**Item Loan Database Queries**

1.Please follow instructions given below.

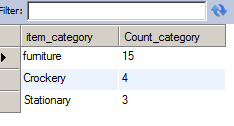
Write a query to display category and number of items in that category. Give the count an alias name of Count\_category. Display the details on the sorted order of count in descending order.

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.

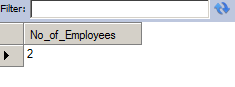
Write a query to display the number of employees in HR department. Give the alias name as No\_of\_Employees.

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.

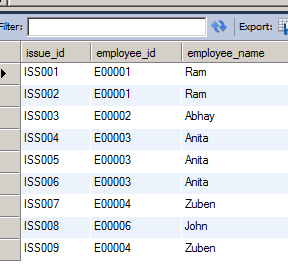
Write a query to display issue\_id, employee\_id, employee\_name.

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.

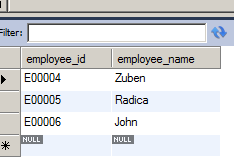
3 rows

SELECT employee\_id, employee\_name

FROM employee\_master

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

order by employee\_id;



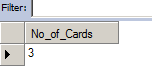
7.Please follow instructions given below.

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'



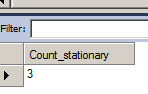
8.Please follow instructions given below.

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'



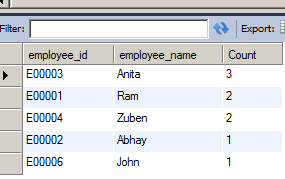
9.Please follow instructions given below.

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.]

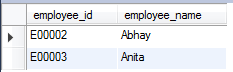
2 rows

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

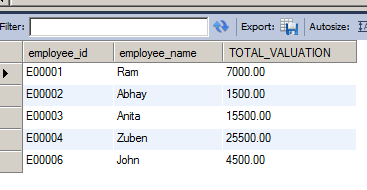
select em.employee\_id,em.employee\_name,sum(im.item\_valuation) as TOTAL\_VALUATION

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.

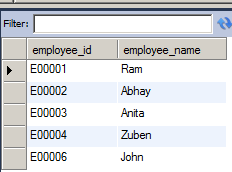
Write a query to display distinct employee id, employee name who kept the item issued for more than a year. Hint: Use Date time function to calculate the difference between item issue and return date. Display the records only if it is more than 365 Days.

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.

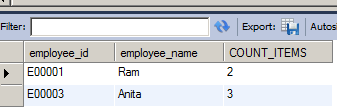
2 rows

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;

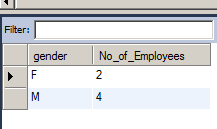


14.Please follow instructions given below.

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

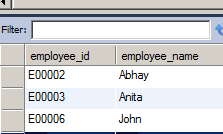
gender order by gender;

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.

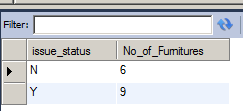
Write a query to get the number of items of the furniture category issued and not issued. The query should display issue status and the number of furniture as No\_of\_Furnitures.

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.

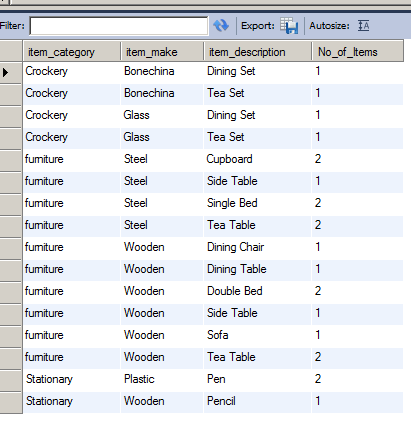
Write a query to find the number of items in each category, make and description. The Query should display Item Category, Make, description and the number of items as No\_of\_Items. Display the records in ascending order based on Item Category, then by item make and then by item description.

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.

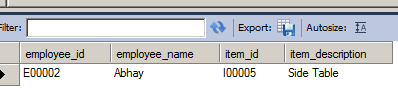
Write a query to display employee id, employee name, item id and item description of employees who were issued item(s) in the month of January 2013. Display the records sorted in order based on employee id and then by item id in ascending order.

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.

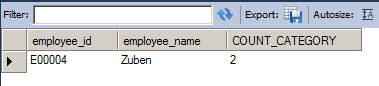
1 row

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;



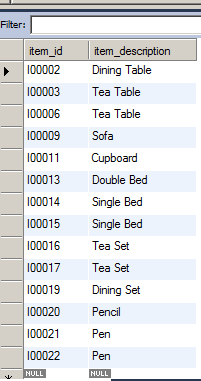
20.Please follow instructions given below.

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

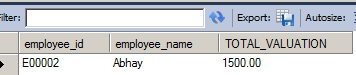
join employee\_issue\_details eid on em.employee\_id=eid.employee\_id join item\_master im on

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

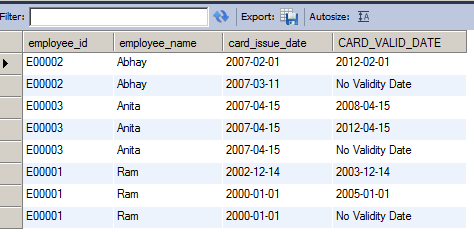
employee\_card\_details ecd

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.

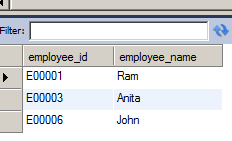
3 rows

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;



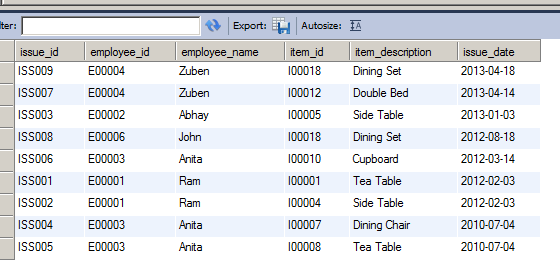
24.Please follow instructions given below.

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);;

