

Bases de datos relacionales (SQL)

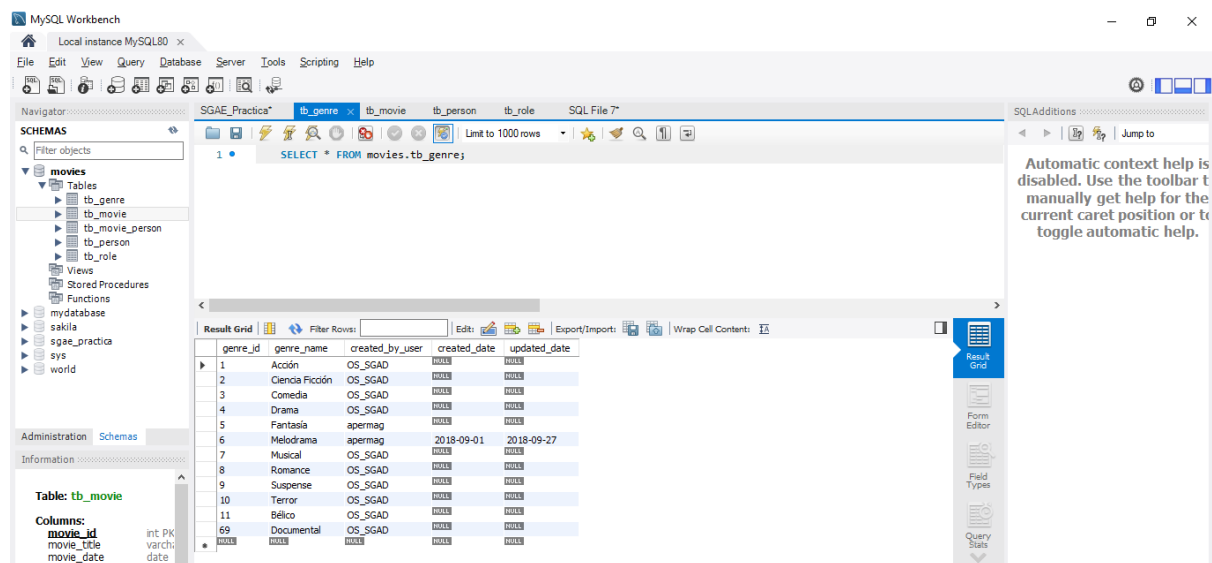
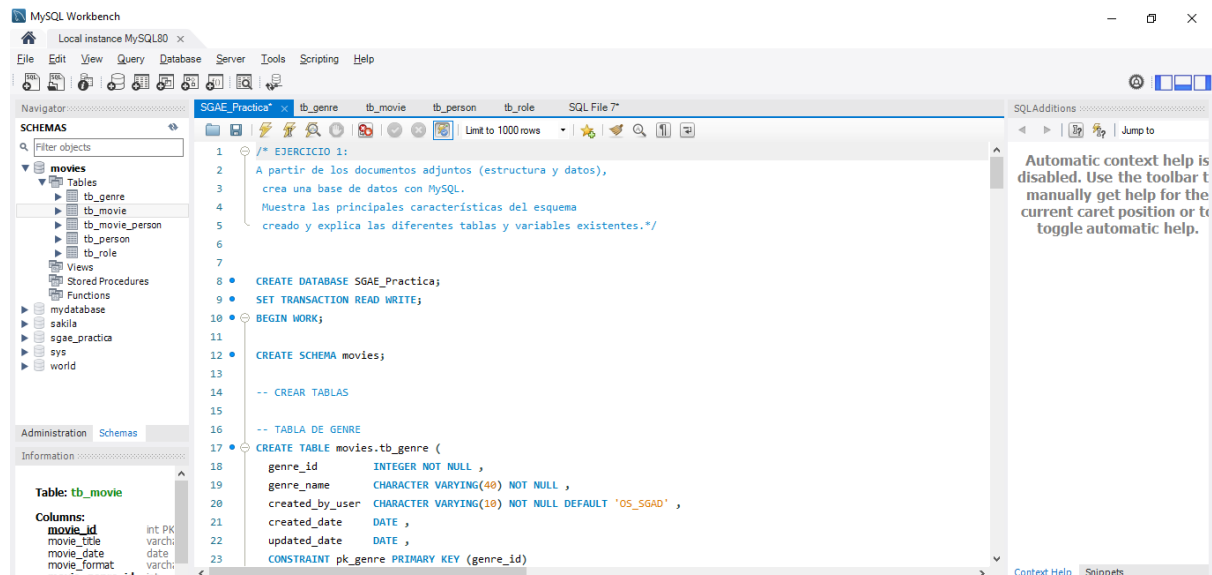
Tarea M1 T01

- Ejercicio 1

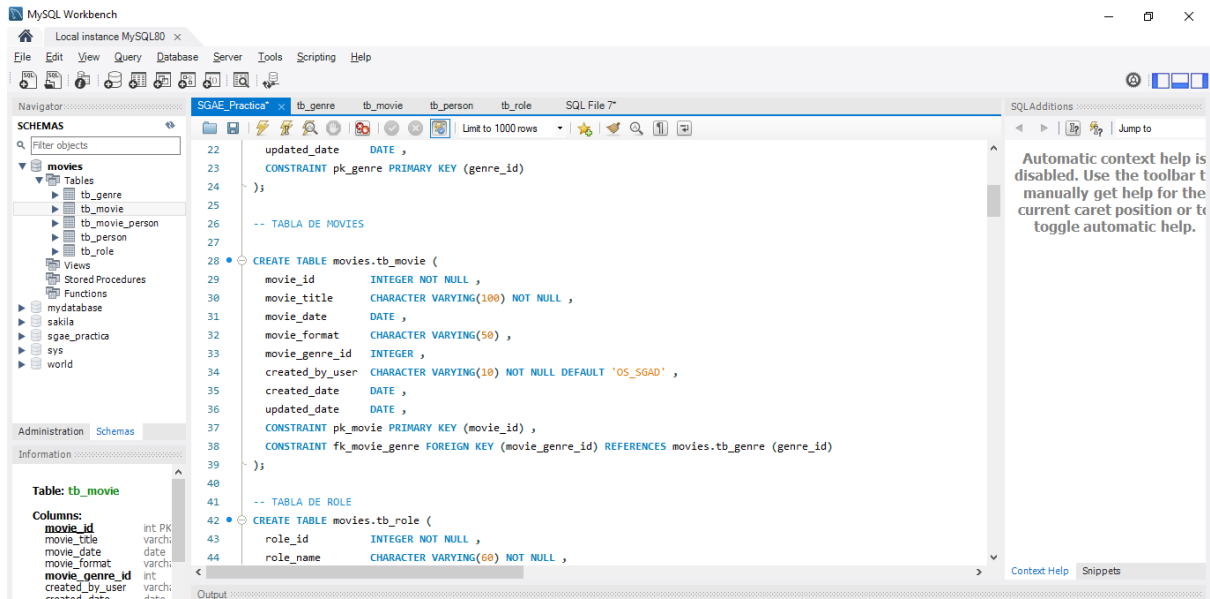
A partir de los documentos adjuntos (estructura y datos), crea una base de datos con MySQL. Muestra las principales características del esquema creado y explica las diferentes tablas y variables existentes.

El esquema creado se llama: **movies**, contiene 5 tablas sobre información de películas como: género, nombres de películas y sus características, personas y sus características, roles de las personas y una relación entre ellas indicando si tienen o no premiación.

- Tabla de género:



- Tabla de movies:



MySQL Workbench interface showing the SQL editor with the schema for the 'movies' database. The schema includes tables for genres, movies, roles, and users, with their respective columns and constraints.

```

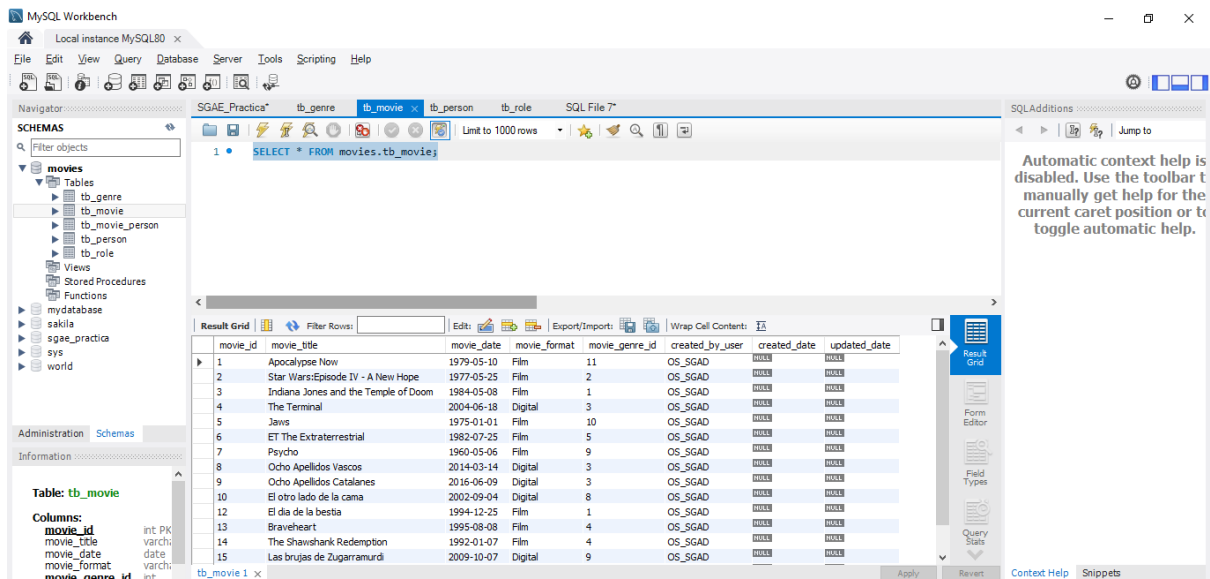
22 updated_date DATE ,
23 CONSTRAINT pk_genre PRIMARY KEY (genre_id)
24 );
25
26 -- TABLA DE MOVIES
27
28 CREATE TABLE movies.tb_movie (
29 movie_id INTEGER NOT NULL ,
30 movie_title CHARACTER VARYING(100) NOT NULL ,
31 movie_date DATE ,
32 movie_format CHARACTER VARYING(50) ,
33 movie_genre_id INTEGER ,
34 created_by_user CHARACTER VARYING(10) NOT NULL DEFAULT 'OS_SGAD' ,
35 created_date DATE ,
36 updated_date DATE ,
37 CONSTRAINT pk_movie PRIMARY KEY (movie_id) ,
38 CONSTRAINT fk_movie_genre FOREIGN KEY (movie_genre_id) REFERENCES movies.tb_genre (genre_id)
39 );
40
41 -- TABLA DE ROLE
42 CREATE TABLE movies.tb_role (
43 role_id INTEGER NOT NULL ,
44 role_name CHARACTER VARYING(50) NOT NULL ,

```

Table: **tb_movie**

Columns:

- movie_id int PK
- movie_title varchar
- movie_date date
- movie_format varchar
- movie_genre_id int
- created_by_user varchar
- created_date date



MySQL Workbench interface showing the SQL editor with the query `SELECT * FROM movies.tb_movie;` and the resulting data grid.

movie_id	movie_title	movie_date	movie_format	movie_genre_id	created_by_user	created_date	updated_date
1	Apocalypse Now	1979-05-10	Film	11	OS_SGAD	NULL	NULL
2	Star Wars:Episode IV - A New Hope	1977-05-25	Film	2	OS_SGAD	NULL	NULL
3	Indiana Jones and the Temple of Doom	1984-05-08	Film	1	OS_SGAD	NULL	NULL
4	The Terminal	2004-06-18	Digital	3	OS_SGAD	NULL	NULL
5	Jaws	1975-01-01	Film	10	OS_SGAD	NULL	NULL
6	ET The Extraterrestrial	1982-07-25	Film	5	OS_SGAD	NULL	NULL
7	Psycho	1960-05-06	Film	9	OS_SGAD	NULL	NULL
8	Ocho Apellidos Vascos	2014-03-14	Digital	3	OS_SGAD	NULL	NULL
9	Ocho Apellidos Catalanes	2016-06-09	Digital	3	OS_SGAD	NULL	NULL
10	El otro lado de la cama	2002-09-04	Digital	8	OS_SGAD	NULL	NULL
12	El dia de la bestia	1994-12-25	Film	1	OS_SGAD	NULL	NULL
13	Braveheart	1995-08-08	Film	4	OS_SGAD	NULL	NULL
14	The Shawshank Redemption	1992-01-07	Film	4	OS_SGAD	NULL	NULL
15	Las brujas de Zugarramundi	2009-10-07	Digital	9	OS_SGAD	NULL	NULL

- Tabla de role:

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with the 'movies' database selected. The main editor window shows the SQL script for creating the 'tb_role' table. The script includes a primary key constraint for 'role_id' and a foreign key constraint for 'movie_id' referencing 'tb_movie'.

```

34 created_by_user CHARACTER VARYING(10) NOT NULL DEFAULT 'OS_SGAD' ,
35 created_date DATE ,
36 updated_date DATE ,
37 CONSTRAINT pk_movie PRIMARY KEY (movie_id) ,
38 CONSTRAINT fk_movie_genre FOREIGN KEY (movie_genre_id) REFERENCES movies.tb_genre (genre_id)
39 );
40
41 -- TABLA DE ROLE
42 CREATE TABLE movies.tb_role (
43   role_id INTEGER NOT NULL ,
44   role_name CHARACTER VARYING(60) NOT NULL ,
45   created_by_user CHARACTER VARYING(10) NOT NULL DEFAULT 'OS_SGAD' ,
46   created_date DATE ,
47   updated_date DATE ,
48   CONSTRAINT pk_role PRIMARY KEY (role_id)
49 );
50
51 -- TABLA DE PERSON
52 CREATE TABLE movies.tb_person (
53   person_id INTEGER NOT NULL ,
54   person_name CHARACTER VARYING(100) NOT NULL ,
55   person_country CHARACTER VARYING(40) ,
56   person_dob DATE NOT NULL ,

```

The right sidebar shows the 'SQL Additions' panel with a message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with the 'movies' database selected. The main editor window shows the SQL script for selecting all data from the 'tb_role' table. The script is: `SELECT * FROM movies.tb_role;`

```

1 SELECT * FROM movies.tb_role;

```

The right sidebar shows the 'SQL Additions' panel with a message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

The bottom panel shows the 'Result Grid' with the following data:

role_id	role_name	created_by_user	created_date	updated_date
1	Actor	OS_SGAD	2023-10-27 10:10:10	2023-10-27 10:10:10
2	Director	OS_SGAD	2023-10-27 10:10:10	2023-10-27 10:10:10
3	Productor	OS_SGAD	2023-10-27 10:10:10	2023-10-27 10:10:10
4	Guionista	OS_SGAD	2023-10-27 10:10:10	2023-10-27 10:10:10
5	Musica	OS_SGAD	2023-10-27 10:10:10	2023-10-27 10:10:10

- Tabla de person:

MySQL Workbench interface showing the SQL editor. The left sidebar displays the Schemas tree with the 'movies' database selected. The main editor window shows the SQL File 7 with the following code:

```

49  );
50
51  -- TABLA DE PERSON
52  CREATE TABLE movies.tb_person (
53      person_id      INTEGER NOT NULL ,
54      person_name     CHARACTER VARYING(100) NOT NULL ,
55      person_country  CHARACTER VARYING(40) ,
56      person_dob      DATE NOT NULL ,
57      person_dod      DATE ,
58      person_parent_id INTEGER ,
59      created_by_user  CHARACTER VARYING(10) NOT NULL DEFAULT 'OS_SGAD' ,
60      created_date     DATE ,
61      updated_date     DATE ,
62      CONSTRAINT pk_person PRIMARY KEY (person_id) ,
63      CONSTRAINT fk_person_parent FOREIGN KEY (person_parent_id) REFERENCES movies.tb_person (person_id)
64  );
65
66  -- TABLA DE MOVIE PERSON
67  CREATE TABLE movies.tb_movie_person (
68      movie_id        INTEGER NOT NULL ,
69      person_id        INTEGER NOT NULL ,
70      role_id          INTEGER NOT NULL ,
71      movie_ward_ind   CHAR(1) NOT NULL ,

```

The right sidebar shows the SQL Additions panel with a message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

MySQL Workbench interface showing the SQL editor. The left sidebar displays the Schemas tree with the 'movies' database selected. The main editor window shows the SQL File 7 with the following code:

```

1  SELECT * FROM movies.tb_person;

```

The right sidebar shows the SQL Additions panel with a message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

The Result Grid displays the data from the query:

person_id	person_name	person_country	person_dob	person_dod	person_parent_id	created_by_user	created_date	updated_date
1	Francis Ford Coppola	United States	1939-04-07	NULL	NULL	OS_SGAD	NULL	NULL
2	Carmin Coppola	United States	1945-07-08	NULL	NULL	OS_SGAD	NULL	NULL
3	Marlon Brando	United States	1924-04-03	2004-07-01	NULL	OS_SGAD	NULL	NULL
4	Robert Duvall	United States	1931-01-05	NULL	NULL	OS_SGAD	NULL	NULL
5	Martin Sheen	United States	1940-08-03	NULL	NULL	OS_SGAD	NULL	NULL
6	Harrison Ford	United States	1942-07-13	NULL	NULL	OS_SGAD	NULL	NULL
7	George Lucas	United States	1944-05-14	NULL	NULL	OS_SGAD	NULL	NULL
8	Gary Kurtz	United States	1940-07-27	NULL	NULL	OS_SGAD	NULL	NULL
9	Steven Spielberg	United States	1946-12-18	NULL	NULL	OS_SGAD	NULL	NULL
10	John Williams	United States	1928-08-08	NULL	NULL	OS_SGAD	NULL	NULL
11	Tom Hanks	United States	1956-07-09	NULL	NULL	OS_SGAD	NULL	NULL
12	Catherine Zeta-Jones	Wales	1969-09-25	NULL	NULL	OS_SGAD	NULL	NULL
13	Alfred Joseph Hitchcock	United Kingdom	1899-08-13	1980-04-29	NULL	OS_SGAD	NULL	NULL

- Tabla movie_person:

The screenshot shows the MySQL Workbench interface with the SQL script for creating the `tb_movie_person` table. The script includes a primary key constraint for `person_id` and foreign key constraints for `movie_id`, `person_id`, and `role_id`.

```

61 updated_date DATE ,
62 CONSTRAINT pk_person PRIMARY KEY (person_id) ,
63 CONSTRAINT fk_person_parent FOREIGN KEY (person_parent_id) REFERENCES movies.tb_person (person_id)
64 );
65
66 -- TABLA DE MOVIE PERSON
67 CREATE TABLE movies.tb_movie_person (
68 movie_id INTEGER NOT NULL ,
69 person_id INTEGER NOT NULL ,
70 role_id INTEGER NOT NULL ,
71 movie_award_ind CHAR(1) NOT NULL ,
72 created_by_user CHARACTER VARYING(10) NOT NULL DEFAULT 'OS_SGAD' ,
73 created_date DATE ,
74 updated_date DATE ,
75 CONSTRAINT pk_movper PRIMARY KEY (movie_id, person_id, role_id) ,
76 CONSTRAINT fk_movper_movie FOREIGN KEY (movie_id) REFERENCES movies.tb_movie (movie_id) ,
77 CONSTRAINT fk_movper_person FOREIGN KEY (person_id) REFERENCES movies.tb_person (person_id) ,
78 CONSTRAINT fk_movper_role FOREIGN KEY (role_id) REFERENCES movies.tb_role (role_id)
79 );
80
81 COMMIT;
82
83 -- INSERTAR DATOS

```

The left sidebar shows the Schemas pane with the `movies` database selected. The right sidebar shows the SQL Additions pane with a message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

The screenshot shows the MySQL Workbench interface with the query `SELECT * FROM tb_movie_person;` executed. The result grid displays the data for the `tb_movie_person` table.

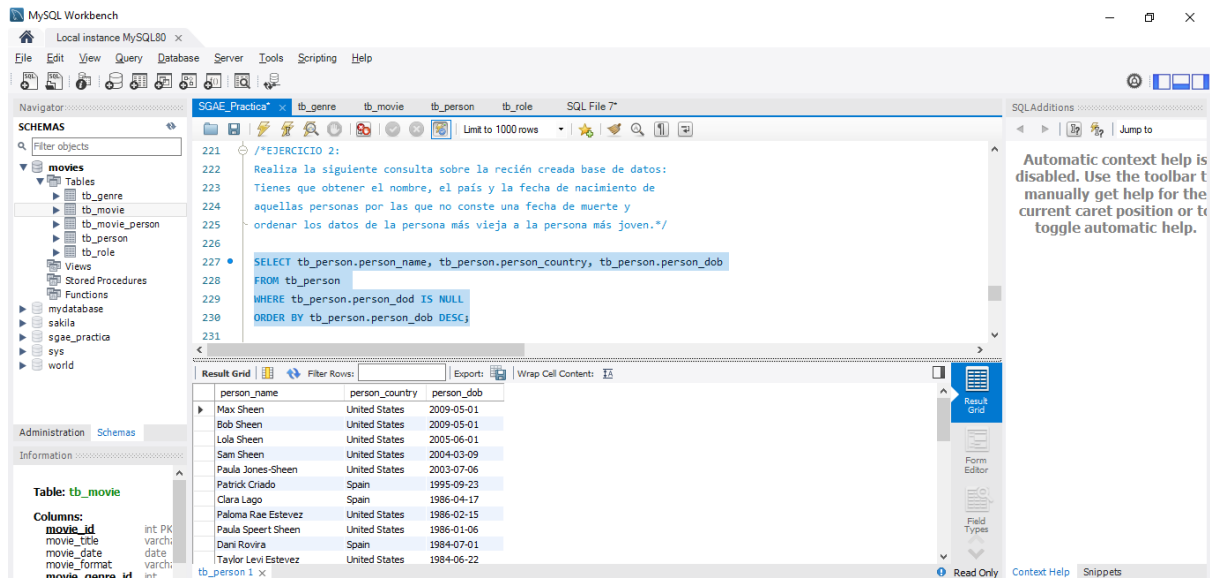
	movie_id	person_id	role_id	movie_award_ind	created_by_user	created_date	updated_date
1	1	2	Y	OS_SGAD	NULL	NULL	NULL
1	1	3	N	OS_SGAD	NULL	NULL	NULL
1	1	5	N	OS_SGAD	NULL	NULL	NULL
1	2	5	N	OS_SGAD	NULL	NULL	NULL
1	3	1	N	OS_SGAD	NULL	NULL	NULL
1	4	1	N	OS_SGAD	NULL	NULL	NULL
1	5	1	Y	OS_SGAD	NULL	NULL	NULL
1	6	1	N	OS_SGAD	NULL	NULL	NULL
1	41	1	N	OS_SGAD	NULL	NULL	NULL
2	6	1	N	OS_SGAD	NULL	NULL	NULL
2	7	2	Y	OS_SGAD	NULL	NULL	NULL
2	8	3	N	OS_SGAD	NULL	NULL	NULL
3	6	1	N	OS_SGAD	NULL	NULL	NULL
3	7	1	N	OS_SGAD	NULL	NULL	NULL

The left sidebar shows the Schemas pane with the `movies` database selected. The right sidebar shows the SQL Additions pane with a message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

- Ejercicio 2

Realiza la siguiente consulta sobre la recién creada base de datos:

Tienes que obtener el nombre, el país y la fecha de nacimiento de aquellas personas por las que no conste una fecha de muerte y ordenar los datos de la persona más vieja a la persona más joven.



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
/*EJERCICIO 2:
Realiza la siguiente consulta sobre la recién creada base de datos:
Tienes que obtener el nombre, el país y la fecha de nacimiento de
aquellas personas por las que no conste una fecha de muerte y
ordenar los datos de la persona más vieja a la persona más joven.*/

SELECT tb_person.person_name, tb_person.person_country, tb_person.person_dob
FROM tb_person
WHERE tb_person.person_dod IS NULL
ORDER BY tb_person.person_dob DESC;
```

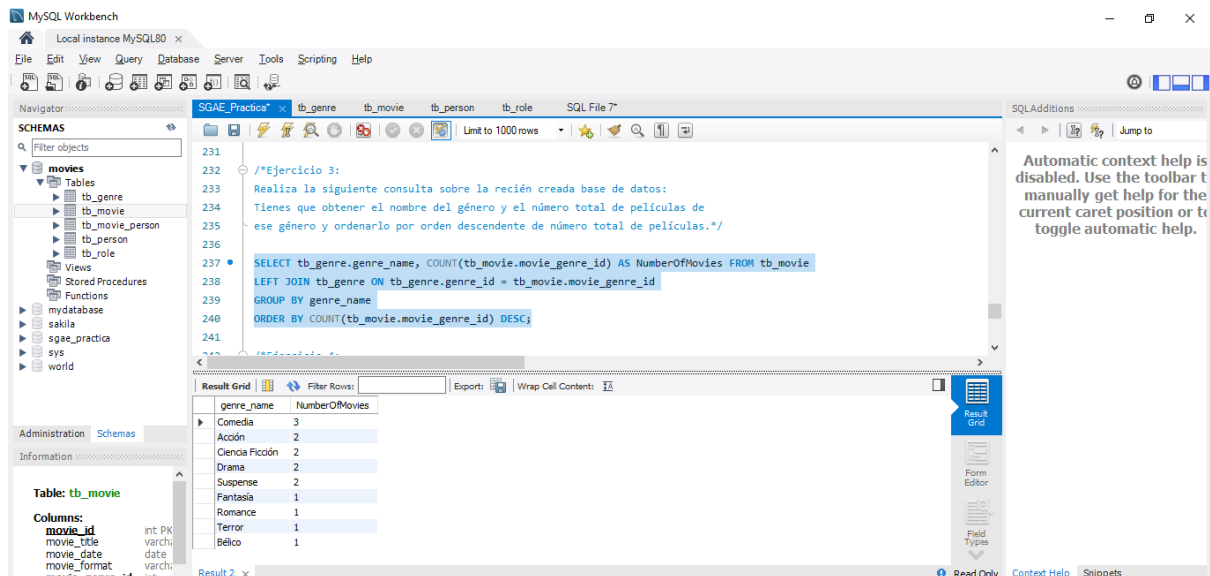
The Result Grid shows the following data:

person_name	person_country	person_dob
Max Sheen	United States	2009-05-01
Bob Sheen	United States	2009-05-01
Lola Sheen	United States	2005-06-01
Sam Sheen	United States	2004-03-09
Paula Jones-Sheen	United States	2003-07-06
Patrick Criado	Spain	1995-09-23
Clara Lago	Spain	1986-04-17
Paloma Rae Estevez	United States	1986-02-15
Paula Speert Sheen	United States	1986-01-06
Dani Rovira	Spain	1984-07-01
Taylor Levi Estevez	United States	1984-06-22

- Ejercicio 3

Realiza la siguiente consulta sobre la recién creada base de datos:

Tienes que obtener el nombre del género y el número total de películas de ese género y ordenarlo por orden descendente de número total de películas.



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
/*Ejercicio 3:
Realiza la siguiente consulta sobre la recién creada base de datos:
Tienes que obtener el nombre del género y el número total de películas de
ese género y ordenarlo por orden descendente de número total de películas.*/

SELECT tb_genres.genre_name, COUNT(tb_movies.movie_genre_id) AS NumberOfMovies
FROM tb_movies
LEFT JOIN tb_genres ON tb_genres.genre_id = tb_movies.movie_genre_id
GROUP BY genre_name
ORDER BY COUNT(tb_movies.movie_genre_id) DESC;
```

The Result Grid shows the following data:

genre_name	NumberOfMovies
Comedia	3
Acción	2
Ciencia Ficción	2
Drama	2
Suspense	2
Fantasia	1
Romance	1
Terror	1
Bélico	1

- Ejercicio 4

Realiza la siguiente consulta sobre la recién creada base de datos:

Tienes que obtener, para cada persona, su nombre y el número máximo de roles distintos que ha tenido en una misma película.

Posteriormente, muestra únicamente a aquellas personas que hayan asumido más de un rol en una misma película.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
/*Ejercicio 4:
Realiza la siguiente consulta sobre la recién creada base de datos:
Tienes que obtener, para cada persona, su nombre y el número máximo
de roles distintos que ha tenido en una misma película.
Posteriormente, muestra únicamente a aquellas personas que hayan
asumido más de un rol en una misma película.*/

SELECT tb_person.person_name, COUNT(DISTINCT tb_movie_person.role_id) AS NumberOfDifRoles FROM tb_movie_person
LEFT JOIN tb_person ON tb_person.person_id = tb_movie_person.person_id
GROUP BY person_name;
```

The Result Grid shows the following data:

person_name	NumberOfDifRoles
Alfred Joseph Hitchcock	3
Anthony Perkins	1
Carmen Machi	1
Carmine Coppola	1
Catherine Zeta-Jones	1
Charlie Sheen	1
Clara Lago	1
Dani Rovira	1
Emilio Martínez Lázaro	1
Francis Ford Coppola	3
Gary Kurtz	1

The screenshot shows the MySQL Workbench interface with the second query in the SQL editor:

```
SELECT tb_person.person_name, COUNT(DISTINCT tb_movie_person.role_id) AS MoreThanOneRole FROM tb_movie_person
LEFT JOIN tb_person ON tb_person.person_id = tb_movie_person.person_id
GROUP BY person_name;
```

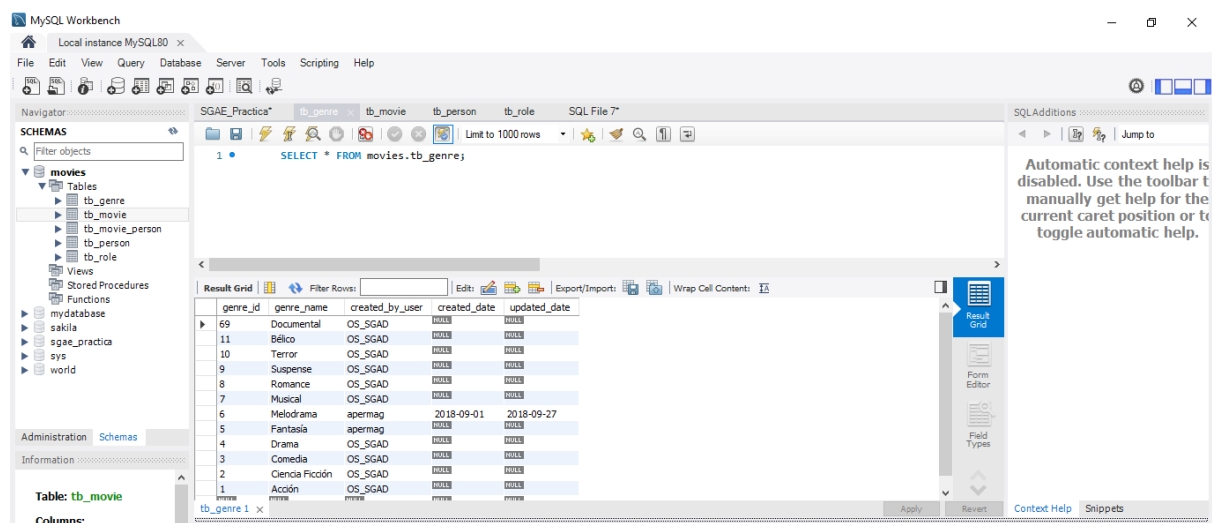
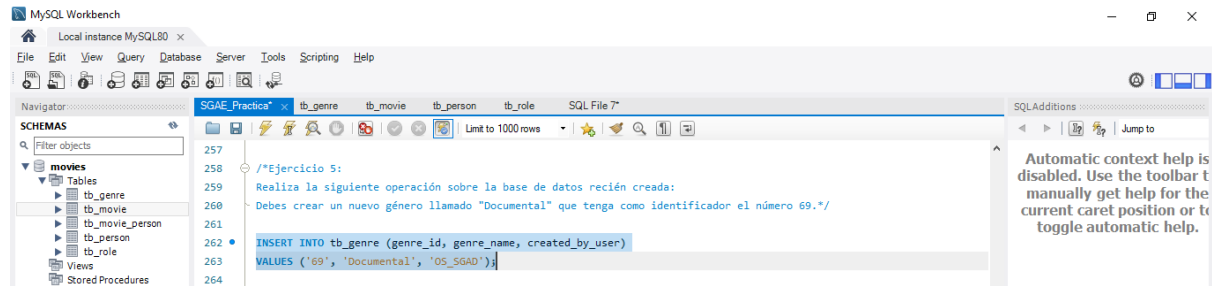
The Result Grid shows the following data:

person_name	MoreThanOneRole
Alfred Joseph Hitchcock	3
Francis Ford Coppola	3
George Lucas	3
Mel Gibson	2
Steven Spielberg	2

- Ejercicio 5

Realiza la siguiente operación sobre la base de datos recién creada:

Debes crear un nuevo género llamado "Documental" que tenga como identificador el número 69.

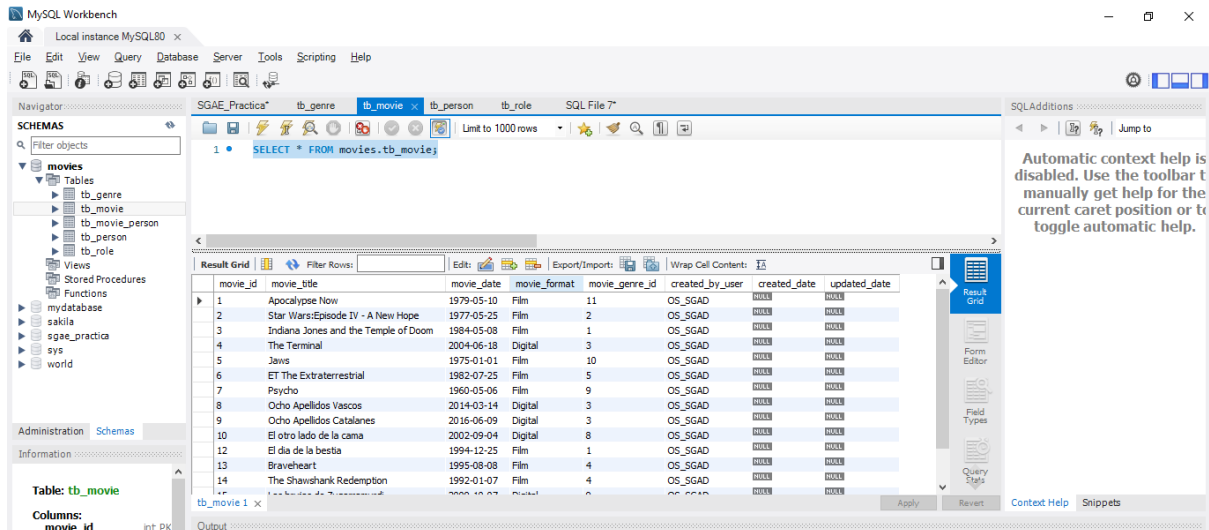


- Ejercicio 6

Realiza la siguiente operación sobre la base de datos recién creada:

Elimina la película "La Gran Familia Española" de la base de datos.





- Ejercicio 7

Realiza la siguiente operación sobre la base de datos recién creada:

Cambia el género de la película "Ocho apellidos catalanes" para que conste como comedia y no como romántica.

