prabhudayala@gmail.com_22

October 28, 2019

SQL Assignment

[1]:
```
import pandas as pd
```

[2]:
```
import sqlite3
con = sqlite3.connect('Db-IMDB.db')
```

1st question

[3]:
```
pd.read_sql_query('select distinct m.mid,p.pid,m.title,trim(p.name),m.year from␣
↪movie m join m_director dir on dir.mid=m.mid join person p on p.pid=dir.pid␣
↪join m_genre gen on gen.mid=m.mid join genre g on g.gid=gen.gid where g.name␣
↪like \'%comedy%\' and ((substr(m.year,-4)%4 = 0 and substr(m.year,-4)%100 <>␣
↪0) or substr(m.year,-4)%400 = 0)', con)
```

[3]:

	MID	PID	title	\
0	tt3726012	nm1179457	Mastizaade	
1	tt0366551	nm0500444	Harold & Kumar Go to White Castle	
2	tt1954470	nm0440604	Gangs of Wasseypur	
3	tt0327437	nm0178997	Around the World in 80 Days	
4	tt0809504	nm0001162	The Accidental Husband	
5	tt2082197	nm1397301	Barfi!	
6	tt0361411	nm0149446	Bride & Prejudice	
7	tt0115641	nm0431918	Beavis and Butt-Head Do America	
8	tt1185420	nm1160374	Dostana	
9	tt4900716	nm2439708	Kapoor & Sons	
10	tt1182937	nm0159147	Rab Ne Bana Di Jodi	
11	tt3679060	nm2539328	Dishoom	
12	tt5197544	nm1224429	Baar Baar Dekho	
13	tt1292703	nm1962313	Oye Lucky! Lucky Oye!	
14	tt2283748	nm1084488	OMG: Oh My God!	
15	tt5108476	nm0159147	Befikre	
16	tt2172071	nm0424103	Student of the Year	
17	tt0347473	nm0007134	Main Hoon Na	
18	tt2317337	nm1999473	Vicky Donor	
19	tt0473367	nm1063072	Jaane Tu... Ya Jaane Na	
20	tt0242519	nm0698184	Hera Pheri	
21	tt5997928	nm3672712	Hotel Salvation	
22	tt1049405	nm0230286	The Other End of the Line	
23	tt2168910	nm1223294	Cocktail	

24	tt0117878	nm0041928	Tere Mere Sapne
25	tt5472758	nm2414127	Happy Bhag Jayegi
26	tt2181931	nm1903006	English Vinglish
27	tt2258337	nm1442514	Eega
28	tt5235880	nm1150656	A Flying Jatt
29	tt5943306	nm0451326	Freaky Ali
..	...	...	...
202	tt2188811	nm4858383	Luv U Soniyo
203	tt2190256	nm0051091	Daal Mein Kuch Kaala Hai
204	tt0442799	nm1323642	Kis Kis Ki Kismat
205	tt0049549	nm0781830	New Delhi
206	tt0178211	nm0660967	Ab Ayega Mazaa
207	tt0237132	nm0538306	Dadar Kirti
208	tt1825665	nm2579028	Bach ke Zara
209	tt0251067	nm0665374	Ek Hota Vidushak
210	tt1132595	nm0155027	Maan Gaye Mughall-E-Azam
211	tt1132606	nm2818464	Ugly Aur Pagli
212	tt0172188	nm0223522	Bol Radha Bol
213	tt0231646	nm0436202	Ghar Ghar Ki Kahani
214	tt0402606	nm0474909	Suno Sasurjee
215	tt0396782	nm1394996	Paisa Vasool
216	tt0390079	nm1212514	Hari Om
217	tt0060317	nm1455833	Do Dooni Char
218	tt0414503	nm0442472	Shaadi Ka Laddoo
219	tt0430480	nm1328817	Popcorn Khao! Mast Ho Jao
220	tt0452236	nm0576454	Ginny Aur Johny
221	tt0157917	nm0154113	Lakhon Ki Baat
222	tt2838554	nm4872976	Khokababu
223	tt1024852	nm0474610	Meerabai Not Out
224	tt0420854	nm1680229	Ranga S.S.L.C
225	tt0374342	nm0025609	Yaad Rakhegi Duniya
226	tt0095857	nm0576587	Pestonjee
227	tt0438153	nm0474891	Let's Enjoy
228	tt1582597	nm2490004	Sathyam
229	tt1087526	nm0667393	Tandoori Love
230	tt2356426	nm4110102	Le Halua Le
231	tt0363011	nm0433892	Raja Aur Rangeeli

	trim(p.name)	year
0	Milap Zaveri	2016
1	Danny Leiner	2004
2	Anurag Kashyap	2012
3	Frank Coraci	2004
4	Griffin Dunne	2008
5	Anurag Basu	2012
6	Gurinder Chadha	2004
7	Mike Judge	1996

8	Tarun Mansukhani	2008
9	Shakun Batra	2016
10	Aditya Chopra	2008
11	Rohit Dhawan	2016
12	Nitya Mehra	2016
13	Dibakar Banerjee	2008
14	Umesh Shukla	2012
15	Aditya Chopra	2016
16	Karan Johar	2012
17	Farah Khan	2004
18	Shoojit Sircar	2012
19	Abbas Tyrewala	2008
20	Priyadarshan	2000
21	Shubhashish Bhutiani	2016
22	James Dodson	2008
23	Homi Adajania	2012
24	Joy Augustine	1996
25	Mudassar Aziz	2016
26	Gauri Shinde	2012
27	S.S. Rajamouli	2012
28	Remo D'Souza	2016
29	Sohail Khan	2016
..	...	...
202	Joe Rajan	2012
203	Anand Balraj	2012
204	Govind Menon	2004
205	Mohan Segal	1956
206	Pankaj Parashar	1984
207	Tarun Majumdar	1980
208	Salim Raza	2008
209	Jabbar Patel	1992
210	Sanjay Chhel	2008
211	Sachin Kamlakar Khot	2008
212	David Dhawan	1992
213	Kalpataru	1988
214	Vimal Kumar	2004
215	Srinivas Bhashyam	2004
216	Ganapathy Bharat	2004
217	Debu Sen	1968
218	Raj Kaushal	2004
219	Kabir Sadanand	2004
220	Mehmood	1976
221	Basu Chatterjee	1984
222	Shankaraiya	2012
223	Chandrakant Kulkarni	2008
224	Yograj Bhat	2004
225	Deepak Anand	1992

```
226           Vijaya Mehta  1988
227  Siddharth Anand Kumar  2004
228        Amma Rajasekhar  2008
229          Oliver Paulus  2008
230            Raja Chanda  2012
231       K.S. Prakash Rao  1996

[232 rows x 5 columns]
```

2nd question

[4]:
```
result = pd.read_sql_query('SELECT distinct Name [Actor Names],title [Movie␣
↪Title], substr(m.year,-4) [Year] from person p join m_cast casts on␣
↪trim(casts.pid)=p.pid\
                           join movie m on m.mid=casts.mid\
                           where m.title like \'%Anand%\' and m.year =1971',␣
↪con)
```

[5]:
```
result
```

[5]:
```
          Actor Names Movie Title  Year
0    Amitabh Bachchan       Anand  1971
1       Rajesh Khanna       Anand  1971
2      Brahm Bhardwaj       Anand  1971
3          Ramesh Deo       Anand  1971
4           Seema Deo       Anand  1971
5          Dev Kishan       Anand  1971
6         Durga Khote       Anand  1971
7       Lalita Kumari       Anand  1971
8        Lalita Pawar       Anand  1971
9        Atam Prakash       Anand  1971
10      Sumita Sanyal       Anand  1971
11     Asit Kumar Sen       Anand  1971
12         Dara Singh       Anand  1971
13      Johnny Walker       Anand  1971
14          Moolchand       Anand  1971
15       Gurnam Singh       Anand  1971
16             Savita       Anand  1971
```

3rd question

[6]:
```
pd.read_sql_query('SELECT DISTINCT [Actor Names] from (SELECT Name [Actor␣
↪Names] from person p join m_cast casts on trim(casts.pid)=p.pid\
                           join movie m on m.mid=casts.mid\
                           where cast(substr(m.year,-4) as int) < 1970␣
↪INTERSECT\
                           SELECT Name [Actor Names] from person p join m_cast␣
↪casts on trim(casts.pid)=p.pid\
                           join movie m on m.mid=casts.mid\
                           where cast(substr(m.year,-4) as int) > 1990)', con)
```

[6]:

	Actor Names
0	A.K. Hangal
1	Aachi Manorama
2	Abbas
3	Abdul
4	Abhi Bhattacharya
5	Achala Sachdev
6	Adil
7	Ajay
8	Ajit
9	Akashdeep
10	Akbar Bakshi
11	Alka
12	Allu Ramalingaiah
13	Altaf
14	Amar
15	Amarnath
16	Ameer
17	Amitabh Bachchan
18	Amjad Khan
19	Amol Sen
20	Amrit
21	Anand
22	Anand Kumar
23	Anand Tiwari
24	Anil
25	Anil Kumar
26	Anil Nagrath
27	Anjali Kadam
28	Anju Mahendru
29	Anoop Kumar
..	...
413	Vishnu
414	Vishwa Mehra
415	Waheeda Rehman
416	Wasi Khan
417	Yash Kumar
418	Yasmin
419	Yunus Parvez
420	Yusuf
421	Zia
422	Zohra Sehgal
423	Zul Vellani
424	Asrani
425	Dev Anand
426	Deven Verma
427	Feroz Khan

```
428          Hema Malini
429          Manoj Kumar
430       Master Bhagwan
431              Mehmood
432     Naseeruddin Shah
433           Prayag Raj
434       Rajendra Kumar
435       Randhir Kapoor
436         Rishi Kapoor
437               Sachin
438        Shammi Kapoor
439        Shashi Kapoor
440         Subhash Ghai
441           Sunil Dutt
442          Vinod Mehra

[443 rows x 1 columns]
```

4th question

[7]:
```
result = pd.read_sql_query('SELECT distinct p.pid,trim(p.name) [Director␣
↪Name],count(m.title) [No. of Movies directed] from person p join m_director␣
↪dir on dir.pid=p.pid\
                           join movie m on m.mid = dir.mid\
                           group by p.pid,p.name\
                           having count(*)>=10\
                           order by count(m.title) desc', con)
```

[8]:
```
result
```

[8]:

	PID	Director Name	No. of Movies directed
0	nm0223522	David Dhawan	39
1	nm0080315	Mahesh Bhatt	35
2	nm0698184	Priyadarshan	30
3	nm0890060	Ram Gopal Varma	30
4	nm0080333	Vikram Bhatt	29
5	nm0611531	Hrishikesh Mukherjee	27
6	nm0007181	Yash Chopra	21
7	nm0154113	Basu Chatterjee	19
8	nm0759662	Shakti Samanta	19
9	nm0007131	Subhash Ghai	18
10	nm0070867	Shyam Benegal	17
11	nm0122216	Abbas Alibhai Burmawalla	17
12	nm0851253	Rama Rao Tatineni	17
13	nm0220828	Manmohan Desai	16
14	nm0347899	Gulzar	16
15	nm0802692	Raj N. Sippy	16
16	nm0438029	Raj Kanwar	15
17	nm0542498	Mahesh Manjrekar	15

18	nm0409791	Indra Kumar	14
19	nm0451846	Raj Khosla	14
20	nm0712541	Rahul Rawail	14
21	nm0764316	Rajkumar Santoshi	14
22	nm0004363	Rakesh Roshan	13
23	nm0007147	Dev Anand	13
24	nm0025629	Vijay Anand	13
25	nm0062614	Harry Baweja	13
26	nm0440604	Anurag Kashyap	13
27	nm0536705	Ananth Narayan Mahadevan	13
28	nm0706484	K. Raghavendra Rao	13
29	nm0063357	Anees Bazmee	12
30	nm0223485	Guddu Dhanoa	12
31	nm0422552	Prakash Jha	12
32	nm0442479	Satish Kaushik	12
33	nm0474398	Nagesh Kukunoor	12
34	nm0576488	Prakash Mehra	12
35	nm0576494	Umesh Mehra	12
36	nm0788858	Anil Sharma	12
37	nm1055105	Madhur Bhandarkar	12
38	nm1460159	Rohit Shetty	12
39	nm0149850	Pramod Chakravorty	11
40	nm0348495	Sanjay Gupta	11
41	nm0403826	Nasir Hussain	11
42	nm0576559	Ketan Mehta	11
43	nm0631539	Govind Nihalani	11
44	nm1887138	Mohit Suri	11
45	nm0004292	Raj Kapoor	10
46	nm0052630	K. Bapaiah	10
47	nm0080235	Vishal Bhardwaj	10
48	nm0151504	N. Chandra	10
49	nm0223606	Tigmanshu Dhulia	10
50	nm0244892	J.P. Dutta	10
51	nm0474837	Mehul Kumar	10
52	nm0576554	Hansal Mehta	10
53	nm0592803	Sudhir Mishra	10
54	nm0613517	K. Muralimohana Rao	10
55	nm0660967	Pankaj Parashar	10
56	nm0695162	J. Om Prakash	10
57	nm0746950	Bimal Roy	10

5th question

5th (a) question

Note: I didnt considered gender which contains None

[9]:
```
pd.read_sql_query('select substr(year,-4) [Year],title from (select␣
↪year,title,gender from (select p.name,p.gender,m.title,m.year from person p␣
↪join m_cast casts\
```

```
                    on trim(casts.pid) =p.pid\
                    join movie m on m.mid = casts.mid where p.gender != \'None\'\
                    )group by substr(year,-4),title,gender\
                    )group by title\
                    having count(gender)=1 and gender like \'female\'',con)
```

[9]:

	Year	title
0	1999	Bindhaast
1	1939	Kala Jigar
2	2018	Pihu
3	2000	Snegithiye

5th (b) question

[10]:
```
pd.read_sql_query('select year [Year],(cast (count(*) as float)/cast (ab as␣
↪float))*100 [percentage of movies],[ab] [No_of_movies_per_year] from (select␣
↪*,(select count(*) from movie m where m.year=substr(t.year,-4)) [ab] from␣
↪(select substr(year,-4) [year],title from (select year,title,gender from␣
↪(select p.name,p.gender,m.title,m.year from person p join m_cast casts\
                    on trim(casts.pid) =p.pid\
                    join movie m on m.mid = casts.mid where p.gender != \'None\'\
                    )group by substr(year,-4),title,gender\
                    )group by title\
                    having count(gender)=1 and gender like \'female\'\
                    ) t) a group by year',con)
```

[10]:

	Year	percentage of movies	No_of_movies_per_year
0	1939	50.000000	2
1	1999	1.515152	66
2	2000	1.562500	64
3	2018	1.075269	93

6th question

[11]:
```
pd.read_sql_query('select t.title [Movie Name], max([No.of cast in movie])␣
↪[Maximum no. of cast in movie] from\
                    (SELECT title,count(distinct pid) [No.of cast in movie] from␣
↪movie m join m_cast casts on casts.mid =m.mid\
                    group by m.title\
                    having count(distinct PID)) t', con)
```

[11]:

	Movie Name	Maximum no. of cast in movie
0	Ocean's Eight	238

7th question

[12]:
```
pd.read_sql_query('select substr(j.year,-4) as Year_start, j.year + 9 as␣
↪Year_end,\
                    count(*) as num_movies\
                    from (select distinct year from movie) j join\
                    movie m\
                    on m.year >= j.year and m.year < j.year + 10\
```

```
                    group by j.year\
                    order by count(*) desc\
                    limit 1;',con)
```

[12]:

	Year_start	Year_end	num_movies
0	2008	2017	1128

8th question

```
pd.read_sql_query('select * from person where pid in (select persons.pid from␣
↪(select t.*,j.*, j.year-t.year [TimeGap] from \
                    (select distinct m.title [title], SUBSTR(m.year,-4) [year],p.
↪pid [pid],\
                    ROW_NUMBER() over(partition by p.pid order by p.pid,SUBSTR(m.
↪year,-4) ) rnk \
                    from movie m join m_cast casts\
                    on casts.mid=m.mid join person p on trim(casts.pid)=p.pid )␣
↪t join\
                    (select distinct m.title [title], SUBSTR(m.year,-4) [year],p.
↪pid [pid],\
                    ROW_NUMBER() over(partition by p.pid order by p.pid,SUBSTR(m.
↪year,-4)) rnk \
                    from movie m join m_cast casts\
                    on casts.mid=m.mid join person p on trim(casts.pid)=p.pid )␣
↪j\
                    on j.pid=t.pid and t.rnk+1=j.rnk and j.year-t.year>3)␣
↪persons)',con)
```

[13]:

	index	PID	Name	Gender
0	10	nm2951768	Freida Pinto	Female
1	11	nm4575116	Rohan Chand	Male
2	39	nm0001162	Griffin Dunne	Male
3	74	nm0949433	Damian Young	Male
4	111	nm1753302	Waris Ahluwalia	Male
5	230	nm6467532	Caroline Christl Long	Female
6	245	nm6071249	Rajeev Pahuja	Male
7	260	nm3491108	Michelle Santiago	Female
8	278	nm0943079	Daniel Wu	Male
9	320	nm6631007	Marian Lorencik	Male
10	325	nm7255636	Celina Nessa	Female
11	336	nm0000375	Robert Downey Jr.	Male
12	337	nm0262635	Chris Evans	Male
13	340	nm0424060	Scarlett Johansson	Female
14	341	nm0719637	Jeremy Renner	Male
15	348	nm0079273	Paul Bettany	Male
16	363	nm0380073	Maximiliano Hernández	Male
17	367	nm0001765	Harry Dean Stanton	Male
18	378	nm4230395	Brent McGee	Male
19	385	nm0498278	Stan Lee	None

```
20      388  nm1364280          Damion Poitier    Male
21      404  nm5241466              Mark Falvo    Male
22      476  nm4663387              Anita Date  Female
23      477  nm3481909     Ronjini Chakraborty  Female
24      478  nm3708961            Deepak Damle    Male
25      483  nm1390115           Harish Khanna    Male
26      486  nm4831740              Mohd Samad    Male
27      490  nm0080232        Nitish Bharadwaj    Male
28      492  nm3000723          Sonali Sachdev  Female
29      496  nm8644385            Lalu Makhija    Male
...     ...        ...                     ...     ...
4352  37865  nm0080173          Master Bhagwan    Male
4353  37872  nm0025603                  Anand     Male
4354  37893  nm1323642            Govind Menon    None
4355  37899  nm2287420               Sunil Pal    Male
4356  37910  nm0219939         Danny Denzongpa    Male
4357  37918  nm0438501            Rishi Kapoor    Male
4358  37954  nm2321771              Smeep Kang    None
4359  37963  nm0423656         Joginder Shelly    Male
4360  37964  nm0220823              Ketan Desai    None
4361  37967  nm0154269           Anil Chaudhary    Male
4362  37975  nm0894354              Tinnu Verma    None
4363  37981  nm0220849              Vikas Desai    Male
4364  37997  nm0756393            Kishore Sahu    Male
4365  38003  nm3749594               Aditya Om    Male
4366  38031  nm0007124                Suhasini  Female
4367  38034  nm0595934              Mohan Babu    Male
4368  38035  nm1130698               Raj Tilak    Male
4369  38056  nm1794338           Aryeman Ramsay    Male
4370  38061  nm0214358                Deepshika  Female
4371  38084  nm0693092           Prathap Pothen    Male
4372  38085  nm0576543              Ashok Mehta    None
4373  38087  nm0333031              O.P. Goyle     None
4374  38138  nm0051091            Anand Balraj    Male
4375  38195  nm0438499           Ranjit Kapoor    None
4376  38200  nm0244864              Anjan Dutt    Male
4377  38202  nm0471443        Manmohan Krishna    Male
4378  38203  nm5758702         V. Ravichandran    Male
4379  38216  nm0433392                Jyothika  Female
4380  38275  nm2606304    Srinivas Sunderrajan    None
4381  38279  nm0007806                   Abbas    Male

[4382 rows x 4 columns]
```

Verification: 1. Let's try with random person 2. Taking nm0433392(Jyothika)

[14]:
```
pd.read_sql_query('select t.*,j.*, j.year-t.year [TimeGap] from \
```

```
                    (select distinct m.title [title], SUBSTR(m.year,-4) [year],p.
↪pid [pid],\
                    ROW_NUMBER() over(partition by p.pid order by p.pid,SUBSTR(m.
↪year,-4) ) rnk \
                    from movie m join m_cast casts\
                    on casts.mid=m.mid join person p on trim(casts.pid)=p.pid␣
↪where p.pid=\'nm0433392\') t join\
                    (select distinct m.title [title], SUBSTR(m.year,-4) [year],p.
↪pid [pid],\
                    ROW_NUMBER() over(partition by p.pid order by p.pid,SUBSTR(m.
↪year,-4)) rnk \
                    from movie m join m_cast casts\
                    on casts.mid=m.mid join person p on trim(casts.pid)=p.pid␣
↪where p.pid=\'nm0433392\') j\
                    on j.pid=t.pid and t.rnk+1=j.rnk and t.title<>j.title',con)
```

[14]:

	title	year	pid	rnk	title	year	\
0	Doli Saja Ke Rakhna	1998	nm0433392	2	Vaalee	1999	
1	Vaalee	1999	nm0433392	4	Snegithiye	2000	
2	Snegithiye	2000	nm0433392	6	Uyirile Kalanthathu	2000	
3	Uyirile Kalanthathu	2000	nm0433392	8	Mass	2004	
4	Mass	2004	nm0433392	10	Vettaiyaadu Vilaiyaadu	2006	

	pid	rnk	TimeGap
0	nm0433392	3	1
1	nm0433392	5	1
2	nm0433392	7	0
3	nm0433392	9	4
4	nm0433392	11	2

verfication: 1. At Starting, she has done movie in 1998, the next movie is in 1999 and there is only one year gap, but after Snegithiye(which is released in 2000). She took 4 years gap to do a movie Mass(Telugu movie) in 2004.

9th question

[ ]:

[15]:
```
pd.read_sql_query('select * from (select [person pid] [Actor ID],dir_pid␣
↪[Director ID],[Max_movies_by_any_director] from (select distinct [person␣
↪pid],dir_pid,count(dir_pid) [Max_movies_by_any_director] from\
                    (select distinct trim(p.name),p.pid [person pid],m.mid,m.
↪title,dir.pid [dir_pid],trim(p.name) [dir_name] from person p\
                    join m_cast casts on trim(casts.pid)=p.pid\
                    join movie m on m.mid=casts.mid join m_director dir on dir.
↪mid=m.mid \
                    where dir.pid not like \'%nm0007181%\')\
                    group by trim([person pid]),trim(dir_pid) ) t\
```

```
                    where t.[Max_movies_by_any_director] =(select distinct␣
↪max([Max_movies_by_any_director]) from (select distinct [person␣
↪pid],dir_pid,count(dir_pid) [Max_movies_by_any_director] from\
                    (select distinct trim(p.name),p.pid [person pid],m.mid,m.
↪title,dir.pid [dir_pid],trim(p.name)  [dir_name] from person p\
                    join m_cast casts on trim(casts.pid)=p.pid\
                    join movie m on m.mid=casts.mid join m_director dir on dir.
↪mid=m.mid\
                    where dir.pid not like \'%nm0007181%\') s\
                    group by trim([person pid]),trim(dir_pid) )\
                    where [person pid]=t.[person pid])) q join \
                    (select * from (select distinct trim([actor id]) [actor␣
↪id],trim(dir_name) [Actor Name],count(dir_pid) [no_of_yash_movies_acted]␣
↪from\
                    (select p.name,p.pid [actor id],m.mid,m.title,dir.pid␣
↪[dir_pid],p.name [dir_name] from person p join m_cast casts on trim(casts.
↪pid)=p.pid\
                    join movie m on m.mid=casts.mid join m_director dir on dir.
↪mid=m.mid\
                    where dir.pid like \'%nm0007181%\' ) s\
                    group by [actor id],dir_name)) f on q.[Actor ID]=f.[actor␣
↪id]\
                    where␣
↪no_of_yash_movies_acted>[Max_movies_by_any_director]',con)
```

[15]:

	Actor ID	Director ID	Max_movies_by_any_director	actor id	\
0	nm0004434	nm0004292	3	nm0004434	
1	nm0004434	nm0006370	3	nm0004434	
2	nm0007181	nm0007134	1	nm0007181	
3	nm0158332	nm0004292	2	nm0158332	
4	nm0158332	nm0025629	2	nm0158332	
5	nm0158332	nm0097899	2	nm0158332	
6	nm0158332	nm0611531	2	nm0158332	
7	nm0158332	nm0756488	2	nm0158332	
8	nm0159165	nm0660936	2	nm0159165	
9	nm0347901	nm0122216	3	nm0347901	
10	nm0407002	nm0007131	4	nm0407002	
11	nm0407002	nm0025629	4	nm0407002	
12	nm0407002	nm0159148	4	nm0407002	
13	nm0471443	nm0159148	4	nm0471443	
14	nm0557645	nm0037022	1	nm0557645	
15	nm0557645	nm0611550	1	nm0557645	
16	nm0557645	nm0839348	1	nm0557645	
17	nm0707271	nm0220828	6	nm0707271	
18	nm0716851	nm0451846	3	nm0716851	
19	nm0755087	nm0025608	3	nm0755087	
20	nm0802183	nm0220828	3	nm0802183	

21	nm0839590	nm0759662	1	nm0839590
22	nm0839590	nm0788391	1	nm0839590
23	nm0839590	nm0836899	1	nm0839590
24	nm0839590	nm1130698	1	nm0839590
25	nm1275889	nm0004570	1	nm1275889
26	nm1275889	nm0706800	1	nm1275889
27	nm1588131	nm0006370	1	nm1588131
28	nm1588131	nm0766166	1	nm1588131
29	nm1767604	nm0474909	1	nm1767604
30	nm3163800	nm0611531	1	nm3163800

	Actor Name	no_of_yash_movies_acted
0	Shashi Kapoor	7
1	Shashi Kapoor	7
2	Yash Chopra	2
3	Leela Chitnis	3
4	Leela Chitnis	3
5	Leela Chitnis	3
6	Leela Chitnis	3
7	Leela Chitnis	3
8	Sudha Chopra	3
9	Rakhee Gulzar	5
10	Iftekhar	9
11	Iftekhar	9
12	Iftekhar	9
13	Manmohan Krishna	10
14	Nissar	2
15	Nissar	2
16	Nissar	2
17	Jagdish Raj	11
18	Waheeda Rehman	5
19	Achala Sachdev	4
20	Neetu Singh	4
21	Surendra Nath	2
22	Surendra Nath	2
23	Surendra Nath	2
24	Surendra Nath	2
25	Raj Hans	2
26	Raj Hans	2
27	Shyam Arora	2
28	Shyam Arora	2
29	Ashok Verma	2
30	Nazir	2

Total 17 actors are there who acted more movies with yash chopra than any other director.
Validation

[16]:
```
pd.read_sql_query('select distinct trim(p.pid),trim(p.name),m.title,m.mid,dir.
↪pid from person p join m_cast casts on trim(casts.pid)=p.pid \
                   join movie m on m.mid=casts.mid join m_director dir on dir.
↪mid=m.mid\
                   where  p.pid = \'nm0159165\'  ',con)
```

[16]:

	trim(p.pid)	trim(p.name)	title	MID	PID
0	nm0159165	Sudha Chopra	Rocky	tt0215132	nm0004570
1	nm0159165	Sudha Chopra	Silsila	tt0083081	nm0007181
2	nm0159165	Sudha Chopra	Desh Premee	tt0083820	nm0220828
3	nm0159165	Sudha Chopra	Prem Rog	tt0084532	nm0004292
4	nm0159165	Sudha Chopra	Pukar	tt0086156	nm0066918
5	nm0159165	Sudha Chopra	Saath Saath	tt0341554	nm0474861
6	nm0159165	Sudha Chopra	Sparsh	tt0079938	nm0660936
7	nm0159165	Sudha Chopra	Katha	tt0085776	nm0660936
8	nm0159165	Sudha Chopra	Trishul	tt0078418	nm0007181
9	nm0159165	Sudha Chopra	Dostana	tt0080653	nm0451846
10	nm0159165	Sudha Chopra	Baseraa	tt0300980	nm0848343
11	nm0159165	Sudha Chopra	Kaala Patthar	tt0079386	nm0007181
12	nm0159165	Sudha Chopra	Anmol	tt0106287	nm0220823
13	nm0159165	Sudha Chopra	Gharaonda	tt0150718	nm0080405
14	nm0159165	Sudha Chopra	Saaheb	tt0089952	nm0304356
15	nm0159165	Sudha Chopra	Griha Pravesh	tt0156581	nm0080345
16	nm0159165	Sudha Chopra	Meera	tt0081147	nm0347899
17	nm0159165	Sudha Chopra	Pyar Jhukta Nahin	tt0087958	nm0755420

Sudha Chopra has acted in 17 movies, out of these movies. She has acted 3 movies with yash chopra. If we observe the result set of original answer. Maximum movies she has done with any director except yash chopra is 2.

[17]:
```
pd.read_sql_query('select distinct trim(p.pid),trim(p.name),m.title,m.mid,dir.
↪pid from person p join m_cast casts on trim(casts.pid)=p.pid \
                   join movie m on m.mid=casts.mid join m_director dir on dir.
↪mid=m.mid\
                   where  p.pid = \'nm0159165\'  and dir.pid=\'nm0007181\'␣
↪',con)
```

[17]:

	trim(p.pid)	trim(p.name)	title	MID	PID
0	nm0159165	Sudha Chopra	Silsila	tt0083081	nm0007181
1	nm0159165	Sudha Chopra	Trishul	tt0078418	nm0007181
2	nm0159165	Sudha Chopra	Kaala Patthar	tt0079386	nm0007181

With Yash chopra she has done 3 movies.

[ ]:
```

```

10th question

[18]:
```
pd.read_sql_query('select * from person p join m_cast casts on trim(casts.
↪pid)=p.pid where  p.name like \'%shah%rukh%Khan%\' ',con)
```

[18]:

	index	PID	Name	Gender	index	MID	PID	\
0	3012	nm0451321	Shah Rukh Khan	Male	3157	tt1188996	nm0451321	
1	3012	nm0451321	Shah Rukh Khan	Male	3462	tt1285241	nm0451321	
2	3012	nm0451321	Shah Rukh Khan	Male	4738	tt0248126	nm0451321	
3	3012	nm0451321	Shah Rukh Khan	Male	4926	tt5946128	nm0451321	
4	3012	nm0451321	Shah Rukh Khan	Male	5347	tt3405236	nm0451321	
5	3012	nm0451321	Shah Rukh Khan	Male	6035	tt0238936	nm0451321	
6	3012	nm0451321	Shah Rukh Khan	Male	6149	tt4559006	nm0451321	
7	3012	nm0451321	Shah Rukh Khan	Male	6738	tt0112870	nm0451321	
8	3012	nm0451321	Shah Rukh Khan	Male	6925	tt0347304	nm0451321	
9	3012	nm0451321	Shah Rukh Khan	Male	7076	tt4535650	nm0451321	
10	3012	nm0451321	Shah Rukh Khan	Male	7342	tt1024943	nm0451321	
11	3012	nm0451321	Shah Rukh Khan	Male	7445	tt0172684	nm0451321	
12	3012	nm0451321	Shah Rukh Khan	Male	7548	tt5997666	nm0451321	
13	3012	nm0451321	Shah Rukh Khan	Male	7694	tt5882970	nm0451321	
14	3012	nm0451321	Shah Rukh Khan	Male	7742	tt0420332	nm0451321	
15	3012	nm0451321	Shah Rukh Khan	Male	7967	tt1562871	nm0451321	
16	3012	nm0451321	Shah Rukh Khan	Male	8145	tt1182937	nm0451321	
17	3012	nm0451321	Shah Rukh Khan	Male	8378	tt0367110	nm0451321	
18	3012	nm0451321	Shah Rukh Khan	Male	8440	tt2176013	nm0451321	
19	3012	nm0451321	Shah Rukh Khan	Male	8882	tt0871510	nm0451321	
20	3012	nm0451321	Shah Rukh Khan	Male	8968	tt2112124	nm0451321	
21	3012	nm0451321	Shah Rukh Khan	Male	9646	tt0213890	nm0451321	
22	3012	nm0451321	Shah Rukh Khan	Male	10599	tt3495026	nm0451321	
23	3012	nm0451321	Shah Rukh Khan	Male	10988	tt0461936	nm0451321	
24	3012	nm0451321	Shah Rukh Khan	Male	11248	tt2461132	nm0451321	
25	3012	nm0451321	Shah Rukh Khan	Male	11479	tt2797242	nm0451321	
26	3012	nm0451321	Shah Rukh Khan	Male	11534	tt1806913	nm0451321	
27	3012	nm0451321	Shah Rukh Khan	Male	12067	tt0347473	nm0451321	
28	3012	nm0451321	Shah Rukh Khan	Male	13070	tt0164538	nm0451321	
29	3012	nm0451321	Shah Rukh Khan	Male	13198	tt0449999	nm0451321	
..	...	...	...	...	...	...	...	
60	3012	nm0451321	Shah Rukh Khan	Male	30168	tt0126871	nm0451321	
61	3012	nm0451321	Shah Rukh Khan	Male	30880	tt0136352	nm0451321	
62	3012	nm0451321	Shah Rukh Khan	Male	30909	tt0109134	nm0451321	
63	3012	nm0451321	Shah Rukh Khan	Male	30959	tt3531852	nm0451321	
64	3012	nm0451321	Shah Rukh Khan	Male	31567	tt0991346	nm0451321	
65	3012	nm0451321	Shah Rukh Khan	Male	32547	tt1182884	nm0451321	
66	3012	nm0451321	Shah Rukh Khan	Male	33896	tt0813996	nm0451321	
67	3012	nm0451321	Shah Rukh Khan	Male	35780	tt1937092	nm0451321	
68	3012	nm0451321	Shah Rukh Khan	Male	36675	tt0172519	nm0451321	
69	3012	nm0451321	Shah Rukh Khan	Male	36962	tt0114231	nm0451321	
70	3012	nm0451321	Shah Rukh Khan	Male	37155	tt0101732	nm0451321	
71	3012	nm0451321	Shah Rukh Khan	Male	37813	tt0172980	nm0451321	
72	3012	nm0451321	Shah Rukh Khan	Male	38990	tt0857381	nm0451321	
73	3012	nm0451321	Shah Rukh Khan	Male	39285	tt0137100	nm0451321	
74	3012	nm0451321	Shah Rukh Khan	Male	39930	tt0172234	nm0451321	

75	3012	nm0451321	Shah Rukh Khan	Male	45301	tt0331639	nm0451321
76	3012	nm0451321	Shah Rukh Khan	Male	46867	tt0227194	nm0451321
77	3012	nm0451321	Shah Rukh Khan	Male	47870	tt0120540	nm0451321
78	3012	nm0451321	Shah Rukh Khan	Male	50100	tt1182908	nm0451321
79	3012	nm0451321	Shah Rukh Khan	Male	51458	tt0114726	nm0451321
80	3012	nm0451321	Shah Rukh Khan	Male	53338	tt0438981	nm0451321
81	3012	nm0451321	Shah Rukh Khan	Male	54500	tt1877691	nm0451321
82	3012	nm0451321	Shah Rukh Khan	Male	56056	tt4228400	nm0451321
83	3012	nm0451321	Shah Rukh Khan	Male	61117	tt0221982	nm0451321
84	3012	nm0451321	Shah Rukh Khan	Male	62246	tt0126234	nm0451321
85	3012	nm0451321	Shah Rukh Khan	Male	63466	tt1538210	nm0451321
86	3012	nm0451321	Shah Rukh Khan	Male	69080	tt0250415	nm0451321
87	3012	nm0451321	Shah Rukh Khan	Male	69272	tt0453748	nm0451321
88	3012	nm0451321	Shah Rukh Khan	Male	77628	tt0286664	nm0451321
89	3012	nm0451321	Shah Rukh Khan	Male	80216	tt1773042	nm0451321

	ID
0	3157
1	3462
2	4738
3	4926
4	5347
5	6035
6	6149
7	6738
8	6925
9	7076
10	7342
11	7445
12	7548
13	7694
14	7742
15	7967
16	8145
17	8378
18	8440
19	8882
20	8968
21	9646
22	10599
23	10988
24	11248
25	11479
26	11534
27	12067
28	13070
29	13198

```
..    ...
60  30168
61  30880
62  30909
63  30959
64  31567
65  32547
66  33896
67  35780
68  36675
69  36962
70  37155
71  37813
72  38990
73  39285
74  39930
75  45301
76  46867
77  47870
78  50100
79  51458
80  53338
81  54500
82  56056
83  61117
84  62246
85  63466
86  69080
87  69272
88  77628
89  80216

[90 rows x 8 columns]
```

Here Shah Rukh Khan has did 90 movies on his carrer upto the recorded data. This is sharukh number 0

[19]:
```
pd.read_sql_query('select distinct trim(pid) [co_actors_for_Sharukh] from␣
↪m_cast where mid in \
                  (select m.mid from movie m join m_cast casts on casts.mid =␣
↪m.mid join person p on p.pid=trim(casts.pid)\
                  where p.name like \'%shah%rukh%Khan%\' )\
                  and trim(pid) not in (\
                  select p.pid from person p join m_cast casts2 on p.
↪pid=trim(casts2.pid) and p.name like \'%shah%rukh%Khan%\')\
                   ',con)
```

[19]:
```
    co_actors_for_Sharukh
0               nm0004418
```

1	nm1995953
2	nm2778261
3	nm0631373
4	nm0241935
5	nm0792116
6	nm1300111
7	nm0196375
8	nm1464837
9	nm2868019
10	nm0906226
11	nm1012529
12	nm0952232
13	nm0665555
14	nm2670109
15	nm2540053
16	nm0632634
17	nm2974366
18	nm3841022
19	nm3840343
20	nm0248474
21	nm2648101
22	nm2574505
23	nm2387689
24	nm4373080
25	nm2006282
26	nm2775421
27	nm2489172
28	nm2927987
29	nm0282948
...	...
2352	nm4968917
2353	nm0442470
2354	nm1267986
2355	nm0710601
2356	nm1236022
2357	nm0361643
2358	nm0759785
2359	nm0958860
2360	nm1942776
2361	nm2146462
2362	nm0474604
2363	nm0795348
2364	nm1886256
2365	nm1889672
2366	nm1894111
2367	nm2161830
2368	nm2145968

```
2369            nm0802369
2370            nm2135417
2371            nm0442479
2372            nm2133802
2373            nm2108016
2374            nm1543016
2375            nm2273524
2376            nm0080426
2377            nm4173451
2378            nm7620177
2379            nm3093045
2380            nm0451154
2381            nm3385526

[2382 rows x 1 columns]
```

2382 actors worked with shahrukh khan in his 90 movies. These people are sharukh number 1.

[20]:
```
pd.read_sql_query('select distinct p.pid ,p.name,p.gender,m.title from m_cast␣
↪casts join person p on p.pid=trim(casts.pid)\
                   join movie m on m.mid=casts.mid\
                   where casts.mid in (select distinct mid from m_cast casts3␣
↪where trim(pid)\
                   in (select distinct trim(pid) from m_cast where mid in \
                   (select distinct m.mid from movie m join m_cast casts on␣
↪casts.mid = m.mid join person p on p.pid=trim(casts.pid)\
                   where p.name like \'%shah%rukh%Khan%\' )\
                   and trim(pid) not in (\
                   select p.pid from person p join m_cast casts2 on p.
↪pid=trim(casts2.pid) and p.name like \'%shah%rukh%Khan%\')\
                   ) and mid not in (select distinct mid from m_cast casts \
                   join person p on p.pid =trim(casts.pid) and p.name like␣
↪\'%shah%rukh%Khan%\'))\
                   and p.pid not in (select distinct trim(pid) from m_cast␣
↪casts3 where trim(pid)\
                   in (select distinct trim(pid) from m_cast where mid in \
                   (select distinct m.mid from movie m join m_cast casts on␣
↪casts.mid = m.mid join person p on p.pid=trim(casts.pid)\
                   where p.name like \'%shah%rukh%Khan%\')\
                   and trim(pid) not in (\
                   select p.pid from person p join m_cast casts2 on p.
↪pid=trim(casts2.pid) and p.name like \'%shah%rukh%Khan%\')\
                   ) and mid not in (select distinct mid from m_cast casts \
                   join person p on p.pid =trim(casts.pid) and p.name like␣
↪\'%shah%rukh%Khan%\'))',con)
```

[20]:

	PID	Name	Gender	title
0	nm2539953	Alicia Vikander	Female	Tomb Raider
1	nm0922035	Dominic West	Male	Tomb Raider
2	nm0324658	Walton Goggins	Male	Tomb Raider
3	nm0943079	Daniel Wu	Male	Tomb Raider
4	nm0000218	Kristin Scott Thomas	Female	Tomb Raider
5	nm0001394	Derek Jacobi	Male	Tomb Raider
6	nm0929654	Alexandre Willaume	Male	Tomb Raider
7	nm3116102	Tamer Burjaq	None	Tomb Raider
8	nm3248891	Adrian Collins	Male	Tomb Raider
9	nm2418809	Keenan Arrison	Male	Tomb Raider
10	nm4491546	Andrian Mazive	Male	Tomb Raider
11	nm4305023	Milton Schorr	Male	Tomb Raider
12	nm4789912	Hannah John-Kamen	Female	Tomb Raider
13	nm9678609	Peter Waison	Male	Tomb Raider
14	nm5786571	Samuel Mak	Male	Tomb Raider
15	nm9478258	Sky Yang	Male	Tomb Raider
16	nm9659535	Civic Chung	Male	Tomb Raider
17	nm1086981	Josef Altin	Male	Tomb Raider
18	nm5037475	Billy Postlethwaite	Male	Tomb Raider
19	nm3788222	Roger Jean Nsengiyumva	Male	Tomb Raider
20	nm1785575	Jaime Winstone	Female	Tomb Raider
21	nm0643394	Michael Obiora	Male	Tomb Raider
22	nm9632976	Shekhar Varma	Male	Tomb Raider
23	nm6946516	Rekha John-Cheriyan	Female	Tomb Raider
24	nm3379444	Antonio Aakeel	Male	Tomb Raider
25	nm8993944	Maisy De Freitas	Female	Tomb Raider
26	nm6917958	Emily Carey	Female	Tomb Raider
27	nm7890799	Gordon Chow	Male	Tomb Raider
28	nm3461674	Duncan Airlie James	Male	Tomb Raider
29	nm7509518	Jandre le Roux	Male	Tomb Raider
...	...	...	...	...
50799	nm0223607	Dhumal	Male	Geet Gaata Chal
50800	nm0329710	V. Gopal	Male	Geet Gaata Chal
50801	nm0576455	Mehmood Jr.	Male	Geet Gaata Chal
50802	nm0839129	Sunder	Male	Geet Gaata Chal
50803	nm0842004	Swati	Female	Geet Gaata Chal
50804	nm0864603	Ramayan Tiwari	Male	Geet Gaata Chal
50805	nm0716851	Waheeda Rehman	Female	Allah-Rakha
50806	nm0792866	Meenakshi Sheshadri	Female	Allah-Rakha
50807	nm0219946	Ramesh Deo	Male	Allah-Rakha
50808	nm0580345	Mushtaq Merchant	Male	Allah-Rakha
50809	nm1231418	Asif	None	Allah-Rakha
50810	nm0451163	Ali Khan	Male	Allah-Rakha
50811	nm0695199	Pran	Male	Allah-Rakha
50812	nm0201711	Harbans Darshan M. Arora	Male	Allah-Rakha
50813	nm0151539	Chandrashekhar	Male	Allah-Rakha

50814	nm1571380	Gorilla	Male	Allah-Rakha
50815	nm0438477	Kamal Kapoor	Male	Allah-Rakha
50816	nm0438510	Trilok Kapoor	Male	Allah-Rakha
50817	nm0613407	Murad	Male	Allah-Rakha
50818	nm0755996	Prem Sagar	None	Allah-Rakha
50819	nm0893142	Venkatesh Daggubati	Male	Anari
50820	nm2371614	Hayley Cleghorn	Female	Come December
50821	nm2675737	Nirvasha Jithoo	None	Come December
50822	nm1045770	Aushim Khetarpal	Male	Come December
50823	nm2370589	Kamal Maharshi	Male	Come December
50824	nm1866356	Mohini Manik	Female	Come December
50825	nm2688404	Jaipreet Nagra	Male	Come December
50826	nm6170213	Ajay Kumar Verma	Male	Come December
50827	nm4468244	Arun Govil	Male	Kanoon
50828	nm0019427	Ishrat Ali	Male	Kanoon

[50829 rows x 4 columns]

There are 50829 actors who worked with co-actors of shahrukh khan. These people are sharukh number 2.