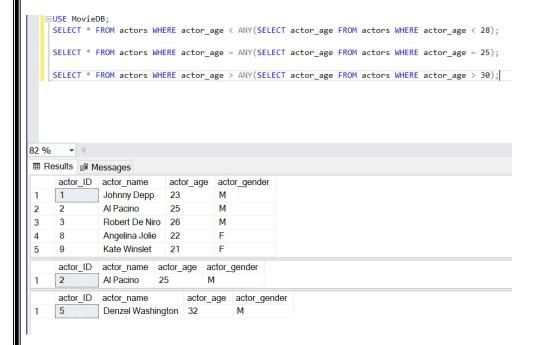
# **DBMS LAB ASSIGNMENT - 5**

NAME: K.V.S.BHARADWAJ

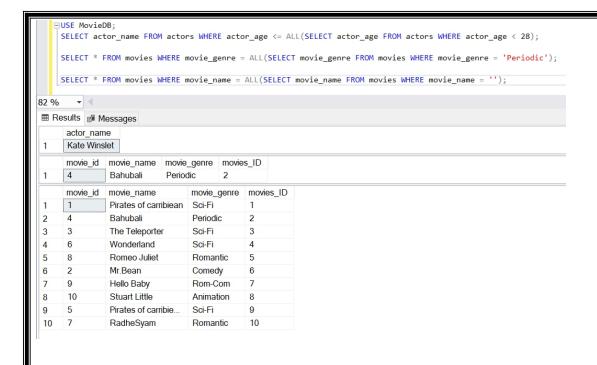
**ROLL NO.**: 19BCS047

Q1) Illustrate logical ANY, ALL and LIKE operator- the queries should be relevant to your respective databases 3 queries for each operator. One query explaining the difference between ANY and ALL.

# **QUERIES FOR "ANY"**



# **QUERIES FOR "ALL"**



# **QUERIES FOR "LIKE"**

```
─USE MovieDB;

     SELECT * FROM movies WHERE movie_id LIKE 4;
     SELECT * FROM actors WHERE actor_name LIKE '%Pitt';
     SELECT actor_age FROM actors WHERE actor_gender LIKE 'M';
82 %
         - 4
■ Results Messages
      movie_id
                movie_name
                              movie_genre
                                            movies_ID
 1
      4
                 Bahubali
                              Periodic
                                            2
      actor_ID
                actor_name
                             actor_age
                                         actor_gender
      7
                 Brad Pitt
                              31
                                         M
 1
      actor_age
      23
 1
 2
      25
 3
      26
      30
 4
       32
 5
      28
 6
       31
 7
```

Q2) One query for each Aggregate function.

The aggregate functions are MIN(), MAX(), COUNT(), AVG(), SUM()

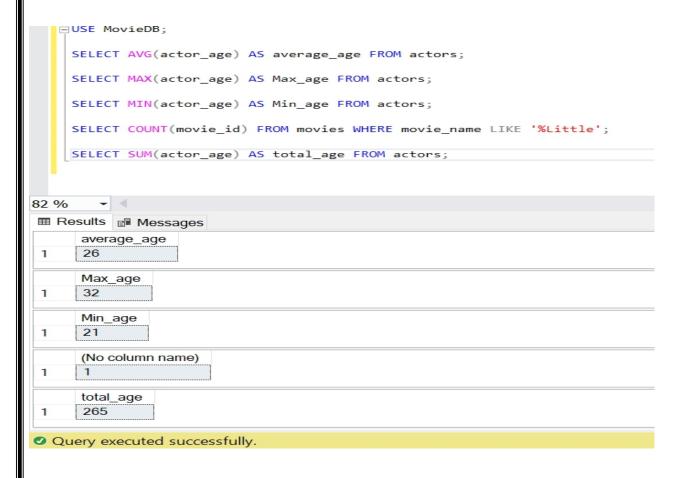
AVG() – return the average of the set

MIN() – returns the minimum value in a set

MAX() – returns the maximum value in set

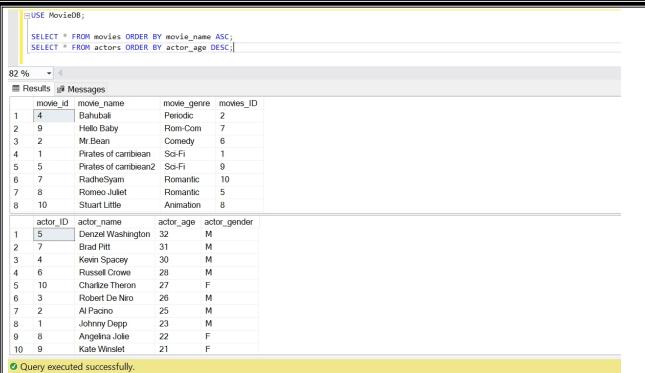
SUM() – returns the sum of all distinct values of a set

COUNT() – returns the number of items in a set



Q3) Illustrate the usage of order by, group by and having clause (2 queries for each case)

# **ORDER BY**

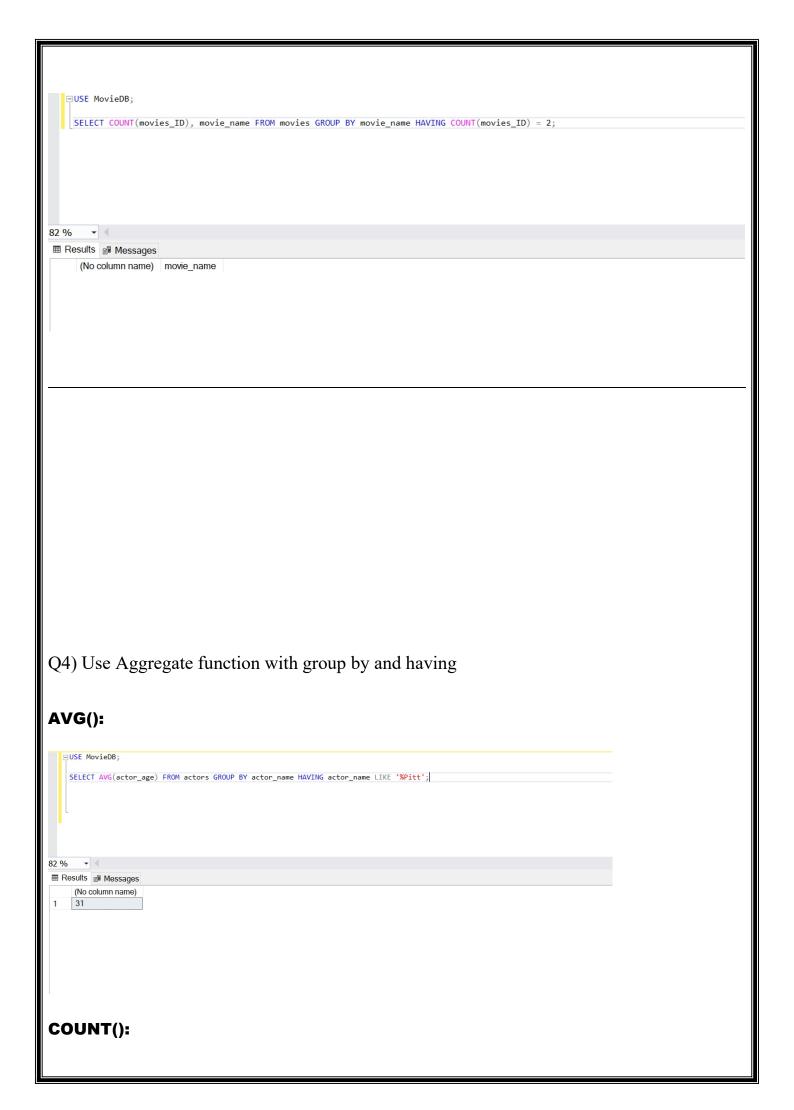


# **GROUP BY**

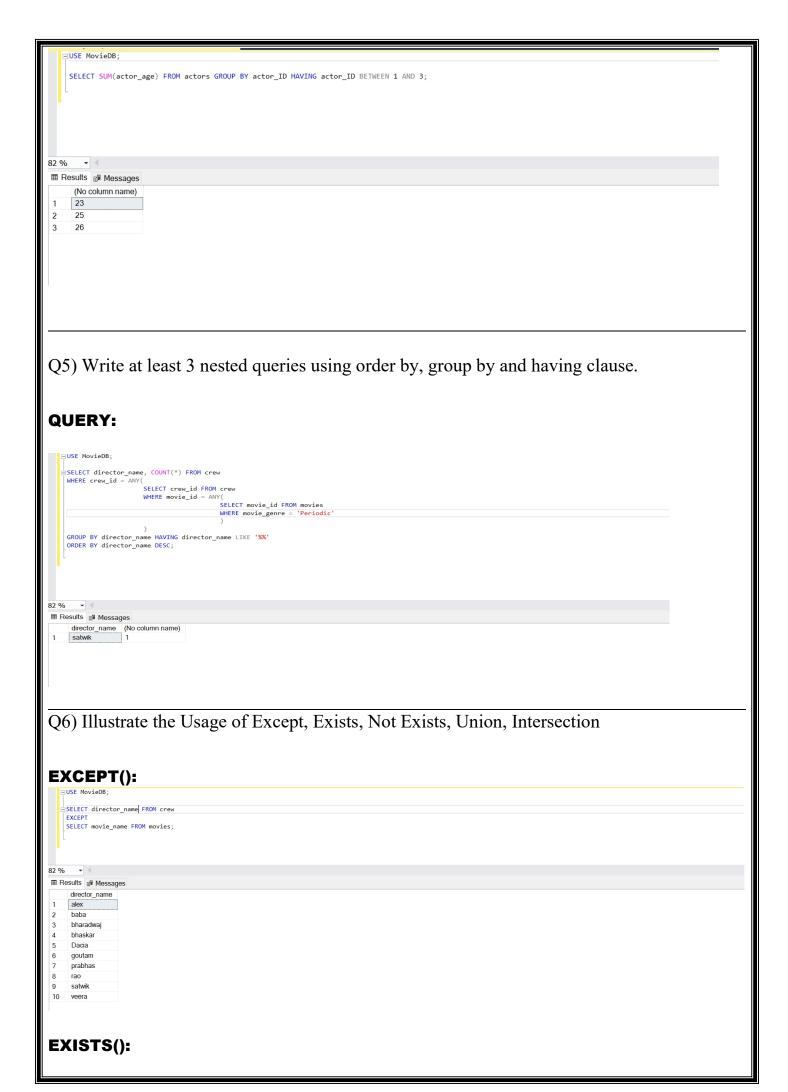
```
□USE MovieDB;
    SELECT actor_age, COUNT(*) AS actor_id FROM actors GROUP BY actor_age;
    SELECT movie_name, COUNT(*) FROM movies GROUP BY movie_name;
82 %
■ Results ■ Messages
      actor_age actor_id
      21
2
      22
3
      23
4
      25
5
      26
6
      27
7
      28
      movie_name (No column name)
     Bahubali
 1
2
      Hello Baby
3
      Mr.Bean
4
      Pirates of c...
5
      Pirates of c...
6
      RadheSyam
7
      Romeo Juli...
8
      Stuart Little
9
      The Telepo...
 10
     Wonderland 1
```

# Query executed successfully.

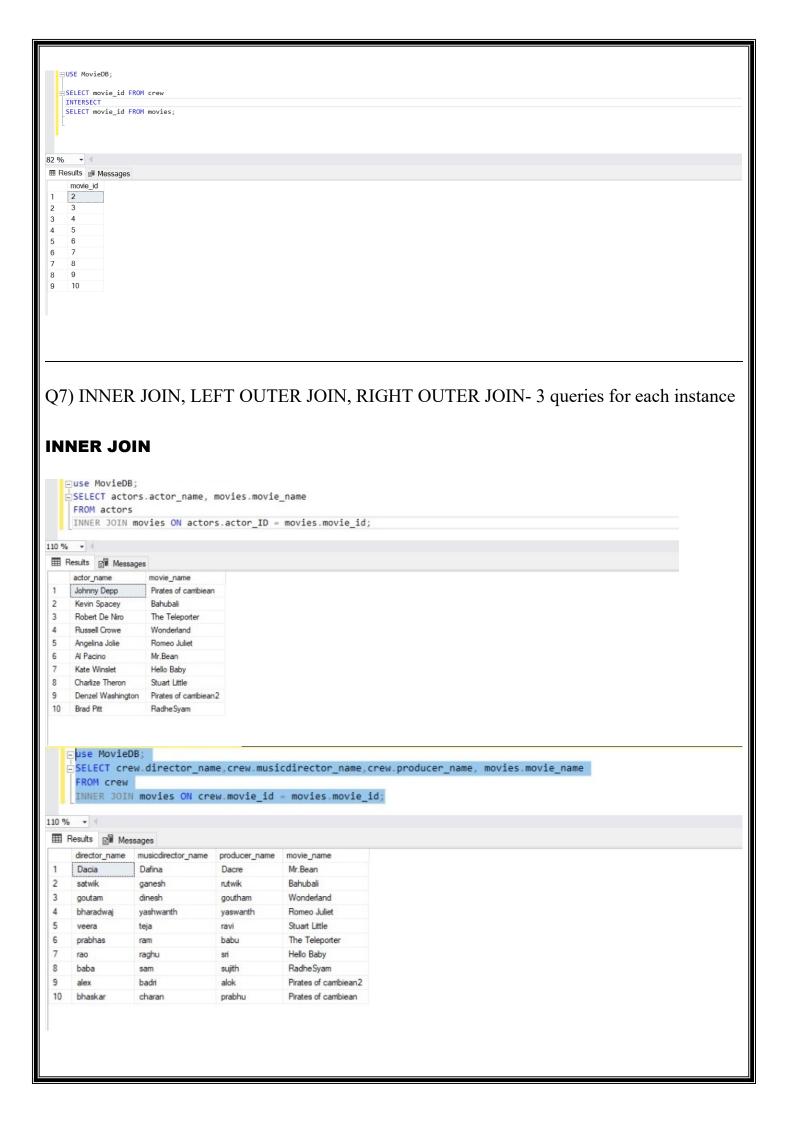
# **HAVING CLAUSE**

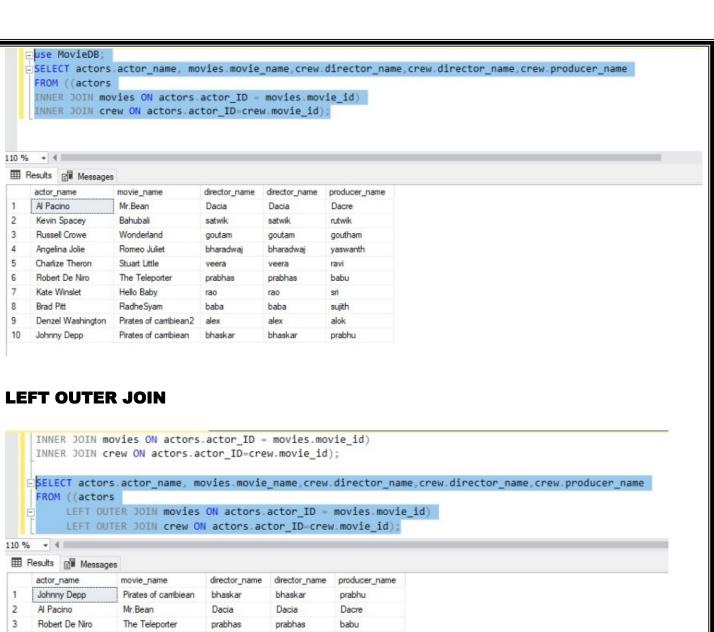


```
SELECT COUNT(actor_ID) FROM actors GROUP BY actor_age HAVING actor_age \geq 130;
      -
82 %
(No column name)
    1
 3
MIN():
 □USE MovieDB;
    SELECT MIN(actor_age) FROM actors GROUP BY actor_gender HAVING actor_gender = 'F';
82 % -
■ Results ■ Messages
    (No column name)
 1 21
MAX():
    SELECT MAX(actor_age) FROM actors GROUP BY actor_gender HAVING actor_gender = 'M';
      ---
82 %
(No column name)
    32
SUM():
```









	actor_name	movie_name	director_name	director_name	producer_name
1	Johnny Depp	Pirates of carribiean	bhaskar	bhaskar	prabhu
2	Al Pacino	Mr.Bean	Dacia	Dacia	Dacre
3	Robert De Niro	The Teleporter	prabhas	prabhas	babu
4	Kevin Spacey	Bahubali	satwik	satwik	rutwik
5	Denzel Washington	Pirates of carribiean2	alex	alex	alok
6	Russell Crowe	Wonderland	goutam	goutam	goutham
7	Brad Pitt	RadheSyam	baba	baba	sujith
8	Angelina Jolie	Romeo Juliet	bharadwaj	bharadwaj	yaswanth
9	Kate Winslet	Hello Baby	rao	rao	sri
10	Charlize Theron	Stuart Little	veera	veera	ravi

