

CICLISMO

38.

SELECT

m.codigo, m.color

FROM

llevar L,

maillot M,

ciclista C,

etapa E

WHERE

NOT EXISTS(SELECT

m.codigo

FROM

maillot M2,

llevar L2,

ciclista C2,

etapa E2

WHERE

m.codigo = l.codigo

AND l.dorsal = c.dorsal

AND l.netapa = e.netapa)

GROUP BY m.codigo , m.color

39,

SELECT

e.netapa, e.salida, e.llegada

FROM

etapa E,

puerto P

WHERE

km IN (SELECT

```
        km
FROM
    etapa
WHERE
    km > 190)
    AND e.netapa = p.netapa
GROUP BY e.netapa
HAVING COUNT(nompuerto) >= 2
```

40,

```
SELECT
    c.dorsal, c.nombre
FROM
    ciclista C
WHERE
    EXISTS( SELECT
        l.dorsal
    FROM
        llevar l
    WHERE
        l.dorsal = 20
        AND NOT EXISTS( SELECT
            l2.dorsal
        FROM
            llevar L2
        WHERE
            l2.dorsal = c.dorsal
            AND l2.codigo = l.codigo))
```

41,

```
SELECT
```

```

        c.dorsal, c.nombre
FROM
    ciclista c,
    llevar l
WHERE
    l.codigo IN (SELECT
        codigo
    FROM
        llevar
    WHERE
        dorsal = 20)
    AND c.dorsal != 20
    AND c.dorsal = l.dorsal
    group by c.dorsal, c.nombre

```

42.

```

SELECT
    c.dorsal, c.nombre
FROM
    ciclista c
WHERE
    c.dorsal NOT IN (SELECT
        l.dorsal
    FROM
        llevar l
    WHERE
        l.codigo IN (SELECT
            l2.codigo
        FROM
            llevar l2
        WHERE

```

l2.dorsal = 20))

43.

```
SELECT c.dorsal, c.nombre
FROM ciclista c
WHERE NOT EXISTS (
    SELECT codigo
    FROM llevar
    WHERE dorsal = 20
    AND codigo NOT IN (
        SELECT codigo
        FROM llevar
        WHERE dorsal = c.dorsal
    )
)
AND c.dorsal !=20;
```

44.

```
SELECT c.dorsal, c.nombre
FROM ciclista c
WHERE NOT EXISTS (
    SELECT codigo
    FROM llevar l1
    WHERE l1.dorsal = 20 AND NOT EXISTS (
        SELECT codigo
        FROM llevar l2
        WHERE l2.dorsal = c.dorsal AND l2.codigo = l1.codigo
    )
) AND NOT EXISTS (
    SELECT codigo
    FROM llevar l3
```

```

WHERE l3.dorsal = c.dorsal AND NOT EXISTS (
    SELECT codigo
    FROM llevar l4
    WHERE l4.dorsal = 20 AND l4.codigo = l3.codigo
)
) and c.dorsal!=20

```

45.

```

CREATE VIEW kms AS

SELECT
    l.dorsal, m.codigo, SUM(e.km) AS kms
FROM
    etapa E,
    llevar L,
    maillot M
WHERE
    e.netapa = l.netapa
    AND m.codigo = l.codigo
GROUP BY l.dorsal , m.codigo

SELECT
    l.dorsal, c.nombre, m.color, SUM(e.km) AS kmss
FROM
    etapa E,
    maillot M,
    llevar L,
    ciclista C
WHERE
    e.netapa = l.netapa
    AND m.codigo = l.codigo
    AND c.dorsal = l.dorsal
GROUP BY l.dorsal , c.nombre , m.color

```

```
HAVING SUM(e.km) = (SELECT  
    MAX(kms)  
FROM  
    kms)
```

46.

```
SELECT dorsal, nombre  
FROM ciclista  
WHERE dorsal IN (  
    SELECT dorsal  
    FROM llevar  
    GROUP BY dorsal  
    HAVING COUNT(DISTINCT codigo) = (  
        SELECT COUNT(DISTINCT codigo) - 3  
        FROM llevar  
        WHERE dorsal = 1  
    )  
    AND dorsal != 1  
)
```

47,

```
SELECT  
    e.netapa, e.km  
FROM  
    etapa E,  
    puerto P  
WHERE  
    p.netapa = e.netapa  
GROUP BY e.netapa , e.km
```

MUSICA

22.

```
select a.nombre
```

```
from artista A, pertenece P, grupo G, club C
```

```
where g.pais='Inglaterra' AND c.num>500 AND a.dni=p.dni AND p.cod=g.cod AND  
c.cod_gru=g.cod
```

```
group by a.nombre
```

23.

```
SELECT
```

```
    c.titulo
```

```
FROM
```

```
    esta E,
```

```
    cancion C,
```

```
    disco D,
```

```
    grupo G
```

```
WHERE
```

```
    e.can = c.cod AND e.cod = d.cod
```

```
    AND g.cod = d.cod_gru
```

```
    AND g.nombre = 'U2'
```

24.

```
SELECT DISTINCT
```

```
    a1.nombre AS 'VOZ', a2.nombre AS 'GUITARRA'
```

```
FROM
```

```
    artista a1, pertenece p1, grupo g1,
```

```
    artista a2, pertenece p2, grupo g2
```

```
WHERE
```

```
    g1.pais = 'España' AND g2.pais = 'España'
```

```
    AND a1.dni = p1.dni AND p1.cod = g1.cod AND p1.funcion = 'VOZ'
```

```
    AND a2.dni = p2.dni AND p2.cod = g2.cod AND p2.funcion = 'GUITARRA'
```

```
AND g1.cod<>g2.cod  
ORDER BY a1.nombre;
```

25.

```
SELECT  
    a.nombre  
FROM  
    grupo G,  
    artista A,  
    pertenece P  
WHERE  
    a.dni = p.dni AND p.cod = g.cod  
GROUP BY a.nombre  
HAVING COUNT(a.nombre) > 1
```

26.

```
NI IDEA  
SELECT distinct titulo  
FROM cancion  
WHERE NOT EXISTS (SELECT MAX(duracion) FROM cancion)
```

27.

```
SELECT  
    nombre, num  
FROM  
    club  
ORDER BY  
    num DESC  
LIMIT  
    1 OFFSET 9;
```


28.

```
SELECT
    a.nombre
FROM
    artista a,
    pertenece p,
    grupo g
WHERE
    p.funcion = 'BAJO' AND a.dni = p.dni
    AND p.cod = g.cod
GROUP BY a.dni
HAVING COUNT(DISTINCT g.cod) = 1
```

29.

```
select max(numcanciones), te.nombre
from(
SELECT
    count(c1.cod) AS NUMCANCIONES, co.nombre
FROM
    cancion C1, esta E1, disco D1, companyia co
WHERE
    e1.can = c1.cod AND e1.cod = d1.cod AND co.cod=d1.cod_comp
GROUP BY co.nombre) te
group by te.nombre
order by 1 desc
limit 1
```

