

Islamic University of Technology

CSE 4308

Database Management Systems Lab

Mukit Mahdin

ID:200042170

Department: Computer Science and Engineering

Program: Software Engineering

Group: B

Tasks:

The initial table creation codes and the codes for inserting the rows in the tables were given in the banking.sql file.

Then the below sql statements will help us to perform the required tasks.

1.

```
SELECT distinct customer_name
from account natural join borrower;
```
2.

```
SELECT distinct customer_name,customer_street,customer_city
from account natural join borrower natural join customer;
```
3.

```
select customer_name, customer_city
from account full join customer
where account_number=(SELECT account_number
from account full join loan
where account_number=null);
```
4.

```
select sum(assets) as tot_assets
from branch;
```
5.

```
select branch.branch_name,count(account_number) as number_of_accounts
from account full join branch
group by branch.branch_name;

select branch.branch_city,count(account_number) as number_of_accounts
from account full join branch
group by branch.branch_city;
```
6.

```
select branch_name,avg(balance) as avg_balance
from account natural join branch
group by branch.branch_name
order by branch_name ascending;
```
7.

```
select branch_name,sum(balance) as tot_balance
from account natural join branch
group by branch.branch_name;
```

8.

```
select branch_name,avg(amount) as avg_loan
from branch natural join loan
group by branch_name
having branch_name not in ('Horseneck');

select branch_name,avg(amount) as avg_loan
from branch natural join loan
where branch_name not in ('Horseneck')
group by branch_name;
```
9.

```
select max(balance) as max_balance,customer_name,account_number
from account natural join customer;
```
10.

```
select distinct customer_name,customer_city,customer_street
from customer natural join branch
where branch_city=customer_city;
```
11.

```
select branch_city,branch_name,avg(amount) as avg_loan
from branch natural join loan
group by branch_city
having avg(amount) >=1500;
```
12.

```
select branch_name
from (select avg(assets) as avg_balance from branch)
as table1,(select branch_name,sum(assets)
as tot_balance from branch
group by branch_name) as table2
where avg_balance<tot_balance;
```
13.

```
select customer_name
from (select customer_name,sum(balance)
as tot_balance,sum(amount) as tot_loan
from customer natural join loan natural join account
group by customer_name) as table1
where tot_balance>=tot_loan;
```
14.

```
select *
from (select account_number, balance,loan_number
from account full join loan) as table1
natural join (select * from branch full join customer) as table2
where account_number=null and loan_number=null
having count(loan_number)>0 and count(account_number);
```