

## DESARROLLO FULL STACK BÁSICO.

UNIDAD 12

Fundamentos de bases de  
datos.

Nombre: Herlendy Alejandro  
Sánchez Gaitán

## Objetivo

Evaluar los conocimientos de administración de bases de datos, conceptos básicos y comandos SQL.

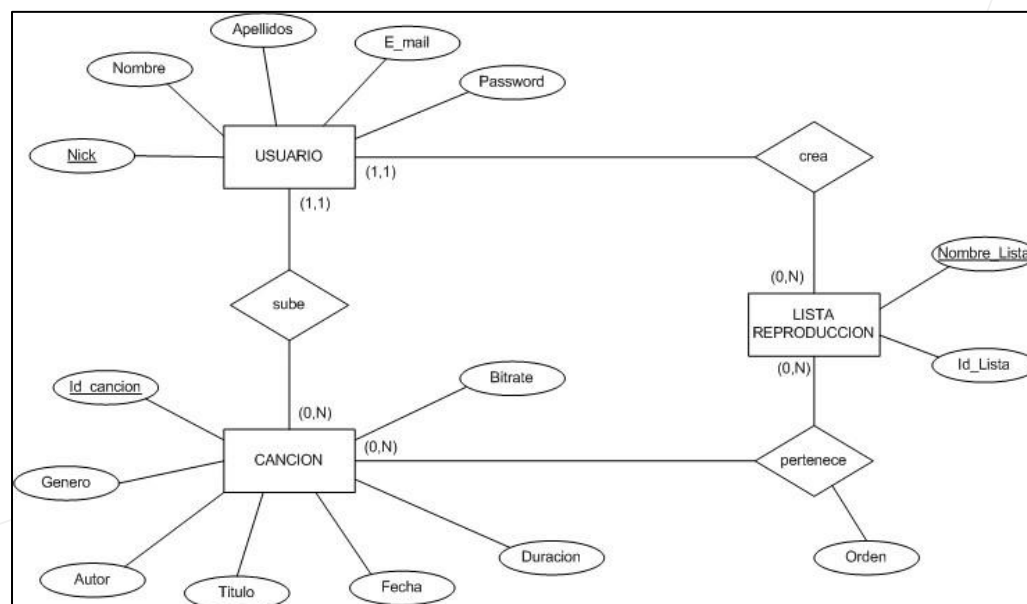
## Temática

Administración de bases de datos, conceptos básicos y comandos SQL.

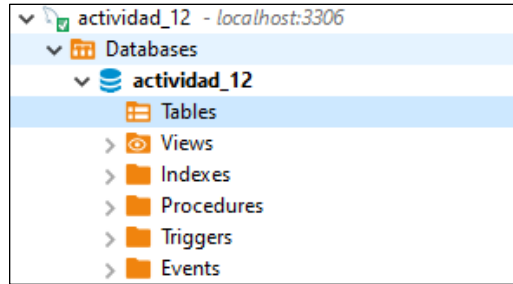
## Descripción de la actividad

Según lo visto en el módulo de fundamentos de bases de datos, responder el siguiente cuestionario de administración de bases de datos, conceptos básicos y comandos SQL.

1. Crear base de datos en phpMyAdmin que corresponda al siguiente modelo entidad relación.

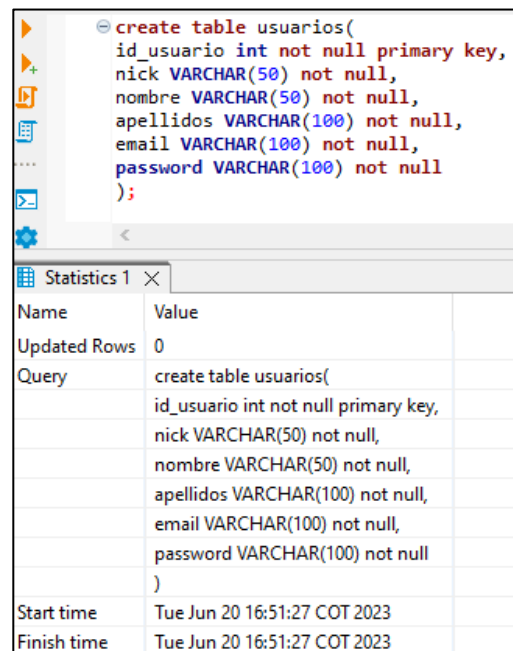


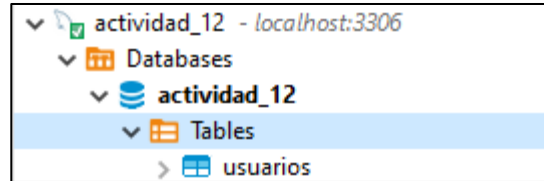
La tabla "muchos" contendrá la columna que actúa como clave foránea, mientras que la tabla "uno" contendrá la clave primaria que será referenciada por la clave foránea en la tabla "muchos".



## Creación tabla Usuarios

```
create table usuarios(
id_usuario int not null primary key,
nick VARCHAR(50) not null,
nombre VARCHAR(50) not null,
apellidos VARCHAR(100) not null,
email VARCHAR(100) not null,
password VARCHAR(100) not null
);
```





## Creación tabla listas\_reproducciones

```
CREATE TABLE listas_reproducciones (

id_lista INT not null primary key,

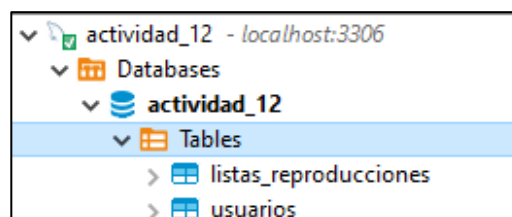
nombre VARCHAR(100) not null,

id_usuario_ INT not null,

foreign key (id_usuario_) references usuarios(id_usuario)

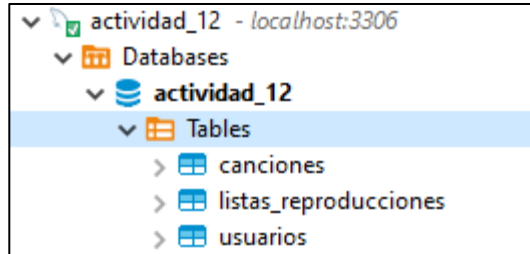
);
```

<pre> /*TABLA LISTAS_REPRODUCCIONES*/ CREATE TABLE listas_reproducciones (   id_lista INT not null primary key,   nombre VARCHAR(100) not null,   id_usuario_ INT not null,   foreign key (id_usuario_) references usuarios(id_usuario) );         </pre>	
Statistics 1	
Name	Value
Updated Rows	0
Query	CREATE TABLE listas_reproducciones (             id_lista INT not null primary key,             nombre VARCHAR(100) not null,             id_usuario_ INT not null,             foreign key (id_usuario_) references usuarios(id_usuario)           )
Start time	Tue Jun 20 17:24:12 COT 2023
Finish time	Tue Jun 20 17:24:12 COT 2023



```
CREATE TABLE canciones (  
  id_cancion INT not null primary key,  
  genero VARCHAR(50) not null,  
  autor VARCHAR(100) not null,  
  titulo VARCHAR(100) not null,  
  fecha DATE not null,  
  duracion DECIMAL(8,2) not null,  
  bitrate DECIMAL(8,2) not null,  
  id_lista_ INT not null,  
  foreign key (id_lista_) references listas_reproducciones (id_lista_)  
);
```

Name	Value
Updated Rows	0
Query	CREATE TABLE canciones ( id_cancion INT not null primary key, genero VARCHAR(50) not null, autor VARCHAR(100) not null, titulo VARCHAR(100) not null, fecha DATE not null, duracion DECIMAL(8,2) not null, bitrate DECIMAL(8,2) not null, id_lista_ INT not null, foreign key (id_lista_) references listas_reproducciones (id_lista_) )
Start time	Tue Jun 20 17:41:40 COT 2023
Finish time	Tue Jun 20 17:41:40 COT 2023



Resultado final



2. Realice 5 ejemplos de sentencias INSERT a cualquier tabla, en donde se evidencien los cambios.

## Tabla usuarios

```
insert into usuarios (id_usuario,nick,nombre,apellidos,email,password)
```

```
values (1,"Pepe","Felipe","Gonzalez","pepe@gmail.com","P3P3@113");
```

```
insert into usuarios (id_usuario,nick,nombre,apellidos,email,password)
```

```
values (2,"Lu","Luisa","Morales","lu@gmail.com","LU_896#");
```

```
insert into usuarios (id_usuario,nick,nombre,apellidos,email,password)
```

```
values (3,"Lau","Laura","Linares","Lau@gmail.com","Lau$2031&");
```

```
insert into usuarios (id_usuario,nick,nombre,apellidos,email,password)
```

```
values (4,"Drako","Alejandro","Sanchez","alejo@gmail.com","Drak$%2983@");
```

```
insert into usuarios (id_usuario,nick,nombre,apellidos,email,password)
```

```
values (5,"Agus","Agustin","Bueno","agus@gmail.com","Agu$2021_123");
```

<pre>insert into usuarios (id_usuario,nick,nombre,apellidos,email,password) values (1,"Pepe","Felipe","Gonzalez","pepe@gmail.com","P3P3@113");</pre>	
Statistics 1	
Name	Value
Updated Rows	1
Query	insert into usuarios (id_usuario,nick,nombre,apellidos,email,password) values (1,"Pepe","Felipe","Gonzalez","pepe@gmail.com","P3P3@113")
Start time	Tue Jun 20 17:09:26 COT 2023
Finish time	Tue Jun 20 17:09:26 COT 2023

Se realiza el mismo proceso con las otras 4 sentencias INSERT

```
insert into usuarios (id_usuario,nick,nombre,apellidos,email,password)
values (2,"Lu","Luisa","Morales","lu@gmail.com","LU_896#");

insert into usuarios (id_usuario,nick,nombre,apellidos,email,password)
values (3,"Lau","Laura","Linares","Lau@gmail.com","Lau$2031&");

insert into usuarios (id_usuario,nick,nombre,apellidos,email,password)
values (4,"Drako","Alejandro","Sanchez","alejo@gmail.com","Drak$%2983@");

insert into usuarios (id_usuario,nick,nombre,apellidos,email,password)
values (5,"Agus","Agustin","Bueno","agus@gmail.com","Agu$2021_123");
```

	id_usuario	nick	nombre	apellidos	email	password
1	1	Pepe	Felipe	Gonzalez	pepe@gmail.com	P3P3@113
2	2	Lu	Luisa	Morales	lu@gmail.com	LU_896#
3	3	Lau	Laura	Linares	Lau@gmail.com	Lau\$2031&
4	4	Drako	Alejandro	Sanchez	alejo@gmail.com	Drak\$%2983@
5	5	Agus	Agustin	Bueno	agus@gmail.com	Agu\$2021_123



## Tabla listas\_reproducciones

```
insert into listas_reproducciones (id_lista,nombre,id_usuario_)
values (1,"Música Clásica",1);

insert into listas_reproducciones (id_lista,nombre,id_usuario_)
values (2,"Salsa Romántica",1);

insert into listas_reproducciones (id_lista,nombre,id_usuario_)
values (3,"Musica Pop en inglés",3);

insert into listas_reproducciones (id_lista,nombre,id_usuario_)
values (4,"Reggaeton Viejo",2);

insert into listas_reproducciones (id_lista,nombre,id_usuario_)
values (5,"Electronica",4);

insert into listas_reproducciones (id_lista,nombre,id_usuario_)
values (6,"Música Cristiana",4);
```

/*EJEMPLOS INSERT TABLA LISTAS_REPRODUCCIONES*/	
<pre>insert into listas_reproducciones (id_lista,nombre,id_usuario_) values (1,"Música Clásica",1);</pre>	
Statistics 1 X	
Name	Value
Updated Rows	1
Query	insert into listas_reproducciones (id_lista,nombre,id_usuario_) values (1,"Música Clásica",1)
Start time	Tue Jun 20 17:25:18 COT 2023
Finish time	Tue Jun 20 17:25:18 COT 2023

Se realiza el mismo proceso con las otras 5 sentencias INSERT



SQL Editor interface showing a list of SQL insert statements for the `listas_reproducciones` table:

```

insert into listas_reproducciones (id_lista,nombre,id_usuario_)
values (2,"Salsa Romántica",1);

insert into listas_reproducciones (id_lista,nombre,id_usuario_)
values (3,"Musica Pop en inglés",3);

insert into listas_reproducciones (id_lista,nombre,id_usuario_)
values (4,"Reggaeton Viejo",2);

insert into listas_reproducciones (id_lista,nombre,id_usuario_)
values (5,"Electronica",4);

insert into listas_reproducciones (id_lista,nombre,id_usuario_)
values (6,"Música Cristiana",4);
  
```

Below the editor, a "Statistics 1" window displays the execution details for the last query:

Name	Value
Updated Rows	1
Query	insert into listas_reproducciones (id_lista,nombre,id_usuario_) values (6,"Música Cristiana",4)
Start time	Tue Jun 20 17:27:06 COT 2023
Finish time	Tue Jun 20 17:27:06 COT 2023

	123 id_lista	ABC nombre	123 id_usuario_
Grid	1	Música Clásica	1
	2	Salsa Romántica	1
	3	Musica Pop en inglés	3
	4	Reggaeton Viejo	2
	5	Electronica	4
	6	Música Cristiana	4

## Tabla canciones

```

insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_)
values (1, 'musica clásica', 'Ludwig van Beethoven', 'Sinfonía No. 5 en Do Menor',
STR_TO_DATE('1808-04-01', '%Y-%m-%d'), 30.0, 320.0, 1);
  
```

```

insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_)
values (2, 'Pop', 'Taylor Swift', 'Shake It Off', STR_TO_DATE('2014-08-18', '%Y-%m-%d'), 3.39, 320,
3);
  
```

```
insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_)
values (3, 'Reguetón', 'Daddy Yankee', 'Gasolina', STR_TO_DATE('2004-07-13', '%Y-%m-%d'), 3.49,
256, 4);

insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_)
values (4, 'Electrónica', 'Daft Punk', 'Around the World', STR_TO_DATE('1997-03-17', '%Y-%m-%d'),
7.08, 320, 5);

insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_)
values (5, 'Música Cristiana', 'Hillsong Worship', 'What a Beautiful Name', STR_TO_DATE('2016-10-
14', '%Y-%m-%d'), 5.43, 192, 6);
```

<pre>insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_) values (1, 'musica clásica', 'Ludwig van Beethoven', 'Sinfonía No. 5 en Do Menor', STR_TO_DATE('1808-04-01', '%Y-%m-%d'), 30.0, 320.0, 1);</pre>	
Statistics 1	
Name	Value
Updated Rows	1
Query	insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_) values (1, 'musica clásica', 'Ludwig van Beethoven', 'Sinfonía No. 5 en Do Menor', STR_TO_DATE('1808-04-01', '%Y-%m-%d'), 30.0, 320.0, 1)
Start time	Tue Jun 20 17:43:29 COT 2023
Finish time	Tue Jun 20 17:43:29 COT 2023

Se realiza el mismo proceso con las otras 4 sentencias INSERT

<pre>/*EJEMPLOS INSERT TABLA CANCIONES*/  insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_) values (1, 'musica clásica', 'Ludwig van Beethoven', 'Sinfonía No. 5 en Do Menor', STR_TO_DATE('1808-04-01', '%Y-%m-%d'), 30.0, 320.0, 1);  insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_) values (2, 'Pop', 'Taylor Swift', 'Shake It Off', STR_TO_DATE('2014-08-18', '%Y-%m-%d'), 3.39, 320, 3);  insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_) values (3, 'Reguetón', 'Daddy Yankee', 'Gasolina', STR_TO_DATE('2004-07-13', '%Y-%m-%d'), 3.49, 256, 4);  insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_) values (4, 'Electrónica', 'Daft Punk', 'Around the World', STR_TO_DATE('1997-03-17', '%Y-%m-%d'), 7.08, 320, 5);  insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_) values (5, 'Música Cristiana', 'Hillsong Worship', 'What a Beautiful Name', STR_TO_DATE('2016-10-14', '%Y-%m-%d'), 5.43, 192, 6);</pre>	
Statistics 1	
Name	Value
Updated Rows	1
Query	insert into canciones (id_cancion, genero, autor, titulo, fecha, duracion, bitrate, id_lista_) values (5, 'Música Cristiana', 'Hillsong Worship', 'What a Beautiful Name', STR_TO_DATE('2016-10-14', '%Y-%m-%d'), 5.43, 192, 6)
Start time	Tue Jun 20 17:54:22 COT 2023
Finish time	Tue Jun 20 17:54:22 COT 2023

	123 id_cancion	ABC genero	ABC autor	ABC titulo	fecha	123 duracion	123 bitrate	123 id_lista_
1	1	musica clásica	Ludwig van Beethoven	Sinfonía No. 5 en Do Menor	1808-04-01	30	320	1
2	2	Pop	Taylor Swift	Shake It Off	2014-08-18	3.39	320	3
3	3	Reguetón	Daddy Yankee	Gasolina	2004-07-13	3.49	256	4
4	4	Electrónica	Daft Punk	Around the World	1997-03-17	7.08	320	5
5	5	Música Cristiana	Hillsong Worship	What a Beautiful Name	2016-10-14	5.43	192	6

- Realice 5 ejemplos de sentencias UPDATE a cualquier tabla, en donde se evidencien los cambios.

## Tabla usuarios

usuarios						
	123 id_usuario	ABC nick	ABC nombre	ABC apellidos	ABC email	ABC password
1	1	Pepe	Felipe	Gonzalez	pepe@gmail.com	P3P3@113
2	2	Lu	Luisa	Morales	lu@gmail.com	LU_896#
3	3	Lau	Laura	Linares	Lau@gmail.com	Lau\$2031&
4	4	Drako	Alejandro	Sanchez	alejo@gmail.com	Drak\$%2983@
5	5	Agus	Agustin	Bueno	agus@gmail.com	Agu\$2021_123

### 1) update usuarios

set apellidos = "Duarte"

where id\_usuario = 5

/*EJEMPLOS UPDATE TABLA USUARIOS*/	
update usuarios	
set apellidos = "Duarte"	
where id_usuario = 5	
Statistics 1	
Name	Value
Updated Rows	1
Query	update usuarios set apellidos = "Duarte" where id_usuario = 5
Start time	Tue Jun 20 17:58:20 COT 2023
Finish time	Tue Jun 20 17:58:20 COT 2023

usuarios   Enter a SQL expression to filter results (use Ctrl+Space)							
	id_usuario	nick	nombre	apellidos	email	password	
1	1	Pepe	Felipe	Gonzalez	pepe@gmail.com	P3P3@113	
2	2	Lu	Luisa	Morales	lu@gmail.com	LU_896#	
3	3	Lau	Laura	Linares	Lau@gmail.com	Lau\$2031&	
4	4	Drako	Alejandro	Sanchez	alejo@gmail.com	Drak\$%2983@	
5	5	Agus	Agustin	Duarte	agus@gmail.com	Agu\$2021_123	

2) **update** usuarios

**set** password = "TFVfODk2Iw=="

**where** nick = "Lu"

usuarios   Enter a SQL expression to filter results (use Ctrl+Space)							
	id_usuario	nick	nombre	apellidos	email	password	
1	1	Pepe	Felipe	Gonzalez	pepe@gmail.com	P3P3@113	
2	2	Lu	Luisa	Morales	lu@gmail.com	LU_896#	
3	3	Lau	Laura	Linares	Lau@gmail.com	Lau\$2031&	
4	4	Drako	Alejandro	Sanchez	alejo@gmail.com	Drak\$%2983@	
5	5	Agus	Agustin	Duarte	agus@gmail.com	Agu\$2021_123	

Suponiendo que la contraseña en muy visible ahora se codifica en base64

<pre>update usuarios set password = "TFVfODk2Iw==" where nick = "Lu"</pre>	
Statistics 1	
Name	Value
Updated Rows	1
Query	update usuarios set password = "TFVfODk2Iw==" where nick = "Lu"
Start time	Tue Jun 20 18:01:56 COT 2023
Finish time	Tue Jun 20 18:01:56 COT 2023

Properties Data ER Diagram							
usuarios Enter a SQL expression to filter results (use Ctrl+Space)							
	id_usuario	nick	nombre	apellidos	email	password	
1	1	Pepe	Felipe	Gonzalez	pepe@gmail.com	P3P3@113	
2	2	Lu	Luisa	Morales	lu@gmail.com	TFvFODk2lw==	
3	3	Lau	Laura	Linares	Lau@gmail.com	Lau\$2031&	
4	4	Drako	Alejandro	Sanchez	alejo@gmail.com	Drak\$%2983@	
5	5	Agus	Agustin	Duarte	agus@gmail.com	Agu\$2021_123	

### 3) update listas\_reproducciones

set nombre = "Música Electronica!!!"

where id\_lista = 5

<pre>update listas_reproducciones set nombre = "Música Electronica!!!" where id_lista = 5</pre>	
Statistics 1	
Name	Value
Updated Rows	1
Query	update listas_reproducciones set nombre = "Música Electronica!!!" where id_lista = 5
Start time	Tue Jun 20 18:05:16 COT 2023
Finish time	Tue Jun 20 18:05:16 COT 2023

	id_lista	nombre	id_usuario_
1	1	Música Clásica	1
2	2	Salsa Romántica	1
3	3	Musica Pop en inglés	3
4	4	Reggaeton Viejo	2
5	5	Música Electronica!!!	4
6	6	Música Cristiana	4

### 4) update listas\_reproducciones

set nombre = "Gospel Music <3"

where nombre = "Música Cristiana"

<pre>update listas_reproducciones set nombre = "Gospel Music &lt;3" where nombre = "Música Cristiana"</pre>	
Statistics 1	
Name	Value
Updated Rows	1
Query	update listas_reproducciones set nombre = "Gospel Music <3" where nombre = "Música Cristiana"
Start time	Tue Jun 20 18:07:33 COT 2023
Finish time	Tue Jun 20 18:07:33 COT 2023

	id_lista	nombre	id_usuario_
1	1	Música Clásica	1
2	2	Salsa Romántica	1
3	3	Musica Pop en inglés	3
4	4	Reggaeton Viejo	2
5	5	Música Electronica!!!	4
6	6	Gospel Music <3	4

```
set autor = 'Wolfgang Amadeus Mozart',
titulo = 'Sinfonía No. 40 en Sol menor',
fecha = STR_TO_DATE('1788-07-25', '%Y-%m-%d'),
duracion = 25.16,
bitrate = 320
where id_cancion = 1
```

canciones <small>Enter a SQL expression to filter results (use Ctrl+Space)</small>								
	123 id_cancion	ABC genero	ABC autor	ABC titulo	fecha	123 duracion	123 bitrate	123 id_lista
1	1	musica clásica	Ludwig van Beethoven	Sinfonía No. 5 en Do Menor	1808-04-01	30	320	1
2	2	Pop	Taylor Swift	Shake It Off	2014-08-18	3.39	320	3
3	3	Reguetón	Daddy Yankee	Gasolina	2004-07-13	3.49	256	4
4	4	Electrónica	Daft Punk	Around the World	1997-03-17	7.08	320	5
5	5	Música Cristiana	Hillsong Worship	What a Beautiful Name	2016-10-14	5.43	192	6

The screenshot displays the SQL editor and the Statistics window in DBeaver. The SQL editor contains the following SQL statement:

```

-- update canciones
set autor = 'Wolfgang Amadeus Mozart',
titulo = 'Sinfonía No. 40 en Sol menor',
fecha = STR_TO_DATE('1788-07-25', '%Y-%m-%d'),
duracion = 25.16,
bitrate = 320
where id_cancion = 1

```

The Statistics window shows the execution details for the query:

Name	Value
Updated Rows	1
Query	<pre> update canciones set autor = 'Wolfgang Amadeus Mozart', titulo = 'Sinfonía No. 40 en Sol menor', fecha = STR_TO_DATE('1788-07-25', '%Y-%m-%d'), duracion = 25.16, bitrate = 320 where id_cancion = 1 </pre>
Start time	Tue Jun 20 18:12:08 COT 2023
Finish time	Tue Jun 20 18:12:08 COT 2023

canciones   Enter a SQL expression to filter results (use Ctrl+Space)									
	123 id_cancion	ABC genero	ABC autor	ABC titulo	🕒 fecha	123 duracion	123 bitrate	123 id_lista	
1	1	musica clásica	Wolfgang Amadeus Mozart	Sinfonía No. 40 en Sol menor	1788-07-25	25.16	320	1	
2	2	Pop	Taylor Swift	Shake It Off	2014-08-18	3.39	320	3	
3	3	Reguetón	Daddy Yankee	Gasolina	2004-07-13	3.49	256	4	
4	4	Electrónica	Daft Punk	Around the World	1997-03-17	7.08	320	5	
5	5	Música Cristiana	Hillsong Worship	What a Beautiful Name	2016-10-14	5.43	192	6	

4. Realice 5 ejemplos de consultas que contengan condicionales tipo WHERE usando:
- INNER JOIN
  - COUNT
  - ORDER BY
  - BETWEEN

## Tabla usuarios inner join

Muestras los usuarios con sus respectivas listas de reproducciones

```
select *
from usuarios as u
inner join listas_reproducciones as ls on u.id_usuario = ls.id_usuario_
where ls.id_usuario_ is not null
```

SQL Editor

```
/*CONSULTA INNER JOIN*/
select *
from usuarios as u
inner join listas_reproducciones as ls on u.id_usuario = ls.id_usuario_
where ls.id_usuario_ is not null
```

usuarios(+) 1 X

select \* from usuarios as u inner join listas\_reproducciones as ls on u.id\_usuario = ls.id\_usuario\_ where ls.id\_usuario\_ is not null

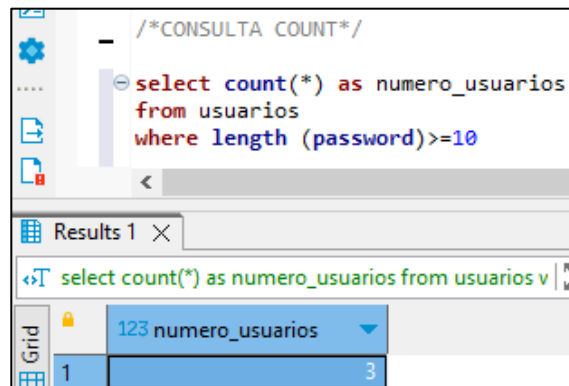
	id_usuario	nick	nombre	apellidos	email	password	id_lista	nombre	id_usuario_
1	1	Pepe	Felipe	Gonzalez	pepe@gmail.com	P3P3@113	1	Música Clásica	1
2	1	Pepe	Felipe	Gonzalez	pepe@gmail.com	P3P3@113	2	Salsa Romántica	1
3	3	Lau	Laura	Linares	Lau@gmail.com	Lau\$2031&	3	Musica Pop en ingl	3
4	2	Lu	Luisa	Morales	lu@gmail.com	TFVfODk2lw==	4	Reggaeton Viejo	2
5	4	Drako	Alejandro	Sanchez	alejo@gmail.com	Drak\$%2983@	5	Música Electronica	4
6	4	Drako	Alejandro	Sanchez	alejo@gmail.com	Drak\$%2983@	6	Gospel Music <3	4

## Tabla usuarios COUNT

Cuenta la cantidad de usuarios cuya contraseña tiene longitud mayor o igual a 10, por seguridad.

```
select count(*) as numero_usuarios
from usuarios
where length (password)>=10
```





## Tabla canciones ORDER BY

Ordena las fechas de manera descendente, es decir, muestra primero las más recientes después del año 2000.

```

select *
from canciones
where fecha >='2000-01-01'
order by fecha desc;
  
```

select \*  
from canciones  
where fecha >='2000-01-01'  
order by fecha desc;

canciones 1 X

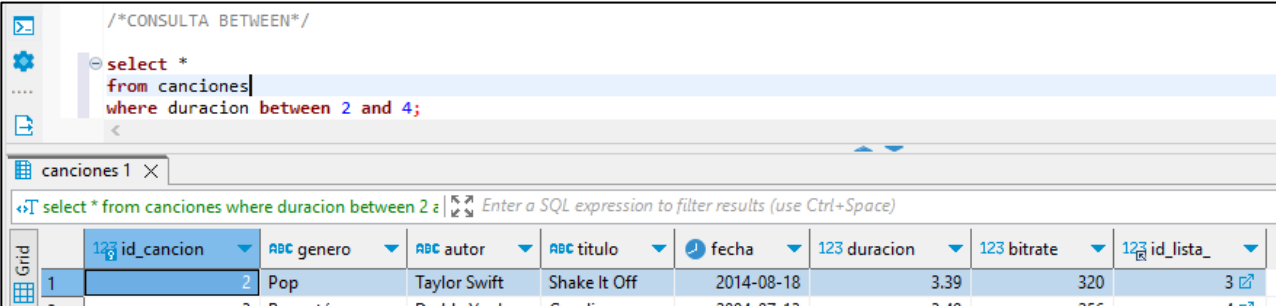
select \* from canciones where fecha >='2000-01-01' Enter a SQL expression to filter results (use Ctrl+Space)

Grid	123 id_cancion	ABC genero	ABC autor	ABC titulo	fecha	123 duracion	123 bitrate	123 id_lista_
1	5	Música Cristiana	Hillsong Worship	What a Beautiful Name	2016-10-14	5.43	192	6
2	2	Pop	Taylor Swift	Shake It Off	2014-08-18	3.39	320	3
3	3	Reguetón	Daddy Yankee	Gasolina	2004-07-13	3.49	256	4

## Tabla listas\_reproducciones BETWEEN

```
select *
from canciones
where duracion between 2 and 4;
```

Selecciona las canciones cuya duración esta entre 2 y 4 minutos



The screenshot shows a database interface with a SQL query editor at the top and a results grid below. The query is: `select * from canciones where duracion between 2 and 4;`. The results grid displays two rows of data from the 'canciones' table.

	id_cancion	genero	autor	titulo	fecha	duracion	bitrate	id_lista_
1	2	Pop	Taylor Swift	Shake It Off	2014-08-18	3.39	320	3
2	3	Reguetón	Daddy Yankee	Gasolina	2004-07-13	3.49	256	4

Github:

Documento:

[https://github.com/AlejandroSanchez01/full\\_stack\\_ejercicios/blob/main/Actividad\\_12/herlendy\\_alejandro\\_sanchez\\_gaitan\\_Actividad\\_Unidad\\_12.pdf](https://github.com/AlejandroSanchez01/full_stack_ejercicios/blob/main/Actividad_12/herlendy_alejandro_sanchez_gaitan_Actividad_Unidad_12.pdf)

Script:

[https://github.com/AlejandroSanchez01/full\\_stack\\_ejercicios/blob/main/Actividad\\_12/Script.txt](https://github.com/AlejandroSanchez01/full_stack_ejercicios/blob/main/Actividad_12/Script.txt)

**Material requerido/ recursos de apoyo**

<https://www.apachefriends.org/es/index.html>

<https://parzibyte.me/blog/2020/09/28/generar-diagrama-base-datos-phpmyadmin/>