

Consultas MySQL #2

#1 Obtener listado de canciones del genero 1

SELECT * FROM canciones **WHERE** genero_id = 1

The screenshot shows the MySQL Workbench interface. The SQL editor contains the query: `SELECT * FROM canciones WHERE genero_id = 1;`. The results pane displays two rows of data:

	id	nombre	artista_id	duracion	genero_id	fecha	disponible
<input type="checkbox"/>	2	positions	1	00:02:57	1	2020-10-22	50
<input type="checkbox"/>	5	Golden	1	00:03:30	1	2019-12-12	50

#2 Mostrar el total de canciones que tiene el artista 1

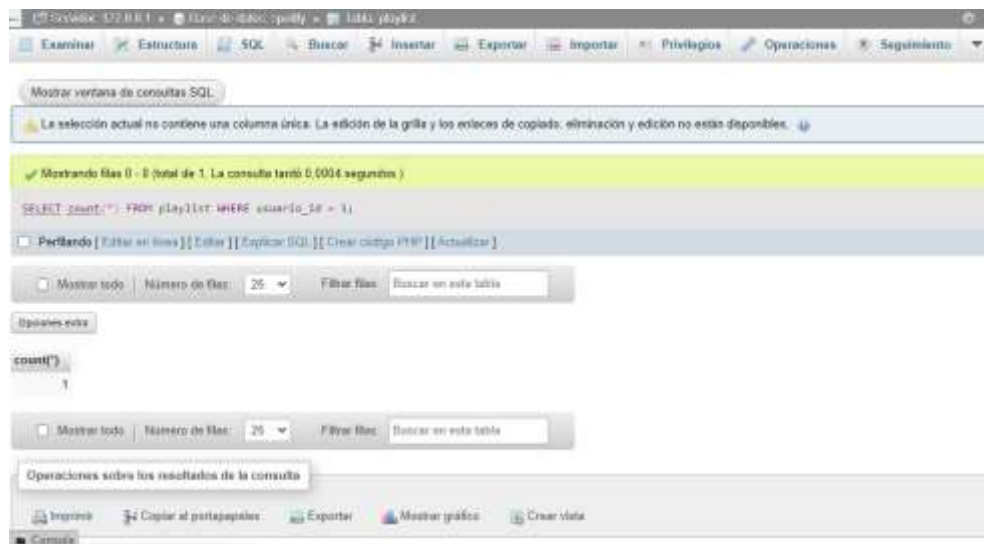
SELECT count(*)FROM canciones **WHERE** artista_id = 1

The screenshot shows the MySQL Workbench interface. The SQL editor contains the query: `SELECT count(*)FROM canciones WHERE artista_id = 1;`. The results pane displays a single row with the count:

	count(*)
<input type="checkbox"/>	2

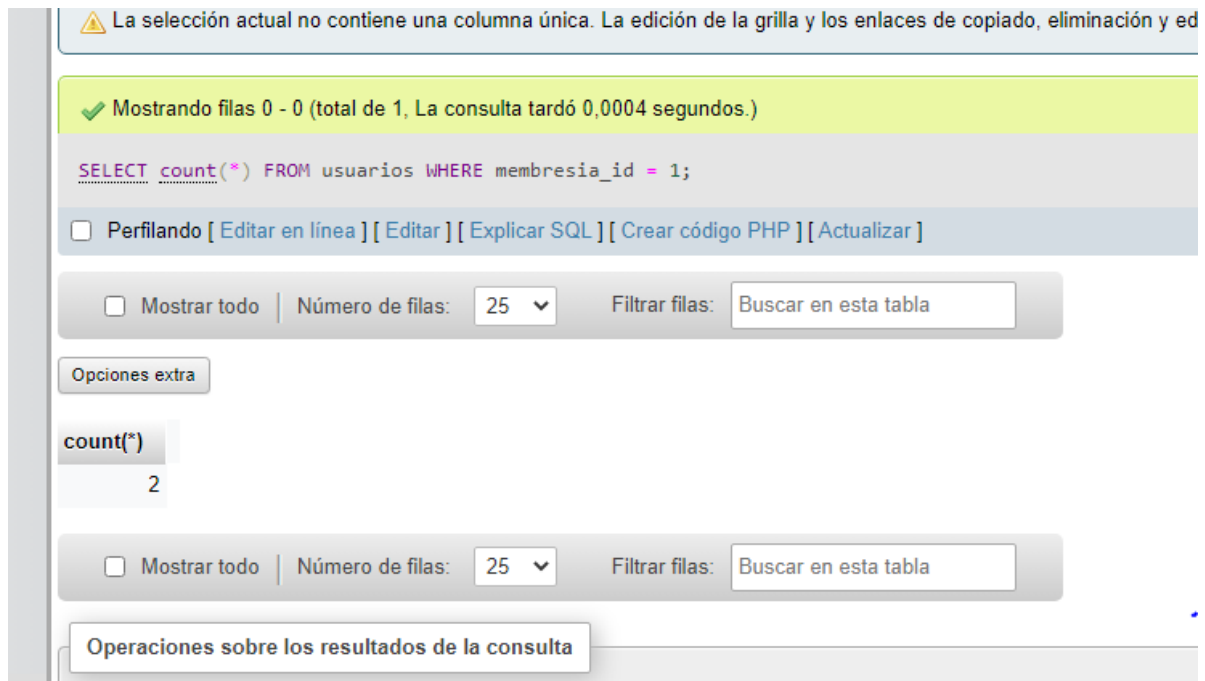
#3 Mostrar el total de playlists del usuario 1

SELECT count(*) **FROM** playlist **WHERE** usuario_id = 1



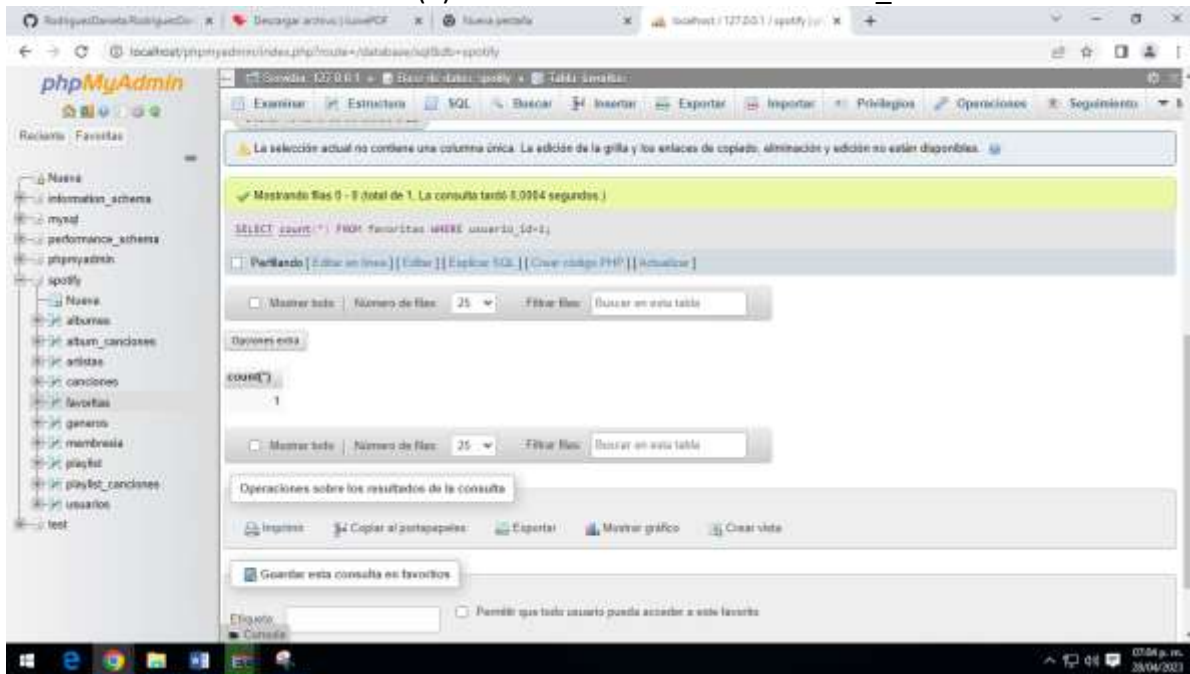
#4 Mostrar el total de usuarios que ha comprado la membresía individual

SELECT count(*) **FROM** usuarios **WHERE** membresía_id = 1



#5 Mostrar el número de canciones que son favoritas del usuario 1

SELECT count(*) **FROM** favoritas **WHERE** usuario_id=1

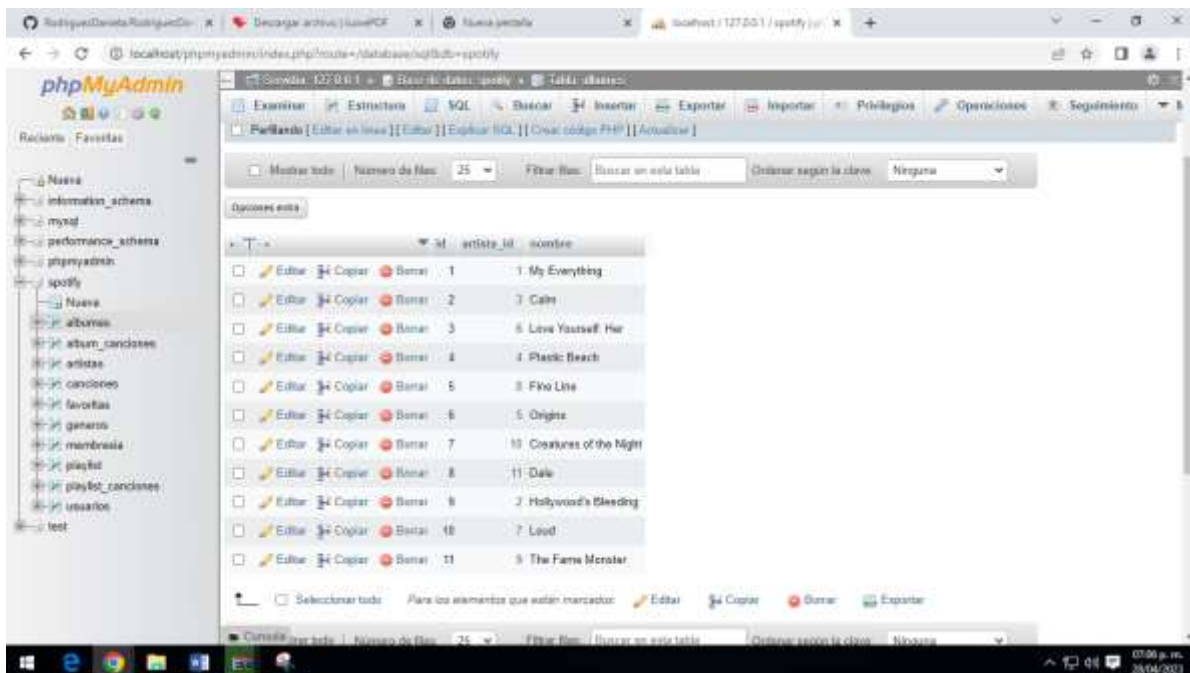


The screenshot shows the phpMyAdmin interface with the 'favoritas' table selected. The SQL query 'SELECT count(*) FROM favoritas WHERE usuario_id=1' is entered in the query box. The result is displayed as a single row with the value '1'.

count(*)
1

#6 Obtener el listado del nombre de todos los albums de la plataforma

SELECT * **FROM** albums



The screenshot shows the phpMyAdmin interface with the 'albums' table selected. The SQL query 'SELECT * FROM albums' is entered in the query box. The result is displayed as a table with 11 rows, showing the album ID, artist ID, and album name.

	id	artista_id	nombre
<input type="checkbox"/>	1	1	My Everything
<input type="checkbox"/>	2	2	Cale
<input type="checkbox"/>	3	3	Love Yourself: Her
<input type="checkbox"/>	4	4	Plastic Beach
<input type="checkbox"/>	5	5	Fixo Line
<input type="checkbox"/>	6	6	Origins
<input type="checkbox"/>	7	10	Creatures of the Night
<input type="checkbox"/>	8	11	Dale
<input type="checkbox"/>	9	2	Hollywood's Bleeding
<input type="checkbox"/>	10	7	Lead
<input type="checkbox"/>	11	9	The Fame Monster

#7 Mostrar el número de canciones con fecha de lanzamiento menor al 2020

SELECT count(*) **FROM** canciones **WHERE** fecha <='2020-12-31'

`SELECT count(*) FROM canciones WHERE fecha <='2020-12-31';`

☐ Perfilando [[Editar en línea](#)] [[Editar](#)] [[Explicar SQL](#)] [[Crear código PHP](#)] [[Actualizar](#)]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

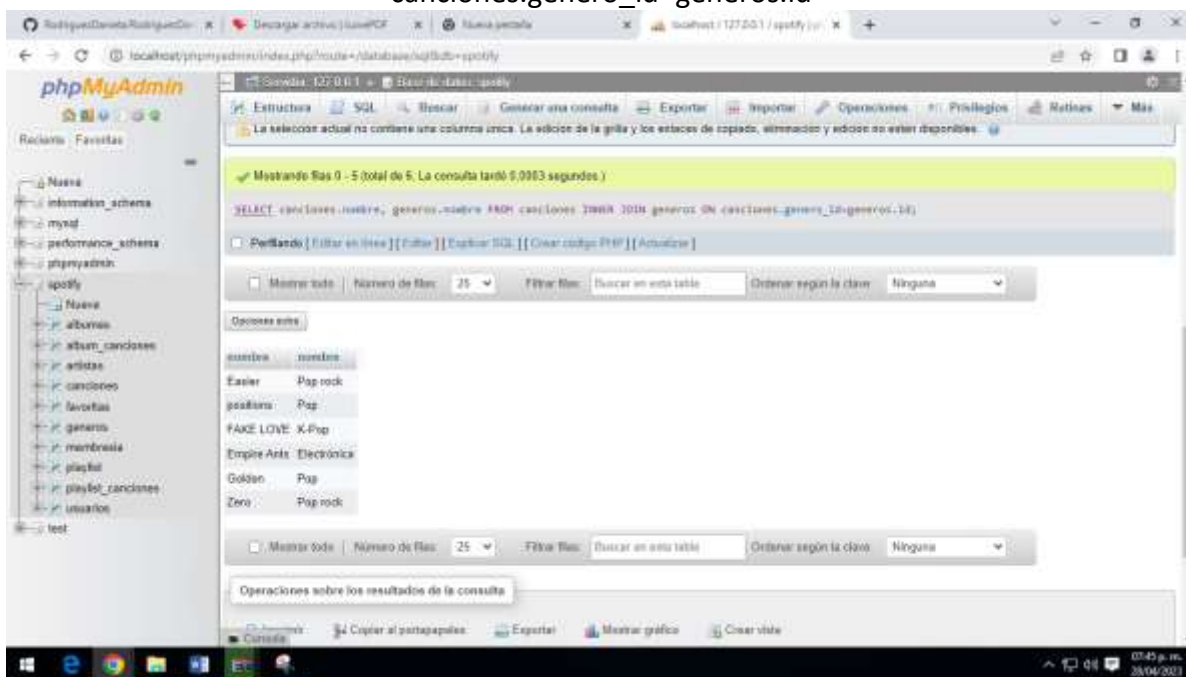
Opciones extra

count(*)
6

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

#8 Obtener el listado de canciones y el nombre de su genero

SELECT canciones.nombre, generos.nombre **FROM** canciones **INNER JOIN** generos **ON** canciones.genero_id=generos.id

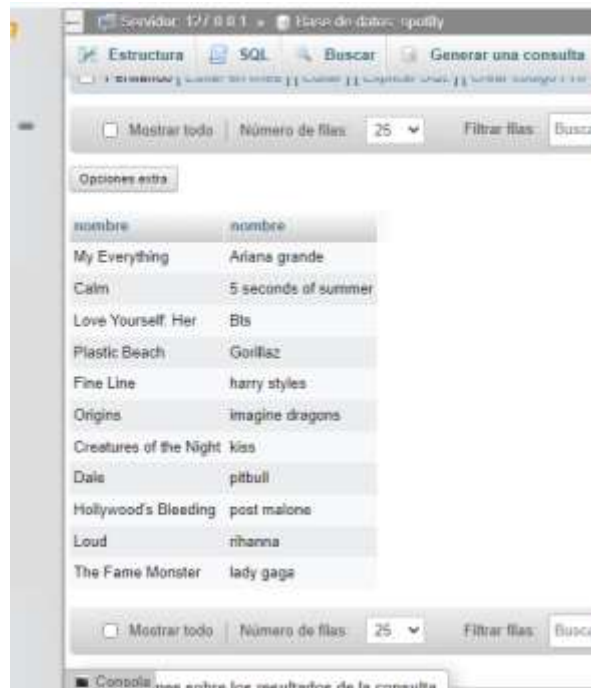


phpMyAdmin interface showing the results of the SQL query. The query is: `SELECT canciones.nombre, generos.nombre FROM canciones INNER JOIN generos ON canciones.genero_id=generos.id`. The results show 5 rows of data.

canciones	nombre
Easy	Pop rock
goldies	Pop
FAKE LOVE	K-Pop
Empire Ants	Electrónica
Golden	Pop
Zero	Pop rock

#9 Obtener listado de álbumes por nombre y el nombre del artista

SELECT albums.nombre, artistas.nombre **FROM** álbumes **INNER JOIN** artistas **ON** albums.artista_id= artistas.id



The screenshot shows a web application interface with a table of albums and artists. The table has two columns: 'nombre' (name) and 'nombre' (name). The data is as follows:

nombre	nombre
My Everything	Ariana grande
Calm	5 seconds of summer
Love Yourself, Her	Bts
Plastic Beach	Gorillaz
Fine Line	harry styles
Origins	imagine dragons
Creatures of the Night	kiss
Dale	pitbull
Hollywood's Bleeding	post malone
Loud	rihanna
The Fame Monster	lady gaga

#10 Mostrar los playlist por nombre y el nombre de los usuarios que lo creo

SELECT playlist.nombre, usuarios.nombre **FROM** playlist **INNER JOIN** usuarios **ON** playlist.usuarios_id=usuarios.id



The screenshot shows a web application interface with a table of playlists and users. The table has two columns: 'nombre' (name) and 'nombre' (name). The data is as follows:

nombre	nombre
baño	Daniela
musiquita	Evelyn
Viaje	Hasly
party	Irma
nuevas	Eduardo

