CONSULTAS VUELTAS CICLISTA - HÉCTOR GARCÍA GONZÁLEZ

Min 4,25 cc 0,5 teoria 3.15 cc 1.15

1)

USE vuelta;

SELECT etapa.numetapa FROM etapa

WHERE etapa.dorsal IN (SELECT ciclista.dorsal FROM ciclista WHERE ciclista.nomequipo IN

(SELECT equipo.nomequipo FROM equipo WHERE equipo.director LIKE 'M%' ));

2)

USE vuelta;

SELECT ciclista.nombre FROM ciclista

INNER JOIN lleva ON lleva.dorsal = ciclista.dorsal AND lleva.codigo EXISTS (SELECT Maillot.codigo FROM maillot WHERE premio < 500);

USE vuelta;

SELECT ciclista.nombre FROM ciclista, lleva

WHERE lleva.dorsal = ciclista.dorsal AND lleva.codigo EXISTS (SELECT Maillot.codigo FROM maillot WHERE premio < 500);

RESPUESTA SIN EXISTS QUE FUNCIONA

USE vuelta;

SELECT ciclista.nombre FROM ciclista, lleva

WHERE lleva.dorsal = ciclista.dorsal

AND lleva.codigo IN (SELECT Maillot.codigo FROM maillot WHERE premio < 500);

USE vuelta;

SELECT distinct ciclista.nombre FROM ciclista, lleva

WHERE lleva.dorsal = ciclista.dorsal

AND lleva.codigo IN (SELECT Maillot.codigo FROM maillot WHERE premio < 500);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*GEMMMA

USE vuelta;

SELECT ciclista.nombre FROM ciclista, lleva

WHERE lleva.dorsal = ciclista.dorsal

AND lleva.codigo IN (SELECT Maillot.codigo FROM maillot WHERE premio < 500) AND Maillot.codigo = lleva.codigo);

USE vuelta;

SELECT ciclista.nombre FROM ciclista, lleva

WHERE lleva.dorsal = ciclista.dorsal AND EXISTS (SELECT Maillot.codigo FROM maillot WHERE premio < 500

AND maillot.codigo = lleva.codigo);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*MEJOR RESPUETSA

USE vuelta;

SELECT DISTINCT ciclista.nombre FROM ciclista, lleva

WHERE lleva.dorsal = ciclista.dorsal AND EXISTS (SELECT Maillot.codigo FROM maillot WHERE premio < 500

AND maillot.codigo = lleva.codigo);

USE vuelta;

SELECT DISTINCT ciclista.nombre FROM ciclista, lleva

WHERE lleva.dorsal = ciclista.dorsal AND EXISTS (SELECT Maillot.codigo FROM maillot WHERE premio < 500

AND maillot.codigo = lleva.codigo);

3)

USE vuelta;

SELECT ciclista.nombre FROM ciclista

where nombre not like 'Miguel Indurain'

union

SELECT ciclista.nomequipo FROM ciclista

where ciclista.nomequipo IN (SELECT ciclista.nomequipo from ciclista where nombre.ciclista LIKE 'Miguel Indurain');

respuesta correcta gemma

USE vuelta;

SELECT DISTINCT c1.nombre FROM ciclista c1, ciclista c2

where c2.nombre = 'Miguel Indurain'

and c1.nomequipo = c2.nomequipo

and c1.edad < c2.edad;

USE vuelta;

SELECT c1.nombre FROM ciclista c1, ciclista c2

where c2.nombre = 'Miguel Indurain'

AND c1.nomequipo = c2.nomequipo

AND c1.edad < c2.edad;

4)

USE vuelta;

SELECT ciclista.nomequipo AS NombreEquipo, COUNT(dorsal) AS NumeroCiclistasEquipo FROM ciclista

GROUP By ciclista.nomequipo;

REPUESTA CORRECTA GEMMA

USE vuelta;

SELECT E.nomequipo AS NombreEquipo, COUNT(c.dorsal) AS NumeroCiclistasEquipo FROM (Equipo E LEFT JOIN ciclista c ON E.nomequipo=c.nomequipo)

GROUP By E.nomequipo;

5)

USE vuelta;

SELECT ciclista.nombre FROM ciclista

INNER JOIN Equipo ON Equipo.nomequipo = ciclista.nomequipo AND equipo.director LIKE 'Manuel Sainz';

6)

Con esto te saca los equipos con media de edad mayor de 25

USE vuelta;

SELECT ciclista.nomequipo,AVG(edad) AS EdadMedia FROM ciclista

GROUP By ciclista.nomequipo

HAVING AVG(edad) > 25

ORDER By AVG(edad);

ESTE ES LA RESPUESTA CORRECTA

USE vuelta;

SELECT ciclista.nomequipo,AVG(edad) AS EdadMedia FROM ciclista

WHERE ciclista.edad > 25

GROUP By ciclista.nomequipo

HAVING COUNT(nombre)>3;

7)

USE vuelta;

SELECT Equipo.nomequipo, AVG(Edad) FROM equipo

INNER JOIN ciclista ON Equipo.nomequipo = ciclista.nomequipo AND edad IN (SELECT AVG(Edad) FROM ciclista

WHERE ciclista.edad = (SELECT AVG(MAX(ciclista.edad)) FROM ciclista )) ;

USE vuelta;

SELECT Equipo.nomequipo, AVG(Edad) FROM equipo

INNER JOIN ciclista ON Equipo.nomequipo = ciclista.nomequipo AND CICLISTA.edad = (SELECT AVG(MAX(ciclista.edad)) FROM ciclista );

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* USE vuelta;

SELECT nomequipo, AVG(Edad) FROM ciclista

GROUP By nomequipo

having avg(edad) >= ALL (select avg(edad) from ciclista GROUP By nomequipo);

otra forma de hacerlo

USE vuelta;

SELECT nomequipo, AVG(Edad) FROM ciclista

GROUP By nomequipo

having avg(edad) =

(select top avg(edad) from ciclista GROUP By nomequipo order by avg(edad) desc);

8)

USE vuelta;

SELECT ciclista.nombre, etapa.kms FROM ciclista

INNER JOIN etapa ON etapa.kms > 240 AND Etapa.dorsal = ciclista.dorsal;

(esto solo me saca algunas pero no todas)

No hay etapas de mas de 270 kms

USE vuelta;

SELECT c.nombre FROM ciclista c

WHERE NOT EXISTS (SELECT \* FROM etapa E

WHERE e.kms > 270 and c.dorsal <> e.dorsal)

AND EXISTS (SELECT \* FROM etapa E2 WHERE E2. KMS > 270);

9)

USE vuelta;

SELECT DISTINCT MAX(maillot.premio) as MaximoPremio, equipo.nomequipo, ciclista.nombre FROM maillot, equipo, ciclista, lleva

WHERE maillot.premio IN (SELECT MAX(maillot.premio) FROM maillot WHERE maillot.codigo IN (SELECT codigo FROM lleva

WHERE lleva.dorsal IN (SELECT ciclista.dorsal FROM ciclista

WHERE ciclista.nomequipo IN (SELECT equipo.nomequipo FROM equipo))))

GROUP By equipo.nomequipo, ciclista.nombre;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*respuestas correcta gemma

USE vuelta;

SELECT MAX(dinero) as MaximoPremio, ciclista.nomequipo

FROM ciclista, (Select dorsal, SUM(Premio) dinero

FROM maillot, lleva

WHERE lleva.codigo = maillot.codigo

GROUP By dorsal) AS tabla

WHERE ciclista.dorsal = tabla.dorsal

GROUP By nomequipo;

USE vuelta;

SELECT MAX(dinero) as MaximoPremio, ciclista.nomequipo --dinero--alias--

FROM ciclista, (Select dorsal, SUM(Premio) dinero

FROM maillot, lleva

WHERE lleva.codigo = maillot.codigo

GROUP By dorsal) AS tabla --tabla seria una tabla temporal--

WHERE ciclista.dorsal = tabla.dorsal

GROUP By nomequipo;