

Please follow instructions given below.

Write a query to display movie names and number of times that movie is issued to customers. Incase 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.

```
SELECT movie_name, count(cid.movie_id) ISSUE_COUNT FROM movies_master mm LEFT JOIN
customer_issue_details cid ON mm.movie_id=cid.movie_id GROUP BY movie_name ORDER BY
Issue_count DESC, movie_name;
```

Please follow instructions given below.

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.

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

Please follow instructions given below.

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.

```
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;
```

Please follow instructions given below.

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

```
SELECT count(customer_id) CUSTOMER_COUNT FROM library_card_master lcm INNER JOIN  
customer_card_details ccd ON lcm.card_id=ccd.card_id WHERE description='Gold Card';
```

Please follow instructions given below.

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.

```
SELECT c.customer_id, c.customer_name, extract(year from c.date_of_registration)  
REGISTERED_YEAR, cd.card_id, cd.issue_date FROM customer_master c join customer_card_details cd on  
c.customer_id=cd.customer_id ORDER BY c.customer_name DESC;
```

Please follow instructions given below.

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.

```
SELECT issue_id ,cid.customer_id, customer_name FROM customer_issue_details cid INNER JOIN
customer_master cm ON cid.customer_id=cm.customer_id WHERE actual_date_of_return>return_date
and customer_name like 'R%' order by customer_name;
```

Please follow instructions given below.

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/52.42 and round it to zero decimal places and display as \$Amount.
Example Assume 500 is the amount then dollar value will be \$10.

Hint: Use AMOUNT_DOLLAR as alias name for amount in dollar.

Display the records in ascending order based on customer name.

```
SELECT ccd.customer_id, customer_name, ccd.card_id, description,concat('$',round(amount/52.42,0))
AMOUNT_DOLLAR FROM customer_master cm INNER JOIN customer_card_details ccd ON
cm.customer_id=ccd.customer_id INNER JOIN library_card_master lcm ON ccd.card_id=lcm.card_id
INNER JOIN customer_issue_details cid ON cid.customer_id = cm.customer_id WHERE
cm.date_of_registration=cid.issue_date order by customer_name;
```

Please follow instructions given below.

Write a query to display the customer id, customer name,contact number and address of customers who have taken movies from library without library card and whose address ends with 'Nagar'.

Display customer name in upper case. Hint: Use CUSTOMER_NAME as alias name for customer name. Display the details sorted in ascending order based on customer name.

```
SELECT customer_id , upper(customer_name) CUSTOMER_NAME,contact_no,contact_address FROM customer_master WHERE customer_id NOT IN ( select customer_id from customer_card_details ) AND customer_id IN ( SELECT customer_id from customer_issue_details ) and contact_address like '%Nagar' order by customer_name ;
```

Please follow instructions given below.

Write a query to display the movie id, movie name,release year,director name of movies acted by the leadactor1 who acted maximum number of movies .Display the records sorted in ascending order based on movie name.

```
select movie_id,movie_name,release_year,director_name from movies_master where lead_actor_name1 in(select lead_actor_name1 from (select lead_actor_name1,count(movie_id)ct from movies_master group by lead_actor_name1)t where t.ct>=all(select count(movie_id) from movies_master group by lead_actor_name1)) order by movie_name;
```

Please follow instructions given below.

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.
Hint: Use MOVIE_COUNT as alias name for number of movies issued.

```
SELECT customer_name, count(movie_id) MOVIE_COUNT FROM customer_master cm LEFT OUTER JOIN customer_issue_details cid ON cm.customer_id=cid.customer_id GROUP BY customer_name ORDER BY customer_name;
```

Please follow instructions given below.

Write a query to display serial number, issue id, customer id, customer name, movie id and movie name of all the videos that are issued and display in ascending order based on serial number.

Serial number can be generated from the issue id , that is last two characters of issue id is the serial number.

For Example Assume the issue id is I00005 then the serial number is 05

Hint: Alias name for serial number is 'SERIAL_NO'


```
SELECT substring(issue_id,5,2) SERIAL_NO,issue_id, cid.customer_id,customer_name, mm.movie_id, movie_name FROM customer_issue_details cid INNER JOIN customer_master cm ON cid.customer_id=cm.customer_id INNER JOIN movies_master mm ON cid.movie_id=mm.movie_id order by SERIAL_NO ASC;
```

Please follow instructions given below.

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.

```
SELECT issue_id, issue_date, cid.customer_id, customer_name, contact_no FROM customer_issue_details cid INNER JOIN customer_master cm ON cid.customer_id=cm.customer_id where extract(year from issue_date)=2013 order by issue_date DESC
```

Please follow instructions given below.

Write a query to display movie id ,movie name and actor names of movies which are not issued to any customers.
 Actors Name to be displayed in the below format.LEAD_ACTOR_ONE space ambersant space LEAD_ACTOR_TWO.

Example: Assume lead actor one's name is "Jack Tomson" and Lead actor two's name is "Maria" then Actors name will be "Jack Tomsom & Maria"Hint:Use ACTORS as alias name for actors name.
 Display the records in ascending order based on movie name.

```
SELECT movie_id, movie_name,concat(lead_actor_name1,' & ',lead_actor_name2) ACTORS FROM
movies_master WHERE movie_id NOT IN ( SELECT movie_id  from customer_issue_details ) order by
movie_name asc;
```

Please follow instructions given below.

Write a query to display the director's name, movie name and lead_actor_name1 of all the movies directed by the director who directed more than one movie. Display the directors 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 and then by movie_name in descending order.

```
SELECT upper(director_name) DIRECTOR_NAME,movie_name,lead_actor_name1 FROM movies_master
WHERE director_name in (SELECT director_name FROM movies_master GROUP BY director_name
HAVING count(movie_id)>1) order by director_name, movie_name desc;
```

Please follow instructions given below.

Write a query to display number of customers who have registered in the library in the year 2012 and who have given/provided contact number.
 Hint:Use NO_OF_CUSTOMERS as alias name for number of customers.

```
SELECT count(customer_id) NO_OF_CUSTOMERS FROM customer_master WHERE extract(year from
date_of_registration)=2012 and contact_no is not null
```

Please follow instructions given below.

Write a query to display the customer's name, contact number, library card id and library card description of all the customers irrespective of customers holding a library card. If customer contact number is not available then display his address. Display the records sorted in ascending order based on customer name. Hint: Use CONTACT_DETAILS as alias name for customer contact.

Please follow instructions given below.

```
SELECT cid.customer_id, customer_name,count(movie_id) NO_OF_TIMES FROM customer_master cm
INNER JOIN customer_issue_details cid ON cm.customer_id=cid.customer_id group by customer_id,
customer_name,movie_id having count(movie_id)>1 order by customer_name desc;
```

Write a query to display customer id, customer name, contact number, movie category and number of movies issued to each customer based on movie category who has been issued with more than one movie in that category. Example: Display contact number as "+91-876-456-2345" format.

Hint: Use NO OF MOVIES as alias name for number of movies column.

Hint: Use CONTACT ISD as alias name for contact number.

 Display the records sorted in ascending order based on customer name and then by movie category.

```
SELECT cid.customer_id,customer_name,concat('+91-',substring(contact_no,1,3),'-','substring(contact_no,4,3),'-',substring(contact_no,7))
CONTACT_ISD,movie_category,count(movie_category) NO_OF_MOVIES FROM customer_issue_details
cid INNER JOIN movies_master mm ON cid.movie_id=mm.movie_id INNER JOIN customer_master cm
ON cm.customer_id=cid.customer_id group by
customer_id,customer_name,CONTACT_ISD,movie_category having count(movie_category)>1 order by
customer_name, movie_category;
```

Please follow instructions given below.

Write a query to display customer id and customer name of customers who has been issued with maximum number of movies and customer who has been issued with minimum no of movies.

For example Assume customer John has been issued 5 movies, Ram has been issued 10 movies and Kumar has been issued 2 movies. The name and id of Ram should be displayed for issuing maximum movies and Kumar should be displayed for issuing minimum movies. Consider only the customers who have been issued with atleast 1 movie Customer(s) who has/have been issued the maximum number of movies must be displayed first followed by the customer(s) who has/have been issued with the minimum number of movies. In case of multiple customers who have been displayed with the maximum or minimum number of movies, display the records sorted in ascending order based on customer name.

```
(select cid.customer_id , customer_name FROM customer_master cm INNER JOIN
customer_issue_details cid
ON cm.customer_id=cid.customer_id
group by customer_id , customer_name
having count(movie_id)>=ALL(select count(movie_id)
FROM customer_issue_details
group by customer_id) order by customer_name)
```


UNION ALL

```
(select cid.customer_id , customer_name FROM customer_master cm INNER JOIN
customer_issue_details cid
ON cm.customer_id=cid.customer_id
group by customer_id , customer_name
having count(movie_id)<=ALL(select count(movie_id)
FROM customer_issue_details
group by customer_id) order by customer_name) ;
```

Please follow instructions given below.

Write a query to display the customer id , customer name and number of times movies have been issued from Comedy category. Display only for customers who has taken more than once.

Hint: Use NO_OF_TIMES as alias name

Display the records in ascending order based on customer name.

```
SELECT cid.customer_id,customer_name,count(cid.movie_id) NO_OF_TIMES FROM customer_master
cm INNER JOIN customer_issue_details cid ON cm.customer_id=cid.customer_id INNER JOIN
movies_master mm
```

```
ON cid.movie_id=mm.movie_id WHERE movie_category='Comedy' GROUP BY customer_name HAVING
count(cid.customer_id)>1 order by customer_name;
```

Please follow instructions given below.

Write a query to display customer id and total rent paid by the customers who are issued with the videos. Need not display the customers who has not taken / issued with any videos. Hint: Alias Name for total rent paid is TOTAL_COST. Display the records sorted in ascending order based on customer id

```
SELECT cid.customer_id, sum(rental_cost) TOTAL_COST FROM customer_issue_details cid INNER JOIN  
movies_master mm ON cid.movie_id=mm.movie_id GROUP BY customer_id order by customer_id;
```