Database Concepts 5COM1052

Coursework 2018-2019

Mega Independent Cinema Experience (MICE)

1. Produce a list of films which have a date of release before August 2016

```
--ql
SELECT FILM_NO, FILM_NAME, DATE_OF_RELEASE FROM FILM
WHERE DATE_OF_RELEASE < TO_DATE('01-AUG-16' , 'dd-mm-yy');
```

1 row updated.

FILM_NO	FILM_NAME	DATE_OF_R
2	Source Code	15-APR-11
3	The raid	10-JUL-11
4	Rat Race	07-FEB-01
5	Enemy	10-AUG-13
6	Dracula Untold	24-AUG-14
7	Shutter Island	15-NOV-10
8	John Wick	06-JUN-14
9	Southpaw	20-MAY-15
10	Captain America: The Winter Soldier	15-MAR-14
11	21 Jump Street	11-OCT-12
12	Transformers	10-JUN-04

FILM_NO	FILM_NAME	DATE_OF_R
13	Big Fish	21-FEB-12
14	Rise of the Planet of the Apes	22-SEP-11
15	Treasure Planet	28-APR-03
16	Inception	06-JAN-10

2. List the full details of the cinemas managed by the employees with the employee number 55 and 52

GIVEN_NAME	FAMILY_NAME	ADDRESS	DOB	SUPERVISOR_EMP_NO	CINEMA_NO
Heather	Watson	42 Ellesmere Leas Campion X4R B2S	05-MAY-90	75032569	3
Gowtham	Kandeepan	129 culver grove Stanmore HA7 2NP	10-JUN-98	94367024	4

```
--q2
SELECT  Given_Name,Family_Name,Address,DOB, Supervisor_Emp_No,Cinema_No
FROM EMPLOYEE
WHERE Emp_No = '55'or Emp_No = '52';
--q3
```

3. List employee details for the employees who work at the same cinema as the employee Joe Bloggs

```
--q3
SELECT Emp_No ,Given_Name,Family_Name,DOB,Address,Supervisor_Emp_No,Cinema_No
FROM EMPLOYEE
WHERE Cinema_No IN (SELECT Cinema_No from Employee where Given_Name like 'Joe');
--q4
```

EMP_NO	GIVEN_NAME	FAMILY_NAME	DOB	ADDRESS	SUPERVISOR_EMP_NO	CINEMA_NO
32	Blake	lively	28-JUL-85	1 Back Irlam Street Irlam 0D2 KG5	49326592	1
43	Joe	Bloggs	16-NOV-96	7 Harding Moorings Malden 9FW NG5	49326592	1
49326592	Angus	Krueger	20-JUL-86	49 Paradise City Hatfeild AL9 432		1

4. Write a SQL statement to count the number of films in the database released before August 2016 (The result should just show the count)

	1 The raid	10-JUL-11
	1 Rat Race	07-FEB-01
	1 Captain America: The Winter Soldier	15-MAR-14
	1 Dracula Untold	24-AUG-14
	1 John Wick	06-JUN-14
	1 Southpaw	20-MAY-15
	1 Rise of the Planet of the Apes	22-SEP-11
	1 Treasure Planet	28-APR-03
	1 Shutter Island	15-NOV-10
	1 21 Jump Street	11-OCT-12
	1 Big Fish	21-FEB-12

ilms before August 2016	FILM_NAME	DATE_OF_R
	1 Inception	06-JAN-10
	1 Transformers	10-JUN-04
	1 Source Code	15-APR-11
	1 Enemy	10-AUG-13

```
--q4
SELECT COUNT (DATE_OF_RELEASE) as "films before August 2016", FILM_NAME, date_of_release
FROM FILM
WHERE DATE_OF_RELEASE < TO_DATE('01-AUG-16' , 'dd-mm-yy')
GROUP BY Film_No, date_of_release,FILM_NAME;
--q5
```

5. Produce a list of Employees who work for the cinema with the cinema number 07 and include the cinema name in the result. Arrange the result in ascending order by employees' surname. Name attributes should be combined into a single output column, and given a sensible header.

```
--q5
SELECT EMPLOYEE.GIVEN_NAME || ' ' ||EMPLOYEE.Family_Name AS "Employee Name",EMPLOYEE.CINEMA_NO,CINEMA.CINEMA_NAME
FROM EMPLOYEE
JOIN CINEMA ON CINEMA.CINEMA_NO = EMPLOYEE.CINEMA_NO
```

Employee Name	CINEMA_NO	CINEMA_NAME
Steve Fox	7	Cineworld
Chay Barton	7	Cineworld
Sharna Lopez	7	Cineworld
Pual Jack	7	Cineworld

6. Employee number 99 has phoned in sick. We need the full details of his supervisor.

```
SELECT Supervisor_emp_no
FROM EMPLOYEE
WHERE emp_no = '99'
```

	SUPERVISOR_EMP_NO
1	45920461

7. Write a SQL statement to list all the film names of films shown between 1st July 2017 and 30th September 2017

```
--Q7
SELECT FILM_NO, FILM_NAME, DATE_OF_RELEASE FROM FILM
WHERE DATE_OF_RELEASE BETWEEN TO_DATE('1/7/2017', 'DD/MM/YYYY') AND TO_DATE('30/09/2017', 'DD/MM/YYYY')
```

FILM_NO	FILM_NAME	DATE_OF_RELEASE
1	King arthur legend of the sword	23-AUG-17

8. Write a SQL statement to list the cinema numbers of cinemas which employ more than four employees

```
--Q8
SELECT CINEMA_NO AS "CINEMA NUMBER", COUNT (*) AS "EMPLOYEES"
FROM EMPLOYEE
GROUP BY CINEMA_NO
HAVING COUNT (*) > 4;
```

CINEMA NUMBER	EMPLOYEES
2	5

9. Produce the takings for the film with the name *Big Fish* when it was shown in cinema 02 in screen no 1

10. Which film has taken the least takings at a performance? Include film name and cinema name in the results

```
--Q10
SELECT FILM.FILM_NAME,CINEMA.CINEMA_NAME,MIN(PERFORMANCE.Total_Takings) AS "Lowest"
from PERFORMANCE
JOIN FILM ON PERFORMANCE.FILM_NO=FILM.FILM_NO
JOIN CINEMA ON PERFORMANCE.CINEMA_no = CINEMA.CINEMA_no
WHERE ROWNUM = 1
GROUP BY FILM.FILM_NAME, CINEMA.CINEMA_NAME
ORDER BY MIN(PERFORMANCE.Total_Takings) ASC;
```

	FILM_NAME	CINEMA_NAME	Lowest
1	Rat Race	ODEON	3442

With all queries part of the mark allocation are for the query and half for the submitted results as shown.

Please remember we check your tables and table content on-line.

Section Three

Grant access on your tables to comtpmm, comrklk, comqhx1, rj14aab

```
GRANT SELECT ON Cinema to comtpmm;
GRANT SELECT ON Employee to comtpmm;
GRANT SELECT ON FILM to comtpmm;
GRANT SELECT ON Screen to comtpmm;
GRANT SELECT ON Showings to comtpmm;
GRANT SELECT ON Performance to comtpmm;

GRANT SELECT ON Cinema to comrklk;
GRANT SELECT ON Employee to comrklk;
GRANT SELECT ON FILM to comrklk;
GRANT SELECT ON Screen to comrklk;
GRANT SELECT ON Showings to comrklk ;
GRANT SELECT ON Performance to  comrklk ;

GRANT SELECT ON Cinema to comqhxl;
GRANT SELECT ON Employee to comqhxl;
GRANT SELECT ON FILM to comqhxl;
GRANT SELECT ON Screen to comqhxl;
GRANT SELECT ON Showings to comqhxl;
GRANT SELECT ON Performance to comqhxl;

GRANT SELECT ON Cinema to rjl4aab;
GRANT SELECT ON Employee to rjl4aab;
GRANT SELECT ON FILM to rjl4aab;
GRANT SELECT ON Screen to rjl4aab;
GRANT SELECT ON Showings to rjl4aab;
GRANT SELECT ON Performance to rjl4aab;
```

5 marks

Table descriptions supplied. **5 marks**

Cinema table

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	CINEMA_NO	NUMBER(2,0)	No	(null)	1	(null)
2	CINEMA_NAME	VARCHAR2(50 BYTE)	Yes	(null)	2	(null)
3	LOCATION	VARCHAR2(90 BYTE)	Yes	(null)	3	(null)
4	MANAGER_EMP_NO	NUMBER(8,0)	Yes	(null)	4	(null)

Employee table

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	EMP_NO	NUMBER(8,0)	No	(null)	1	(null)
2	GIVEN_NAME	VARCHAR2(40 BYTE)	Yes	(null)	2	(null)
3	FAMILY_NAME	VARCHAR2(50 BYTE)	Yes	(null)	3	(null)
4	ADDRESS	VARCHAR2(40 BYTE)	Yes	(null)	4	(null)
5	DOB	DATE	Yes	(null)	5	(null)
6	SUPERVISOR_EMP_NO	NUMBER(8,0)	Yes	(null)	6	(null)
7	CINEMA_NO	NUMBER(2,0)	Yes	(null)	7	(null)

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMEN
1	FILM_NO	NUMBER(2,0)	No	(null)	1	(null)
2	FILM_NAME	VARCHAR2(50 BYTE)	No	(null)	2	(null)
3	DATE_OF_RELEASE	DATE	Yes	(null)	3	(null)

Film table

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMME
1	PERFORMANCE_ID	NUMBER(2,0)	No	(null)	1	(null)
2	CINEMA_NO	NUMBER(2,0)	Yes	(null)	2	(null)
3	SCREEN_NO	NUMBER(2,0)	Yes	(null)	3	(null)
4	FILM_NO	NUMBER(2,0)	Yes	(null)	4	(null)
5	DATES_NO	DATE	Yes	(null)	5	(null)
6	TOTAL_TAKINGS	NUMBER(6,2)	Yes	(null)	6	(null)

Performance table

Screen table

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	CINEMA_NO	NUMBER(2,0)	No	(null)	1	(null)
2	SCREEN_NO	NUMBER(2,0)	No	(null)	2	(null)
3	CAPACITY	NUMBER(3,0)	Yes	(null)	3	(null)

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	CINEMA_NO	NUMBER(2,0)	No	(null)	1	(null)
2	SCREEN_NO	NUMBER(2,0)	No	(null)	2	(null)
3	FILM_NO	NUMBER(2,0)	No	(null)	3	(null)
4	START_DATE	DATE	Yes	(null)	4	(null)
5	END_DATE	DATE	Yes	(null)	5	(null)

Showings table