SELECT MAX(categ) - 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 proyectos.codP
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

```
1.
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) >= ALL
      (SELECT SUM(Tiempo*preciohora)
      FROM trabajos INNER JOIN maquinas
      ON trabajos.codM = maquinas.codM
      GROUP BY CodP);
```

```
2.
CREATE VIEW sumasProyecto (cliente, descrip, totalPtas, totalEuros)
AS SELECT cliente, descrip, SUM(tiempo*preciohora),
      SUM(tiempo*preciohora) / 166.386
FROM proyectos INNER JOIN trabajos ON proyectos.codP = trabajos.codP
      INNER JOIN maquinas ON trabajos.codM = maquinas.codM
GROUP BY descrip, cliente;
```

```
SELECT *
FROM sumasProyecto
WHERE totalEuros IN (SELECT MAX(totalEuros)
      FROM sumasProyecto)
```

```
/* DROP VIEW sumasProyecto; */
```

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

```
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='Arganda'
GROUP BY trabajos.codc
HAVING COUNT(DISTINCT trabajos.codp) = (SELECT COUNT(*)
                                         FROM proyectos
                                         WHERE localidad = '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 aquél proyecto que figure con más partes de trabajo.

```
SELECT P.CodP, descrip, cliente, COUNT(*) As "Número de Trabajos"
FROM proyectos P, trabajos T
WHERE P.codP = T.codP
GROUP BY P.CodP, descrip, cliente
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);
```