

EJERCICIO RESUELTO

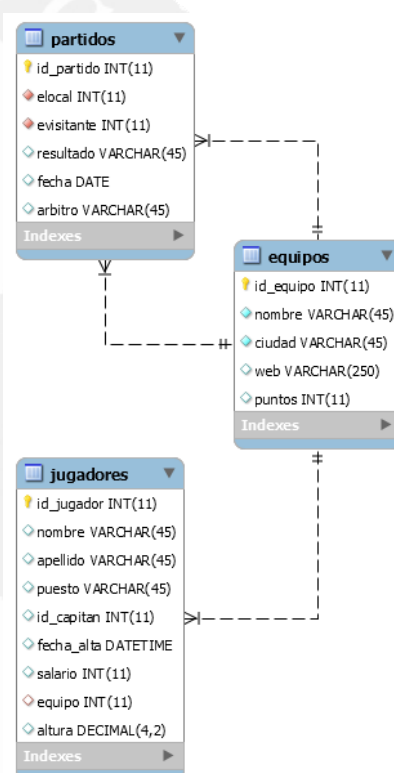
Módulo: Bases de Datos

Consultas con varias tablas

Descripción:

Sobre el caso práctico de la base de datos del diagrama siguiente realizar las consultas que se piden:

- Nombre de jugador, nombre de equipo y puesto del mismo.
- Datos de equipo y número de partidos que han jugado como locales.
- Datos de equipos con más de tres jugadores registrados.
- Nombre de todos los equipos y datos de sus partidos como locales en caso de haberlos.
- Datos de equipos y salario máximo entre sus jugadores.
- Datos del partido con mayor puntuación.
- Nombre y número de victorias de cada equipo.
- Nombre de equipo, nombre de su capitán para cada equipo.
- Explique desde el punto de vista de la optimización las diferencias entre usar EXISTS en una consulta o usar combinación de tablas.



Objetivos:

- Realizar consultas en las que intervienen varias tablas

Resolución:

- Nombre de jugador, nombre de equipo y puesto del mismo.

```
SELECT j.nombre, e.nombre, j.puesto
FROM jugadores j JOIN equipos e ON (j.equipo=e.id_equipo);
```

- Datos de equipo y número de partidos que han jugado como locales.

```
SELECT *, COUNT(*) AS nlocales
FROM equipos e JOIN partidos p ON (e.id_equipo=p.elocal)
GROUP BY p.elocal;
```

- Datos de equipos con más de tres jugadores registrados.

```
SELECT e.*
FROM jugadores j JOIN equipos e ON(j.equipo=e.id_equipo)
GROUP BY j.equipo HAVING COUNT(*)>3;
```

- Nombre de todos los equipos y datos de sus partidos como locales en caso de haberlos.

```
SELECT e.nombre, p.*
FROM equipos e JOIN partidos p ON (e.id_equipo=p.elocal);
```

- Datos de equipos y salario máximo entre sus jugadores.

```
SELECT e.*, MAX(j.salario) AS maxsal
FROM equipos e JOIN jugadores j ON (e.id_equipo=j.equipo)
GROUP BY j.equipo;
```

- Datos del partido con mayor puntuación.

El campo de resultado es de tipo cadena por lo que debemos usar alguna función de cadena para calcular el resultado numérico. Para ello tenemos la función SUBSTRING_INDEX que nos permite extraer una parte de una cadena separada por un delimitador, en nuestro caso un guion de este modo obtendremos, en primer lugar, la puntuación de cada partido junto con los datos del partido. En segundo lugar, usamos dicha consulta para extraer los datos del partido con más puntuación de entre los calculados anteriormente. Quedaría finalmente:

#Puntuación de cada partido

```
SELECT p.*, SUBSTRING_INDEX( resultado, '-', 1) +
SUBSTRING_INDEX(resultado, '-',1) AS puntos
FROM partidos p;
```

#Datos del partido con más puntuación

```

SELECT t.*
FROM (
    SELECT      p.*,   SUBSTRING_INDEX(p.resultado, '-',1)   +
    SUBSTRING_INDEX( p.resultado, '-',-1) AS puntos FROM partidos p)
AS t
WHERE t.puntos IN (
    SELECT      MAX(SUBSTRING_INDEX(k.resultado, '-',1)   +
    SUBSTRING_INDEX(k.resultado, '-',-1)) AS maxpuntos
    FROM partidos k );

```

- Nombre y número de victorias de cada equipo.

Necesitamos combinar las tablas partidos y equipos para determinar aquellos equipos cuyo resultado es superior al del contrincante dependiendo de si juegan o no en casa.

```

SELECT e.nombre, COUNT(*)
FROM equipos e JOIN partidos p
ON      (e.id_equipo=p.elocal) OR (e.id_equipo=p.evisitante) AND (
SUBSTRING_INDEX(resultado, '-',1)-SUBSTRING_INDEX(resultado, '-',-1)>0
AND e.id_ equipo=p.elocal OR SUBSTRING_INDEX(resultado, '-',1)-
SUBSTRING_INDEX(resultado, '-',-1)<0 AND e.id_ equipo=p.evisitante )
GROUP BY e.nombre;

```

- Nombre de equipo, nombre de su capitán para cada equipo.

```

SELECT DISTINCT(e.nombre),j.id_capitan FROM jugadores j JOIN equipos e
ON(j.equipo=e.id_equipo);

```

- Explique desde el punto de vista de la optimización las diferencias entre usar EXISTS en una consulta o usar combinación de tablas.

Cuando combinamos tablas el sistema debe hacer el producto cartesiano formando tablas más pesadas y por tanto requiriendo más memoria.

En el caso de usar EXISTS, al ser un tipo de subconsulta y por su forma de procesamiento (cada fila de la tabla más externa debe compararse con el predicado EXISTS) lo que se requiere es más proceso o cpu.

Como siempre, usar una u otra opción dependerá de nuestros requisitos y del hardware de que dispongamos.