Solución actividad de aprendizaje 5-07

Se trata de:

• Realizar consultas SELECT que de diversos tipos contra la base de datos **concursomusica.** Fundamentalmente se utilizarán subconsultas, aunque no es obligatorio usarlas.

Objetivos:

- Editar y ejecutar instrucciones SELECT sobre combinaciones de tablas.
- Obtener los resultados de funciones de agrupamiento sobre todas las filas agrupadas por una o varias columnas.
- Obtener resultado de consultas condicionadas a lo obtenido en una función de agrupamiento.
- Realizar subconsultas para obtener valores usados en cláusulas WHERE y HAVING
- Realizar subconsultas que generen una hoja de resultados fuente para otra consulta.

Ejercicios

Antes de comenzar los ejercicios de esta actividad, debes eliminar la columna **total_votos** de la tabla **canciones** para evitar que se use esa columna en cualquier consulta relacionada con obtener los votos que tiene una canción.

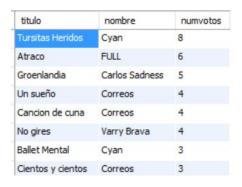
1.- A partir del contenido de la tabla **votos**, obtén cuántos votos tiene cada canción que ha sido votada. Se deben mostrar los títulos de las canciones, los nombres de los grupos a los que pertenecen las canciones y el número de votos.

SELECT titulo, nombre, count(*)
FROM grupos INNER JOIN canciones ON grupo=codgrupo
INNER JOIN votos ON cancion=numCancion
GROUP BY numCancion;

titulo	nombre	count(*)
Cancion de cuna	Correos	4
Cientos y cientos	Correos	3
Un sueño	Correos	4
Jovenes	Correos	3
No gires	Varry Brava	4
Calor	Varry Brava	1

2.- A partir de la consulta anterior, obtén los títulos de las 8 canciones más votadas.

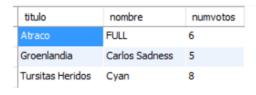
SELECT titulo, nombre, count(*) AS numvotos
FROM grupos INNER JOIN canciones ON grupo=codgrupo
INNER JOIN votos ON cancion=numCancion
GROUP BY numCancion
ORDER BY numvotos DESC
LIMIT 8;



3.- Obtén el título de todas las canciones que han recibido más votos que la canción titulada **No gires**, el número de votos recibidos y el nombre del grupo al que pertenecen.

SELECT titulo, nombre, count(*) AS numvotos
FROM canciones INNER JOIN votos ON cancion=numCancion
INNER JOIN grupos ON grupo=codgrupo
GROUP BY titulo
HAVING numvotos >

(SELECT count(*) FROM canciones INNER JOIN votos ON cancion=numCancion WHERE titulo='no gires');



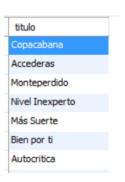
4.- Obtén los títulos de las canciones que no han recibido aún votos (los de las canciones que no están en la tabla votos). Resuelve este ejercicio de dos formas: con LEFT JOIN y con subconsulta.

Solución con LEFT JOIN:

SELECT titulo FROM canciones LEFT JOIN votos ON cancion=numcancion WHERE usuario IS NULL;

Solución con subconsulta:

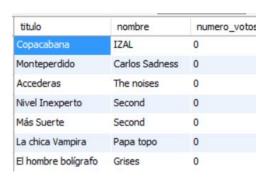
SELECT titulo
FROM canciones
WHERE numcancion NOT IN
(SELECT DISTINCT cancion FROM votos);



Es recomendable usar DISTINCT en esta subconsulta para evitar que NOT IN tenga que buscar entre muchísimos valores.

5.- Obtén los títulos de las canciones que no han recibido aún votos junto con el nombre del grupo al que pertenecen. Debe salir además una columna con valor cero para indicar el número de votos.

SELECT titulo, nombre, 0 AS numero_votos FROM canciones INNER JOIN grupos ON grupo=codgrupo WHERE numcancion NOT IN (SELECT DISTINCT cancion FROM votos);



6.- Obtén cuantos votos tiene cada una de las canciones, el título y el nombre del grupo al que pertenecen. Deben salir todas las canciones, no sólo las votadas.

SELECT titulo, nombre, count(*) AS numero_votos
FROM grupos INNER JOIN canciones ON grupo=codgrupo
INNER JOIN votos ON cancion=numCancion
GROUP BY numCancion
UNION

SELECT titulo, nombre, 0 AS numero_votos
FROM canciones INNER JOIN grupos ON grupo=codgrupo
WHERE numcancion NOT IN
(SELECT DISTINCT cancion FROM votos);

titulo	nombre	numero_votos
Todo nos parece	Astrud	1
Luz Artificial	Kitai	1
Clouds Factory	Rufus T. Fi	1
Hallo	Mäbu	1
Copacabana	IZAL	0
Monteperdido	Carlos Sad	0
Accederas	The noises	n

Con left

SELECT titulo, nombre, count(usuario) AS numero_votos FROM grupos inner JOIN canciones ON grupo=codgrupo left JOIN votos ON cancion=numCancion GROUP BY numCancion

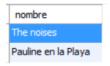
7.- Obtén los nombres de grupos que no hayan recibido votos, es decir, que todas sus canciones no hayan recibido votos.

SELECT nombre

FROM grupos

WHERE codgrupo NOT IN

(SELECT DISTINCT grupo FROM votos INNER JOIN canciones ON cancion=numcancion);



SELECT nombre FROM canciones LEFT JOIN votos ON cancion=numcancion INNER JOIN grupos ON grupo=codgrupo GROUP BY nombre HAVING COUNT(usuario)=0;

8.- Obtén los nombres y apellidos de los usuarios que han votado la canción titulada **Cancion de cuna.**

SELECT usuarios.nombre,apellidos FROM usuarios INNER JOIN votos ON usuario=user INNER JOIN canciones ON cancion=numcancion WHERE titulo='cancion de cuna';

CFGS DAM Bases de Datos. Curso 22-23

nombre	apellidos
Eustaquio	García García
Alberto	Cantúa Vitienes
Fernando	Sierra Torre
Daniel	Tresgallo Ceballos

9.- Obtén los nombres y apellidos de todos los usuarios que han votado a alguno de los grupos votados por **Elsa Frutos Núñez**.

SELECT nombre, apellidos

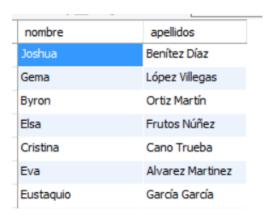
FROM votos INNER JOIN canciones ON cancion=numcancion

INNER JOIN usuarios ON user=usuario

WHERE grupo IN

(SELECT grupo FROM votos INNER JOIN canciones ON cancion=numcancion

INNER JOIN usuarios ON user=usuario
WHERE usuarios.nombre='elsa' AND apellidos='frutos núñez');



10.- Obtén las fechas de enero en que se han dado menos votos que los dados el día 2 de enero de este 2018.

SELECT fecha, count(*) AS numvotos
FROM votos
WHERE month(fecha)=1
GROUP BY fecha
HAVING numvotos <
(SELECT count(*) FROM votos WHERE fecha='2018-01-02');

fecha	numvotos
2018-01-01	2
2018-01-08	1
2018-01-09	1
2018-01-10	2
2018-01-13	1
2018-01-20	1

Podrás comprobar que no salen las fechas en las que no se han dado votos. Eso, por ahora, es imposible obtenerlo. Podremos obtenerlo cuando realicemos los procedimientos o PROCEDURES.