

# Sprint01\_Tasca01

August 17, 2022

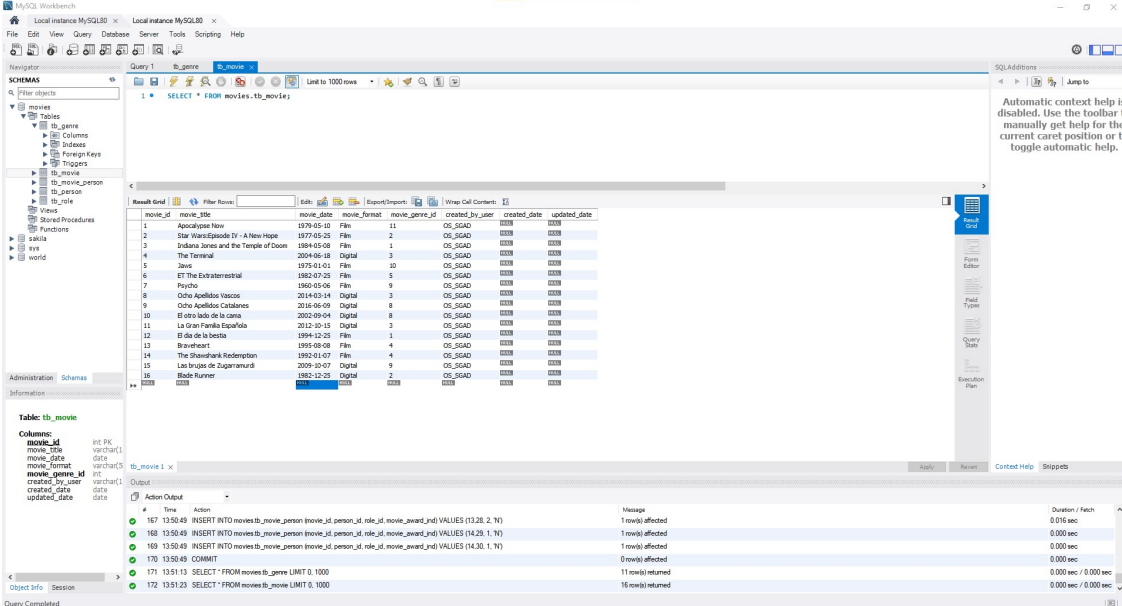
## Tasca 01

### Exercici 1

A partir dels documents adjunts (estructura i dades), crea una base de dades amb MySQL. Mostra les característiques principals de l'esquema creat i explica les diferents taules i variables que hi ha.

```
[1]: from IPython import display
from base64 import b64decode
display.Image(b64decode(base64_data))
```

[1]:



movie_id	movie_title	movie_date	movie_format	movie_genre_id	created_by_user	created_date	updated_date
1	Star Wars Episode IV: A New Hope	1977-05-25	Film	11	OS_SGAD	0000	0000
2	Indiana Jones and the Temple of Doom	1984-05-28	Film	1	OS_SGAD	0000	0000
3	The Terminator	2004-06-18	Digital	3	OS_SGAD	0000	0000
4	Jaws	1975-01-01	Film	10	OS_SGAD	0000	0000
5	ET: The Extraterrestrial	1982-07-05	Film	1	OS_SGAD	0000	0000
6	Psycho	1960-05-06	Film	9	OS_SGAD	0000	0000
7	Ocho Apellidos Vascos	2014-03-14	Digital	3	OS_SGAD	0000	0000
8	Ocho Apellidos Catalanes	2016-06-09	Digital	8	OS_SGAD	0000	0000
9	El otro lado de la cama	2002-09-04	Digital	8	OS_SGAD	0000	0000
10	La Gran Familia Española	2012-10-15	Digital	3	OS_SGAD	0000	0000
11	El día de la bestia	1994-12-25	Film	1	OS_SGAD	0000	0000
12	Braveheart	1995-08-08	Film	4	OS_SGAD	0000	0000
13	The Shawshank Redemption	1993-01-07	Film	4	OS_SGAD	0000	0000
14	Las brujas de Zugamundi	2009-10-07	Digital	9	OS_SGAD	0000	0000
15	Blade Runner	1982-12-25	Digital	2	OS_SGAD	0000	0000
16							

La base de dades movies està formada per 5 taules: 1. tb\_genre: Conté els generes de les pel·lícules, en total 11. - genre\_id: Clau primària - genre\_name: camp amb el nom del gènere

2. tb\_movie: Conté les diferents pel·lícules, amb les columnes:

- movie\_id: Clau primària
- movie\_title: Títol de la pel·lícula
- movie\_date: Data d'estrena
- movie\_format: Format de la pel·lícula

- movie\_genre: el id del gènere. Relacionada amb la clau primària camp genre\_id de la taula tb\_genre.
3. tb\_person: Persones involucrades en la filmació de les pel·lícules
- person\_id: Clau primària
  - person\_name: Nom de l'actor
  - person\_country: País d'on és l'actor
  - person\_dob: Data de naixement
  - person\_dod: Data de defunció
  - person\_parent\_id: Person\_id d'un dels progenitors si aquest també està a la taula
4. tb\_movie\_person: Relaciona els actors de la taula tb\_person amb les pel·lícules a tb\_movie:
- movie\_id:
  - person\_id: Relacionada amb la clau primària person\_id de l'actor en la taula tb\_person
  - role\_id: Relacionada amb la clau primària role\_id de la taula tb\_role
  - movie\_award\_ind: Si ha rebut algun premi
5. tb\_role: Conté els rols que pot tindre cada una de les persones involucrades en les pel·lícules
- role\_id:
  - role\_name: nom del rol

Cal connectar la base de dades de mysql amb Python mitjançant el següent codi:

```
[2]: import mysql.connector

mydb = mysql.connector.connect(
    host='localhost',
    user='root',
    password='ItAcademy383',
    database='movies'
)

print(mydb)

cursor = mydb.cursor() # Necessari per executar declaracions sql
```

<mysql.connector.connection.MySQLConnection object at 0x000001F1E8329430>

## Exercici 2

```
[3]: # Exercici 2
query = """
SELECT person_name, person_country, person_dob FROM tb_person
WHERE person_dod is NULL
ORDER BY person_dob
"""

cursor.execute(query)
rows = cursor.fetchall()
for row in rows:
```

```
print(row)
```

```
('John Williams', 'United States', datetime.date(1928, 8, 8))
('Vera Miles', 'United States', datetime.date(1929, 8, 23))
('Sean Connery', 'Scotland', datetime.date(1930, 7, 8))
('Robert Duvall', 'United States', datetime.date(1931, 1, 5))
('Morgan Freeman', 'United States', datetime.date(1935, 10, 1))
('Francis Ford Coppola', 'United States', datetime.date(1939, 4, 7))
('Gary Kurtz', 'United States', datetime.date(1940, 7, 27))
('Martin Sheen', 'United States', datetime.date(1940, 8, 3))
('Harrison Ford', 'United States', datetime.date(1942, 7, 13))
('George Lucas', 'United States', datetime.date(1944, 5, 14))
('Carmine Coppola', 'United States', datetime.date(1945, 7, 8))
('Steven Spielberg', 'United States', datetime.date(1946, 12, 18))
('Tim Robbins', 'United States', datetime.date(1949, 6, 7))
('Mel Gibson', 'Australia', datetime.date(1950, 8, 9))
('Tom Hanks', 'United States', datetime.date(1956, 7, 9))
('Emilio Martinez Lazaro', 'Spain', datetime.date(1956, 9, 9))
('Karra Elejalde', 'Spain', datetime.date(1960, 3, 6))
('Emilio Estevez', 'United States', datetime.date(1962, 5, 12))
('Ramón Estevez', 'United States', datetime.date(1963, 8, 7))
('Carmen Machi', 'Spain', datetime.date(1964, 8, 9))
('Charlie Sheen', 'United States', datetime.date(1965, 9, 3))
('René Estevez', 'United States', datetime.date(1967, 4, 2))
('Catherine Zeta-Jones', 'Wales', datetime.date(1969, 9, 25))
('Robert Alamo', 'Spain', datetime.date(1970, 5, 6))
('Daniel Sanchez Arevalo', 'Spain', datetime.date(1970, 6, 8))
('Quim Gutierrez', 'Spain', datetime.date(1981, 3, 27))
('Veronica Echegui', 'Spain', datetime.date(1983, 3, 14))
('Taylor Levi Estevez', 'United States', datetime.date(1984, 6, 22))
('Dani Rovira', 'Spain', datetime.date(1984, 7, 1))
('Paula Speert Sheen', 'United States', datetime.date(1986, 1, 6))
('Paloma Rae Estevez', 'United States', datetime.date(1986, 2, 15))
('Clara Lago', 'Spain', datetime.date(1986, 4, 17))
('Patrick Criado', 'Spain', datetime.date(1995, 9, 23))
('Paula Jones-Sheen', 'United States', datetime.date(2003, 7, 6))
('Sam Sheen', 'United States', datetime.date(2004, 3, 9))
('Lola Sheen', 'United States', datetime.date(2005, 6, 1))
('Bob Sheen', 'United States', datetime.date(2009, 5, 1))
('Max Sheen', 'United States', datetime.date(2009, 5, 1))
```

### Exercici 3

Has d'obtenir el nom del gènere i el nombre total de pel·lícules d'aquest gènere i ordenar-ho per ordre descendent de nombre total de pel·lícules.

```
[4]: query = """
      SELECT genre_name, COUNT(tb_movie.movie_id) as num_movies
```

```

FROM tb_genre
LEFT JOIN tb_movie ON tb_genre.genre_id = tb_movie.movie_genre_id
GROUP BY genre_name
ORDER BY num_movies DESC
"""
cursor.execute(query)
rows = cursor.fetchall()
for row in rows:
    print(row)

```

```

('Comedia', 3)
('Acción', 2)
('Ciencia Ficción', 2)
('Drama', 2)
('Romance', 2)
('Suspense', 2)
('Fantasía', 1)
('Terror', 1)
('Bélico', 1)
('Melodrama', 0)
('Musical', 0)

```

**Exercici 4** Has d'obtenir, per a cada persona, el seu nom i el nombre màxim de rols diferents que ha tingut en una mateixa pel·lícula.

Posteriorment, mostra únicament aquelles persones que hagin assumit més d'un rol en una mateixa pel·lícula.

```

[5]: query = """
SELECT person_name, COUNT(tb_person.person_name) as max_rols
FROM tb_person
LEFT JOIN tb_movie_person ON tb_person.person_id = tb_movie_person.person_id
GROUP BY person_name

"""
cursor.execute(query)
rows = cursor.fetchall()
for row in rows:
    print(row)

```

```

('Francis Ford Coppola', 3)
('Carmine Coppola', 1)
('Marlon Brando', 1)
('Robert Duvall', 1)
('Martin Sheen', 1)
('Harrison Ford', 3)
('George Lucas', 3)
('Gary Kurtz', 1)

```

```

('Steven Spielberg', 5)
('John Williams', 1)
('Tom Hanks', 1)
('Catherine Zeta-Jones', 1)
('Alfred Joseph Hitchcock', 3)
('Anthony Perkins', 1)
('Vera Miles', 1)
('Emilio Martinez Lazaro', 3)
('Dani Rovira', 2)
('Clara Lago', 2)
('Carmen Machi', 2)
('Karra Elejalde', 2)
('Daniel Sanchez Arevalo', 2)
('Quim Gutierrez', 1)
('Robert Alamo', 1)
('Hector Colome', 1)
('Veronica Echegui', 1)
('Patrick Criado', 1)
('Sean Connery', 1)
('Mel Gibson', 2)
('Morgan Freeman', 1)
('Tim Robbins', 1)
('Charlie Sheen', 1)
('Emilio Estevez', 1)
('Ramón Estevez', 1)
('Reneé Estevez', 1)
('Paula Speert Sheen', 1)
('Bob Sheen', 1)
('Max Sheen', 1)
('Sam Sheen', 1)
('Lola Sheen', 1)
('Paula Jones-Sheen', 1)
('Paloma Rae Estevez', 1)
('Taylor Levi Estevez', 1)

```

```

[6]: query = """
SELECT person_name, COUNT(tb_person.person_name) as max_roles
FROM tb_person
LEFT JOIN tb_movie_person ON tb_person.person_id = tb_movie_person.person_id
GROUP BY person_name
HAVING COUNT(tb_person.person_name) > 1

"""

cursor.execute(query)
rows = cursor.fetchall()
for row in rows:

```

```
print(row)
```

```
('Francis Ford Coppola', 3)
('Harrison Ford', 3)
('George Lucas', 3)
('Steven Spielberg', 5)
('Alfred Joseph Hitchcock', 3)
('Emilio Martinez Lazaro', 3)
('Dani Rovira', 2)
('Clara Lago', 2)
('Carmen Machi', 2)
('Karra Elejalde', 2)
('Daniel Sanchez Arevalo', 2)
('Mel Gibson', 2)
```

### Exercici 5

Has de crear un nou gènere anomenat “Documental” el qual tingui com a identificador el nombre 69.

```
[7]: query = """
INSERT INTO tb_genre (genre_id, genre_name)
VALUES (69, 'Documental');

"""
cursor.execute(query)
```

```
[8]: query = """
Select genre_id, genre_name
From tb_genre

"""
cursor.execute(query)
rows = cursor.fetchall()
for row in rows:
    print(row)
```

```
(1, 'Acción')
(2, 'Ciencia Ficción')
(3, 'Comedia')
(4, 'Drama')
(5, 'Fantasía')
(6, 'Melodrama')
(7, 'Musical')
(8, 'Romance')
(9, 'Suspense')
(10, 'Terror')
```

```
(11, 'Bélico')
(69, 'Documental')
```

### Exercici 6

Elimina la pel·lícula “La Gran Familia Española” de la base de dades.

```
[9]: query = """
DELETE FROM tb_movie_person
WHERE movie_id = 11;
"""

cursor.execute(query)

query = """
DELETE FROM tb_movie
WHERE movie_title = "La Gran Familia Española";
"""

cursor.execute(query)
```

```
[10]: query = """
Select movie_title
From tb_movie

"""

cursor.execute(query)
rows = cursor.fetchall()
for row in rows:
    print(row)
```

```
('Apocalypse Now',)
('Star Wars:Episode IV - A New Hope',)
('Indiana Jones and the Temple of Doom',)
('The Terminal',)
('Jaws',)
('ET The Extraterrestrial',)
('Psycho',)
('Ocho Apellidos Vascos',)
('Ocho Apellidos Catalanes',)
('El otro lado de la cama',)
('El dia de la bestia',)
('Braveheart',)
('The Shawshank Redemption',)
('Las brujas de Zugarramurdi',)
('Blade Runner',)
```

### Exercici 7

Canvia el gènere de la pel·lícula “Ocho apellidos catalanes” perquè consti com a comèdia i no com a romàntica.

```
[11]: query = """
UPDATE tb_movie
SET movie_genre_id = 3
WHERE movie_title = 'Ocho apellidos catalanes';
"""
cursor.execute(query)
```

```
[12]: query = """
Select movie_title, genre_id, genre_name
From tb_movie
LEFT JOIN tb_genre ON tb_movie.movie_genre_id = tb_genre.genre_id

"""
cursor.execute(query)
rows = cursor.fetchall()
for row in rows:
    print(row)
```

```
('Apocalypse Now', 11, 'Bélico')
('Star Wars:Episode IV - A New Hope', 2, 'Ciencia Ficción')
('Indiana Jones and the Temple of Doom', 1, 'Acción')
('The Terminal', 3, 'Comedia')
('Jaws', 10, 'Terror')
('ET The Extraterrestrial', 5, 'Fantasía')
('Psycho', 9, 'Suspense')
('Ocho Apellidos Vascos', 3, 'Comedia')
('Ocho Apellidos Catalanes', 3, 'Comedia')
('El otro lado de la cama', 8, 'Romance')
('El dia de la bestia', 1, 'Acción')
('Braveheart', 4, 'Drama')
('The Shawshank Redemption', 4, 'Drama')
('Las brujas de Zugarramurdi', 9, 'Suspense')
('Blade Runner', 2, 'Ciencia Ficción')
```

Els canvis introduïts a la base de dades, realment no s'han produït en mysql, cal exectur el codi següent:

```
db.commit()
db.close()
```