In [1]:

```
import sqlite3
import pandas as pd

conn = sqlite3.connect('Db-IMDB.db')
```

In [2]:

```
sql = """ UPDATE genre SET name=trim(name) """
cur = conn.cursor()
cur.execute(sql)
conn.commit()
```

In [3]:

```
#Questions for the assignment
#https://drive.google.com/file/d/1b1Wd29cjNSszHP8OeK__GBohxyuBSt0r/view?usp=sharing
```

1) 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 [4]:

```
query = pd.read_sql_query('''select distinct p.name, m.title, m.year from person p, movie m join
    (select md.pid as pid, mg.mid as mid from m_director md join m_genre mg on mg.mid=md.mid
    where mg.gid IN
    (select g.gid from genre g where g.name like "%Comedy%")) as temp on temp.pid=p.pid and
temp.mid=m.mid group by m.title
    having m.year%4==0 and m.year%100!=0 or m.year%400==0''', conn)
query
```

Out[4]:

	Name	title	year
0	Rabi Kinagi	100% Love	2012
1	Ajai Sinha	3 Bachelors	2012
2	Remo D'Souza	A Flying Jatt	2016
3	Pankaj Parashar	Ab Ayega Mazaa	1984
4	Sachin Kundalkar	Aiyyaa	2012
5	Saeed Akhtar Mirza	Albert Pinto Ko Gussa Kyon Ata Hai	1980
6	Basu Chatterjee	Apne Paraye	1980
7	Frank Coraci	Around the World in 80 Days	2004
8	Rajpal Yadav	Ata Pata Lapatta	2012
9	Nitya Mehra	Baar Baar Dekho	2016
10	Salim Raza	Bach ke Zara	2008
11	Siddharth Anand	Bachna Ae Haseeno	2008
12	Tarun Majumdar	Balika Badhu	1976
13	Anurag Basu	Barfi!	2012
14	Hrishikesh Mukherjee	Bawarchi	1972
15	Mike Judge	Beavis and Butt-Head Do America	1996
16	Aditya Chopra	Befikre	2016
17	Rama Rao Tatineni	Beti No. 1	2000
18	Vivek Sharma	Bhoothnath	2008

	Name	Title	year
19	J.K. Bihari	Biwi Ho To Aisi	1988
20	Rohit Shetty	Bol Bachchan	2012
21	David Dhawan	Bol Radha Bol	1992
22	Brij	Bombay 405 Miles	1980
23	Nagesh Kukunoor	Bombay to Bangkok	2008
24	S. Ramanathan	Bombay to Goa	1972
25	Bhappi Sonie	Brahmachari	1968
26	Gurinder Chadha	Bride & Prejudice	2004
27	Jagdish Rajpurohit	Bumboo	2012
28	Sachin Yardi	C Kkompany	2008
29	Hriday Shetty	Chaalis Chauraasi	2012
...	...	...	...
199	Ramesh Sippy	Seeta Aur Geeta	1972
200	Raj Kaushal	Shaadi Ka Laddoo	2004
201	Prakash Mehra	Sharaabi	1984
202	Anees Bazmee	Singh Is Kinng	2008
203	Ashwani Dhir	Son of Sardaar	2012
204	Onir	Sorry Bhai!	2008
205	Karan Johar	Student of the Year	2012
206	Rohit Shetty	Sunday	2008
207	Vimal Kumar	Suno Sasurjee	2004
208	P. Sambasiva Rao	Swayamvar	1980
209	Oliver Paulus	Tandoori Love	2008
210	T.L.V. Prasad	Tauba Tauba	2004
211	Abhishek Sharma	Tere Bin Laden: Dead Or Alive	2016
212	Joy Augustine	Tere Mere Sapne	1996
213	Mandeep Kumar	Tere Naal Love Ho Gaya	2012
214	Kunal Kohli	Teri Meri Kahaani	2012
215	Griffin Dunne	The Accidental Husband	2008
216	James Dodson	The Other End of the Line	2008
217	Sunil K. Reddy	Thikka	2016
218	Kunal Kohli	Thoda Pyaar Thoda Magic	2008
219	Guddu Dhanoa	Tu Chor Main Sipahi	1996
220	Milind Dhaimade	Tu Hai Mera Sunday	2016
221	Vijay	Tutak Tutak Tutiya	2016
222	Sachin Kamlakar Khot	Ugly Aur Pagli	2008
223	Shoojit Sircar	Vicky Donor	2012
224	Brij	Victoria No. 203	1972
225	Shyam Benegal	Welcome to Sajjanpur	2008
226	Aditya Datt	Will You Marry Me	2012
227	Deepak Anand	Yaad Rakhegi Duniya	1992
228	Subhash Ghai	Yuvvraaj	2008

229 rows × 3 columns

In [5]:

```
act = pd.read_sql_query('''select distinct p.name from person p''',conn)
act
```

Out[5]:

Name

	Name
0	Christian Bale
1	Cate Blanchett
2	Benedict Cumberbatch
3	Naomie Harris
4	Andy Serkis
5	Peter Mullan
6	Jack Reynor
7	Eddie Marsan
8	Tom Hollander
9	Matthew Rhys
10	Freida Pinto
11	Rohan Chand
12	Keveshan Pillay
13	Louis Ashbourne Serkis
14	Moonsamy Narasigadu
15	Soobrie Govender
16	Gopal Singh
17	Kista Munsami
18	Mahomed Araf Cassim
19	Riaz Mansoor
20	Roshan Jayesh Patel
21	T'khai Phillips
22	Sachin Soni
23	Hridhay Somera
24	Ethaniel Jaden Moonsamy
25	Gareth Ryan Benjamin
26	Nirvayesh Chakravorty Thanendra
27	Adiyan Ahmed Choudhury
28	Amara Motala
29	Diyara Prakash
...	...
35513	Rakesh Chaturvedi
35514	Swapna Joshi
35515	Shukla Barnali Ray
35516	Pavithran
35517	Vara Mullapoodi
35518	D. Sumana Kittur
35519	Abhishek Chhadha
35520	Arup Dutta
35521	Illangkannan
35522	Visakh G S
35523	Sandip Ray
35524	S.V. Krishna Reddy
35525	R.K. Selvamani
35526	Amma Rajasekhar
35527	Sanjay Talreja
35528	Rajatesh Nayyar
35529	Murali Nair
35530	Pryas Gupta
35531	Shivamani

	Name
35532	Oliver Paulus
35533	Vishal Inamdar
35534	Kumar Shahani
35535	Avtandil Varsimashvili
35536	G. Ram Prasad
35537	Raja Chanda
35538	Deepak Ramteke
35539	Kamika Verma
35540	Dhorairaj Bhagavan
35541	Nasir Shaikh
35542	Adrian Fulle

35543 rows × 1 columns

In [6]:

```
sql = """ UPDATE m_cast SET pid=trim(pid) """
cur = conn.cursor()
cur.execute(sql)
conn.commit()
```

2. List the names of all the actors who played in the movie 'Anand' (1971)

In [7]:

```
df = pd.read_sql_query('''SELECT name from Person where pid IN
                        (SELECT pid FROM m_cast where mid IN
                        (SELECT mid from movie where title = 'Anand'));''', conn)
```

In [8]:

```
print("all the actors who played in the movie 'Anand' (1971)")
df
```

all the actors who played in the movie 'Anand' (1971)

Out[8]:

	Name
0	Amitabh Bachchan
1	Rajesh Khanna
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	Name

In [9]:

```
sql = '''UPDATE Movie SET year=SUBSTR(year,-4) WHERE year LIKE '% %' '''
cur = conn.cursor()
cur.execute(sql)
conn.commit()
```

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

In [10]:

```
actors = pd.read_sql_query('''SELECT Name from Person WHERE PID NOT IN
(SELECT p.PID from Person p, Movie m
INNER JOIN M_Cast mc ON mc.MID = m.MID AND mc.PID = p.PID
WHERE CAST(m.year AS INTEGER) BETWEEN 1970 AND 1990)
AND PID IN (SELECT p1.PID from Person p1, Movie m1
INNER JOIN M_Cast mc1 ON mc1.MID = m1.MID AND mc1.PID = p1.PID
WHERE CAST(m1.year AS INTEGER) < 1970
INTERSECT
SELECT p2.PID from Person p2, Movie m2
INNER JOIN M_Cast mc2 ON mc2.MID = m2.MID AND mc2.PID = p2.PID
WHERE CAST(m2.year AS INTEGER) > 1990)''',conn)
actors
```

Out[10]:

	Name
0	Zohra Sehgal
1	Rakesh Sharma
2	Ric Young
3	Jeremy Child
4	Zul Vellani
5	Sitaram
6	Gopal Dutt
7	P. Kumar
8	Rohit Chopra
9	Sivakumar
10	Saraswathi
11	Begum Para
12	Nilofar
13	Saroj Khan
14	Sudarshan
15	Desai
16	Sudhir Kumar
17	Gani
18	Prem Prakash
19	Raj Dutt
20	Veera
21	Rakesh
22	Dilip Raj
23	Salim Khan
24	Manohar Deepak
25	Shamsher Singh
26	Rajshree

	Name
27	Raj Kamal
28	Kalpana Kartik
29	Brij Bhushan
30	Mohamad Ali
31	Master Amar
32	Manish
33	Lata Mangeshkar
34	Gummadi
35	Kaveri
36	Ismail
37	Miss Firoza
38	Dube
39	Shekhar
40	Jamila Massey
41	Suryakantham

4. 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 [11]:

```
query = '''
    select a.name, count(b.mid) num_movies from person a left join m_director b
    on a.pid=b.pid group by a.pid having num_movies>10 order by num_movies desc
'''
result = pd.read_sql_query(query, conn)
result
```

Out[11]:

	Name	num_movies
0	David Dhawan	78
1	Mahesh Bhatt	70
2	Ram Gopal Varma	60
3	Vikram Bhatt	58
4	Hrishikesh Mukherjee	54
5	Yash Chopra	42
6	Basu Chatterjee	38
7	Shakti Samanta	38
8	Subhash Ghai	36
9	Shyam Benegal	34
10	Abbas Alibhai Burmawalla	34
11	Manmohan Desai	32
12	Gulzar	32
13	Raj N. Sippy	32
14	Raj Kanwar	30
15	Mahesh Manjrekar	30
16	Priyadarshan	30
17	Indra Kumar	28
18	Raj Khosla	28
19	Rahul Rawail	28
20	Rajkumar Santoshi	28

	Name	num_movies
21	Rakesh Roshan	26
22	Dev Anand	26
23	Vijay Anand	26
24	Harry Baweja	26
25	Anurag Kashyap	26
26	Ananth Narayan Mahadevan	26
27	K. Raghavendra Rao	26
28	Anees Bazmee	24
29	Guddu Dhanoa	24
...	...	...
82	Puri Jagannadh	14
83	Sangeeth Sivan	14
84	K. Viswanath	14
85	Ashu Trikha	14
86	Imtiaz Ali	14
87	Saawan Kumar Tak	14
88	Manoj Kumar	12
89	Deepak Anand	12
90	Aparna Sen	12
91	Sooraj R. Barjatya	12
92	Brij	12
93	Ravi Chopra	12
94	Rituparno Ghosh	12
95	Karan Johar	12
96	Sohail Khan	12
97	Sandesh Kohli	12
98	Mohan Kumar	12
99	Jag Mundhra	12
100	K. Ravi Shankar	12
101	Shantaram Rajaram Vankudre	12
102	Babbar Subhash	12
103	Deepak Tijori	12
104	Onir	12
105	Vipul Amrutlal Shah	12
106	Kabir Khan	12
107	Shaad Ali	12
108	Samir Karnik	12
109	A.R. Murugadoss	12
110	Vivek Agnihotri	12
111	Nila Madhab Panda	12

112 rows × 2 columns

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

In [12]:

```
fr = pd.read_sql_query('''select z.year, count(*)
from Movie z
where not exists (select *
                from Person x, M_Cast xy
```

```
                                    where x.PID = xy.PID and xy.MID = z.MID and x.Gender!='Female')
group by z.year''', conn)
fr
```

Out[12]:

	year	count(*)
0	1939	1
1	1999	1
2	2000	1
3	2009	1
4	2012	1
5	2018	2

5. 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 [13]:

```
perc_query = pd.read_sql_query('''select a.year, a.c*100.00/b.c as percentage, b.c as
total_overall
from (select z.year, count(*) as c
      from Movie z
      where not exists (select *
                        from Person x, M_Cast xy
                        where x.PID = xy.PID and xy.MID = z.MID and x.Gender!='Female')
      group by z.year) a,
      (select z.year, count(*) as c from Movie z group by z.year) b
where a.year=b.year
order by a.year''', conn)
perc_query
```

Out[13]:

	year	percentage	total_overall
0	1939	50.000000	2
1	1999	1.515152	66
2	2000	1.562500	64
3	2009	0.909091	110
4	2012	0.900901	111
5	2018	1.923077	104

6. 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 [14]:

```
l_cast_query = pd.read_sql_query('''select x.title, count(distinct xy.PID) as c
from Movie x, M_Cast xy
where x.MID = xy.MID
group by x.MID, x.title
having not exists (select uv.MID
                   from M_Cast uv
                   group by uv.MID
                   having count(distinct uv.PID) > count(distinct xy.PID))''', conn)
l_cast_query
```

	title	c
0	Ocean's Eight	238

7. 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 [15]:

```
decade_query = pd.read_sql_query('''select y.year, count(*)
from (select distinct x.year from Movie x) y,
     Movie z
where y.year <= z.year and z.year < y.year+10
group by y.year
having not exists (select y1.year
                   from (select distinct x1.year from Movie x1) y1, Movie z1
                   where y1.year <= z1.year and z1.year < y1.year+10
                   group by y1.year
                   having count(z1.MID) > count(z.MID))''', conn)
decade_query
```

Out[15]:

	year	count(*)
0	2008	1205

8. Find the actors that were never unemployed for more than 3 years at a stretch. (Assume that the actors remain unemployed between two consecutive movies).

In [16]:

```
unemployed_query = pd.read_sql_query('''select Name from Person where PID
    not in(select distinct(PID)from M_Cast as C1 natural join Movie as
    M1 where exists(select MID from M_Cast as C2 natural join Movie as M2 where
    C1.PID = C2.PID and (M2.year - 3) > M1.year and
    not exists(select MID from M_Cast as C3 natural join
    Movie as M3 where C1.PID = C3.PID and M1.year < M3.year and M3.year < M2.year)))''', conn)
unemployed_query
```

Out[16]:

	Name
0	Christian Bale
1	Cate Blanchett
2	Benedict Cumberbatch
3	Naomie Harris
4	Andy Serkis
5	Peter Mullan
6	Jack Reynor
7	Eddie Marsan
8	Tom Hollander
9	Matthew Rhys

	Name
10	Freida Pinto
11	Rohan Chand
12	Keveshan Pillay
13	Louis Ashbourne Serkis
14	Moonsamy Narasigadu
15	Soobrie Govender
16	Gopal Singh
17	Kista Munsami
18	Mahomed Araf Cassim
19	Riaz Mansoor
20	Roshan Jayesh Patel
21	T'khai Phillips
22	Sachin Soni
23	Hridhay Somera
24	Ethaniel Jaden Moonsamy
25	Gareth Ryan Benjamin
26	Nirvayesh Chakravorty Thanendra
27	Adiyan Ahmed Choudhury
28	Amara Motala
29	Diyara Prakash
...	...
38255	Sandip Ray
38256	S.V. Krishna Reddy
38257	R.K. Selvamani
38258	Amma Rajasekhar
38259	Rahat Kazmi
38260	Rohit Gupta
38261	Bela Negi
38262	Sanjay Talreja
38263	Rajatesh Nayyar
38264	Murali Nair
38265	Pryas Gupta
38266	Shivamani
38267	Oliver Paulus
38268	Vishal Inamdar
38269	Kumar Shahani
38270	Ka-Fai Wai
38271	Avtandil Varsimashvili
38272	G. Ram Prasad
38273	Raja Chanda
38274	Deepak Ramteke
38275	Srinivas Sunderrajan
38276	Kamika Verma
38277	Dhorairaj Bhagavan
38278	Nasir Shaikh
38279	Abbas
38280	Kannan
38281	Adrian Fulle
38282	Gulshan Kumar
38283	Iqbal
38284	Sushma Shiromani

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

In [17]:

```
# # #number of movies between an actor and director
# act =  pd.read_sql_query( '''SELECT pd.name as Director, pa.name as actor, COUNT(*)
# FROM M_director d JOIN
# Person pd
# ON pd.pid = d.pid JOIN
# M_cast c
# ON c.MID = d.MID JOIN
# Person pa
# ON pa.PID = c.PID
# GROUP BY pd.name, pa.name''',conn)
# act
```

In [18]:

```
actors = pd.read_sql_query('''SELECT da.*
    FROM (SELECT pd.pid, pd.name as Director, pa.pid, pa.name as actor, COUNT(*) as cnt,
            RANK() OVER (PARTITION BY pa.pid ORDER BY COUNT(*) DESC) as seqnum
    FROM M_director d JOIN
            Person pd
            ON pd.pid = d.pid JOIN
            M_cast c
            ON c.MID = d.MID JOIN
            Person pa
            ON pa.PID = c.PID
     GROUP BY pd.pid, pd.name, pa.pid, pa.name
     ) da
    WHERE director = 'Yash Chopra' AND
     seqnum = 1 ''',conn)
actors
```

Out[18]:

	pid	Director	pid:1	actor	cnt	seqnum
0	nm0007181	Yash Chopra	nm0004434	Shashi Kapoor	28	1
1	nm0007181	Yash Chopra	nm0007181	Yash Chopra	8	1
2	nm0007181	Yash Chopra	nm0015296	Akhtar-Ul-Iman	2	1
3	nm0007181	Yash Chopra	nm0019463	Murad Ali	2	1
4	nm0007181	Yash Chopra	nm0046230	Badri Prasad	2	1
5	nm0007181	Yash Chopra	nm0049403	Balam	2	1
6	nm0007181	Yash Chopra	nm0052570	Saira Banu	2	1
7	nm0007181	Yash Chopra	nm0080266	Raj Bharti	2	1
8	nm0007181	Yash Chopra	nm0080385	Ashwini Bhave	2	1
9	nm0007181	Yash Chopra	nm0081070	Andrew Bicknell	2	1
10	nm0007181	Yash Chopra	nm0085944	Paul Blackwell	2	1
11	nm0007181	Yash Chopra	nm0101508	Steve Box	2	1
12	nm0007181	Yash Chopra	nm0158332	Leela Chitnis	6	1

	pid	Chopra Director	pid:1	actor	cnt	seqnum
13	nm0007181	Yash Chopra	nm0159158	Pamela Chopra	2	1
14	nm0007181	Yash Chopra	nm0159165	Sudha Chopra	6	1
15	nm0007181	Yash Chopra	nm0223044	Karan Dewan	2	1
16	nm0007181	Yash Chopra	nm0223563	Poonam Dhillon	4	1
17	nm0007181	Yash Chopra	nm0239270	Satyadev Dubey	2	1
18	nm0007181	Yash Chopra	nm0307231	Simi Garewal	2	1
19	nm0007181	Yash Chopra	nm0347901	Rakhee Gulzar	10	1
20	nm0007181	Yash Chopra	nm0348507	Sumati Gupte	2	1
21	nm0007181	Yash Chopra	nm0407002	Iftekhar	18	1
22	nm0007181	Yash Chopra	nm0411327	Ismail	2	1
23	nm0007181	Yash Chopra	nm0415612	Gajanan Jagirdar	4	1
24	nm0007181	Yash Chopra	nm0424102	I.S. Johar	2	1
25	nm0007181	Yash Chopra	nm0433945	Ravikant	8	1
26	nm0007181	Yash Chopra	nm0441258	Kathana	2	1
27	nm0007181	Yash Chopra	nm0441321	Yashodra Katju	2	1
28	nm0007181	Yash Chopra	nm0451387	Padma Khanna	4	1
29	nm0007181	Yash Chopra	nm0464521	Poonam Sinha	2	1
...	...	...	...	...	...	...
217	nm0007181	Yash Chopra	nm5171679	Ricky Rajpal	2	1
218	nm0007181	Yash Chopra	nm5171778	Shane Salter	2	1
219	nm0007181	Yash Chopra	nm5171814	Narin Oz	2	1
220	nm0007181	Yash Chopra	nm5221129	Jasmine Jardot	2	1
221	nm0007181	Yash Chopra	nm5235227	Michael Patil	2	1
222	nm0007181	Yash Chopra	nm5270502	Barrie Martin	2	1
223	nm0007181	Yash Chopra	nm5276839	Morris Minelli	2	1
224	nm0007181	Yash Chopra	nm5330549	Steven Baker	2	1
225	nm0007181	Yash Chopra	nm5359722	Anick Wiget	2	1
226	nm0007181	Yash Chopra	nm5390815	Varun Thajur	2	1
227	nm0007181	Yash Chopra	nm5391624	Amarinder Sodhi	2	1
228	nm0007181	Yash Chopra	nm5424345	Amarinder Sodhi	2	1
229	nm0007181	Yash Chopra	nm5424365	Varun Thakur	2	1
230	nm0007181	Yash Chopra	nm5477035	Iola Nguyen	2	1
231	nm0007181	Yash Chopra	nm5483899	Stephanie Aslan	2	1

	pid	Chopra Director Yash Chopra	pid:1	actor	cnt	seqnum
232	nm0007181	Yash Chopra	nm5624434	Claire Greasley	2	1
233	nm0007181	Yash Chopra	nm5904446	Laura Bernardeschi	2	1
234	nm0007181	Yash Chopra	nm6189661	Magnus Emaka Einang	2	1
235	nm0007181	Yash Chopra	nm6217431	Martin Crossingham	2	1
236	nm0007181	Yash Chopra	nm6265025	David Clement-Horton	2	1
237	nm0007181	Yash Chopra	nm6308379	Taj Gill	2	1
238	nm0007181	Yash Chopra	nm7074701	Dolly Jagdeo	2	1
239	nm0007181	Yash Chopra	nm7076286	Katy Kartwheel	2	1
240	nm0007181	Yash Chopra	nm7150152	Abbie Murphy	2	1
241	nm0007181	Yash Chopra	nm7624361	Richard Broom	2	1
242	nm0007181	Yash Chopra	nm7760187	Saul George	6	1
243	nm0007181	Yash Chopra	nm7768629	Vinita Sharma	2	1
244	nm0007181	Yash Chopra	nm8533115	Sean Moon	2	1
245	nm0007181	Yash Chopra	nm8737993	Varun Thakur	2	1
246	nm0007181	Yash Chopra	nm9036653	Surendra Rahi	6	1

247 rows × 6 columns

10. The Shahrukh number of an actor is the length of the shortest path between the actor andShahrukh Khan in the "co-acting" graph. That is, Shahrukh Khan has Shahrukh number 0; allactors 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 [21]:

```
sql = """ UPDATE Person SET Name=trim(Name) """
cur = conn.cursor()
cur.execute(sql)
conn.commit()
```

In [22]:

```
srk_num_2_actors = pd.read_sql_query("""
WITH SHAHRUKH_0 AS
    (
        SELECT
            TRIM(P.PID) PID
        FROM
            Person P
        WHERE
            Trim(P.Name) like '%Shahrukh%'
    ),
    SHAHRUKH_1_MOVIES AS
    (
        SELECT
            DISTINCT
            TRIM(MC.MID) MID,
            S0.PID
        FROM
```

```
                M_Cast MC,
                SHAHRUKH_0 S0
            WHERE
                TRIM(MC.PID) = S0.PID
    ),
    SHAHRUKH_1_ACTORS AS
    (
        SELECT
            DISTINCT
            TRIM(MC.PID) PID
        FROM
            M_Cast MC,
            SHAHRUKH_1_MOVIES S1M
        WHERE
            TRIM(MC.MID) = S1M.MID AND
            TRIM(MC.PID) <> S1M.PID
    ),
    SHAHRUKH_2_MOVIES AS
    (
        SELECT
            DISTINCT
            TRIM(MC.MID) MID,
            S1A.PID
        FROM
            M_Cast MC,
            SHAHRUKH_1_ACTORS S1A
        WHERE
            TRIM(MC.PID) = S1A.PID
    )
    SELECT
        DISTINCT
        TRIM(MC.PID) PID,
        TRIM(P.Name) ACTOR_NAME
    FROM
        Person P,
        M_Cast MC,
        SHAHRUKH_2_MOVIES S2M
    WHERE
            TRIM(MC.PID) = TRIM(P.PID) AND
            TRIM(MC.MID) = S2M.MID AND
            TRIM(MC.PID) <> S2M.PID;

""", conn)
```

In [23]:

```
srk_num_2_actors
```

Out[23]:

	PID	ACTOR_NAME
0	nm2951768	Freida Pinto
1	nm6467532	Caroline Christl Long
2	nm6071249	Rajeev Pahuja
3	nm3491108	Michelle Santiago
4	nm7509518	Jandre le Roux
5	nm5951787	Raj Awasti
6	nm5525290	Michael Chapman
7	nm8232648	James Heron
8	nm7247557	Alex Jaep
9	nm6631007	Marian Lorencik
10	nm7255636	Celina Nessa
11	nm5721141	James Pimenta
12	nm4964257	M'laah Kaur Singh
13	nm0380073	Maximiliano Hernández
14	nm3630374	Sohum Shah

	PID	ACTOR_NAME
15	nm3708961	Deepak Damle
16	nm8334880	Piyush Kaushik
17	nm1390115	Harish Khanna
18	nm3818286	Sushant Singh Rajput
19	nm0080232	Nitish Bharadwaj
20	nm8644385	Lalu Makhija
21	nm6661769	Mir Sarwar
22	nm4731677	Ayushmann Khurrana
23	nm0007102	Tabu
24	nm2331000	Radhika Apte
25	nm0223521	Anil Dhawan
26	nm2435847	Manav Vij
27	nm1817397	Ashwini Kalsekar
28	nm3777127	Chhaya Kadam
29	nm1664541	Zakir Hussain
...	...	...
15881	nm2019990	Pradhan Manjari
15882	nm2021136	Poonam Jha
15883	nm1459053	Sunila Karambelkar
15884	nm1763457	Arup Ganguli
15885	nm1760405	Laxmi Patel
15886	nm1760399	Meena Pankaj
15887	nm1686154	Pratap
15888	nm2184586	Vidya Shenoy
15889	nm2182462	Jeetendra Khanna
15890	nm2177257	K.L. Sethi
15891	nm2084775	Malaika Shinoy
15892	nm2465676	Poonam Bajwa
15893	nm1688585	Kishin Punjabi
15894	nm1082198	Surjeet Redi
15895	nm0695939	Premji
15896	nm1693988	Kamu
15897	nm5578623	Monal
15898	nm1524755	Ushma Rathod
15899	nm1567918	Shilpi
15900	nm1946131	Zubeda Khan
15901	nm2519512	N. Sagar
15902	nm4042918	Habib Tanvar
15903	nm1881395	Mohd. Zahiruddin
15904	nm2522571	Muktha George
15905	nm0030135	Anjuman
15906	nm3099317	Dhruv Shetty
15907	nm2371614	Hayley Cleghorn
15908	nm2675737	Nirvasha Jithoo
15909	nm2370589	Kamal Maharshi
15910	nm1866356	Mohini Manik

15911 rows × 2 columns