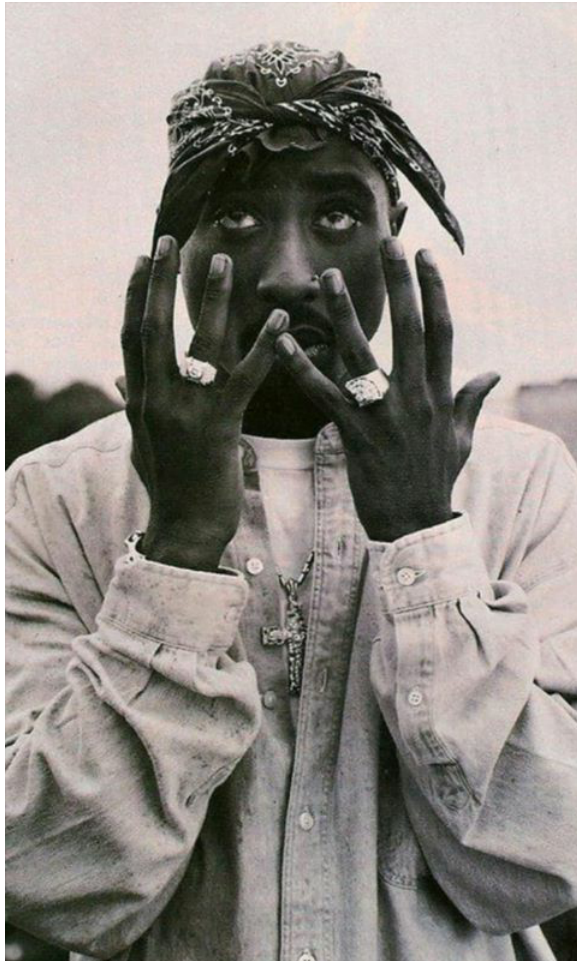


Proyecto de base de datos

HISTORIA DEL HIP HOP



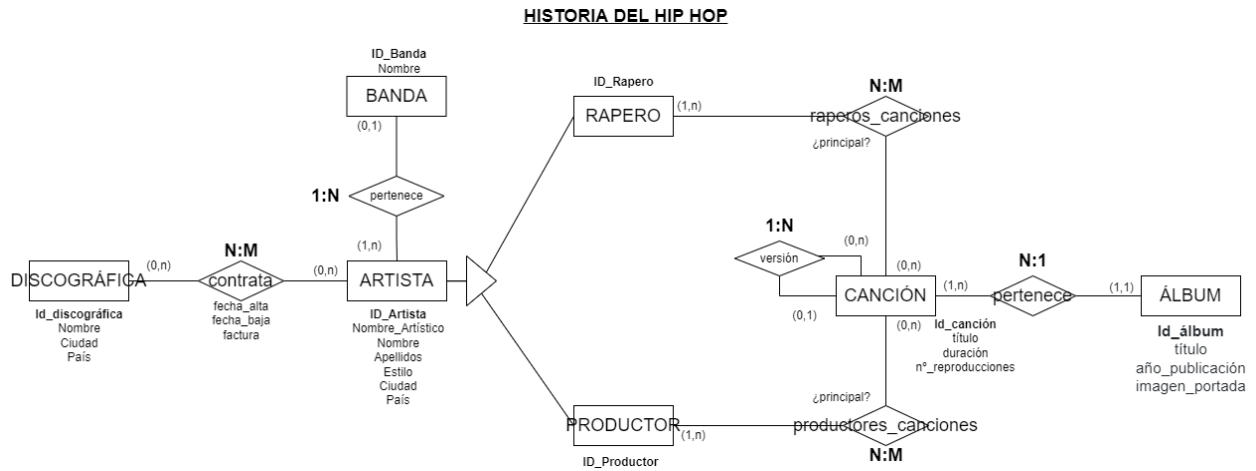
DESCRIPCIÓN

Este proyecto trata sobre las distintas acciones que han hecho los artistas del Hip Hop (raperos y productores), como las canciones que han grabado, los álbumes que han realizado, la discográfica a la que pertenece, su contrato con ella y la banda a la que pertenecen.

ÍNDICE

- I. [Diagrama MER extendido](#)
- II. [Modelo relacional](#)
- III. [Esquema Workbench](#)
- IV. [Carga masiva de datos](#)
- V. [Consultas multitable](#)
- VI. [Vistas](#)
- VII. [Funciones y procedimientos](#)
- VIII. [Triggers](#)
- IX. [Valoración personal](#)

I. DIAGRAMA MER EXTENDIDO



II. MODELO RELACIONAL

Discográfica: (Id_Discográfica PK, Nombre, Ciudad, País)

contrata: (Id_Discográfica PK FK, ID_Artista PK FK, fecha_alta, fecha_baja, factura)

Artista: (Id_Artista PK, ID_Banda FK, Nombre_Artístico, Nombre, Apellidos, Estilo, Ciudad, País)

Banda: (ID_Banda PK, Nombre)

Rapero: (Id_Rapero PK, Id_Artista FK)

Productor: (Id_Productor PK, Id_Artista FK)

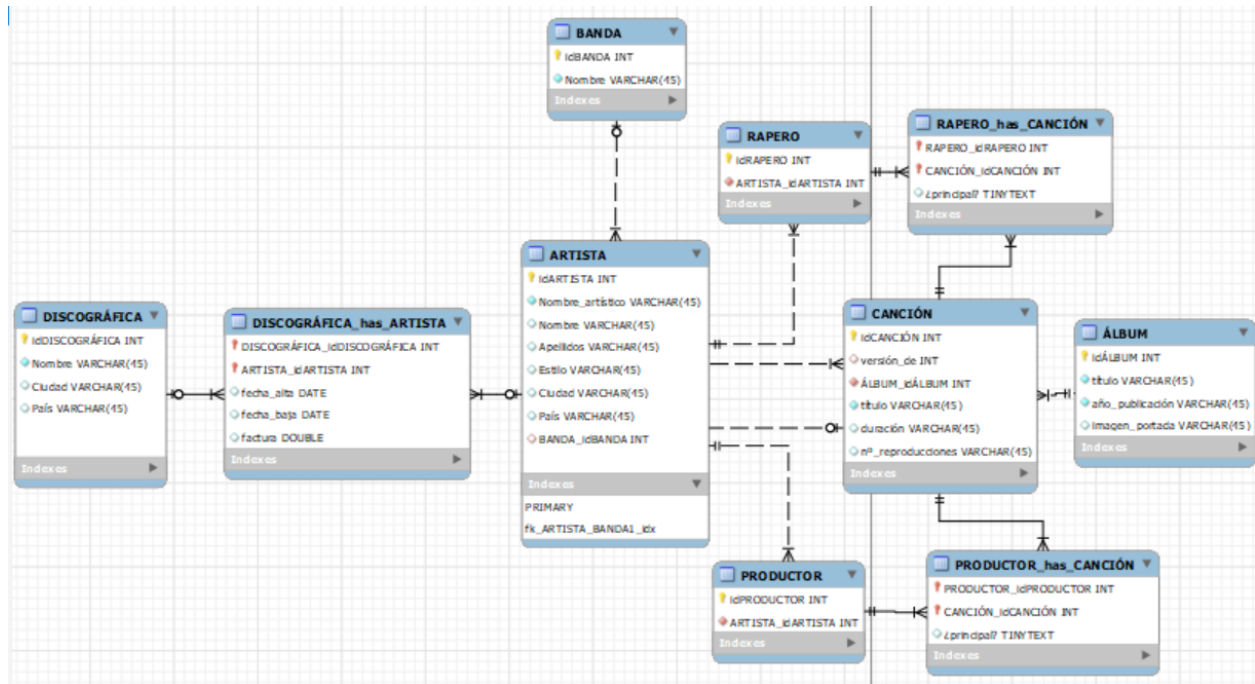
raperos_canciones: (Id_Rapero PK FK, Id_canción PK FK, ¿principal?)

productores_canciones: (Id_Productores PK FK, Id_canción PK FK, ¿principal?)

Canción: (Id_canción PK, versión_de FK, Id_álbum FK, Título, Duración, n°_reproducciones)

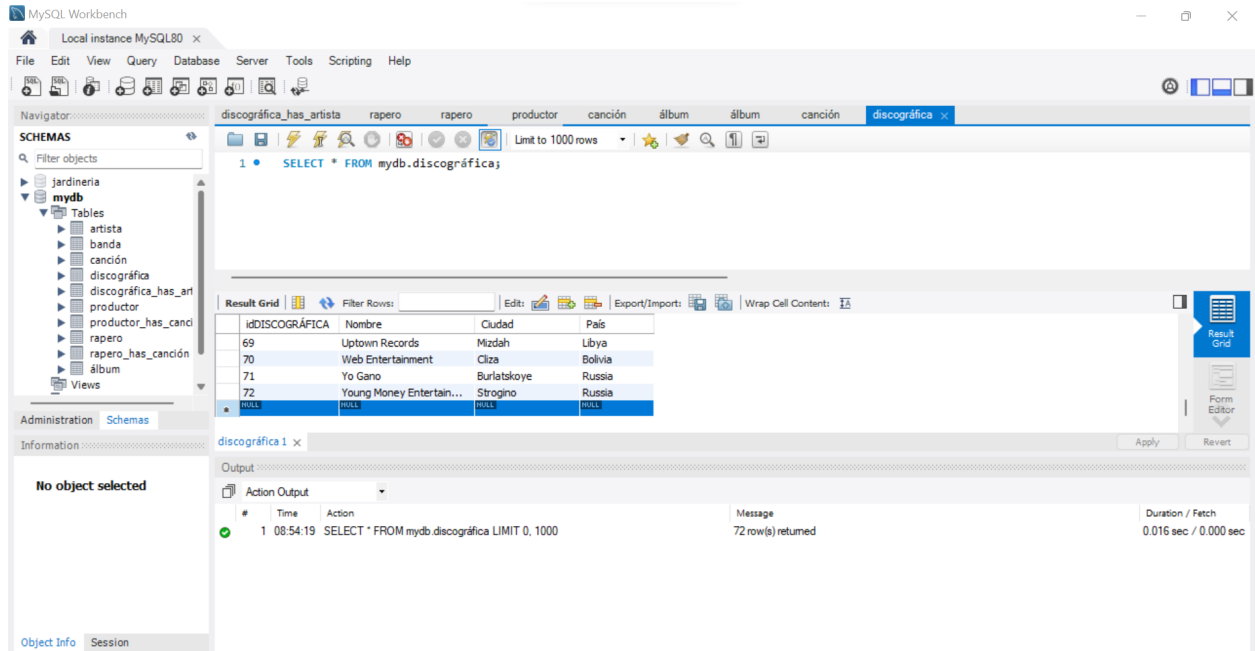
Álbum: (Id_álbum, título, año_publicación, imagen_portada)

III. ESQUEMA WORKBENCH



IV. CARGA MASIVA DE DATOS

Para realizar la carga masiva de datos he utilizado la web Mockaroo para generar datos aleatorios y la web Kaggle para tener la lista de artistas. Los genera en formato CSV, por lo tanto; se pueden copiar en un Excel. Para introducirlos en mi base de datos he utilizado MySQL Workbench. En el apartado MySQL Connections he seleccionado Local instance MySQL 80, ya que ese es mi servidor de bases de datos. He localizado la mía, he ido introduciendo datos tabla por tabla, copiando mis datos guardados anteriormente en Excel y pegándolos en cada una de las tablas con click derecho y "Paste Row", aplico los cambios y se me actualizan en mi base de datos.



V. CONSULTAS MULTITABLA

1. Devuelve los álbumes que no tengan ninguna canción asociada.

select á.*

from álbum á left join canción c

on á.idÁLBUM = c.ÁLBUM_idÁLBUM

where c.ÁLBUM_idÁLBUM is null;

	idÁLBUM	título	año_publicación	imagen_portada
1	6	Rolling Papers	2011	[NULL]
2	7	17	2017	[NULL]
3	8	Shoot For The Stars Aim For The Moon	2020	[NULL]

2. Devuelve los artistas que tienen una canción llamada KK y no son los artistas principales de esa canción.

select a.*

from artista a inner join rapero r on a.idARTISTA = r.ARTISTA_idARTISTA

inner join rapero_has_canción rhc on r.idRAPERO = rhc.RAPERO_idRAPERO

inner join canción c on rhc.CANCIÓN_idCANCIÓN = c.idCANCIÓN

where c.título = 'KK' and rhc.`¿principal?` = 'false';

	idARTISTA	Nombre_artístico	Nombre	Apellidos	Estilo	Ciudad	País	BANDA_idBANDA
1	35	Project Pat	Sara-ann	Dunton	Crunk	Los Angeles	China	2
2	57	Juicy J	Valenka	Astall	Pop Rap	Fresno	Syria	1

3. Devuelve los datos de los artistas que tienen una factura mayor o igual que la media.

select a.*

from artista a inner join discográfica_has_artista dha

on a.idARTISTA = dha.ARTISTA_idARTISTA

where dha.factura >=(select avg(dha2.factura) from discográfica_has_artista dha2);

	idARTISTA	Nombre_artístico	Nombre	Apellidos	Estilo	Ciudad	País	BANDA_idBANDA
1	34	Grandmaster Flash & The Furious Five	Rivalee	Quibell	Old school	Fort Smith	China	[NULL]
2	49	EPMD	Charis	Pagelsen	Old school	Erie	China	7
3	39	Ice Cube	Raphael	Blumire	Hardcore	Denver	China	8
4	3	Run DMC	Isidoro	Rossant	Hardcore	Lansing	Syria	6
5	4	GLOBE	Evyn	Dawtrej	Crunk	Asheville	Ivory Coast	5
6	33	NWA	Clarette	Portman	Trap	Tacoma	China	[NULL]
7	63	Fedd the God	Clarissa	Crack	Gangsta Rap	Sparks	France	1
8	47	Lupe Fiasco	Mariann	Cana	Old school	Little Rock	China	5
9	16	50 Cent	Merv	Majury	Hardcore	Portland	Portugal	6
10	22	Kanye West	Elwin	Juza	Old school	Lincoln	China	[NULL]
11	66	Sledgren	Lauraine	MacGilpatrick	Pop Rap	Honolulu	Indonesia	1
12	21	DMX	Avery	Petronis	Trap	Washington	China	6
13	30	OutKast	Clevey	Caffery	Crunk	Oklahoma Cit	China	10
14	65	Kris Hollis	Mattie	Jiracek	Hardcore	Huntington	Japan	1
15	57	Juicy J	Valenka	Astall	Pop Rap	Fresno	Syria	1
16	61	Berner	Bryon	Alf	G-Funk	Inglewood	Portugal	1
17	41	The Pharcyde	Agnes	Sidwell	Gangsta rap	Clearwater	China	9
18	27	Wizkid	Jacquette	Fitzer	Pop rap	Springfield	Finland	4
19	60	Chevy Woods	Shepperd	Brimner	Crunk	New York City	Indonesia	1
20	13	Grandmaster Flash	Gaylor	Mickan	Pop Rap	Dayton	China	8
21	68	RMB Justize	Moyna	Hallock	Crunk	El Paso	Indonesia	1
22	29	Mobb Deep	Maribeth	Presdie	Crunk	Jackson	China	5
23	10	JAY-Z	Augusto	Braybrook	Hardcore	Port Saint Luc	Colombia	4

4. Devuelve el número de canciones que tiene cada artista.

```
select a.Nombre_artístico , count(rhc.RAPERO_idRAPERO)

from artista a inner join rapero r on a.idARTISTA = r.ARTISTA_idARTISTA

inner join rapero_has_canción rhc on r.idRAPERO = rhc.RAPERO_idRAPERO

group by a.Nombre_artístico ;
```

	Nombre_artístico	count(rhc.RAPERO_idRAPERO)
1	Public Enemy	5
2	Wiz Khalifa	8
3	Ty Dolla Sign	1
4	Chevy Woods	1
5	Project Pat	1
6	Juicy J	2
7	A Tribe Called Quest	5
8	Dr Dre	5
9	Snoop Doggy Dogg	7

5. Devuelve los artistas que llevan más de 5 años en cualquier discográfica.

```
select a.Nombre_artístico

from discográfica_has_artista dha inner join artista a

on dha.ARTISTA_idARTISTA = a.idARTISTA

where dha.fecha_baja is not null and datediff(fecha_baja,fecha_alta)/365>=5;
```

	Nombre_artístico
1	Grandmaster Flash & The Furious Five
2	Ice Cube
3	Run DMC
4	GLOBE
5	NWA
6	The Fugees
7	Stop The Violence Movement
8	TM88
9	O.C.
10	Lupe Fiasco
11	Public Enemy
12	JAY-Z
13	Kanye West
14	Snoop Doggy Dogg
15	Geto Boys
16	Canibus
17	Sledgren
18	DMX
19	Kris Hollis
20	Slum Village
21	Juicy J
22	Sosamann
23	DJ Bonics
24	Wizkid

VI. VISTAS

1. Vista 1

create view Vista1 as

select a.Nombre_artístico

from discográfica_has_artista dha inner join artista a

on dha.ARTISTA_idARTISTA = a.idARTISTA

where dha.fecha_baja is not null and datediff(fecha_baja,fecha_alta)/365>=5;

2. Vista 2

create view Vista2 as

select a.*

from artista a inner join rapero r on a.idARTISTA = r.ARTISTA_idARTISTA

inner join rapero_has_canción rhc on r.idRAPERO = rhc.RAPERO_idRAPERO

inner join canción c on rhc.CANCIÓN_idCANCIÓN = c.idCANCIÓN

where c.título = 'KK' and rhc.`¿principal?` = 'false';

VII. FUNCIONES Y PROCEDIMIENTOS

1. Función que calcula el número de canciones que tiene un artista.

DELIMITER \$\$

CREATE function calcular_canciones_artistas (IDARTISTA int) returns int

deterministic

begin

declare canciones_raperos int;

declare canciones_productores int;

declare total_canciones int;

select count(rhc.CANCIÓN_idCANCIÓN) into canciones_raperos

from artista a inner join rapero r on a.idARTISTA = r.ARTISTA_idARTISTA

inner join rapero_has_canción rhc on rhc.RAPERO_idRAPERO = r.idRAPERO

where a.idARTISTA = IDARTISTA


```

group by rhc.RAPERO_idRAPERO ;

select count(phc.CANCIÓN_idCANCIÓN) into canciones_productores

from artista a inner join productor p on a.idARTISTA = p.ARTISTA_idARTISTA

inner join productor_has_canción phc on phc.PRODUCTOR_idPRODUCTOR =
p.idPRODUCTOR

where a.idARTISTA = IDARTISTA

group by phc.PRODUCTOR_idPRODUCTOR ;

if (canciones_raperos is null) then

        set canciones_raperos = 0;

end if;

if (canciones_productores is null) then

        set canciones_productores = 0;

end if;

set total_canciones = canciones_raperos + canciones_productores ;

return total_canciones ;

end$$

DELIMITER ;

```

```

select calcular_canciones_artistas (55);

```

	123	calcular_canciones_artistas (55)	↕
1			8

2. Función que devuelve el número de artistas que tiene una banda.

DELIMITER \$\$

CREATE function calcular_artistas_bandas (banda int) returns int

deterministic

begin

declare num_artistas int;

select count(a.BANDA_idBANDA) into num_artistas

from banda b inner join artista a on b.idBANDA = a.BANDA_idBANDA

where b.idBANDA = banda

group by b.idBANDA ;

return num_artistas ;

END\$\$

DELIMITER ;

select calcular_artistas_bandas (1);

	123	calcular_artistas_bandas (1)	↕
1			15

3. Procedimiento que suma 100 \$ si el artista que se le pasa tiene más de 5 canciones.

DELIMITER \$\$

CREATE PROCEDURE aplicar_factura_artista (IDARTISTA int)

```

begin

    declare num_canciones int;

    set num_canciones = calcular_canciones_artistas (IDARTISTA);

    if (num_canciones>5) then

        update discográfica_has_artista dha

        set dha.factura = dha.factura + 100

        where dha.ARTISTA_idARTISTA = IDARTISTA ;

    end if;

end$$

DELIMITER ;

```

```
call aplicar_factura_artista (55);
```

Name	Value
Updated Rows	3
Query	call aplicar_factura_artista (55)
Finish time	Thu Mar 24 18:57:59 CET 2022

92	68 ↗	55 ↗	1992-12-26	[NULL]	699.454,23
----	----------------------	----------------------	------------	--------	------------

92	68 ↗	55 ↗	1992-12-26	[NULL]	699.554,23
----	----------------------	----------------------	------------	--------	------------

4. Cursor que muestre los datos concatenados de cada artista en registros diferentes.

```
DELIMITER $$
```

```
CREATE PROCEDURE cursor_datos_concatenados()
```

begin

```
DECLARE done INT DEFAULT FALSE;
```

```
declare datos varchar (1000);
```

```
DECLARE cur CURSOR FOR select concat('ID: ',a.idARTISTA ,' Nombre artistico: ',a.Nombre_artístico,' Nombre: ',a.Nombre, ' Apellido: ',a.Apellidos,' Estilo: ',a.Estilo,' Ciudad: ',a.Ciudad,' Pais: ',a.País) from artista a;
```

```
DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
```

```
open cur ;
```

```
while done = false do
```

```
    FETCH cur INTO datos;
```

```
    select datos;
```

```
end while;
```

```
close cur;
```

```
end$$
```

```
DELIMITER ;
```

```
call cursor_datos_concatenados ;
```

call cursor_datos_concatenados	
Data filter is not supported	
Grilla	datos
1	ID: 63 Nombre artistico: Fedd the God Nombre: Clarissa Apellido: Crack Estilo: Gangsta Rap Ciudad: Sparks Pais: France
Record	

Resultados 1 (63)	Resultados 1 (64) ×	Resultados 1 (65)	Resultados 1 (66)	Resultados 1 (67)	Resultados 1 (68)	Resultados 1 (69)	»62
call cursor_datos_concatenados <i>Data filter is not supported</i>							
Grilla							
1 ID: 64 Nombre artistico: Young Deji Nombre: Heida Apellido: Tripney Estilo: Hardcore Ciudad: Salt Lake City Pais: Colombia							
Record							

Resultados 1 (63)	Resultados 1 (64)	Resultados 1 (65) ×	Resultados 1 (66)	Resultados 1 (67)	Resultados 1 (68)	Resultados 1 (69)	»62
call cursor_datos_concatenados <i>Data filter is not supported</i>							
Grilla							
1 ID: 65 Nombre artistico: Kris Hollis Nombre: Mattie Apellido: Jiracek Estilo: Hardcore Ciudad: Huntington Pais: Japan							
Record							

- Procedimiento que recibido el código de una discográfica y un número n devuelva los artistas que pertenezcan a esa discográfica ordenados por la factura correspondiente a cada artista de mayor a menor hasta n registros.

DELIMITER \$\$

CREATE PROCEDURE lista_artistas_factura_maxima (disco int , n int)

begin

select d.idDISCOGRÁFICA , a.idARTISTA , a.Nombre_artístico , dha.factura

from artista a inner join discográfica_has_artista dha on a.idARTISTA =
dha.ARTISTA_idARTISTA

inner join discográfica d on dha.DISCOGRÁFICA_idDISCOGRÁFICA = d.idDISCOGRÁFICA

where d.idDISCOGRÁFICA = disco

order by dha.factura desc limit n;

end\$\$

DELIMITER ;

call lista_artistas_factura_maxima (29, 4);

	<small>123</small> idDISCOGRÁFICA <small>↕</small>	<small>123</small> idARTISTA <small>↕</small>	<small>ABC</small> Nombre_artístico <small>↕</small>	<small>123</small> factura <small>↕</small>
1	29	13	Grandmaster Flash	687.075,58
2	29	68	RMB Justize	514.788
3	29	44	A Tribe Called Quest	185.313,13
4	29	24	Drake	141.139,25

VIII. TRIGGERS

1. Trigger que al insertar un artista compruebe que no existe ese mismo nombre artístico en otro registro.

DELIMITER \$\$

CREATE TRIGGER artista_mismo_nombre

BEFORE INSERT ON artista FOR EACH ROW

begin

declare nombre_artista varchar(50);

select a.Nombre_artístico into nombre_artista

from artista a

where a.Nombre_artístico = new.Nombre_artístico ;

if(nombre_artista like new.Nombre_artístico) then

SIGNAL SQLSTATE '45000'

SET MESSAGE_TEXT='Ese nombre lo tiene ya otro artista',

MYSQL_ERRNO = 1643 ;

end if;

END\$\$

DELIMITER ;

idARTISTA	Nombre_artístico	Nombre	Apellidos	Estilo	Ciudad	País	BANDA_idBANDA
47	Lupe Fiasco	Mariann	Cana	Old school	Little Rock	China	5
48	Canibus	Dawna	Gallemore	Gangsta rap	Saint Paul	United States	8
<div><div><div><div><div></div><div>Error synchronizing data with database</div></div><div><div>Razón:</div><div>SQL Error [1643] [45000]: Ese nombre lo tiene ya otro artista</div></div></div><div><div>Aceptar</div><div>1 Detalles</div></div></div><div>SQL Error [1643] [45000]: Ese nombre lo tiene ya otro artista Ese nombre lo tiene ya otro artista</div></div>							
68	RMB Justize	Moyna	Hallock	Crunk	El Paso	Indonesia	1
69	RMB Justize	Mamerto	Galán	Crunk	Bormujos	España	2

2. Trigger que al eliminar una banda no deje eliminarla si esa banda tiene más de 10 artistas incluidos.

DELIMITER \$\$

CREATE TRIGGER banda_con_mas_de_10_artistas

BEFORE DELETE ON banda FOR EACH ROW

begin

declare num_artistas int;

set num_artistas = calcular_artistas_bandas (old.idBANDA);

if (num_artistas>10) then

SIGNAL SQLSTATE '45000'

SET MESSAGE_TEXT='Esta banda tiene más de 10 artistas incluidos',

