

MINI PROJECT - 1

NAME: Nandhini V

DATE: 14-07-2025

Table Creation:

Query:

Create table CUSTOMER_MASTER

(

```
CUSTOMER_ID Varchar(10),  
CUSTOMER_NAME Varchar(30) NOT NULL,  
CONTACT_NO Varchar(10),  
CONTACT_ADD Varchar(100),  
DATE_OF_REGISTRATION Date NOT NULL,  
AGE Varchar(15) NOT NULL,  
Constraint MT_cts1 PRIMARY KEY(CUSTOMER_ID)
```

);

desc CUSTOMER_MASTER;

Output:

Result Grid						
Filter Rows:		Export:		Wrap Cell Content:		
	Field	Type	Null	Key	Default	Extra
▶	CUSTOMER_ID	varchar(10)	NO	PRI	NULL	
	CUSTOMER_NAME	varchar(30)	NO		NULL	
	CONTACT_NO	varchar(10)	YES		NULL	
	CONTACT_ADD	varchar(100)	YES		NULL	
	DATE_OF_REGISTRATION	date	NO		NULL	
	AGE	varchar(15)	NO		NULL	

Query:

Create table MOVIES_MASTER

(

```
MOVIE_ID Varchar(10),  
MOVIE_NAME Varchar(80) NOT NULL,  
RELEASE_DATE Varchar(30) NOT NULL,  
LANGUAGE Varchar(30),  
RATING int,  
DURATION_In_Minutes VARCHAR(10) NOT NULL,  
MOVIE_TYPE Varchar(100),
```

```

MOVIE_CATEGORY VARCHAR(40) NOT NULL,
DIRECTOR VARCHAR(60) NOT NULL,
LEAD_Actor_name1 Varchar(50) NOT NULL,
LEAD_Actor_name2 VARCHAR(60) NOT NULL,
RENTAL_COST INT,
Constraint MT_cts4 PRIMARY KEY(MOVIE_ID)
);
desc MOVIES_MASTER;

```

Output:

Field	Type	Null	Key	Default	Extra
MOVIE_ID	varchar(10)	NO	PRI	NULL	
MOVIE_NAME	varchar(80)	NO		NULL	
RELEASE_DATE	varchar(30)	NO		NULL	
LANGUAGE	varchar(30)	YES		NULL	
RATING	int	YES		NULL	
DURATION_In_Minutes	varchar(10)	NO		NULL	
MOVIE_TYPE	varchar(100)	YES		NULL	
MOVIE_CATEGORY	varchar(40)	NO		NULL	
DIRECTOR	varchar(60)	NO		NULL	
LEAD_Actor_name1	varchar(50)	NO		NULL	
LEAD_Actor_name2	varchar(60)	NO		NULL	
RENTAL_COST	int	YES		NULL	




Query:

```

Create table CUSTOMER_ISSUE_DETAILS
(
    ISSUE_ID Varchar(10) NOT NULL,
    CUSTOMER_ID Varchar(10) NOT NULL,
    MOVIE_ID VARCHAR(10),
    ISSUE_DATE Date NOT NULL,
    RETURN_DATE Date NOT NULL,
    ACTUAL_DATE_RETURN Date NOT NULL,
    Constraint MT_cts5 PRIMARY KEY(ISSUE_ID),
    Constraint MT_Mem FOREIGN KEY(CUSTOMER_ID) References
CUSTOMER_MASTER(CUSTOMER_ID),
    Constraint MT_Mem1 FOREIGN KEY(MOVIE_ID) References
MOVIES_MASTER(MOVIE_ID)
);
desc CUSTOMER_ISSUE_DETAILS;

```

Output:

Result Grid		 Filter Rows:	<input type="text"/>	Export:		Wrap Cell Content:	
	Field	Type	Null	Key	Default	Extra	
▶	ISSUE_ID	varchar(10)	NO	PRI	NULL		
	CUSTOMER_ID	varchar(10)	NO	MUL	NULL		
	MOVIE_ID	varchar(10)	YES	MUL	NULL		
	ISSUE_DATE	date	NO		NULL		
	RETURN_DATE	date	NO		NULL		
	ACTUAL_DATE_RETURN	date	NO		NULL		


Query:


Create table LIBRARY_CARD_MASTER


```
(
    CARD_ID Varchar(10),
    DESCRIPTION Varchar(30) NOT NULL,
    AMOUNT Double,
    NUMBER_OF_YEARS int NOT NULL,
    Constraint MT_cts2 PRIMARY KEY(CARD_ID)
);
desc LIBRARY_CARD_MASTER;
```

Output:

Result Grid


Filter Rows:

Export:


Wrap Cell Content:


	Field	Type	Null	Key	Default	Extra
▶	CARD_ID	varchar(10)	NO	PRI	NULL	
	DESCRIPTION	varchar(30)	NO		NULL	
	AMOUNT	double	YES		NULL	
	NUMBER_OF_YEARS	int	NO		NULL	

Query:

Create table CUSTOMER_CARD_DETAILS

```
(
    CUSTOMER_ID Varchar(10),
    CARD_ID VARCHAR(10),
    ISSUE_DATE DATE NOT NULL,
    Constraint MT_cts3 PRIMARY KEY(CUSTOMER_ID),
    Constraint MT_CTS41 FOREIGN KEY(CUSTOMER_ID) References
CUSTOMER_MASTER(CUSTOMER_ID),
    Constraint MT_CTS42 FOREIGN KEY(CARD_ID) References
LIBRARY_CARD_MASTER(CARD_ID)
);
desc CUSTOMER_CARD_DETAILS;
```

Output:

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Field	Type	Null	Key	Default	Extra
▶	CUSTOMER_ID	varchar(10)	NO	PRI	NULL	
	CARD_ID	varchar(10)	YES	MUL	NULL	
	ISSUE_DATE	date	NO		NULL	

Value Insertion:

Query:

insert into customer_master values

```
('C00001',' NITIN ','9830354218',' A/122, KALKAJI',' 2012-10-15', 22),
('C00002',' AGNESH ','8923156781',' 9/1,ANDHERI EAST',' 2012-11-01',
35),
('C00003',' T RAMACHANDRAN','9831289761','9/1,NANDABAKKAM',' 2012-11-02',
25),
('C00004',' ,RAJIB MITRA' ,',9830356781',' H/56, BLOCK1,JADAVPUR','
2012-11-21', 45),
('C00005',' SHIV PRASAD', NULL,', 2/2 PHASE II JAWAHAR NAGAR' ,',2012-12-25',
30),
('C00006',' 'AJAY GHOSH' ,',8763478901' ,',N/2,GANDHI COLONY DUM DUM',
'2012-12-30' ,20),
('C00007',' 'GEETHA REDDY' ,',8976167890',' AH 1/1 T NAGAR' ,',2012-12-31'
,30),
('C00008',' 'RIA NATRAJAN' ,',9856723190' ,',A/B GANDHI COLONY' ,',2013-01-01'
,45),
('C00009',' 'RAJAN PILLAI' ,',NULL,' A 1/66 KODAMBAKKAM ',',2013-01-02',
40),
('C00010',' 'RAGHAV SINGH', ',9675167890' ,',A/6 NEHRU JAWAHAR NAGAR',
'2013-03-02' ,50),
('C00011',' 'RAJ SEKHANRAN','8423178906' ,',A/1 MAYUR KUNJ' ,',2013-03-15' ,25);
```

```
select * from customer_master;
```

Output:

Result Grid						
Filter Rows:		Edit:		Export/Import:		Wrap Cell Content:
	CUSTOMER_ID	CUSTOMER_NAME	CONTACT_NO	CONTACT_ADD	DATE_OF_REGISTRATION	AGE
▶	C00001	NITIN	9830354218	A/122, KALKAJI	2012-10-15	22
	C00002	AGNESH	8923156781	9/1,ANDHERI EAST	2012-11-01	35
	C00003	T RAMACHANDRAN	9831289761	9/1,NANDABAKKAM	2012-11-02	25
	C00004	RAJIB MITRA	9830356781	H/56, BLOCK1,JADAVPUR	2012-11-21	45
	C00005	SHIV PRASAD	NULL	2/2 PHASE II JAWAHAR NAGAR	2012-12-25	30
	C00006	AJAY GHOSH	8763478901	N/2,GANDHI COLONY DUM DUM	2012-12-30	20
	C00007	GEETHA REDDY	8976167890	AH 1/1 T NAGAR	2012-12-31	30
	C00008	RIA NATRAJAN	9856723190	A/B GANDHI COLONY	2013-01-01	45
	C00009	RAJAN PILLAI	NULL	A 1/66 KODAMBAKKAM	2013-01-02	40
	C00010	RAGHAV SINGH	9675167890	A/6 NEHRU JAWAHAR NAGAR	2013-03-02	50
	C00011	RAJ SEKHANRAN	8423178906	A/1 MAYUR KUNJ	2013-03-15	25
*	NULL	NULL	NULL	NULL	NULL	NULL

Query:

insert into movies_master values

('M00001',' DIE HARD',' 1998 ','ENGLISH ',4 ,120 ,'UNIVERSAL','
ACTION',
'JOHN MCTIERNAN' ,'BRUCE WILLIS' , 'BONNIE BEDELIA' ,100),

('M00002',' THE DARK KNIGHT','2008',' ENGLISH', 5 ,90 , 'PARENTAL
GUIDENCE',' ACTION',
CHRISTOPHER NOLAN' ,
'CHRISTIAN BALE','HEALTH LEDGER', 100),

('M00003',' THE MATRIX',1999,' ENGLISH', 4, 120,' UNIVERSAL',' ACTION',
ANDY LARRY',' KEANU REEVES ',
'CARRIE-ANEE MOSS', 100),

('M00004',' INCEPTION', 2010 , 'ENGLISH ',5 ,120,' PARENTAL GUIDANCE
, 'ACTION',
'CHRISTOPHER NOLAN', 'LEONARDO DICAPRIO', JOSEPH GORDAN' ,100),

('M00005','OFFICE SPACE', 1999 , 'ENGLISH', 4, 95 , 'UNIVERSAL ', 'COMEDY',
,
'MIKE JUDGE', ' RON LIVINGSTON', ' JENNIFER ANISTON ',100),

('M00006', 'YOUNG FRANKENSTEIN','1974',' ENGLISH', 4 ,130,' UNIVERSAL',
COMEDY',
'MEL BROOKS', ' GENE WILDER', ' TERI GARR', 100),

```

('M00007',' SHAUN OF THE DEAD', 2004 , 'ENGLISH', 4 ,95,' UNIVERSAL',
'COMEDY',
'EDGAR WRIGHT',' SIMON PEGG','KATE ASHFIELD' ,100),

('M00008',' CASABLANCA', 1942,' ENGLISH', 3, 120 , 'UNIVERSAL','
ROMANCE',' MICHAEL
CURTIZ',' HUMPREY BOGART',' INGRID BERGMAN', 1000),

('M00009 ', 'THE NOTEBOOK', 2004 , 'ENGLISH', 3 ,120 , 'PARENTAL
GUIDENCE',' ROMANCE ',
'NICK CASSAVETES',' RYAN GOSLING',' RACHEL MCADAMS', 100),

('M00010 ', 'GONE WITH THE WIND ',1939 , 'ENGLISH', 3 ,120,' PARENTAL
GUIDENCE ', 'ROMANCE',
'VICTOR FLEMMING', 'CLARK GABLE',' VIVIEN LEIGH ',100),

('M00011 ', 'TITANIC', 1997 , 'ENGLISH ',3 ,120 , 'PARENTAL GUIDENCE','
ROMANCE'
,'JAMES CAMERON',' LEONARDO DICAPRIO ', 'KATE WINSLET', 100);

```

```
select * from movies_master;
```

Output:

MOVIE_ID	MOVIE_NAME	RELEASE_DATE	LANGUAGE	RATING	DURATION_In_Minutes	MOVIE_TYPE	MOVIE_CATEGORY	DIRECTOR	LEAD_Actor_name1	LEAD_Actor_name2	RENTAL_COST
M00001	DIE HARD	1998	ENGLISH	4	120	UNIVERSAL	ACTION	JOHN MCTIERNAN	BRUCE WILLIS	BONNIE BEDELIA	100
M00002	THE DARK KNIGHT	2008	ENGLISH	5	90	PARENTAL GUIDENCE	ACTION	CHRISTOPHER NOLAN	CHRISTIAN BALE	HEALTH LEDGER	100
M00003	THE MATRIX	1999	ENGLISH	4	120	UNIVERSAL	ACTION	ANDY LARRY	KEANU REEVES	CARRIE-ANEE MOSS	100
M00004	INCEPTION	2010	ENGLISH	5	120	PARENTAL GUIDENCE	ACTION	CHRISTOPHER NOLAN	LEONARDO DICAPRIO	JOSEPH GORDAN	100
M00005	OFFICE SPACE	1999	ENGLISH	4	95	UNIVERSAL	COMEDY	MIKE JUDGE	RON LIVINGSTON	JENNIFER ANISTON	100
M00006	YOUNG FRANKENSTEIN	1974	ENGLISH	4	130	UNIVERSAL	COMEDY	MEL BROOKS	GENE WILDER	TERI GARR	100
M00007	SHAUN OF THE DEAD	2004	ENGLISH	4	95	UNIVERSAL	COMEDY	EDGAR WRIGHT	SIMON PEGG	KATE ASHFIELD	100
M00008	CASABLANCA	1942	ENGLISH	3	120	UNIVERSAL	ROMANCE	MICHAEL CURTIZ	HUMPREY BOGART	INGRID BERGMAN	1000
M00009	THE NOTEBOOK	2004	ENGLISH	3	120	PARENTAL GUIDENCE	ROMANCE	NICK CASSAVETES	RYAN GOSLING	RACHEL MCADAMS	100
M00010	GONE WITH THE WIND	1939	ENGLISH	3	120	PARENTAL GUIDENCE	ROMANCE	VICTOR FLEMMING	CLARK GABLE	VIVIEN LEIGH	100
M00011	TITANIC	1997	ENGLISH	3	120	PARENTAL GUIDENCE	ROMANCE	JAMES CAMERON	LEONARDO DICAPRIO	KATE WINSLET	100

Query:

```
insert into customer_issue_details values
```

```

('I00001','C00001','M00001','2012-10-15','2012-10-17','2012-10-17'),
('I00002', 'C00002','M00002','2012-11-02' , '2012-11-04','2012-11-05'),
('I00003','C00002','M00002','2012-12-02','2012-12-04','2012-12-03'),
('I00004','C00003','M00003','2012-11-02','2012-11-04','2012-11-10'),
('I00005','C00003','M00004','2012-11-10','2012-11-12','2012-11-12'),
('I00006','C00003','M00005','2012-11-12','2012-11-14','2012-11-14'),
('I00007','C00004','M00006 ', '2012-11-21','2012-11-23','2012-11-24'),
('I00008','C00010','M00008','2013-03-02','2013-03-04','2013-03-05'),
('I00009','C00011','M00010 ', '2013-03-16','2013-03-18','2013-03-18'),

```

```
(I00010',C00004'      ,M00007','2012-11-25','2012-11-27','2012-11-27'),
(I00011',C00004'      ,M00007','2012-11-28','2012-11-30','2012-11-30'),
(I00012',C00001',M00001','2013-11-28','2013-11-30','2013-11-30'),
(I00013',C00003',M00001','2012-12-03','2012-12-05','2012-12-05'),
(I00014',C00003',M00010  ','2013-01-02','2013-01-04','2013-01-05'),
(I00015',C00003',M00011  ','2013-02-03','2013-02-05','2013-02-06'),
(I00016',C00003',M00011  ','2013-03-05','2013-03-07','2013-03-07'),
(I00017',C00003',M00008','2013-05-13','2013-04-17','2013-04-17'),
(I00018',C00002',M00010  ','2015-01-15','2015-01-17','2015-01-17'),
(I00019',C00004',M00001','2012-11-15      ','2012-11-17','2012-11-17');
```

```
select * from customer_issue_details;
```

Output:

Result Grid						
Filter Rows:		Edit:		Export/Import:		Wrap Cell Content: IA
	ISSUE_ID	CUSTOMER_ID	MOVIE_ID	ISSUE_DATE	RETURN_DATE	ACTUAL_DATE_RETURN
▶	I00001	C00001	M00001	2012-10-15	2012-10-17	2012-10-17
	I00002	C00002	M00002	2012-11-02	2012-11-04	2012-11-05
	I00003	C00002	M00002	2012-12-02	2012-12-04	2012-12-03
	I00004	C00003	M00003	2012-11-02	2012-11-04	2012-11-10
	I00005	C00003	M00004	2012-11-10	2012-11-12	2012-11-12
	I00006	C00003	M00005	2012-11-12	2012-11-14	2012-11-14
	I00007	C00004	M00006	2012-11-21	2012-11-23	2012-11-24
	I00008	C00010	M00008	2013-03-02	2013-03-04	2013-03-05
	I00009	C00011	M00010	2013-03-16	2013-03-18	2013-03-18
	I00010	C00004	M00007	2012-11-25	2012-11-27	2012-11-27
	I00011	C00004	M00007	2012-11-28	2012-11-30	2012-11-30
	I00012	C00001	M00001	2013-11-28	2013-11-30	2013-11-30
	I00013	C00003	M00001	2012-12-03	2012-12-05	2012-12-05
	I00014	C00003	M00010	2013-01-02	2013-01-04	2013-01-05
	I00015	C00003	M00011	2013-02-03	2013-02-05	2013-02-06
	I00016	C00003	M00011	2013-03-05	2013-03-07	2013-03-07
	I00017	C00003	M00008	2013-05-13	2013-04-17	2013-04-17
	I00018	C00002	M00010	2015-01-15	2015-01-17	2015-01-17
	I00019	C00004	M00001	2012-11-15	2012-11-17	2012-11-17
*	NULL	NULL	NULL	NULL	NULL	NULL

Query:

```
insert into library_card_master values
```

```
('CRD001', 'SILVER CARD'      ,1000, 1),
('CRD002', 'GOLD CARD ',2000 ,2),
('CRD003', 'PLATINUM CARD'    ,3000, 3),
('CRD004', 'DIAMOND CARD',    4000 ,5);
```

```
select * from library_card_master;
```

Output:

Result Grid	Filter Rows:	Edit:	Export/Import:	Wrap Cell Content:
CARD_ID	DESCRIPTION	AMOUNT	NUMBER_OF_YEARS	
CRD001	SILVER CARD	1000	1	
CRD002	GOLD CARD	2000	2	
CRD003	PLATINUM CARD	3000	3	
CRD004	DIAMOND CARD	4000	5	
NULL	NULL	NULL	NULL	

Query:

Insert into CUSTOMER_CARD_DETAILS Values

('C00001','CRD001','2012-05-13'),

('C00002','CRD002','2012-05-13'),

('C00003','CRD002','2013-05-13'),

('C00004','CRD003','2013-05-13'),

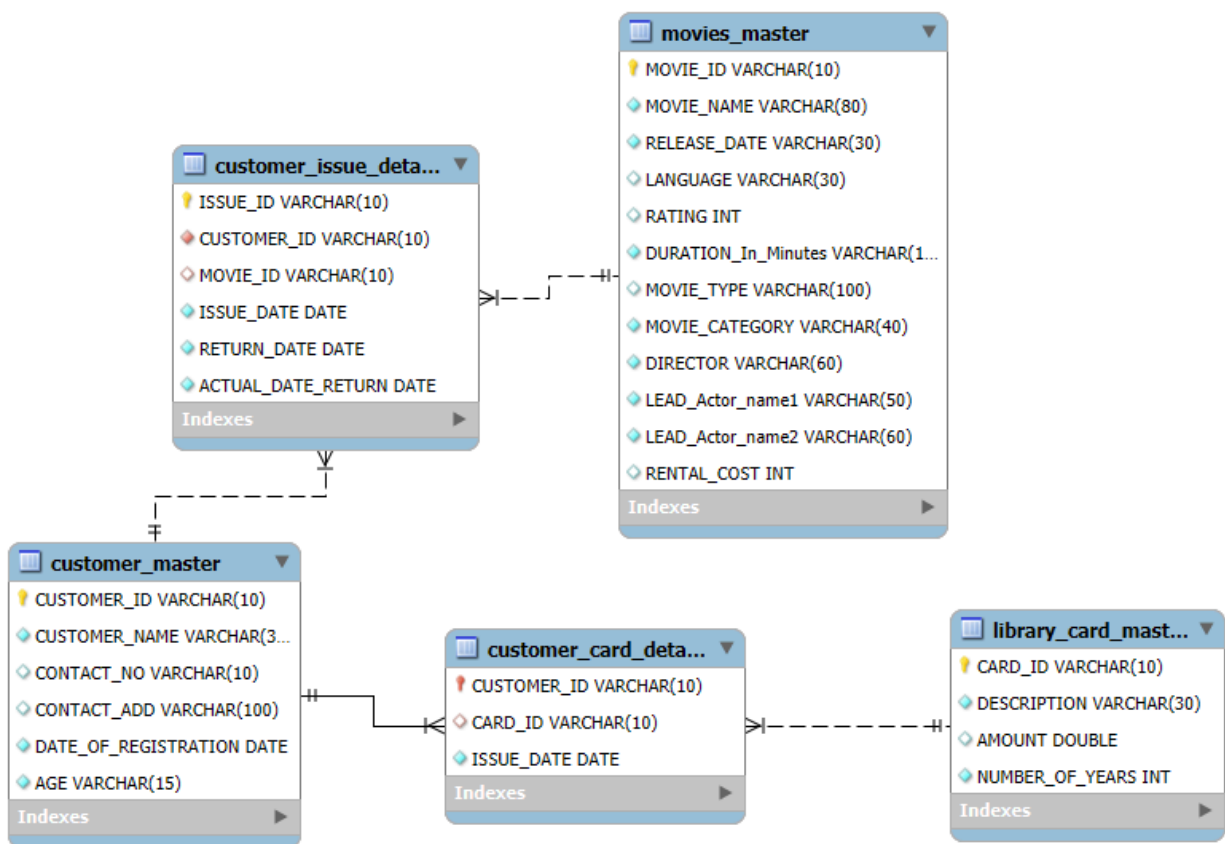
('C00005','CRD003','2012-05-13');

select * from CUSTOMER_CARD_DETAILS;

Output:

Result Grid	Filter Rows:	Edit:	Export/Import:	Wrap Cell Content:
CUSTOMER_ID	CARD_ID	ISSUE_DATE		
C00001	CRD001	2012-05-13		
C00002	CRD002	2012-05-13		
C00003	CRD002	2013-05-13		
C00004	CRD003	2013-05-13		
C00005	CRD003	2012-05-13		
NULL	NULL	NULL		

ER Diagram:



Questions & Solutions:

1. Write a query to display movie names and number of times that movie is issued to customers. In case movies are never issued to customers display number of times as 0.

Display the details in sorted order based on number of times (in descending order) and then by movie name (in ascending order).

The Alias name for the number of movies issued is ISSUE_COUNT.

Query:

```
select movie_name, count(issue_id) ISSUE_COUNT from customer_issue_details right join
movies_master using(movie_id) group by movie_id order by ISSUE_COUNT desc, movie_name
asc;
```

Output:

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
movie_name	ISSUE_COUNT		
▶ DIE HARD	4		
GONE WITH THE WIND	3		
CASABLANCA	2		
SHAUN OF THE DEAD	2		
THE DARK KNIGHT	2		
TITANIC	2		
INCEPTION	1		
THE MATRIX	1		
OFFICE SPACE	1		
YOUNG FRANKENSTEIN	1		
THE NOTEBOOK	0		

2. Write a query to display id, name, age, contact no of customers whose age is greater than 25 and who have registered in the year 2012. Display contact no in the below format +91-XXX-XXX-XXXX example +91-987-678-3434 and use the alias name as "CONTACT_ISD". If the contact no is null then display as 'N/A' Sort all the records in ascending order based on age and then by name.

Query:

```
select customer_id, customer_name, age, coalesce(CONCAT('+91-',
SUBSTRING(contact_no, 1, 3), '-',
SUBSTRING(contact_no, 4, 3), '-',
SUBSTRING(contact_no, 7, 4)), 'N/A') CONTACT_ISD
from customer_master where age > 25 and year(DATE_OF_REGISTRATION) = 2012 order by
age asc, customer_name asc;
```

Output:

	customer_id	customer_name	age	CONTACT_ISD
▶	C00007	GEETHA REDDY	30	+91-897-616-7890
	C00005	SHIV PRASAD	30	N/A
	C00002	AGNESH	35	+91-892-315-6781
	C00004	RAJIB MITRA	45	+91-983-035-6781

3. Write a query to display the movie category and number of movies in that category. Display records based on number of movies from higher to lower order and then by movie category in ascending order.

Hint: Use NO_OF_MOVIES as alias name for number of movies.

Query:

```
select movie_category, count(movie_id) NO_OF_MOVIES from movies_master group by  
MOVIE_CATEGORY order by NO_OF_MOVIES desc, movie_category asc;
```

Output:

	movie_category	NO_OF_MOVIES
▶	ACTION	4
	ROMANCE	4
	COMEDY	3

4. Write a query to display the number of customers having card with description "Gold card". Use CUSTOMER_COUNT as alias name for number of customers.

Query:

```
select count(customer_id) CUSTOMER_COUNT from library_card_master join  
customer_card_details using(card_id) where DESCRIPTION = 'GOLD CARD';
```

Output:



	CUSTOMER_COUNT
▶	2

5. Write a query to display the customer id, customer name, year of registration, library card id, card issue date of all the customers who hold library card. Display the records sorted by customer name in descending order. Use REGISTERED_YEAR as alias name for year of registration.

Query:

```
select customer_id, customer_name, year(date_of_registration) REGISTERED_YEAR, card_id, issue_date from customer_master join customer_card_details using(customer_id) order by CUSTOMER_NAME desc;
```

Output:

Result Grid					
Filter Rows: <input type="text"/>					
Export:  Wrap Cell Content: 					
	customer_id	customer_name	REGISTERED_YEAR	card_id	issue_date
▶	C00005	SHIV PRASAD	2012	CRD003	2012-05-13
	C00004	RAJIB MITRA	2012	CRD003	2013-05-13
	C00003	T RAMACHANDRAN	2012	CRD002	2013-05-13
	C00001	NITIN	2012	CRD001	2012-05-13
	C00002	AGNESH	2012	CRD002	2012-05-13



6. Write a query to display issue id, customer id, customer name for the customers who have paid fine and whose name starts with 'R'. Fine is calculated based on return date and actual date of return. If the date of actual return is after date of return then fine need to be paid by the customer.

Display the records sorted in ascending order based on customer name.

Query:

```
select issue_id, customer_id, customer_name from customer_issue_details join customer_master using(customer_id) where actual_date_return > return_date and CUSTOMER_NAME like 'R%' order by CUSTOMER_NAME;
```

Output:

Result Grid			
Filter Rows: <input type="text"/>			
Export:  Wrap Cell Content: 			
	issue_id	customer_id	customer_name
▶	I00008	C00010	RAGHAV SINGH
	I00007	C00004	RAJIB MITRA

7. Write a query to display customer id, customer name, card id, card description and card amount in dollars of customers who have taken movie on the same day the library card is registered. For Example Assume John registered a library card on 12th Jan 2013 and he took a movie on 12th Jan 2013 then display his details. $AMOUNT_DOLLAR = amount/85.8$ and round it to zero decimal places and display as \$Amount. Example Assume 500 is the amount then dollar value will be \$10. Use AMOUNT_DOLLAR as alias name for amount in dollar. Display the records in ascending order based on customer name.

Query:

```
select customer_id, customer_name, card_id, description, concat('$',round(amount/85.8,0))
AMOUNT_DOLLAR from customer_master join customer_issue_details cid
using(customer_id)join customer_card_details ccd using(customer_id) join library_card_master
using(card_id) where cid.ISSUE_DATE = ccd.ISSUE_DATE order by customer_name;
```

Output:

Result Grid		Filter Rows:		Export:		Wrap Cell Content:	
	customer_id	customer_name	card_id	description	AMOUNT_DOLLAR		
▶	C00003	T RAMACHANDRAN	CRD002	GOLD CARD	\$23		

8. Write a query to display the customer name and number of movies issued to that customer sorted by customer name in ascending order. If a customer has not been issued with any movie then display 0. Use MOVIE_COUNT as alias name for number of movies issued.

Query:

```
select customer_name, count(issue_id) MOVIE_COUNT from customer_master left join
customer_issue_details using(customer_id) group by CUSTOMER_ID order by
customer_name;
```

Output:

Result Grid		Filter Rows:		Export:		Wrap Cell Content:	
	customer_name	MOVIE_COUNT					
▶	AGNESH	3					
	NITIN	2					
	T RAMACHANDRAN	8					
	AJAY GHOSH	0					
	GEETHA REDDY	0					
	RAGHAV SINGH	1					
	RAJ SEKHANRAN	1					
	RAJAN PILLAI	0					
	RAJIB MITRA	4					
	RIA NATRAJAN	0					
	SHIV PRASAD	0					

9. Write a query to display the issue id, issue date, customer id, customer name and contact number for videos that are issued in the year 2013. Display the records in descending order based on issue date of the video.

Query:

```
select issue_id, issue_date, customer_id, customer_name, contact_no from customer_master
join customer_issue_details using(customer_id) where year(ISSUE_DATE) = 2013 order by
ISSUE_DATE desc;
```

Output:

Result Grid			Filter Rows:	<input type="text"/>	Export:		Wrap Cell Content:	
	issue_id	issue_date	customer_id	customer_name	contact_no			
▶	I00012	2013-11-28	C00001	NITIN	9830354218			
	I00017	2013-05-13	C00003	T RAMACHANDRAN	9831289761			
	I00009	2013-03-16	C00011	RAJ SEKHANRAN	8423178906			
	I00016	2013-03-05	C00003	T RAMACHANDRAN	9831289761			
	I00008	2013-03-02	C00010	RAGHAV SINGH	9675167890			
	I00015	2013-02-03	C00003	T RAMACHANDRAN	9831289761			
	I00014	2013-01-02	C00003	T RAMACHANDRAN	9831289761			

10. Write a query to display the director's name, number of movies directed by the director who directed more than one movie. Display the director name in capital letters. Use DIRECTOR_NAME as alias name for director name column Display the records sorted in ascending order based on director_name.

Query:

```
select upper(director) DIRECTOR_NAME, count(movie_id) from movies_master group by
director having count(movie_id) > 1 order by director;
```

Output:

Result Grid			Filter Rows:	<input type="text"/>	Export:		Wrap Cell Content:	
	DIRECTOR_NAME	count(movie_id)						
▶	CHRISTOPHER NOLAN	2						