SQL Assignment

In [1]:

```
import pandas as pd
import sqlite3
```

In [4]:

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call
drive.mount("/content/drive", force_remount=True).
```

In [34]:

```
conn = sqlite3.connect("/content/drive/My Drive/sql_ass/old/Db-IMDB.db")
```

In [6]:

```
pd.set_option('display.max_columns', None)
```

Sample Code

In [ ]:

```
%%time
# Write your sql query below

query = """
        SELECT TRIM(Movie.title) AS 'Movie_Name'
        FROM Movie
        WHERE Movie.rating < 3

        """

q = pd.read_sql_query(query, conn)
print(q.shape)
q.head()
```

```
(85, 1)
Wall time: 57.8 ms
```

Out[ ]:

	Movie_Name
0	Mastizaade
1	Dragonball Evolution
2	Loveyatri
3	Race 3
4	Gunday

Q1 --- List all the directors who directed a 'Comedy' movie in a leap year. (You need to check that the genre is 'Comedy’ and year is a leap year) Your query should return director name, the movie name, and the year.

In [7]:

```
%%time
# Write your sql query below

query = """SELECT p.Name Director,m.title Movie_Title,m.year Year,r.Name Genre
FROM Movie m , M_Director md,Genre r,M_Genre gg,Person p
ON m.MID = gg.MID AND m.MID = md.MID AND r.Name LIKE '%Comedy%' AND md.PID=p.PID
AND m.year%4=0 group by p.Name,m.title
        """

q1 = pd.read_sql_query(query, conn)
print(q1.shape)
print(q1.head())
```

```
(946, 4)
           Director            Movie_Title    Year  \
0      A. Bhimsingh                  Aadmi    1968
1      A. Bhimsingh        Joroo Ka Ghulam    1972
2      A. Bhimsingh     Sadhu Aur Shaitaan    1968
3          A. Muthu  Tera Jadoo Chal Gayaa    2000
4   A.R. Murugadoss                  Akira  I 2016

                                 Genre
0  Comedy, Horror, Musical
1  Comedy, Horror, Musical
2  Comedy, Horror, Musical
3  Comedy, Horror, Musical
4  Comedy, Horror, Musical
CPU times: user 274 ms, sys: 8.66 ms, total: 283 ms
Wall time: 290 ms
```

Q2 --- List the names of all the actors who played in the movie 'Anand' (1971)

In [10]:

```
%%time
# Write your sql query below

query = """
        SELECT Name Actor from Person p JOIN M_Cast c ON TRIM(p.PID) = TRIM(c.PID) WHERE MID IN
        (SELECT MID from Movie WHERE title = 'Anand')
        """

q2 = pd.read_sql_query(query, conn)
print(q2.shape)
print(q2)
```

```
(17, 1)
                 Actor
0        Rajesh Khanna
1     Amitabh Bachchan
2        Sumita Sanyal
3          Ramesh Deo
4           Seema Deo
5       Asit Kumar Sen
6          Dev Kishan
7        Atam Prakash
8        Lalita Kumari
9              Savita
10      Brahm Bhardwaj
11        Gurnam Singh
12        Lalita Pawar
13         Durga Khote
14          Dara Singh
15       Johnny Walker
16           Moolchand
CPU times: user 127 ms, sys: 4.14 ms, total: 131 ms
Wall time: 135 ms
```

Q3 --- List all the actors who acted in a film before 1970 and in a film after 1990. (That is: < 1970 and > 1990.)

In [12]:

```
%%time
# Write your sql query below

query = """SELECT name Actor FROM Person WHERE TRIM(PID) IN
(SELECT TRIM(PID) FROM M_Cast WHERE MID IN
(SELECT MID FROM Movie m1 WHERE m1.year > 1990)
AND PID IN (SELECT PID FROM M_Cast WHERE MID IN
(SELECT MID FROM Movie m2 WHERE m2.year < 1970)))"""

q3 = pd.read_sql_query(query, conn)
print(q3.shape)
print(q3.head())
```

```
(333, 1)
               Actor
0       Rishi Kapoor
1   Amitabh Bachchan
2             Asrani
3       Zohra Sehgal
4    Parikshat Sahni
CPU times: user 116 ms, sys: 3.85 ms, total: 119 ms
Wall time: 121 ms
```

Q4 --- List all directors who directed 10 movies or more, in descending order of the number of movies they directed. Return the directors' names and the number of movies each of them directed.

In [15]:

```
%%time
# Write your sql query below

query = """SELECT DISTINCT p.Name Director,COUNT(*) movies FROM Person p
JOIN M_Director d on TRIM(p.PID) = TRIM(d.PID)
GROUP BY TRIM(d.PID) HAVING COUNT(*) >=10 ORDER BY movies DESC"""

q4 = pd.read_sql_query(query, conn)
print(q4.shape)
print(q4)
```

```
(58, 2)
                     Director  movies
0                David Dhawan      39
1                Mahesh Bhatt      35
2                Priyadarshan      30
3             Ram Gopal Varma      30
4                Vikram Bhatt      29
5         Hrishikesh Mukherjee      27
6                 Yash Chopra      21
7             Basu Chatterjee      19
8              Shakti Samanta      19
9                Subhash Ghai      18
10              Shyam Benegal      17
11     Abbas Alibhai Burmawalla      17
12            Rama Rao Tatineni      17
13             Manmohan Desai      16
14                     Gulzar      16
15               Raj N. Sippy      16
16                 Raj Kanwar      15
17           Mahesh Manjrekar      15
18                Indra Kumar      14
19                 Raj Khosla      14
20               Rahul Rawail      14
21           Rajkumar Santoshi      14
22              Rakesh Roshan      13
23                  Dev Anand      13
24                Vijay Anand      13
```

```
25               Harry Baweja      13
26             Anurag Kashyap      13
27   Ananth Narayan Mahadevan      13
28         K. Raghavendra Rao      13
29               Anees Bazmee      12
30               Guddu Dhanoa      12
31                Prakash Jha      12
32             Satish Kaushik      12
33            Nagesh Kukunoor      12
34              Prakash Mehra      12
35                Umesh Mehra      12
36                Anil Sharma      12
37          Madhur Bhandarkar      12
38               Rohit Shetty      12
39         Pramod Chakravorty      11
40               Sanjay Gupta      11
41              Nasir Hussain      11
42                Ketan Mehta      11
43            Govind Nihalani      11
44                 Mohit Suri      11
45                 Raj Kapoor      10
46                 K. Bapaiah      10
47            Vishal Bhardwaj      10
48                 N. Chandra      10
49           Tigmanshu Dhulia      10
50                 J.P. Dutta      10
51                Mehul Kumar      10
52               Hansal Mehta      10
53              Sudhir Mishra      10
54        K. Muralimohana Rao      10
55            Pankaj Parashar      10
56              J. Om Prakash      10
57                  Bimal Roy      10
CPU times: user 21.6 s, sys: 2.93 ms, total: 21.6 s
Wall time: 21.6 s
```

Q5.a --- For each year, count the number of movies in that year that had only female actors.

In [16]:

```
%%time
# Write your sql query below

query = """SELECT m.year Year,count(*) Count FROM Movie m
WHERE NOT EXISTS
(SELECT * FROM M_Cast c ,Person p WHERE p.gender='Male' and c.MID = m.MID
and c.PID = p.PID ) GROUP BY m.year"""

q5a = pd.read_sql_query(query, conn)
print(q5a.shape)
print(q5a)
```

```
(125, 2)
          Year  Count
0         1931      1
1         1936      3
2         1939      2
3         1941      1
4         1943      1
..         ...    ...
120    IV 2011      1
121    IV 2017      1
122     V 2015      1
123    VI 2015      1
124  XVII 2016      1

[125 rows x 2 columns]
CPU times: user 20 s, sys: 2.53 s, total: 22.5 s
Wall time: 22.5 s
```

Q5.b --- Now include a small change: report for each year the percentage of movies in that year with only female actors, and the total number of movies made that year. For example, one answer will be: 1990 31.81 13522 meaning that in 1990 there were 13,522 movies, and 31.81% had only female actors. You do not need to round your answer.

In [18]:

```
%%time
# Write your sql query below

query = """SELECT no_of_females.year Year,
((no_of_females.Total_movies_with_only_female_leads)*100)/total_count.Total Percentage FROM
((SELECT m.year Year,count(*) Total_movies_with_only_female_leads FROM movie m WHERE NOT EXISTS
( SELECT * FROM M_Cast c , person p WHERE c.mid = m.MID and c.PID = p.PID AND p.gender='Male' )
GROUP BY m.year) no_of_females,
(SELECT m.year,count(*) as Total FROM movie m group by m.year) total_count)
WHERE no_of_females.year=total_count.year"""

q5b = pd.read_sql_query(query, conn)
print(q5b.shape)
print(q5b)
```

```
(125, 2)
          Year  Percentage
0         1931         100
1         1936         100
2         1939         100
3         1941         100
4         1943         100
..         ...         ...
120    IV 2011         100
121    IV 2017         100
122     V 2015         100
123    VI 2015         100
124  XVII 2016         100

[125 rows x 2 columns]
CPU times: user 20.1 s, sys: 3.54 s, total: 23.7 s
Wall time: 23.7 s
```

Q6 --- Find the film(s) with the largest cast. Return the movie title and the size of the cast. By "cast size" we mean the number of distinct actors that played in that movie: if an actor played multiple roles, or if it simply occurs multiple times in casts, we still count her/him only once.

In [20]:

```
%%time
# Write your sql query below

query = """SELECT m.title Movie_Title,count(distinct(c.PID)) Cast_Size FROM Movie m JOIN M_Cast c
ON c.MID = m.MID GROUP BY m.MID ORDER BY Cast_Size desc"""

q6 = pd.read_sql_query(query, conn)
print(q6.shape)
print(q6.head())
```

```
(3475, 2)
                 Movie_Title  Cast_Size
0               Ocean's Eight        238
1                    Apaharan        233
2                        Gold        215
3              My Name Is Khan        213
4  Captain America: Civil War        191
CPU times: user 182 ms, sys: 6.99 ms, total: 189 ms
Wall time: 191 ms
```

Q7 --- A decade is a sequence of 10 consecutive years. For example, say in

Q7 --- A decade is a sequence of 10 consecutive years. For example, say in your database you have movie information starting from 1965. Then the first decade is 1965, 1966, ..., 1974; the second one is 1967, 1968, ..., 1976 and so on. Find the decade D with the largest number of films and the total number of films in D.

In [37]:

```
%%time
# Write your sql query below

query = """SELECT m1.year Start, m1.year+9 End, count(*) films FROM
(SELECT DISTINCT year from Movie) m1 JOIN Movie m2 ON m2.year >= Start and m2.year<= End
GROUP BY End ORDER BY films desc LIMIT 1"""

q7 = pd.read_sql_query(query, conn)
print(q7.shape)
print(q7.head())
```

```
(1, 3)
  Start   End  films
0  2008  2017   1128
CPU times: user 96.9 ms, sys: 47 µs, total: 96.9 ms
Wall time: 106 ms
```

Q8 --- Find all the actors that made more movies with Yash Chopra than any other director.

In [36]:

```
%%time
# Write your sql query below

query = """SELECT DISTINCT  Actor, Count(*) Movies_with_YashChopra
FROM(SELECT DISTINCT p1.Name as Director, m1.title as Movie
FROM Person p1 Inner Join M_Director md on TRIM(md.PID)=p1.PID
Inner Join Movie m1 on TRIM(md.MID)=m1.MID and  p1.Name LIKE 'Yash%' Group By p1.Name, m1.title) t
1
Inner Join  (SELECT DISTINCT p2.Name as Actor,m2.title as Movie from Person p2
Inner Join M_Cast mc on TRIM(mc.PID)=p2.PID
Inner Join Movie m2 on TRIM(mc.MID)=m2.MID Group By p2.Name, m2.title) t2 on t1.Movie=t2.Movie
Group By t2.Actor Order By Movies_with_YashChopra DESC"""

q8 = pd.read_sql_query(query, conn)
print(q8.shape)
print(q8.head())
```

```
(514, 2)
               Actor  Movies_with_YashChopra
0        Jagdish Raj                      11
1   Manmohan Krishna                      10
2   Manmohan Krishna                      10
3          Iftekhar                        9
4         Madan Puri                       8
CPU times: user 633 ms, sys: 69.9 ms, total: 703 ms
Wall time: 710 ms
```

Q9 --- The Shahrukh number of an actor is the length of the shortest path between the actor and Shahrukh Khan in the "co-acting" graph. That is, Shahrukh Khan has Shahrukh number 0; all actors who acted in the same film as Shahrukh have Shahrukh number 1; all actors who acted in the same film as some actor with Shahrukh number 1 have Shahrukh number 2, etc. Return all actors whose Shahrukh number is 2.

In [35]:

```
%%time
# Write your sql query below
```

```
query = """SELECT DISTINCT TRIM(name) Name
FROM Person p INNER JOIN M_Cast c on p.PID = TRIM(c.PID) INNER JOIN Movie m ON m.MID = c.MID AND T
RIM(p.Name)!='Shah Rukh Khan'
and m.title in (SELECT DISTINCT title FROM Person p3 INNER JOIN M_Cast c3 on p3.PID = TRIM(c3.PID)
AND TRIM(p3.Name) = p3.Name
INNER JOIN Movie m3 ON m3.MID = c3.MID AND p3.Name IN (SELECT DISTINCT Name FROM Person p2 INNER J
OIN M_Cast c2 ON p2.PID = TRIM(c2.PID)
INNER JOIN Movie m2 ON m2.MID = c2.MID AND TRIM(p2.Name)!='Shah Rukh Khan' AND m2.title IN
(SELECT DISTINCT title FROM Person p3 INNER JOIN M_Cast c3 ON p3.PID = TRIM(c3.PID) AND TRIM(p3.Na
me) = 'Shah Rukh Khan'
INNER JOIN Movie m3 ON m3.MID = c3.MID))) ORDER BY Name"""

q9 = pd.read_sql_query(query, conn)
print(q9.shape)
print(q9.head())
```

```
(16165, 1)
                         Name
0  'Musafir' Radio Performing
1              A'Ali de Sousa
2             A. Abdul Hameed
3                   A. Darpan
4                   A. Gabibi
CPU times: user 751 ms, sys: 56.4 ms, total: 808 ms
Wall time: 827 ms
```