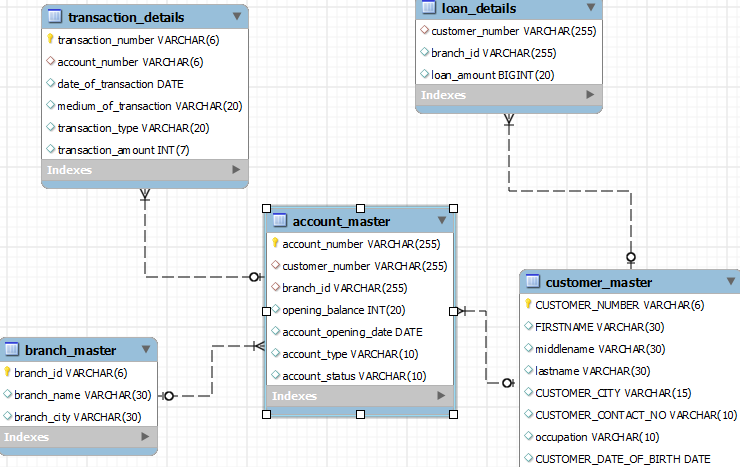
**Bank Management System Queries:**

**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 ld.customer\_number, firstname, lastname**

**FROM customer\_master cm INNER JOIN loan\_details ld**

**ON cm.customer\_number=ld.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 then C00013.**

**SELECT ld.customer\_number, firstname, lastname**

**FROM customer\_master cm INNER JOIN loan\_details ld**

**ON cm.customer\_number=ld.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 ld.customer\_number, firstname,branch\_id, loan\_amount**

**FROM customer\_master cm INNER JOIN loan\_details ld**

**ON cm.customer\_number=ld.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.**

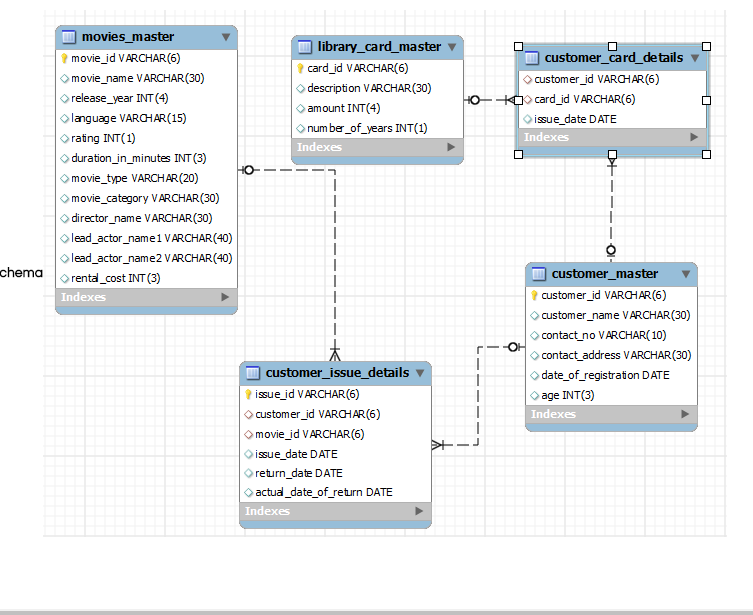
**SELECT firstname, customer\_city,account\_number**

**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. 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.

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 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”. <br/>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\_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;**

|  |  |  |  |
| --- | --- | --- | --- |
| MOVIE\_ID | MOVIE\_NAME | RELEASE\_YEAR | DIRECTOR\_NAME |
| M00004 | INCEPTION | 2010 | CHRISTOPHER NOLAN |
| M00011 | TITANIC | 1997 | JAMES CAMERON |

9.Please follow instructions given below.

<br>

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. <br>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. <br> 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. <br> 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 NOLAN | INCEPTION | LEONARDO 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. <br> 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 decending 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.&nbsp;

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.

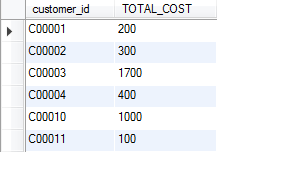
Write a query to display customer id and total rent paid by the customers who are issued with the videos. Need not display the customers who has not taken / issued with any videos. Hint: Alias Name for total rent paid is TOTAL\_COST. Display the records sorted in ascending order based on customer id

6 rows

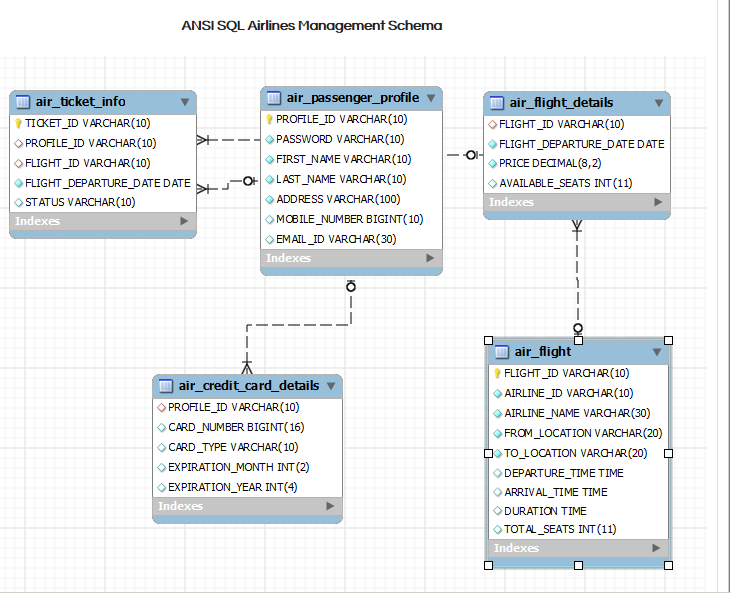
select cid.customer\_id,sum(mm.rental\_cost) as TOTAL\_COST from customer\_issue\_details cid

join movies\_master mm

on cid.movie\_id=mm.movie\_id group by customer\_id order by customer\_id;



Airline Filght 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 ST,ERNAKULAM-12 | 8 |

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;

|  |  |  |
| --- | --- | --- |
| ROFILE\_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;t

|  |  |  |
| --- | --- | --- |
| 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 | DELAYED\_DEPARTURE\_TIME | DELAYED\_ARRIVAL\_TIME |
| PFL005 | SIVA | KUMAR | 916 | 2013-05-06 | 19:55:00 | 21:00:00 | 20:55:00 | 22:00:00 |

|  |  |
| --- | --- |
| 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.

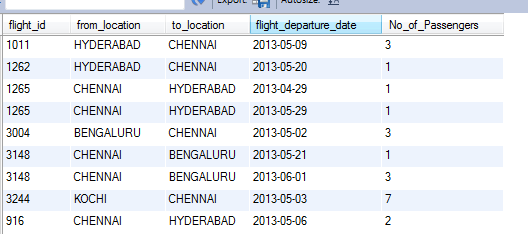
Display the records sorted in ascending order based on flight id and then by flight departure date.

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;



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

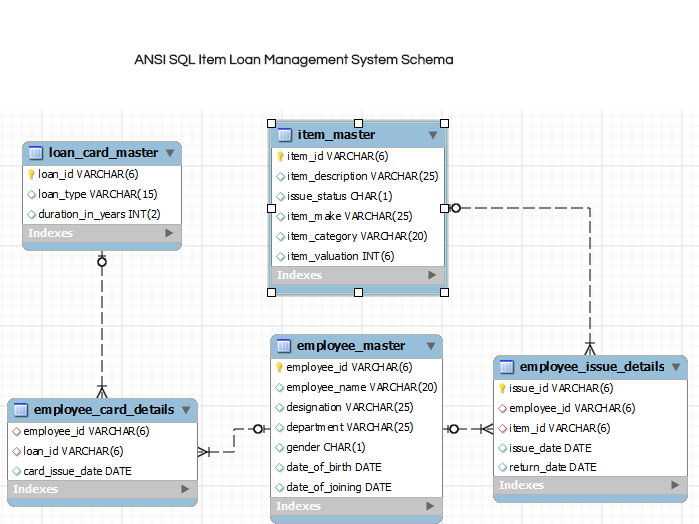
select af.flight\_id,af.from\_location,af.to\_location,count(afd.flight\_departure\_date) as No\_of\_Services,

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 | 3444.500000 |
| 1265 | CHENNAI |  | HYDERABAD | 4 | 3499.250000 |
| 916 | CHENNAI |  | HYDERABAD | 4 | 3699.500000 |
| 1011 | HYDERABAD |  | CHENNAI | 3 | 4108.333333 |

****

**Item Loan Database Queries**

1.Please follow instructions given below.

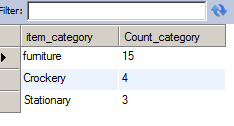
Write a query to display category and number of items in that category. Give the count an alias name of Count\_category. Display the details on the sorted order of count in descending order.

3 rows

**SELECT item\_category , count(item\_id) Count\_category**

**FROM item\_master**

**GROUP BY item\_category order by count\_category DESC;**



2.Please follow instructions given below.

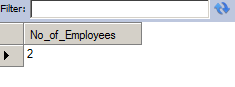
Write a query to display the number of employees in HR department. Give the alias name as No\_of\_Employees.

1 row

SELECT count(employee\_id) AS No\_of\_Employees

FROM employee\_master

WHERE department= 'HR'



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;

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;



5.Please follow instructions given below.

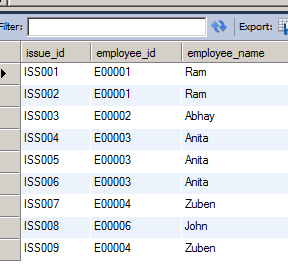
Write a query to display issue\_id, employee\_id, employee\_name.

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

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.

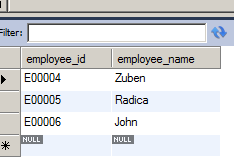
3 rows

SELECT employee\_id, employee\_name

FROM employee\_master

WHERE employee\_id NOT IN ( SELECT employee\_id FROM employee\_card\_details )

order by employee\_id;



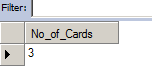
7.Please follow instructions given below.

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'



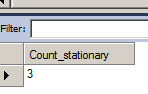
8.Please follow instructions given below.

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'



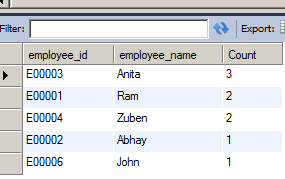
9.Please follow instructions given below.

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;

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.]

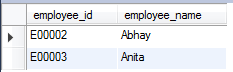
2 rows

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;



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

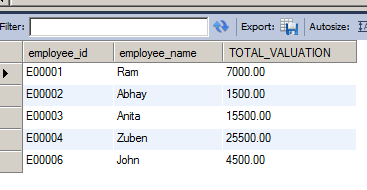
select em.employee\_id,em.employee\_name,sum(im.item\_valuation) as TOTAL\_VALUATION

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;



12.Please follow instructions given below.

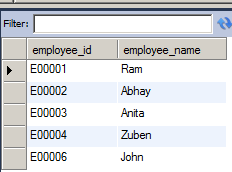
Write a query to display distinct employee id, employee name who kept the item issued for more than a year. Hint: Use Date time function to calculate the difference between item issue and return date. Display the records only if it is more than 365 Days.

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

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.

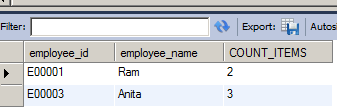
2 rows

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;

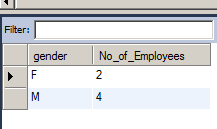


14.Please follow instructions given below.

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

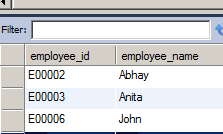
gender order by gender;

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

order by employee\_id;

16.Please follow instructions given below.

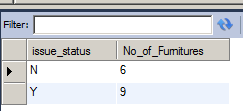
Write a query to get the number of items of the furniture category issued and not issued. The query should display issue status and the number of furniture as No\_of\_Furnitures.

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;



17.Please follow instructions given below.

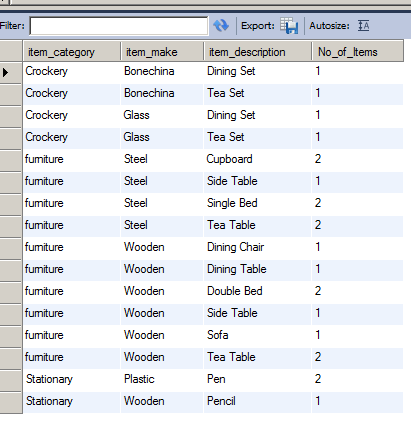
Write a query to find the number of items in each category, make and description. The Query should display Item Category, Make, description and the number of items as No\_of\_Items. Display the records in ascending order based on Item Category, then by item make and then by item description.

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;



18.Please follow instructions given below.

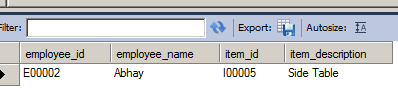
Write a query to display employee id, employee name, item id and item description of employees who were issued item(s) in the month of January 2013. Display the records sorted in order based on employee id and then by item id in ascending order.

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

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.

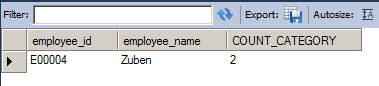
1 row

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;



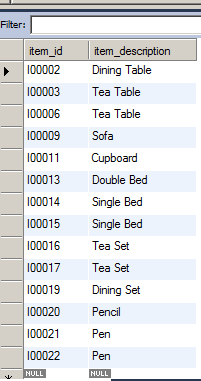
20.Please follow instructions given below.

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;



21.Please follow instructions given below.

Write a query to display the employee id, employee name and&nbsp;&nbsp;total valuation&nbsp;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

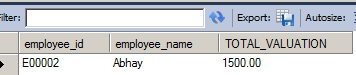
join employee\_issue\_details eid on em.employee\_id=eid.employee\_id join item\_master im on

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;



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

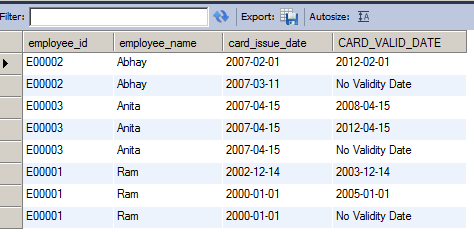
employee\_card\_details ecd

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;



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.

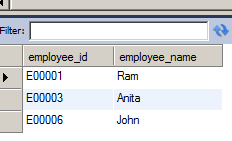
3 rows

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;



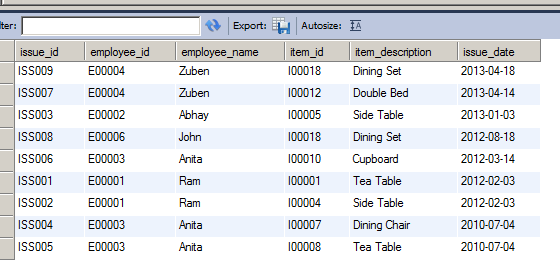
24.Please follow instructions given below.

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;

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.&nbsp; Give the alias name for total valuation as TOTAL\_VALUATION.&nbsp;

<br>[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);;

