

Bank Management System Queries:

Bank Management System Queries:

1. Please follow instructions given below.

Write a query to display account number, customer's number, customer's firstname, lastname, account opening date.

Display the records sorted in ascending order based on account number.

```

SELECT account_number, am.customer_number, firstname, lastname, account_opening_date
FROM customer_master cm JOIN account_master am
ON cm.customer_number=am.customer_number
ORDER BY account_number;
    
```

ACCOUNT_NUMBER	CUSTOMER_NUMBER	FIRSTNAME	LASTNAME	ACCOUNT_OPENING_DATE
A00001	C00001	RAMESH	SHARMA	2012-12-15
A00002	C00002	AVINASH	MINHA	2012-06-12
A00003	C00003	RAHUL	RASTOGI	2012-05-17
A00004	C00002	AVINASH	MINHA	2013-01-27
A00005	C00006	CHITRESH	BARWE	2012-12-17
A00006	C00007	AMIT	BORKAR	2010-08-12
A00007	C00007	AMIT	BORKAR	2012-10-02
A00008	C00001	RAMESH	SHARMA	2009-11-09
A00009	C00003	RAHUL	RASTOGI	2008-11-30
A00010	C00004	PARUL	GANDHI	2013-03-01

2.Please follow instructions given below.

Write a query to display the number of customer's from Delhi. Give the count an alias name of Cust_Count.

SELECT count(customer_number) Cust_Count

FROM customer_master

WHERE customer_city='Delhi'

CUST_COUNT
4

3.Please follow instructions given below.

Write a query to display the customer number, customer firstname,account number for the customer's whose accounts were created after 15th of any month.

Display the records sorted in ascending order based on customer number and then by account number.

```
SELECT am.customer_number, firstname, account_number
FROM customer_master cm JOIN account_master am
ON cm.customer_number=am.customer_number
WHERE day(account_opening_date)>15
ORDER BY am.customer_number, account_number
```

CUSTOMER_NUMBER	FIRSTNAMEMiddle_Name	ACCOUNT_NUMBER
C00002	AVINASH	A00004
C00003	RAHUL	A00003
C00003	RAHUL	A00009
C00006	CHITRESH	A00005

4.Please follow instructions given below.

Write a query to display customer number, customer's first name, account number where the account status is terminated.

Display the records sorted in ascending order based on customer number and then by account number.

```
SELECT am.customer_number,firstname, account_number
FROM customer_master cm JOIN account_master am
ON cm.customer_number=am.customer_number
WHERE account_status='Terminated'
ORDER BY am.customer_number, account_number
```

CUSTOMER_NUMBER	FIRSTNAME	ACCOUNT_NUMBER
C00001	RAMESH	A00008
C00003	RAHUL	A00009

5. Please follow instructions given below.

Write a query to display the total number of withdrawals and total number of deposits being done by customer whose customer number ends with 001. The query should display transaction type and the number of transactions. Give an alias name as Trans_Count for number of transactions.

Display the records sorted in ascending order based on transaction type.

```

SELECT transaction_type, count(transaction_number) Trans_Count
FROM account_master am JOIN transaction_details td
ON am.account_number=td.account_number
WHERE customer_number like '%001'
GROUP BY transaction_type
ORDER BY transaction_type

```

TRANSACTION_TYPE	TRANS_COUNT
DEPOSIT	3
WITHDRAWAL	3

6. Please follow instructions given below.

Write a query to display the number of customers who have registration but no account in the bank.

Give the alias name as Count_Customer for number of customers.

```

SELECT count(customer_number) Count_Customer
FROM customer_master
WHERE customer_number NOT IN (SELECT customer_number FROM account_master)

```

COUNT_CUSTOMER
4

7.Please follow instructions given below.

Write a query to display account number and total amount deposited by each account holder (Including the opening balance). Give the total amount deposited an alias name of Deposit_Amount. Display the records in sorted order based on account number.

```

SELECT td.account_number, opening_balance+sum(transaction_amount) Deposit_Amount
FROM account_master am INNER JOIN transaction_details td
ON am.account_number=td.account_number
WHERE transaction_type='deposit'
GROUP BY account_number
ORDER BY account_number

```

ACCOUNT_NUMBER	DEPOSIT_AMOUNT
A00001	10000
A00002	6000
A00007	17000

8.Please follow instructions given below.

Write a query to display the number of accounts opened in each city .The Query should display Branch City and number of accounts as No_of_Accounts.For the branch city where we don't have any accounts opened display 0. Display the records in sorted order based on branch city.

```
select branch_master.branch_city, count(account_master.account_number) as No_of_Accounts from  
branch_master left join account_master on account_master.branch_id=branch_master.branch_id
```

```
group by branch_master.branch_city order by branch_city;
```

BRANCH_CITY	NO_OF_ACCOUNTS
CHENNAI	0
DELHI	6
KOLKATA	0
MUMBAI	4

9.Please follow instructions given below.

Write a query to display the firstname of the customers who have more than 1 account. Display the records in sorted order based on firstname.

```
select firstname
```

```
FROM customer_master cm INNER JOIN account_master am
```

```
ON cm.customer_number=am.customer_number
```

```
group by firstname
```

```
having count(account_number)>1
```

```
order by firstname;
```

FIRSTNAME

AMIT
AVINASH
RAHUL
RAMESH

10. Please follow instructions given below.

Write a query to display the customer number, customer firstname, customer lastname who has taken loan from more than 1 branch.

Display the records sorted in order based on customer number.

```

SELECT Id.customer_number, firstname, lastname
FROM customer_master cm INNER JOIN loan_details Id
ON cm.customer_number=Id.customer_number
GROUP BY customer_number
HAVING count(branch_id)>1
ORDER BY customer_number

```

CUSTOMER_NUMBER	FIRSTNAME	LASTNAME
C00001	RAMESH	SHARMA
C00002	AVINASH	MINHA

11. Please follow instructions given below.

Write a query to display the customer's number, customer's firstname, customer's city and branch city where the city of the customer and city of the branch is different.

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

```

select customer_master.customer_number, firstname, customer_city, branch_city
from account_master inner join customer_master on account_master.customer_number =
customer_master.customer_number

inner join branch_master on account_master.branch_id = branch_master.branch_id

where customer_city != branch_city order by customer_master.customer_number;

```

CUSTOMER_NUMBER	FIRSTNAME	CUSTOMER_CITY	BRANCH_CITY
C00002	AVINASH	DELHI	MUMBAI
C00003	RAHUL	DELHI	MUMBAI
C00007	AMIT	MUMBAI	DELHI

12. Please follow instructions given below.

Write a query to display the number of clients who have asked for loans but they don't have any account in the bank though they are registered customers. Give the count an alias name of Count.

```

SELECT count(ld.customer_number) Count

```

```

FROM customer_master cm INNER JOIN loan_details ld

```

```

ON cm.customer_number=ld.customer_number

```

```

WHERE cm.customer_number NOT IN ( SELECT customer_number FROM account_master)

```

(Or)

```

select count(customer_number) as Count from customer_master where customer_number not in

```

```

(select customer_number from account_master) and customer_number in

```

```

(select customer_number from loan_details);

```

COUNT
2

13. Please follow instructions given below.

Write a query to display the account number who has done the highest transaction.

For example the account A00023 has done 5 transactions i.e. suppose 3 withdrawal and 2 deposits. Whereas the account A00024 has done 3 transactions i.e. suppose 2 withdrawals and 1 deposit. So account number of A00023 should be displayed.

In case of multiple records, display the records sorted in ascending order based on account number.

```
SELECT td.account_number
FROM account_master am INNER JOIN transaction_details td
ON am.account_number=td.account_number
group by td.account_number
having count(td.transaction_number)>=ALL
(SELECT count(td.transaction_number)
FROM account_master am INNER JOIN transaction_details td
ON am.account_number=td.account_number
group by td.account_number) order by am.account_number;
```

ACCOUNT_NUMBER
A00001

14. Please follow instructions given below.

Write a query to show the branch name, branch city where we have the maximum customers.

For example the branch B00019 has 3 customers, B00020 has 7 and B00021 has 10. So branch id B00021 is having maximum customers. If B00021 is Koramangla branch Bangalore, Koramangla branch should be displayed along with city name Bangalore.

In case of multiple records, display the records sorted in ascending order based on branch name.

```
select branch_name,branch_city
```

```

FROM branch_master INNER JOIN account_master
ON branch_master.branch_id=account_master.branch_id
group by branch_name
having count(customer_number)>=ALL
(select count(customer_number)
FROM branch_master INNER JOIN account_master
ON branch_master.branch_id=account_master.branch_id
group by branch_name) order by branch_name;

```

BRANCH_NAME	BRANCH_CITY
ASAF ALI ROAD	DELHI

15.Please follow instructions given below.

Write a query to display all those account number, deposit, withdrawal where withdrawal is more than deposit amount. Hint: Deposit should include opening balance as well.

For example A00011 account opened with Opening Balance 1000 and A00011 deposited 2000 rupees on 2012-12-01 and 3000 rupees on 2012-12-02. The same account i.e A00011 withdrawn 3000 rupees on 2013-01-01 and 7000 rupees on 2013-01-03. So the total deposited amount is 6000 and total withdrawal amount is 10000. So withdrawal amount is more than deposited amount for account number A00011.

Display the records sorted in ascending order based on account number.

```

select am.account_number,opening_balance+sum(case when transaction_type='Deposit' then
transaction_amount end) as Deposit,sum(case when transaction_type='withdrawal' then
transaction_amount end) as Withdrawal from account_master am join transaction_details td
on am.account_number=td.account_number group by am.account_number having
Withdrawal>Deposit;

```

ACCOUNT_NUMBER	DEPOSIT	WITHDRAWAL
A00001	10000	12000
A00002	6000	7000

16. Please follow instructions given below.

Write a query to show the balance amount for account number that ends with 001.

Note: Balance amount includes account opening balance also. Give alias name as Balance_Amount.

For example A00015 is having an opening balance of 1000. A00015 has deposited 2000 on 2012-06-12 and deposited 3000 on 2012-07-13. The same account has drawn money of 500 on 2012-08-12, 500 on 2012-09-15, 1000 on 2012-12-17. So balance amount is 4000 i.e (1000 (opening balance)+2000+3000) – (500+500+1000).

```
SELECT (SUM(CASE WHEN transaction_type='Deposit'
THEN transaction_amount END)) -
(SUM(CASE WHEN transaction_type='Withdrawal'
THEN transaction_amount END))+(select opening_balance
from account_master where account_number like '%001') AS Balance_Amount
FROM transaction_details where account_number like '%001'
```

BALANCE_AMOUNT
-2000

17. Please follow instructions given below.

Display the customer number, customer's first name, account number and number of transactions being made by the customers from each account. Give the alias name for number of transactions as Count_Trans. Display the records sorted in ascending order based on customer number and then by account number.

**SELECT cm.customer_number,firstname, am.account_number,count(transaction_number)
Count_Trans**

FROM customer_master cm inner JOIN account_master am

ON cm.customer_number=am.customer_number

INNER JOIN transaction_details td

ON am.account_number=td.account_number

group by am.account_number order by cm.customer_number, am.account_number

CUSTOMER_NUMBER	FIRSTNAME	ACCOUNT_NUMBER	COUNT_TRANS
C00001	RAMESH	A00001	6
C00002	AVINASH	A00002	3
C00007	AMIT	A00007	3

18.Please follow instructions given below.

**Write a query to display the customer's firstname who have multiple accounts (atleast 2 accounts).
Display the records sorted in ascending order based on customer's firstname.**

SELECT firstname

FROM customer_master INNER JOIN account_master

ON customer_master.customer_number=account_master.customer_number

GROUP BY firstname

having count(firstname)>=2 order by firstname;

FIRSTNAME
AMIT
AVINASH

RAHUL
RAMESH

19. Please follow instructions given below.

Write a query to display the customer number, firstname, lastname for those client where total loan amount taken is maximum and at least taken from 2 branches.

For example the customer C00012 took a loan of 100000 from bank branch with id B00009 and C00012

Took a loan of 500000 from bank branch with id B00010. So total loan amount for customer C00012 is

600000. C00013 took a loan of 100000 from bank branch B00009 and 200000 from bank branch B00011.

So total loan taken is 300000. So loan taken by C00012 is more than C00013.

SELECT Id.customer_number, firstname, lastname

FROM customer_master cm INNER JOIN loan_details Id

ON cm.customer_number=Id.customer_number

group by customer_number

having count(branch_id)>=2 and sum(loan_amount)>=All(select sum(loan_amount) from loan_details group by customer_number)

CUSTOMER_NUMBER	FIRSTNAME	LASTNAME
C00002	AVINASH	MINHA

20. Please follow instructions given below.

Write a query to display the customer's number, customer's firstname, branch id and loan amount for people who have taken loans..

Display the records sorted in ascending order based on customer number and then by branch id and then by loan amount.

```
SELECT Id.customer_number, firstname,branch_id, loan_amount
FROM customer_master cm INNER JOIN loan_details Id
ON cm.customer_number=Id.customer_number order by cm.customer_number, branch_id,
loan_amount
```

CUSTOMER_NUMBER	FIRSTNAME	BRANCH_ID	LOAN_AMOUNT
C00001	RAMESH	B00001	100000
C00001	RAMESH	B00003	600000
C00002	AVINASH	B00001	600000
C00002	AVINASH	B00002	200000
C00009	ABHISHEK	B00008	400000
C00010	SHANKAR	B00009	500000

21.Please follow instructions given below.

Write a query to display city name and count of branches in that city. Give the count of branches an alias name of Count_Branch.

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

```
SELECT branch_city, count(branch_id) Count_Branch
FROM branch_master
GROUP BY branch_city
ORDER BY branch_city
```

BRANCH_CITY	COUNT_BRANCH
CHENNAI	1
DELHI	4
KOLKATA	1
MUMBAI	3

22. Please follow instructions given below.

Write a query to display account id, customer's firstname, customer's lastname for the customer's whose account is Active.

Display the records sorted in ascending order based on account id /account number.

```

SELECT account_number, firstname, lastname
FROM customer_master cm INNER JOIN account_master am
ON cm.customer_number=am.customer_number
WHERE account_status='Active'
ORDER BY account_number

```

ACCOUNT_NUMBER	FIRSTNAME	LASTNAME
A00001	RAMESH	SHARMA
A00002	AVINASH	MINHA
A00003	RAHUL	RASTOGI
A00004	AVINASH	MINHA
A00005	CHITRESH	BARWE
A00007	AMIT	BORKAR

A00010	PARUL	GANDHI
--------	-------	--------

23.Please follow instructions given below.

Write a query to display customer's number, first name and middle name. For the customers who don't have middle name, display their last name as middle name. Give the alias name as Middle_Name.

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

```
SELECT customer_number,firstname,coalesce(middlename,lastname) Middle_Name
```

```
FROM customer_master order by customer_number
```

CUSTOMER_NUMBER	FIRSTNAME	MIDDLE_NAME
C00001	RAMESH	CHANDRA
C00002	AVINASH	SUNDER
C00003	RAHUL	RASTOGI
C00004	PARUL	GANDHI
C00005	NAVEEN	CHANDRA
C00006	CHITRESH	BARWE
C00007	AMIT	KUMAR
C00008	NISHA	DAMLE
C00009	ABHISHEK	DUTTA
C00010	SHANKAR	NAIR

24.Please follow instructions given below.

Write a query to display the customer number , firstname, customer's date of birth . Display the records sorted in ascending order of date of birth year and within that sort by firstname in ascending order.

```
SELECT customer_number,firstname,customer_date_of_birth
FROM customer_master order by year(customer_date_of_birth), firstname;
```

CUSTOMER_NUMBER	FIRSTNAME	CUSTOMER_DATE_OF_BIRTH
C00009	ABHISHEK	1973-05-22
C00002	AVINASH	1974-10-16
C00008	NISHA	1975-12-03
C00005	NAVEEN	1976-09-19
C00004	PARUL	1976-11-03
C00001	RAMESH	1976-12-06
C00010	SHANKAR	1976-07-12
C00007	AMIT	1981-09-06
C00003	RAHUL	1981-09-26
C00006	CHITRESH	1992-11-06

25.Please follow instructions given below.

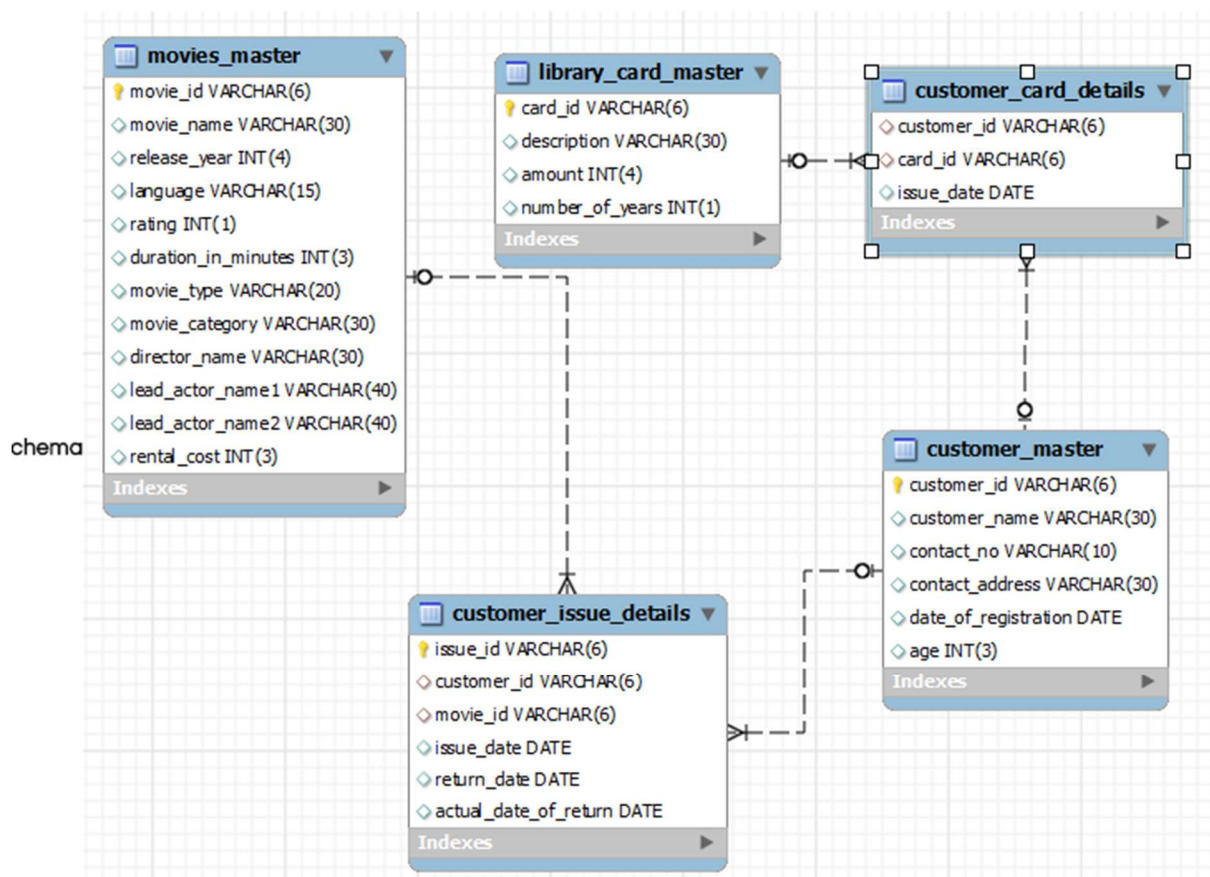
Write a query to display the customers firstname, city and account number whose occupation are not into Business, Service or Student.

Display the records sorted in ascending order based on customer first name and then by account number.

```
SELECT firstname, customer_city,account_number
FROM customer_master cm INNER JOIN account_master am
ON cm.customer_number=am.customer_number
```

**WHERE occupation != 'Service' and occupation != 'Student' and occupation != 'Business' order by
firstname, account_number**

FIRSTNAME	CUSTOMER_CITY	ACCOUNT_NUMBER
PARUL	DELHI	A00010



Video Management database queries:

1. Please follow instructions given below.

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.

11 rows

```
select mm.movie_name, count(cid.issue_id) as ISSUE_COUNT
from movies_master mm left outer join customer_issue_details
cid on mm.movie_id=cid.movie_id group by mm.movie_name
order by ISSUE_COUNT desc,mm.movie_name asc;
```

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
OFFICE SPACE	1
THE MATRIX	1
YOUNG FRANKENSTEIN	1

THE NOTEBOOK	0
--------------	---

2. 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.

4 rows

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

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. 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.

3 rows

Ans:

select movie_category,count(movie_id) as NO_OF_MOVIES from movies_master group by movie_category

order by NO_OF_MOVIES desc,movie_category asc;

MOVIE_CATEGORY	NO_OF_MOVIES
ACTION	4
ROMANCE	4
COMEDY	3

4.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

1 row

select count(ccd.customer_id) as CUSTOMER_COUNT from customer_card_details ccd join library_card_master lcd on ccd.card_id=lcd.card_id where lcd.description='Gold Card';

CUSTOMER_COUNT
2

4.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.

5 rows

select cm.customer_id,cm.customer_name,year(cm.date_of_registration) as REGISTERED_YEAR,ccd.card_id,ccd.issue_date

from customer_master cm join customer_card_details ccd on cm.customer_id=ccd.customer_id

order by cm.customer_name desc;

CUSTOMER_ID	CUSTOMER_NAME	REGISTERED_YEAR	CARD_ID	ISSUE_DATE
C00003	T RAMACHANDRAN	2012	CRD002	2012-11-02
C00005	SHIV PRASAD	2012	CRD003	2012-12-26
C00004	RAJIB MITRA	2012	CRD003	2012-11-21
C00001	NITIN	2012	CRD001	2012-10-15
C00002	AGNESH	2012	CRD002	2012-12-01

5.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.

2 rows

```
select cid.issue_id,cid.customer_id,cm.customer_name from customer_issue_details cid join  
customer_master cm on cid.customer_id=cm.customer_id where cm.customer_name like 'R%'  
and cid.actual_date_return>cid.return_date order by cm.customer_name;
```

ISSUE_ID	CUSTOMER_ID	CUSTOMER_NAME
I00008	C00010	RAGHAV SINGH
I00007	C00004	RAJIB MITRA

6.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;
```

CUSTOMER_ID	CUSTOMER_NAME	CARD_ID	DESCRIPTION	AMOUNT_DOLLAR
C00001	NITIN	CRD001	SILVER CARD	\$19
C00004	RAJIB MITRA	CRD003	PLATINUM CARD	\$57
C00003	T RAMACHANDRAN	CRD002	GOLD CARD	\$38

7.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 ;
```

CUSTOMER_ID	CUSTOMER_NAME	CONTACT_NO	CONTACT_ADDRESS
C00010	RAGHAV SINGH	9675167890	A/6 NEHRU JAWAHAR NAGAR

8.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 gmaster 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.rct>=all(select count(movie_id) from movies_master group by lead_actor_name1))order by movie_name;**

MOVIE_ID	MOVIE_NAME	RELEASE_YEAR	DIRECTOR_NAME
M00004	INCEPTION	2010	CHRISTOPHER NOLAN
M00011	TITANIC	1997	JAMES CAMERON

9.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.

11 rows

select cm.customer_name,count(cid.movie_id) as MOVIE_COUNT from customer_master cm left join customer_issue_details cid on cm.customer_id=cid.customer_id group by cm.customer_name order by cm.customer_name;

CUSTOMER_NAME	MOVIE_COUNT
AGNESH	3
AJAY GHOSH	0
GEETHA REDDY	0
NITIN	2
RAGHAV SINGH	1
RAJ SEKHANRAN	1
RAJAN PILLAI	0
RAJIB MITRA	4
RIA NATRAJAN	0
SHIV PRASAD	0
T RAMACHANDRAN	8

10. 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'

19 rows

```
select substring(cid.issue_id,5,2) as
SERIAL_NO,cid.issue_id,cid.customer_id,cm.customer_name,mm.movie_id,mm.movie_name
from customer_issue_details cid join customer_master cm on cm.customer_id=cid.customer_id
```

join movies_master mm on cid.movie_id=mm.movie_id group by
SERIAL_NO,cid.customer_id,mm.movie_id

order by SERIAL_NO;

SERIAL_NO	ISSUE_ID	CUSTOMER_ID	CUSTOMER_NAME	MOVIE_ID	MOVIE_NAME
01	I00001	C00001	NITIN	M00001	DIE HARD
02	I00002	C00002	AGNESH	M00002	THE DARK KNIGHT
03	I00003	C00002	AGNESH	M00002	THE DARK KNIGHT
04	I00004	C00003	T RAMACHANDRAN	M00003	THE MATRIX
05	I00005	C00003	T RAMACHANDRAN	M00004	INCEPTION
06	I00006	C00003	T RAMACHANDRAN	M00005	OFFICE SPACE
07	I00007	C00004	RAJIB MITRA	M00006	YOUNG FRANKENSTEIN
08	I00008	C00010	RAGHAV SINGH	M00008	CASABLANCA
09	I00009	C00011	RAJ SEKHANRAN	M00010	GONE WITH THE WIND
10	I00010	C00004	RAJIB MITRA	M00007	SHAUN OF THE DEAD
11	I00011	C00004	RAJIB MITRA	M00007	SHAUN OF THE DEAD
12	I00012	C00001	NITIN	M00001	DIE HARD
13	I00013	C00003	T RAMACHANDRAN	M00001	DIE HARD
14	I00014	C00003	T RAMACHANDRAN	M00010	GONE WITH THE WIND
15	I00015	C00003	T RAMACHANDRAN	M00011	TITANIC
16	I00016	C00003	T RAMACHANDRAN	M00011	TITANIC
17	I00017	C00003	T RAMACHANDRAN	M00008	CASABLANCA

18	I00018	C00002	AGNESH	M00010	GONE WITH THE WIND
19	I00019	C00004	RAJIB MITRA	M00001	DIE HARD

11.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 decending order based on issue date of the video.

7 rows

```
select cid.issue_id,cid.issue_date,cid.customer_id,cm.customer_name,cm.contact_no
from customer_issue_details cid join customer_master cm on cid.customer_id=cm.customer_id
where year(issue_date)=2013 group by issue_id,issue_date,customer_id order by
issue_date desc;
```

ISSUE_ID	ISSUE_DATE	CUSTOMER_ID	CUSTOMER_NAME	CONTACT_NO
I00012	2013-11-28	C00001	NITIN	9830354218
I00017	2013-04-15	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

12.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.

1 row

```
select movie_id,movie_name,concat(lead_actor_name1,' & ',lead_actor_name2) as ACTORS
from movies_master where movie_id
not in (select movie_id from customer_issue_details) order by
movie_name;
```

MOVIE_ID	MOVIE_NAME	ACTORS
M00009	THE NOTEBOOK	RYAN GOSLING & RACHEL MCADAMS

13.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.

2 rows

```
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;
```

DIRECTOR_NAME	MOVIE_NAME	LEAD_ACTOR_NAME1
CHRISTOPHER NOLAN	THE DARK KNIGHT	CHRISTIAN BALE
CHRISTOPHER	INCEPTION	LEONARDO

NOLAN		DICAPRIO
-------	--	----------

14.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.

1 row

```
select count(customer_id) as NO_OF_CUSTOMERS from customer_master where
year(date_of_registration)
=2012 and contact_no != 'NULL'
```

NO_OF_CUSTOMERS
6

15.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.

11 rows

```
select cm.customer_name,ifnull(cm.contact_no,cm.contact_add) as
CONTACT_DETAILS,lcd.card_id,lcd.description from customer_master cm
left join customer_card_details ccd on cm.customer_id=ccd.customer_id
left join library_card_master lcd on ccd.card_id=lcd.card_id group by
customer_name,description,CONTACT_DETAILS
order by customer_name;
```

CUSTOMER_NAME	CONTACT_DETAILS	CARD_ID	DESCRIPTION
AGNESH	8923156781	CRD002	GOLD CARD

AJAY GHOSH	8763478901	NULL	NULL
GEETHA REDDY	8976167890	NULL	NULL
NITIN	9830354218	CRD001	SILVER CARD
RAGHAV SINGH	9675167890	NULL	NULL
RAJ SEKHANRAN	8423178906	NULL	NULL
RAJAN PILLAI	A 1/66 KODAMBAKKAM	NULL	NULL
RAJIB MITRA	9830356781	CRD003	PLATINUM CARD
RIA NATRAJAN	9856723190	NULL	NULL
SHIV PRASAD	2/2 PHASE II, JAWAHAR NAGAR	CRD003	PLATINUM CARD
T RAMACHANDRAN	9831289761	CRD002	GOLD CARD

16. Please follow instructions given below.

Write a query to display the customer id, customer name and number of times the same movie is issued to the same customers who have taken same movie more than once. Display the records sorted by customer name in descending order. For Example: Assume customer John has taken Titanic three times and customer Ram has taken Die hard only once then display the details of John. Hint: Use NO_OF_TIMES as alias name for number of times

4 rows

```
select cm.customer_id,cm.customer_name,count(cid.movie_id) as NO_OF_TIMES from
customer_master
```

```
cm join customer_issue_details cid on cm.customer_id=cid.customer_id group by
customer_id,movie_id having
```

```
count(movie_id)>1 order by customer_name desc;
```

CUSTOMER_ID	CUSTOMER_NAME	NO_OF_TIMES
C00003	T RAMACHANDRAN	2
C00004	RAJIB MITRA	2
C00001	NITIN	2
C00002	AGNESH	2

17.Please follow instructions given below.

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.

5 rows

```
select cid.customer_id,cm.customer_name,
concat('+91-',substring(cm.contact_no,1,3),'-',substring(cm.contact_no,4,3),'-',
substring(cm.contact_no,7,4)) as CONTACT_ISD,
mm.movie_category,count(mm.movie_category) as NO_OF_MOVIES from customer_master
cm join customer_issue_details cid
on cm.customer_id=cid.customer_id join movies_master mm on cid.movie_id=mm.movie_id
group by mm.movie_category,cm.customer_name having count(movie_category)>1
order by cm.customer_name,mm.movie_category;
```

CUSTOMER_ID	CUSTOMER_NAME	CONTACT_ISD	MOVIE_CATEGORY	NO_OF_MOVIES
-------------	---------------	-------------	----------------	--------------

C00002	AGNESH	+91-892-315-6781	ACTION	2
C00001	NITIN	+91-983-035-4218	ACTION	2
C00004	RAJIB MITRA	+91-983-035-6781	COMEDY	3
C00003	T RAMACHANDRAN	+91-983-128-9761	ACTION	3
C00003	T RAMACHANDRAN	+91-983-128-9761	ROMANCE	4

18. 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.

3 rows

```
(select cm.customer_id,cm.customer_name from customer_master cm
join customer_issue_details cid
on cm.customer_id=cid.customer_id group by cm.customer_id
having count(cid.issue_id) >= all (select count(cid.issue_id) from customer_master cm
join customer_issue_details cid
on cm.customer_id=cid.customer_id group by cm.customer_id) order by cm.customer_name)
union all
(select cm.customer_id,cm.customer_name from customer_master cm
```


join customer_issue_details cid

on cm.customer_id=cid.customer_id group by cm.customer_id

having count(cid.issue_id) <= all (select count(cid.issue_id) from customer_master cm

join customer_issue_details cid

on cm.customer_id=cid.customer_id group by cm.customer_id) order by cm.customer_name)

CUSTOMER_ID	CUSTOMER_NAME
C00003	T RAMACHANDRAN
C00010	RAGHAV SINGH
C00011	RAJ SEKHANRAN

19.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.

1 row

select cm.customer_id,cm.customer_name,count(mm.movie_id) as NO_OF_TIMES from
customer_master cm

join customer_issue_details cid on cm.customer_id=cid.customer_id join

movies_master mm on cid.movie_id=mm.movie_id where mm.movie_category='comedy' group by
customer_id

order by customer_name>1;

CUSTOMER_ID	CUSTOMER_NAME	NO_OF_TIMES
C00004	RAJIB MITRA	3

20. Please follow instructions given below.

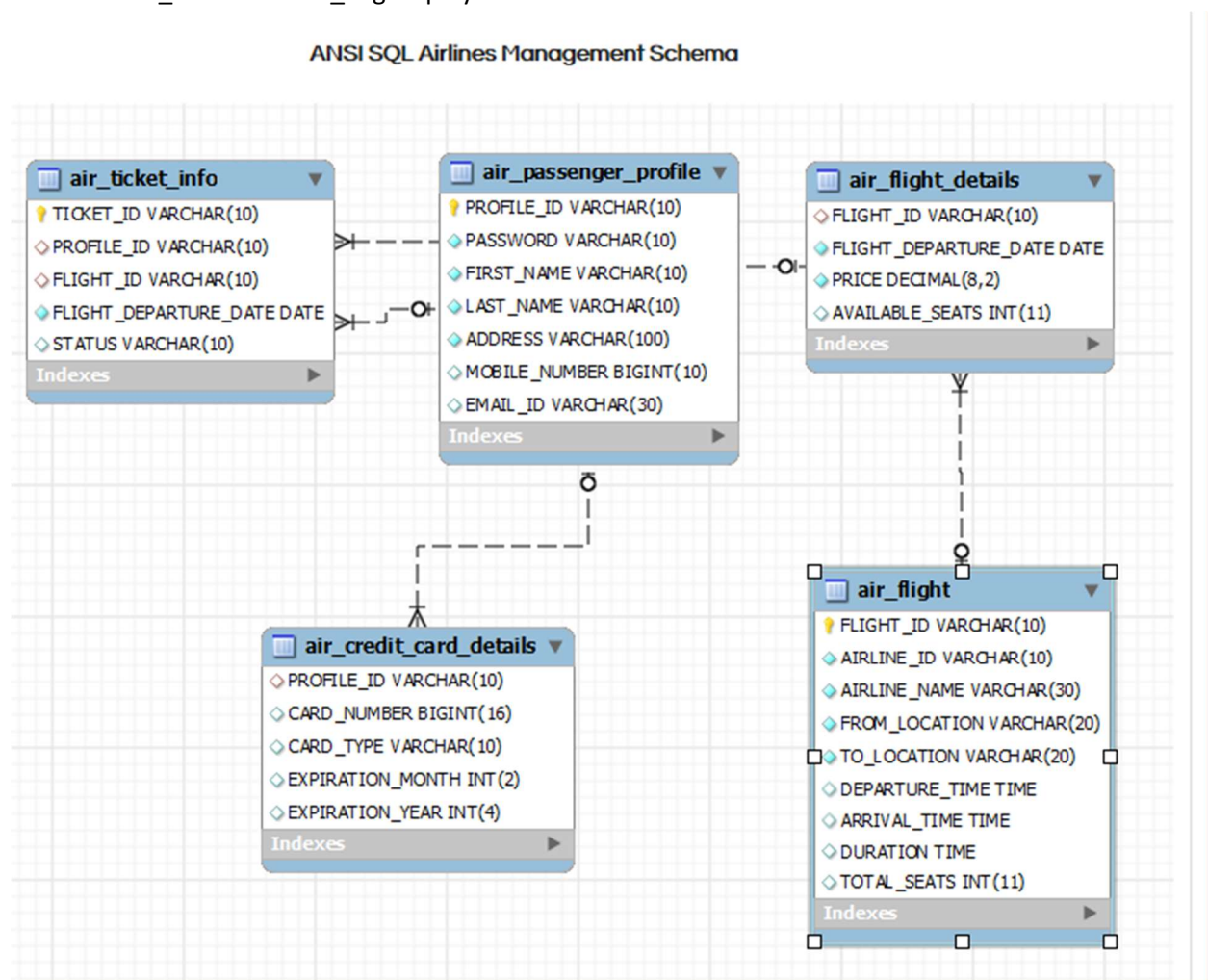
Write a query to display customer id and total rent paid by the customers who are issued with the videos. Need not display the customers who has not taken / issued with any videos. Hint: Alias Name for total rent paid is TOTAL_COST. Display the records sorted in ascending order based on customer id

6 rows

```
select cid.customer_id, sum(mm.rental_cost) as TOTAL_COST from customer_issue_details cid
```

```
join movies_master mm
```

```
on cid.movie_id=mm.movie_id group by cu
```



```
ustomer_id order by customer_id;
```

	customer_id	TOTAL_COST
▶	C00001	200
	C00002	300
	C00003	1700
	C00004	400
	C00010	1000
	C00011	100

Airline Flight Management :

1. Write a query to display the average monthly ticket cost for each flight in ABC Airlines. The query should display the Flight_Id, From_location, To_location, Month Name as "Month_Name" and average price as "Average_Price"

Display the records sorted in ascending order based on flight id and then by Month Name.

15 rows

```
select f.flight_id, f.from_location, f.to_location, monthname(fd.flight_departure_date) as
Month_name, avg(fd.price) as Average_price from air_flight f join air_flight_details fd
on f.flight_id=fd.flight_id group by f.flight_id, Month_name order by f.flight_id, Month_name;
```

FLIGHT_ID	FROM_LOCATION	TO_LOCATION	MONTH_NAME	AVERAGE_PRICE
1011	HYDERABAD	CHENNAI	APRIL	4614.000000
1011	HYDERABAD	CHENNAI	MAY	3855.500000
1262	HYDERABAD	CHENNAI	MAY	3444.500000
1265	CHENNAI	HYDERABAD	APRIL	4086.000000
1265	CHENNAI	HYDERABAD	MAY	3303.666667

289	CHENNAI	KOCHI	MAY	3257.750000
3004	BENGALURU	CHENNAI	MAY	3319.666667
3013	CHENNAI	BENGALURU	MAY	3257.750000
3148	CHENNAI	BENGALURU	JUNE	2773.000000
3148	CHENNAI	BENGALURU	MAY	3052.000000
3241	CHENNAI	KOCHI	MAY	3303.666667
3244	KOCHI	CHENNAI	MAY	3371.500000
3307	BENGALURU	CHENNAI	MAY	3309.000000
916	CHENNAI	HYDERABAD	APRIL	4086.000000
916	CHENNAI	HYDERABAD	MAY	3570.666667

2. Write a query to display the customer(s) who has/have booked least number of tickets in ABC Airlines. The Query should display profile_id, customer's first_name, Address and Number of tickets booked as "No_of_Tickets"

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

1 row

```

select apf.profile_id, apf.first_name, apf.address, count(ati.ticket_id) as No_of_Tickets
from air_passenger_profile apf
join air_ticket_info ati on apf.profile_id=ati.profile_id group by apf.profile_id having
count(ati.ticket_id) <= all

```

(select count(ati.ticket_id) from air_passenger_profile apf

join air_ticket_info ati on apf.profile_id=ati.profile_id group by apf.profile_id) order by

first_name;

PROFILE_ID	FIRST_NAME	ADDRESS	NO_OF_TICKETS
PFL008	GANESH	45 3RD ST, HYDERABAD-24	1

3. Write a query to display the number of flight services between locations in a month. The Query should display From_Location, To_Location, Month as "Month_Name" and number of flight services as "No_of_Services".

Hint: The Number of Services can be calculated from the number of scheduled departure dates of a flight.

The records should be displayed in ascending order based on From_Location and then by To_Location and then by month name

9 rows

```
select af.from_location, af.to_location, monthname(afd.flight_departure_date)
as Month_Name,
count(afd.flight_departure_date) as No_of_Services from air_flight af join
air_flight_details afd
on af.flight_id=afd.flight_id group by
af.from_location, af.to_location, month_name order by
from_location, to_location, month_name;
```

FROM_LOCATION	TO_LOCATION	MONTH_NAME	NO_OF_SERVICES
BENGALURU	CHENNAI	MAY	7
CHENNAI	BENGALURU	JUNE	1
CHENNAI	BENGALURU	MAY	6

CHENNAI	HYDERABAD	APRIL	2
CHENNAI	HYDERABAD	MAY	6
CHENNAI	KOCHI	MAY	7
HYDERABAD	CHENNAI	APRIL	1
HYDERABAD	CHENNAI	MAY	4
KOCHI	CHENNAI	MAY	2

4. Write a query to display the customer(s) who has/have booked maximum number of tickets in ABC Airlines. The Query should display profile_id, customer's first_name, Address and Number of tickets booked as "No_of_Tickets"

Display the records in ascending order based on customer's first name.

1 row

```
select app.profile_id,app.first_name,app.address,count(ati.ticket_id) as No_of_Tickets
from air_passenger_profile app
join air_ticket_info ati on app.profile_id=ati.profile_id join air_flight af on ati.flight_id=af.flight_id
where af.airline_name= 'ABC Airlines' group by app.profile_id
having count(ati.ticket_id) >= all (select count(ati.ticket_id) from air_passenger_profile app
join air_ticket_info ati on app.profile_id=ati.profile_id join air_flight af on ati.flight_id=af.flight_id
where af.airline_name= 'ABC Airlines' group by app.profile_id) order by app.first_name;
```

PROFILE_ID	FIRST_NAME	ADDRESS	NO_OF_TICKETS
PFL009	RAM	119 2ND CROSS	8

		ST,ERNAKULAM-12	
--	--	-----------------	--

5. Write a query to display the number of tickets booked from Chennai to Hyderabad. The Query should display passenger profile_id, first_name, last_name, Flight_Id, Departure_Date and number of tickets booked as "No_of_Tickets".

Display the records sorted in ascending order based on profile id and then by flight id and then by departure date.

3 rows

```
select
ati.profile_id, app.first_name, app.last_name, ati.flight_id, ati.flight_departure_date, count(ati.ticket_id)
as No_of_Tickets from air_ticket_info ati join air_passenger_profile app on ati.profile_id=
app.profile_id join air_flight af on ati.flight_id=af.flight_id
where af.from_location='chennai' and af.to_location='hyderabad' group by ati.profile_id,
ati.flight_id, ati.flight_departure_date order by
ati.profile_id,
ati.flight_id, ati.flight_departure_date;
```

PROFILE_ID	FIRST_NAME	LAST_NAME	FLIGHT_ID	FLIGHT_DEPARTURE_DATE	NO_OF_TICKETS
PFL001	LATHA	SANKAR	1265	2013-04-29	1
PFL004	AARTHI	RAMESH	1265	2013-05-29	1
PFL005	SIVA	KUMAR	916	2013-05-06	2

6. Write a query to display flight id, from location, to location and ticket price of flights whose departure is in the month of april.

3 rows

Display the records sorted in ascending order based on flight id and then by from location.

```
select af.flight_id,af.from_location,af.to_location,afd.price from
air_flight af
join air_flight_details afd on af.flight_id=afd.flight_id
where monthname(afd.flight_departure_date)='april' order by
flight_id,from_location;
```

FLIGHT_ID	FROM_LOCATION	TO_LOCATION	PRICE
1011	HYDERABAD	CHENNAI	4614.00
1265	CHENNAI	HYDERABAD	4086.00
916	CHENNAI	HYDERABAD	4086.00

7. Write a query to display the average cost of the tickets in each flight on all scheduled dates. The query should display flight_id, from_location, to_location and Average price as "Price".

Display the records sorted in ascending order based on flight id and then by from_location and then by to_location.

11 rows

```
select af.flight_id,af.from_location,af.to_location,avg(afd.price)
from air_flight af join air_flight_details afd
on af.flight_id=afd.flight_id group by af.flight_id,af.from_location,af.to_location
order by af.flight_id,af.from_location,af.to_location;
```

FLIGHT_ID	FROM_LOCATION	TO_LOCATION	PRICE
1011	HYDERABAD	CHENNAI	4108.333333
1262	HYDERABAD	CHENNAI	3444.500000
1265	CHENNAI	HYDERABAD	3499.250000

289	CHENNAI	KOCHI	3257.750000
3004	BENGALURU	CHENNAI	3319.666667
3013	CHENNAI	BENGALURU	3257.750000
3148	CHENNAI	BENGALURU	2959.000000
3241	CHENNAI	KOCHI	3303.666667
3244	KOCHI	CHENNAI	3371.500000
3307	BENGALURU	CHENNAI	3309.000000
916	CHENNAI	HYDERABAD	3699.500000

8. Write a query to display the customers who have booked tickets from Chennai to Hyderabad. The query should display profile_id, customer_name (combine first_name & last_name with comma in b/w), address of the customer.

Give an alias to the name as customer_name.

Hint: Query should fetch unique customers irrespective of multiple tickets booked.

Display the records sorted in ascending order based on profile id.

3 rows

```
select app.profile_id, concat(app.first_name,',',app.last_name) as customer_name,app.address
from air_passenger_profile app join air_ticket_info ati on app.profile_id=ati.profile_id
join air_flight af on ati.flight_id=af.flight_id where af.from_location='chennai'
and af.to_location='hyderabad' group by app.profile_id order by app.profile_id;
```

PROFILE_ID	CUSTOMER_NAME	ADDRESS
PFL001	LATHA,SANKAR	123 BROAD CROSS ST,CHENNAI-48

PFL004	AARTHI,RAMESH	343 6TH STREET,HYDERABAD-76
PFL005	SIVA,KUMAR	125 8TH STREET,CHENNAI-46

9. Write a query to display profile id of the passenger(s) who has/have booked maximum number of tickets.

In case of multiple records, display the records sorted in ascending order based on profile id.

2 rows

```
select profile_id from air_ticket_info group by profile_id having
count(ticket_id) >= all (select count(ticket_id)
from air_ticket_info group by profile_id) order by profile_id;
```

PROFILE_ID
PFL002
PFL007

10. Write a query to display the total number of tickets as “No_of_Tickets” booked in each flight in ABC Airlines. The Query should display the flight_id, from_location, to_location and the number of tickets.

Display only the flights in which atleast 1 ticket is booked.

Display the records sorted in ascending order based on flight id.

7 rows

```
select af.flight_id,af.from_location,af.to_location,count(ati.ticket_id) as
No_of_Tickets
from air_flight af join air_ticket_info ati on af.flight_id=ati.flight_id
group by af.flight_id having count(ati.ticket_id) >= 1;
```

IGHT_ID	FROM_LOCATION	TO_LOCATION	NO_OF_TICKETS
1011	HYDERABAD	CHENNAI	4
1262	HYDERABAD	CHENNAI	1
1265	CHENNAI	HYDERABAD	2
3004	BENGALURU	CHENNAI	3
3148	CHENNAI	BENGALURU	7
3244	KOCHI	CHENNAI	7
916	CHENNAI	HYDERABAD	2

11. Write a query to display the no of services offered by each flight and the total price of the services. The Query should display flight_id, number of services as “No_of_Services” and the cost as “Total_Price” in the same order.

Order the result by Total Price in descending order and then by flight_id in descending order.

Hint: The number of services can be calculated from the number of scheduled departure dates of the flight

11 rows

```
select af.flight_id, count(afd.flight_departure_date) as No_of_Services, sum(afd.price) as
Total_Price from air_flight af join air_flight_details afd on af.flight_id=afd.flight_id
group by flight_id
order by total_price desc, flight_id desc;
```

FLIGHT_ID	NO_OF_SERVICES	TOTAL_PRICE
916	4	14798.00

1265	4	13997.00
3307	4	13236.00
3013	4	13031.00
289	4	13031.00
1011	3	12325.00
3004	3	9959.00
3241	3	9911.00
3148	3	8877.00
1262	2	6889.00
3244	2	6743.00

12. Write a query to display the number of passengers who have travelled in each flight in each scheduled date. The Query should display flight_id, flight_departure_date and the number of passengers as “No_of_Passengers” in the same order.

Display the records sorted in ascending order based on flight id and then by flight departure date.

9 rows

```

SELECT flight_id,
       flight_departure_date,
       COUNT(ticket_id) AS No_of_Passengers
FROM   air_ticket_info
GROUP BY flight_id,
         flight_departure_date
ORDER BY flight_id, flight_departure_date;

```

FLIGHT_ID	FLIGHT_DEPARTURE_DATE	NO_OF_PASSENGERS
1011	2013-05-09	4
1262	2013-05-20	1
1265	2013-04-29	1
1265	2013-05-29	1
3004	2013-05-02	3
3148	2013-05-21	2
3148	2013-06-01	5
3244	2013-05-03	7
916	2013-05-06	2

13. Write a query to display profile id of passenger(s) who booked minimum number of tickets.

In case of multiple records, display the records sorted in ascending order based on profile id.

1 row

```
select profile_id from air_ticket_info group by profile_id having count(profile_id) <= all
```

```
(select count(profile_id) from air_ticket_info group by profile_id) order by profile_id;
```

PROFILE_ID
PFL008

14. Write a query to display unique passenger profile id, first name, mobile number and email address of passengers who booked ticket to travel from HYDERABAD to CHENNAI.

Display the records sorted in ascending order based on profile id.

4 rows

```
select distinct ati.profile_id,app.first_name,app.mobile_number,app.email_id
from air_ticket_info
ati join air_passenger_profile app on ati.profile_id=app.profile_id join air_flight af
on ati.flight_id=af.flight_id
where af.from_location='hyderabad' and af.to_location='chennai' order by profile_id;
```

PROFILE_ID	FIRST_NAME	MOBILE_NUMBER	EMAIL_ID
PFL001	LATHA	9876543210	LATHA@GMAIL.COM
PFL004	AARTHI	9595652530	AARTHI@GMAIL.COM
PFL005	SIVA	9884416986	SIVA@GMAIL.COM
PFL008	GANESH	9375237890	GANESH@GMAIL.COM

15. Write a query to intimate the passengers who are boarding Chennai to Hyderabad Flight on 6th May 2013 stating the delay of 1hr in the departure time. The Query should display the passenger's profile_id, first_name, last_name, flight_id, flight_departure_date, actual departure time, actual arrival time, delayed departure time as "Delayed_Departure_Time", delayed arrival time as "Delayed_Arrival_Time". Hint: Distinct Profile ID should be displayed irrespective of multiple tickets booked by the same profile.

Display the records sorted in ascending order based on passenger's profile id.

1 row

```
select distinct app.profile_id,app.first_name,app.last_name,ati.flight_id,ati.flight_departure_date,
af.departure_time,af.arrival_time, af.departure_time ,ADDTIME(af.departure_time,'1:00:00') as
Delayed_Departure_Time,
ADDTIME(af.arrival_time,'1:00:00') as Delayed_Arrival_Time from air_passenger_profile app
```

join air_ticket_info ati on app.profile_id=ati.profile_id join air_flight af on

ati.flight_id=af.flight_id where ati.flight_departure_date='2013-05-06' order by app.profile_id;

PROFILE_ ID	FIRST _NAME	LAST_NAME	FLIGHT _ID	FLIGHT_ DEPARTURE _DATE	DEPARTURE_TIME	ARRIVAL _TIME
PFL005	SIVA	KUMAR	916	2013-05-06	19:55:00	21:00:00

DELAYED_DEPARTURE_TIME	DELAYED_ARRIVAL_TIME
20:55:00	22:00:00

16. Write a query to display the number of tickets as “No_of_Tickets” booked by Kochi Customers. The Query should display the Profile_Id, First_Name, Base_Location and number of tickets booked.

Hint: Use String functions to get the base location of customer from their Address and give alias name as “Base_Location”

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

2 rows

```
select
ap.profile_id, ap.first_name, substring_index(substring_index(ap.address, ',', -1), '-', 1)
as base_location, count(at.ticket_id) as No_of_Tickets from
air_passenger_profile ap join air_ticket_info at
on at.profile_id=ap.profile_id
where substring_index(substring_index(ap.address, ',', -1), '-', 1) = 'kochi'
group by ap.profile_id order by first_name
```

PROFILE_ID	FIRST_NAME	BASE_LOCATION	NO_OF_TICKETS
------------	------------	---------------	---------------

PFL003	AMIT	KOCHI	3
PFL006	RAMESH	KOCHI	4

17. Write a query to display the flight_id, from_location, to_location, number of Services as “No_of_Services” offered in the month of May.

Hint: The number of services can be calculated from the number of scheduled departure dates of the flight

Display the records sorted in ascending order based on flight id.

11 rows

```
select af.flight_id,af.from_location,af.to_location,count(afd.flight_departure_date)
as No_of_Services from air_flight af join air_flight_details afd
on af.flight_id=afd.flight_id where month(afd.flight_departure_date)='05'
group by flight_id order by flight_id;
```

FLIGHT_ID	FROM_LOCATION	TO_LOCATION	NO_OF_SERVICES
1011	HYDERABAD	CHENNAI	2
1262	HYDERABAD	CHENNAI	2
1265	CHENNAI	HYDERABAD	3
289	CHENNAI	KOCHI	4
3004	BENGALURU	CHENNAI	3
3013	CHENNAI	BENGALURU	4
3148	CHENNAI	BENGALURU	2
3241	CHENNAI	KOCHI	3

3244	KOCHI	CHENNAI	2
3307	BENGALURU	CHENNAI	4
916	CHENNAI	HYDERABAD	3

18. Write a query to display profile id, last name, mobile number and email id of passengers whose base location is chennai.

Display the records sorted in ascending order based on profile id.

2 rows

```
select profile_id,last_name,mobile_number,email_id from air_passenger_profile where
substring_index(substring_index(address,',',-1),'-',1)='chennai'
order by profile_id;
```

PROFILE_ID	LAST_NAME	MOBILE_NUMBER	EMAIL_ID
PFL001	SANKAR	9876543210	LATHA@GMAIL.COM
PFL005	KUMAR	9884416986	SIVA@GMAIL.COM

18. Write a query to display number of flights between 6.00 AM and 6.00 PM from chennai. Hint Use FLIGHT_COUNT as alias name.

1 row

```
select count(flight_id) as FLIGHT_COUNT from air_flight where departure_time between
'6:00:00' and '18:00:00' and from_location='chennai';;
```

FLIGHT_COUNT

3

19. Write a query to display unique profile id, first name, email id and contact number of passenger(s) who travelled on flight with id 3148. Display the records sorted in ascending order based on first name.

2 rows

```
select distinct app.profile_id, app.first_name, app.email_id, app.mobile_number from  
air_passenger_profile app
```

```
join air_ticket_info ati on app.profile_id=ati.profile_id
```

```
where ati.flight_id= 3148 group by app.first_name order by app.first_name;
```

PROFILE_ID	FIRST_NAME	EMAIL_ID	MOBILE_NUMBER
PFL002	ARUN	ARUN@AOL.COM	8094564243
PFL007	GAYATHRI	GAYATHRI@GMAIL.COM	8073245678

20. Write a query to display the flights available in Morning, Afternoon, Evening & Night. The Query should display the Flight_Id, From_Location, To_Location, Departure_Time, time of service as "Time_of_Service".

Time of Service should be calculated as: From 05:00:01 Hrs to 12:00:00 Hrs - Morning, 12:00:01 to 18:00:00 Hrs - Afternoon, 18:00:01 to 24:00:00 - Evening and 00:00:01 to 05:00:00 - Night

Display the records sorted in ascending order based on flight id.

11 rows

```
select flight_id, from_location, to_location, departure_time,  
case when departure_time between '05:00:01' and '12:00:00' then 'Morning'  
when departure_time between '12:00:01' and '18:00:00' then 'Afternoon'  
when departure_time between '18:00:01' and '24:00:00' then 'Evening'
```

when departure_time between '00:00:01' and '05:00:00' then 'Night'

end as Time_of_Service

from air_flight order by flight_id;

FLIGHT_ID	FROM_LOCATION	TO_LOCATION	DEPARTURE_TIME	TIME_OF_SERVICE
1011	HYDERABAD	CHENNAI	12:30:00	AFTERNOON
1262	HYDERABAD	CHENNAI	06:00:00	MORNING
1265	CHENNAI	HYDERABAD	21:25:00	EVENING
289	CHENNAI	KOCHI	08:40:00	MORNING
3004	BENGALURU	CHENNAI	09:05:00	MORNING
3013	CHENNAI	BENGALURU	07:40:00	MORNING
3148	CHENNAI	BENGALURU	20:15:00	EVENING
3241	CHENNAI	KOCHI	10:40:00	MORNING
3244	KOCHI	CHENNAI	21:10:00	EVENING
3307	BENGALURU	CHENNAI	18:45:00	EVENING
916	CHENNAI	HYDERABAD	19:55:00	EVENING

21.Please follow instructions given below.

Write a query to display flight id,departure date,flight type of all flights. Flight type can be identified based on the following rules : if ticket price is less than 3000 then 'AIR PASSENGER',ticket price between 3000 and less than 4000 'AIR BUS' and ticket price between 4000 and greater than 4000 then 'EXECUTIVE PASSENGER'. Hint use FLIGHT_TYPE as alias name.

Display the records sorted in ascendeing order based on flight_id and then by departure date.

36 rows

select flight_id,flight_departure_date,

case when price<3000 then 'AIR PASSENGER'

when price>=3000 and price<=4000 then 'AIR BUS'

when price>4000 then 'EXECUTIVE PASSENGER'

end as FLIGHT_TYPE from air_flight_details order by flight_id,flight_departure_date;

FLIGHT_ID	FLIGHT_DEPARTURE_DATE	FLIGHT_TYPE
1011	2013-04-30	EXECUTIVE PASSENGER
1011	2013-05-09	EXECUTIVE PASSENGER
1011	2013-05-21	AIR BUS
1262	2013-05-20	AIR BUS
1262	2013-05-29	AIR BUS
1265	2013-04-29	EXECUTIVE PASSENGER
1265	2013-05-14	AIR BUS
1265	2013-05-18	EXECUTIVE PASSENGER
1265	2013-05-29	AIR PASSENGER
289	2013-05-06	AIR BUS
289	2013-05-08	AIR BUS
289	2013-05-20	AIR BUS
289	2013-05-31	AIR PASSENGER
3004	2013-05-02	AIR BUS

3004	2013-05-19	AIR BUS
3004	2013-05-24	AIR BUS
3013	2013-05-04	AIR BUS
3013	2013-05-06	AIR BUS
3013	2013-05-22	AIR BUS
3013	2013-05-30	AIR PASSENGER
3148	2013-05-16	AIR BUS
3148	2013-05-21	AIR BUS
3148	2013-06-01	AIR PASSENGER
3241	2013-05-01	EXECUTIVE PASSENGER
3241	2013-05-13	AIR BUS
3241	2013-05-27	AIR PASSENGER
3244	2013-05-03	AIR BUS
3244	2013-05-15	AIR BUS
3307	2013-05-03	AIR BUS
3307	2013-05-03	AIR BUS
3307	2013-05-23	AIR BUS
3307	2013-05-29	AIR BUS
916	2013-04-28	EXECUTIVE PASSENGER

916	2013-05-01	EXECUTIVE PASSENGER
916	2013-05-06	AIR BUS
916	2013-05-12	AIR BUS

22.Please follow instructions given below.

Write a query to display the credit card type and no of credit cards used on the same type. Display the records sorted in ascending order based on credit card type.

Hint: Use CARD_COUNT AS Alias name for no of cards.

3 rows

```
SELECT CARD_TYPE,count(card_type) CARD_COUNT FROM air_credit_card_details group by CARD_TYPE
order by CARD_TYPE;
```

CARD_TYPE	CARD_COUNT
GOLD	3
INSTANT	2
PLATINIUM	3

23.Please follow instructions given below.

Write a Query to display serial no, first name,mobile number,email id of all the passengers who holds email address from gmail.com.

The Serial No will be the last three digits of profile ID.

Hint: Use SERIAL_NO as Alias name for serial number.

Display the records sorted in ascending order based on name.

6 rows

```
select substring(profile_id,4) as SERIAL_NO,first_name,mobile_number,email_id
from air_passenger_profile where email_id like '%gmail.com' order by first_name;
```

SERIAL_NO	FIRST_NAME	MOBILE_NUMBER	EMAIL_ID
004	AARTHI	9595652530	AARTHI@GMAIL.COM
008	GANESH	9375237890	GANESH@GMAIL.COM
007	GAYATHRI	8073245678	GAYATHRI@GMAIL.COM
001	LATHA	9876543210	LATHA@GMAIL.COM
006	RAMESH	9432198760	RAMESH@GMAIL.COM
005	SIVA	9884416986	SIVA@GMAIL.COM

24.Please follow instructions given below.

Write a query to display the flight(s) which has least number of services in the month of May. The Query should fetch flight_id, from_location, to_location, least number of Services as “No_of_Services” Hint: Number of services offered can be calculated from the number of scheduled departure dates of a flight

If there are multiple flights, display them sorted in ascending order based on flight id.

4 rows

```
select af.flight_id,af.from_location,af.to_location,count(afd.flight_departure_date) as
No_of_Services from air_flight af join air_flight_details afd on
af.flight_id=afd.flight_id where month(afd.flight_departure_date)='05' group by af.flight_id
having count(afd.flight_departure_date)
<= all (select count(afd.flight_departure_date) from air_flight af join air_flight_details afd on
af.flight_id=afd.flight_id where month(afd.flight_departure_date)='05' group by af.flight_id)
```

order by af.flight_id;

LIGHT_ID	FROM_LOCATION	TO_LOCATION	NO_OF_SERVICES
1011	HYDERABAD	CHENNAI	2
1262	HYDERABAD	CHENNAI	2
3148	CHENNAI	BENGALURU	2
3244	KOCHI	CHENNAI	2

25.Please follow instructions given below.

Write a query to display the number of flights flying from each location. The Query should display the from location and the number of flights to other locations as “No_of_Flights”.

Hint: Get the distinct from location and to location.

Display the records sorted in ascending order based on from location.

4 rows

```
select distinct from_location,count(to_location) as No_of_Flights from  
air_flight
```

```
group by from_location order by from_location;
```

FROM_LOCATION	NO_OF_FLIGHTS
BENGALURU	2
CHENNAI	6
HYDERABAD	2
KOCHI	1

26.Please follow instructions given below.

Write a query to display the number of passengers traveled in each flight in each scheduled date. The Query should display flight_id,from_location,To_location, flight_departure_date and the number of passengers as “No_of_Passengers”.

Hint: The Number of passengers inclusive of all the tickets booked with single profile id.

Display the records sorted in ascending order based on flight id and then by flight departure date.

9 rows

```
select af.flight_id,af.from_location,af.to_location,ati.flight_departure_date,count(ati.ticket_id)
as No_of_Passengers from air_flight af join air_ticket_info ati on af.flight_id=ati.flight_id
group by af.flight_id,ati.flight_departure_date order by af.flight_id,ati.flight_departure_date;
```

flight_id	from_location	to_location	flight_departure_date	No_of_Passengers
1011	HYDERABAD	CHENNAI	2013-05-09	3
1262	HYDERABAD	CHENNAI	2013-05-20	1
1265	CHENNAI	HYDERABAD	2013-04-29	1
1265	CHENNAI	HYDERABAD	2013-05-29	1
3004	BENGALURU	CHENNAI	2013-05-02	3
3148	CHENNAI	BENGALURU	2013-05-21	1
3148	CHENNAI	BENGALURU	2013-06-01	3
3244	KOCHI	CHENNAI	2013-05-03	7
916	CHENNAI	HYDERABAD	2013-05-06	2

27.Please follow instructions given below.

Write a query to display the flight details in which more than 10% of seats have been booked. The query should display Flight_Id, From_Location, To_Location,Total_Seats, seats booked as “No_of_Seats_Booked” .

Display the records sorted in ascending order based on flight id and then by No_of_Seats_Booked.

1 row

```
select af.flight_id,af.from_location,af.to_location,af.total_seats,(af.total_seats-afd.available_seats)
as No_of_Seats_Booked from air_flight af join air_flight_details afd on af.flight_id=
```

afd.flight_id where (af.total_seats-afd.available_seats)>(af.total_seats*0.1) group by flight_id order by
flight_id,No_of_Seats_Booked;

FLIGHT_ID	FROM_LOCATION	TO_LOCATION	TOTAL_SEATS	NO_OF_SEATS_BOOKED
3244	KOCHI	CHENNAI	50	7

28.Please follow instructions given below.

Write a query to display the Flight_Id, Flight_Departure_Date, From_Location,To_Location and Duration of all flights which has duration of travel less than 1 Hour, 10 Minutes.

Display the records sorted in ascending order based on flight id and then by flight departure date.

14 rows

```
select af.flight_id,afd.flight_departure_date,af.from_location,af.to_location,af.duration
from air_flight af join air_flight_details afd on af.flight_id=afd.flight_id
where duration<'1:10:00' group by af.flight_id,afd.flight_departure_date
order by af.flight_id,afd.flight_departure_date;
```

FLIGHT_ID	FLIGHT_DEPARTURE_DATE	FROM_LOCATION	TO_LOCATION	DURATION
3013	2013-05-04	CHENNAI	BENGALURU	01:05:00
3013	2013-05-06	CHENNAI	BENGALURU	01:05:00
3013	2013-05-22	CHENNAI	BENGALURU	01:05:00
3013	2013-05-30	CHENNAI	BENGALURU	01:05:00
3148	2013-05-16	CHENNAI	BENGALURU	01:05:00

3148	2013-05-21	CHENNAI	BENGALURU	01:05:00
3148	2013-06-01	CHENNAI	BENGALURU	01:05:00
3307	2013-05-03	BENGALURU	CHENNAI	01:00:00
3307	2013-05-23	BENGALURU	CHENNAI	01:00:00
3307	2013-05-29	BENGALURU	CHENNAI	01:00:00
916	2013-04-28	CHENNAI	HYDERABAD	01:05:00
916	2013-05-01	CHENNAI	HYDERABAD	01:05:00
916	2013-05-06	CHENNAI	HYDERABAD	01:05:00
916	2013-05-12	CHENNAI	HYDERABAD	01:05:00

29. Please follow instructions given below.

Write a query to display the flight_id, from_location, to_location, number of services as "No_of_Services", average ticket price as "Average_Price" whose average ticket price is greater than the total average ticket cost of all flights. Order the result by lowest average price.

4 rows

```

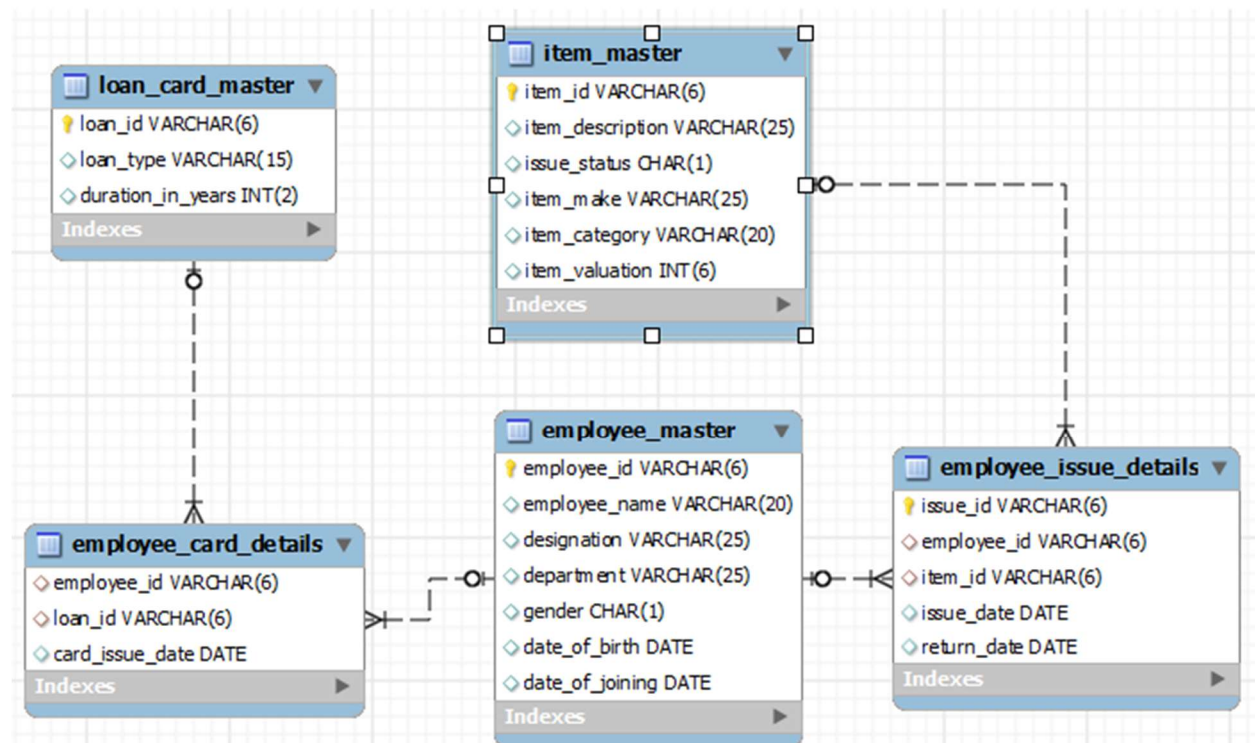
select af.flight_id, af.from_location, af.to_location, count(afd.flight_departure_date) as
No_of_Services,
avg(afd.price) as Average_Price from air_flight af join air_flight_details afd
on af.flight_id=afd.flight_id group by af.flight_id having avg(afd.price)>
(select avg(afd.price) from air_flight_details afd) order by afd.price;

```

FLIGHT_ID	FROM_LOCATION	TO_LOCATION	NO_OF_SERVICES	AVERAGE_PRICE
1262	HYDERABAD	CHENNAI	2	3444.50000
1265	CHENNAI	HYDERABAD	4	3499.25000
916	CHENNAI	HYDERABAD	4	3699.50000

1011	HYDERABAD	CHENNAI	3	4108.3333
------	-----------	---------	---	-----------

ANSI SQL Item Loan Management System Schema



Item Loan Database Queries

1. Please follow instructions given below.

Write a query to display category and number of items in that category. Give the count an alias name of Count_category. Display the details on the sorted order of count in descending order.

3 rows

```

SELECT item_category , count(item_id) Count_category
FROM item_master
GROUP BY item_category order by count_category DESC;

```

Filter:		
item_category	Count_category	
furniture	15	
Crockery	4	
Stationary	3	

2. Please follow instructions given below.

Write a query to display the number of employees in HR department. Give the alias name as No_of_Employees.

1 row

```
SELECT count(employee_id) AS No_of_Employees
FROM employee_master
WHERE department= 'HR'
```

Filter:		
No_of_Employees		
2		

3. Please follow instructions given below.

Write a query to display employee id, employee name, designation and department for employees who have never been issued an item as a loan from the company. Display the records sorted in ascending order based on employee id.

1 row

```
select employee_id, employee_name, designation, department from employee_master
where employee_id
```

not in (select employee_id from employee_issue_details) order by employee_id;

	employee_id	employee_name	designation	department
▶	E00005	Radica	Manager	HR
*	NULL	NULL	NULL	NULL

4. Please follow instructions given below.

Write a query to display the employee id, employee name who was issued an item of highest valuation.

In case of multiple records, display the records sorted in ascending order based on employee id.

[Hint Suppose an item called dinning table is of 22000 and that is the highest price of the item that has been issued. So display the employee id and employee name who issued dinning table whose price is 22000.]

1 row

```
select em.employee_id,em.employee_name from employee_master em join employee_issue_details  
eid
```

```
on em.employee_id=eid.employee_id join item_master im on eid.item_id=im.item_id
```

```
and im.item_valuation>=all(select im.item_valuation from employee_master em
```

```
join employee_issue_details eid
```

```
on em.employee_id=eid.employee_id join item_master im on eid.item_id=im.item_id)
```

```
order by employee_id;
```

	employee_id	employee_name
▶	E00004	Zuben

5. Please follow instructions given below.

Write a query to display issue_id, employee_id, employee_name.

Display the records sorted in ascending order based on issue id.

9 rows

```
select eid.issue_id,eid.employee_id,em.employee_name from employee_issue_details eid join
```

```
employee_master em on eid.employee_id=em.employee_id group by eid.issue_id,eid.employee_id
```

Filter:		Export:	
	issue_id	employee_id	employee_name
▶	ISS001	E00001	Ram
	ISS002	E00001	Ram
	ISS003	E00002	Abhay
	ISS004	E00003	Anita
	ISS005	E00003	Anita
	ISS006	E00003	Anita
	ISS007	E00004	Zuben
	ISS008	E00006	John
	ISS009	E00004	Zuben

order by eid.issue_id;

6.Please follow instructions given below.

Write a query to display employee id, employee name who don't have loan cards.

Display the records sorted in ascending order based on employee id.

3 rows

SELECT employee_id, employee_name

FROM employee_master

WHERE employee_id NOT IN (SELECT employee_id FROM employee_card_details)

order by employee_id;

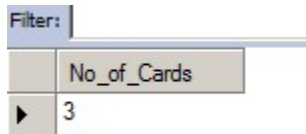
Filter:		
	employee_id	employee_name
▶	E00004	Zuben
	E00005	Radica
	E00006	John
*	NULL	NULL

7. Please follow instructions given below.

Write a query to count the number of cards issued to an employee “Ram”. Give the count an alias name as No_of_Cards.

1 row

```
select count(eid.loan_id) as No_of_Cards from employee_card_details eid join employee_master em
on eid.employee_id=em.employee_id where em.employee_name='Ram'
```



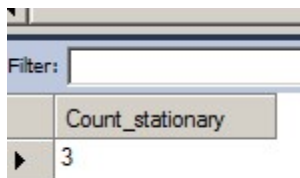
Filter:	
No_of_Cards	
▶	3

8. Please follow instructions given below.

Write a query to display the count of customers who have gone for loan type stationary. Give the count an alias name as Count_stationary.

1 row

```
select count(ecd.employee_id) as Count_Stationary from employee_card_details ecd
join loan_card_master lcm on ecd.loan_id=lcm.loan_id where lcm.loan_type='Stationary'
```



Filter:	
Count_stationary	
▶	3

9. Please follow instructions given below.

Write a query to display the employee id, employee name and number of items issued to them. Give the number of items an alias name as Count. Display the details in descending order of count and then by employee id in ascending order. Consider only employees who have been issued atleast 1 item.

5 rows

```
select em.employee_id,em.employee_name,count(eid.item_id) as Count from employee_master em
join
employee_issue_details eid on em.employee_id=eid.employee_id group by em.employee_id having
```


count(eid.item_id)>=1 order by Count desc,employee_id asc;



employee_id	employee_name	Count
E00003	Anita	3
E00001	Ram	2
E00004	Zuben	2
E00002	Abhay	1
E00006	John	1

10.Please follow instructions given below.

Write a query to display the employee id, employee name who was issued an item of minimum valuation.

In case of multiple records, display them sorted in ascending order based on employee id.

[Hint Suppose an item called pen is of rupees 20 and that is the lowest price. So display the employee id and employee name who issued pen where the valuation is 20.]

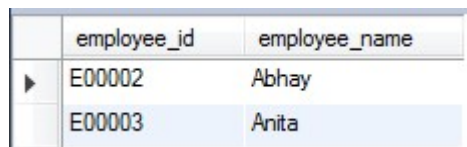
2 rows

```
select em.employee_id,em.employee_name from employee_master em join employee_issue_details  
eid
```

```
on em.employee_id=eid.employee_id join item_master im on eid.item_id=im.item_id
```

```
and im.item_valuation<=all (select im.item_valuation from employee_master em join  
employee_issue_details eid
```

```
on em.employee_id=eid.employee_id join item_master im on eid.item_id=im.item_id) order by  
employee_id;
```



employee_id	employee_name
E00002	Abhay
E00003	Anita

11.Please follow instructions given below.

Write a query to display the employee id, employee name and total valuation of the product issued to each employee. Give the alias name as TOTAL_VALUATION.

Display the records sorted in ascending order based on employee id.

Consider only employees who have been issued atleast 1 item.

5 rows

```
select em.employee_id,em.employee_name,sum(im.item_valuation) as TOTAL_VALUATION
from employee_master em
join employee_issue_details eid on em.employee_id=eid.employee_id join item_master im
on eid.item_id=im.item_id group by em.employee_id having count(im.item_valuation)>=1
order by em.employee_id;
```



The screenshot shows a database query result window with a table containing 5 rows. The table has three columns: employee_id, employee_name, and TOTAL_VALUATION. The data is as follows:

employee_id	employee_name	TOTAL_VALUATION
E00001	Ram	7000.00
E00002	Abhay	1500.00
E00003	Anita	15500.00
E00004	Zuben	25500.00
E00006	John	4500.00

12. Please follow instructions given below.

Write a query to display distinct employee id, employee name who kept the item issued for more than a year. Hint: Use Date time function to calculate the difference between item issue and return date. Display the records only if it is more than 365 Days.

Display the records sorted in ascending order based on employee id.

5 rows

```
select distinct em.employee_id,em.employee_name from employee_master em join
employee_issue_details eid
on em.employee_id=eid.employee_id where datediff(return_date,issue_date)>365 order by
```

Filter:		
	employee_id	employee_name
▶	E00001	Ram
	E00002	Abhay
	E00003	Anita
	E00004	Zuben
	E00006	John

employee_id;

13.Please follow instructions given below.

Write a query to display employee id, employee name and count of items of those who asked for more than 1 furniture. Give the alias name for count of items as COUNT_ITEMS.

Display the records sorted in ascending order on employee id.

2 rows

select em.employee_id,em.employee_name,count(im.item_id) as COUNT_ITEMS from
employee_master em

join employee_issue_details eid on em.employee_id=eid.employee_id join item_master im

on eid.item_id=im.item_id where item_category='furniture' group by employee_id having

count(COUNT_ITEMS)>1 order by employee_id;

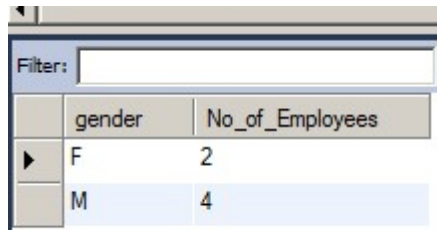
Filter:		Export:	Autos
	employee_id	employee_name	COUNT_ITEMS
▶	E00001	Ram	2
	E00003	Anita	3

14.Please follow instructions given below.

Write a query to display the number of men & women Employees. The query should display the gender and number of Employees as No_of_Employees. Display the records sorted in ascending order based on gender.

2 rows

select gender,count(employee_id) as No_of_Employees from employee_master group by



gender	No_of_Employees
F	2
M	4

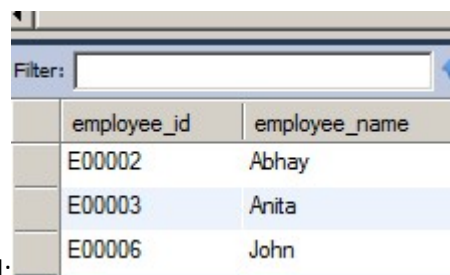
gender order by gender;

15.Please follow instructions given below.

Write a query to display employee id, employee name who joined the company after 2005. Display the records sorted in ascending order based on employee id.

3 rows

select employee_id,employee_name from employee_master where year(date_of_joining)>2005



employee_id	employee_name
E00002	Abhay
E00003	Anita
E00006	John

order by employee_id;

16.Please follow instructions given below.

Write a query to get the number of items of the furniture category issued and not issued. The query should display issue status and the number of furniture as No_of_Furnitures.

Display the records sorted in ascending order based on issue_status.

2 rows

select issue_status,count(item_id) as No_of_Furnitures from item_master where
item_category='furniture' group by issue_status order by

issue_status;

Filter:		
	issue_status	No_of_Fumitures
►	N	6
	Y	9

17.Please follow instructions given below.

Write a query to find the number of items in each category, make and description. The Query should display Item Category, Make, description and the number of items as No_of_Items. Display the records in ascending order based on Item Category, then by item make and then by item description.

16 rows

```
select item_category,item_make,item_description,count(item_id) as No_of_Items from
item_master im group by item_category,item_make,item_description order by
item_category,item_make,item_description;
```

Filter:		Export:		Autosize:	
	item_category	item_make	item_description	No_of_Items	
►	Crockery	Bonechina	Dining Set	1	
	Crockery	Bonechina	Tea Set	1	
	Crockery	Glass	Dining Set	1	
	Crockery	Glass	Tea Set	1	
	fumiture	Steel	Cupboard	2	
	fumiture	Steel	Side Table	1	
	fumiture	Steel	Single Bed	2	
	fumiture	Steel	Tea Table	2	
	fumiture	Wooden	Dining Chair	1	
	fumiture	Wooden	Dining Table	1	
	fumiture	Wooden	Double Bed	2	
	fumiture	Wooden	Side Table	1	
	fumiture	Wooden	Sofa	1	
	fumiture	Wooden	Tea Table	2	
	Stationary	Plastic	Pen	2	
	Stationary	Wooden	Pencil	1	

18. Please follow instructions given below.

Write a query to display employee id, employee name, item id and item description of employees who were issued item(s) in the month of January 2013. Display the records sorted in order based on employee id and then by item id in ascending order.

1 row

```
select em.employee_id,em.employee_name,im.item_id,im.item_description from employee_master em
join
```

```
employee_issue_details eid on em.employee_id=eid.employee_id join item_master im on
```

```
eid.item_id=im.item_id where year(eid.issue_date)=2013 and month(eid.issue_date)=01 order by
```



The screenshot shows a database query result window. At the top, there is a 'Filter:' text box, an 'Export:' button with a printer icon, and an 'Autosize:' button with a vertical double-headed arrow icon. Below these is a table with four columns: 'employee_id', 'employee_name', 'item_id', and 'item_description'. The table contains one row of data: 'E00002', 'Abhay', 'I00005', and 'Side Table'.

employee_id	employee_name	item_id	item_description
E00002	Abhay	I00005	Side Table

```
em.employee_id,im.item_id;
```

19. Please follow instructions given below.

Write a query to display the employee id, employee name and count of item category of the employees who have been issued items in at least 2 different categories.

Give the alias name for category count as COUNT_CATEGORY.

Display the records sorted in ascending order based on employee id.

1 row

```
select em.employee_id,em.employee_name,count(distinct im.item_category) as COUNT_CATEGORY
from employee_master em
```

```
join employee_issue_details eid on em.employee_id=eid.employee_id join item_master im
```

```
on eid.item_id=im.item_id group by em.employee_id having COUNT_CATEGORY>=2
```

```
order by em.employee_id;
```

Filter:		Export:	Autosize:
	employee_id	employee_name	COUNT_CATEGORY
▶	E00004	Zuben	2

20. Please follow instructions given below.

Write a query to display the item id , item description which was never issued to any employee. Display the records sorted in ascending order based on item id.

14 rows

select item_id,item_description from item_master where item_id not in (select item_id
from employee_issue_details) order by item_id;

Filter:		
	item_id	item_description
▶	I00002	Dining Table
	I00003	Tea Table
	I00006	Tea Table
	I00009	Sofa
	I00011	Cupboard
	I00013	Double Bed
	I00014	Single Bed
	I00015	Single Bed
	I00016	Tea Set
	I00017	Tea Set
	I00019	Dining Set
	I00020	Pencil
	I00021	Pen
	I00022	Pen
*	NULL	NULL

21. Please follow instructions given below.

Write a query to display the employee id, employee name and total valuation for the employees who has issued minimum total valuation of the product. Give the alias name for total valuation as TOTAL_VALUATION.

[Hint: Suppose an employee E00019 issued item of price 5000, 10000, 12000 and E00020 issue item of price 2000, 7000 and 1000. So the valuation of items taken by E00019 is 27000 and for E00020 it is 10000. So the employee id, employee name of E00020 should be displayed.]

1 row

```
select em.employee_id,em.employee_name,sum(im.item_valuation) as TOTAL_VALUATION from
employee_master em
```

```
join employee_issue_details eid on em.employee_id=eid.employee_id join item_master im on
```

```
eid.item_id=im.item_id group by em.employee_id having sum(im.item_valuation) <= all
```

```
(select sum(im.item_valuation) from employee_master em
```

```
join employee_issue_details eid on em.employee_id=eid.employee_id join item_master im on
```

```
eid.item_id=im.item_id group by em.employee_id) order by employee_id;
```

filter:		Export:	Autosize:
employee_id	employee_name	TOTAL_VALUATION	
E00002	Abhay	1500.00	

22.Please follow instructions given below.

Write a query to display the employee id, employee name, card issue date and card valid date.

Order by employee name and then by card valid date. Give the alias name to display the card valid date as CARD_VALID_DATE.

[Hint: Validity in years for the loan card is given in loan_card_master table. Validity date is calculated by adding number of years in the loan card issue date. If the duration of year is zero then display AS 'No Validity Date'.]

```
SELECT ecd.employee_id,employee_name,
```

```
card_issue_date, if(lcd.duration_in_years=0, 'NO-VALIDITY DATE', date_add(ec.card_issue_date,
interval duration_in_years year)) as CARD_VALIDITY_DATE
```

```
FROM employee_master em INNER JOIN
```

```
employee_card_details ecd
```



```

ON em.employee_id=ecd.employee_id

INNER JOIN loan_card_master lcd

ON ecd.loan_id=lcd.loan_id

order by employee_name, CARD_VALID_DATE;

```

Filter:		Export:	Autosize:
employee_id	employee_name	card_issue_date	CARD_VALID_DATE
E00002	Abhay	2007-02-01	2012-02-01
E00002	Abhay	2007-03-11	No Validity Date
E00003	Anita	2007-04-15	2008-04-15
E00003	Anita	2007-04-15	2012-04-15
E00003	Anita	2007-04-15	No Validity Date
E00001	Ram	2002-12-14	2003-12-14
E00001	Ram	2000-01-01	2005-01-01
E00001	Ram	2000-01-01	No Validity Date

23.Please follow instructions given below.

Write a query to display the employee id, employee name who have not issued with any item in the year 2013. Hint: Exclude those employees who was never issued with any of the items in all the years. Display the records sorted in ascending order based on employee id.

3 rows

```

select distinct em.employee_id,em.employee_name from employee_master em join
employee_issue_details eid on

em.employee_id=eid.employee_id where em.employee_id not in

(select employee_id from employee_issue_details where year(issue_date)=2013)

order by employee_id;

```

Filter:	
employee_id	employee_name
E00001	Ram
E00003	Anita
E00006	John

24. Please follow instructions given below.

Write a query to display issue id, employee id, employee name, item id, item description and issue date. Display the data in descending order of date and then by issue id in ascending order.

9 rows

```
select eid.issue_id,em.employee_id,em.employee_name,im.item_id,im.item_description,eid.issue_date
from employee_issue_details eid join employee_master em on eid.employee_id=em.employee_id
join item_master im on eid.item_id=im.item_id order by eid.issue_date desc,eid.issue_id;
```

Filter:		Export:		Autosize:	
issue_id	employee_id	employee_name	item_id	item_description	issue_date
ISS009	E00004	Zuben	I00018	Dining Set	2013-04-18
ISS007	E00004	Zuben	I00012	Double Bed	2013-04-14
ISS003	E00002	Abhay	I00005	Side Table	2013-01-03
ISS008	E00006	John	I00018	Dining Set	2012-08-18
ISS006	E00003	Anita	I00010	Cupboard	2012-03-14
ISS001	E00001	Ram	I00001	Tea Table	2012-02-03
ISS002	E00001	Ram	I00004	Side Table	2012-02-03
ISS004	E00003	Anita	I00007	Dining Chair	2010-07-04
ISS005	E00003	Anita	I00008	Tea Table	2010-07-04

25. Write a query to display the employee id, employee name and total valuation for employee who has issued maximum total valuation of the product. Give the alias name for total valuation as TOTAL_VALUATION.

[Hint: Suppose an employee E00019 issued item of price 5000, 10000, 12000 and E00020 issue item of price 2000, 7000, and 1000. So the valuation of items taken by E00019 is 27000 and for E00020 it is 10000. So the employee id, employee name and total valuation of E00019 should display.]

1 row

```
select em.employee_id,em.employee_name,sum(im.item_valuation) as TOTAL_VALUATION
from employee_master em join employee_issue_details eid on em.employee_id=eid.employee_id
join item_master im on eid.item_id=im.item_id group by em.employee_id having
sum(im.item_valuation)
>= all (select sum(im.item_valuation) from employee_master em join employee_issue_details eid on
em.employee_id=eid.employee_id
join item_master im on eid.item_id=im.item_id group by em.employee_id);;
```

Iter: <input type="text"/>	Export: 	Autosize: 
employee_id	employee_name	TOTAL_VALUATION
E00004	Zuben	25500.00