

MySQL Workbench

Local instance MySQL80

FileEditViewQueryDatabaseServerToolsScriptingHelp

SQLFileDatabaseQueryToolsServerToolsScriptingHelp

Limit to 1000 rows

StarIconMagnifyingGlassIcon

SQL File 1\*

SCHEMAS

Filter objects

general\_inicio

movies

Tables

tb\_genre

tb\_movie

tb\_movie\_person

tb\_person

tb\_role

Views

Stored Procedures

Functions

AdministrationSchemas

No object selected

SQL File 1\*

2

3 • use movies

4 ;

5

6

7 • select person\_name, person\_country, person\_dob from tb\_person

8 where person\_dob is null

9 order by person\_dob asc

10 ;

11

12

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

person\_nameperson\_countryperson\_dob

John WilliamsUnited States1928-08-08

Vera MilesUnited States1929-08-23

Sean ConneryScotland1930-07-08

Robert DuvallUnited States1931-01-05

Morgan FreemanUnited States1935-10-01

Francis Ford CoppolaUnited States1939-04-07

Gary KurtzUnited States1940-07-27

Martin SheenUnited States1940-08-03

Harrison FordUnited States1942-07-13

George LucasUnited States1944-05-14

Carmine CoppolaUnited States1945-07-08

Steven SpielbergUnited States1946-12-18

tb\_person 1

Read OnlyContext HelpSnippets

Object Info

Session

Output



MySQL Workbench

Local instance MySQL80

FileEditViewQueryDatabaseServerToolsScriptingHelp

SQL

SQL

Navigator

SQL File 1\*SQL File 3\*

SQLAdditions

SCHEMAS

Filter objects

general\_inicio

movies

Tables

tb\_genre

tb\_movie

tb\_movie\_person

tb\_person

tb\_role

Views

Stored Procedures

Functions

Administration

Schemas

Information

No object selected

Limit to 1000 rows

select p.person\_name, count(mp.role\_id)

from tb\_movie\_person mp

inner join tb\_person p on mp.person\_id=p.person\_id

inner join tb\_movie m on mp.movie\_id=m.movie\_id

group by p.person\_name, mp.movie\_id

order by count(mp.role\_id) desc

;

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

person_name	count(mp.role_id)
Alfred Joseph Hitchcock	3
Francis Ford Coppola	3
Steven Spielberg	2
George Lucas	2
Mel Gibson	2
Harrison Ford	1
George Lucas	1
Gary Kurtz	1
Tom Hanks	1
Catherine Zeta-Jones	1
Emilio Martinez Lazaro	1
Dani Rovira	1

Result 2

Read Only

Context Help

Snippets

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

MySQL Workbench

Local instance MySQL80

FileEditViewQueryDatabaseServerToolsScriptingHelp

SQL

SQL

Navigator

SQL File 1\*SQL File 3\*

SQLAdditions

SCHEMAS

Filter objects

general\_inicio

movies

Tables

tb\_genre

tb\_movie

tb\_movie\_person

tb\_person

tb\_role

Views

Stored Procedures

Functions

Administration

Schemas

Information

No object selected

Execute the statement under the keyboard cursor

--segunda parte ejercicio--;

select p.person\_name, count(mp.role\_id)

from tb\_movie\_person mp

inner join tb\_person p on mp.person\_id=p.person\_id

inner join tb\_movie m on mp.movie\_id=m.movie\_id

group by p.person\_name, mp.movie\_id

having count(mp.role\_id)>1

order by count(mp.role\_id) asc

;

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	person_name	count(mp.role_id)
▶	George Lucas	2
	Steven Spielberg	2
	Mel Gibson	2
	Alfred Joseph Hitchcock	3
	Francis Ford Coppola	3

Result 3

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid

Form Editor

Field Types

Read Only

Context Help

Snippets

Object Info

Session

Output



MySQL Workbench

Local instance MySQL80

FileEditViewQueryDatabaseServerToolsScriptingHelp

SQL

SQL

SQL File 1\*

SQL File 3\*

Limit to 1000 rows

Navigator

SCHEMAS

Filter objects

general\_inicio

movies

Tables

tb\_genre

tb\_movie

tb\_movie\_person

tb\_person

tb\_role

Views

Stored Procedures

Functions

Administration

Schemas

Information

No object selected

SQL File 3\*

1

-- EJERCICIO 6 --

2

\*\*\*Añadir a la estructura de las tablas ON DELETE CASCADE y ON UPDATE CASCADE,

3

a continuación de la codificación de las FOREIGN KEY,

4

rompiendo así la relación con otra tabla\*\*\*--

5

;

6

7

8

delete from tb\_movie where movie\_title='La Gran Familia Española'

9

;

10

11

12

select movie\_title from tb\_movie where movie\_title='La Gran Familia Española'

13

;

14

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

movie\_title

tb\_movie 7

Read Only

Context Help

Snippets

SQLAdditions

Jump to

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Object Info

Session

Output

MySQL Workbench

Local instance MySQL80

FileEditViewQueryDatabaseServerToolsScriptingHelp

SQLFileNewQueryRecent

Limit to 1000 rows

StarIcon

SQL File 1\*SQL File 3\*

1-- EJERCICIO 7 --

2-----

3----UP\*DAT\*E ya añadido en el anterior ejercicio----

4;

5

6

7 • update tb\_movie set movie\_genre\_id='3' where movie\_id=9;

8

9

10 • select genre\_name as GÉNERO from tb\_movie m

11 inner join tb\_genre g on g.genre\_id=m.movie\_genre\_id

12 where m.movie\_title='Ocho Apellidos Catalanes'

13 group by movie\_genre\_id

14 ;

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

Result Grid

Form Editor

Result 8

Read Only

Context Help

Snippets

Object Info

Session

Output

SCHEMAS

Filter objects

general\_inicio

movies

Tables

tb\_genre

tb\_movie

tb\_movie\_person

tb\_person

tb\_role

Views

Stored Procedures

Functions

Administration

Schemas

No object selected

SQLAdditions

Jump to

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.