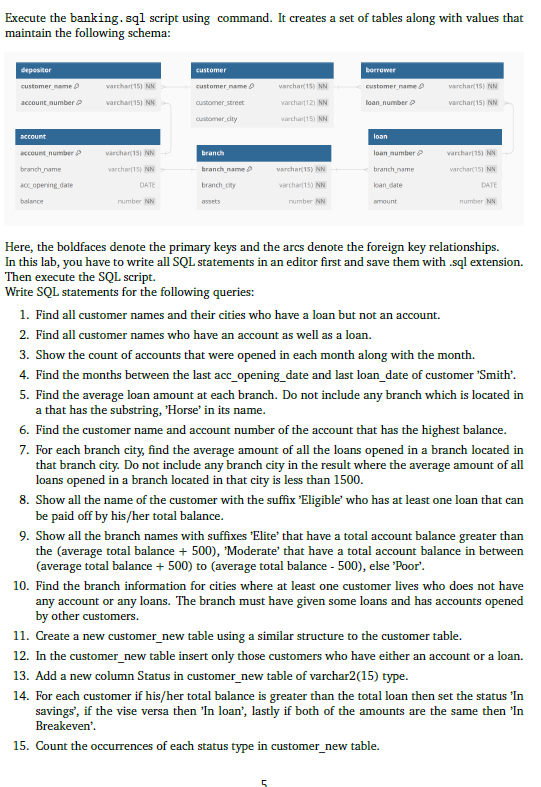
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | |
|  | Report on Lab-05  DATABASE MANAGEMENT SYSTEMS LAB | | | | | |  | |
|  |  | | | | | | |  | | |
| Submitted by  Adid-Al-Mahamud Shazid  Student Id: 210042172  Department: CSE  Programme: SWE  Course Title: CSE 4308  Submitted to  Zannatun Naim Sristy  Lecturer, Department of CSE  September 14, 2023 | |  | | | | |
|  |
|  | | |  |
|  | |  | | | |  | | | |

**Introduction**

In this lab class, we were given tasks based on advanced data manipulation techniques to solve using SQL command line to understand the basics of data definition and data manipulation. The given .sql file named banking.sql was executed before doing the following tasks.

**Task**



**Solution**

--Task 01--

select customer.customer\_name, customer.customer\_city

from customer, borrower

where customer.customer\_name=borrower.customer\_name

minus

select customer.customer\_name, customer.customer\_city

from customer, depositor

where customer.customer\_name=depositor.customer\_name;

--Task 02--

select depositor.customer\_name

from customer, depositor

where customer.customer\_name=depositor.customer\_name

intersect

select borrower.customer\_name

from borrower, customer

where customer.customer\_name=borrower.customer\_name;

--Task 03--

select extract (month from acc\_opening\_date)as months,count(\*) as count

from account

group by extract (month from acc\_opening\_date);

--Task 04--

select months\_between

(

    (select max(account.acc\_opening\_date)

    from depositor, account

    where depositor.account\_number=account.account\_number

    and depositor.customer\_name= 'Smith'),

    (select max(loan.loan\_date)

    from borrower, loan

    where borrower.loan\_number=loan.loan\_number

    and borrower.customer\_name= 'Smith')

)month from dual;

--Task 05--

select a.branch\_name, a.avg\_amount from

(select branch\_name, avg(amount) as avg\_amount from loan group by branch\_name) a,

branch where branch.branch\_name=a.branch\_name and

branch.branch\_city not like '%HORSE%';

--Task 06--

select customer\_name,account\_number from depositor

where account\_number in

(

    select account\_number from account

    where balance=(select max(balance) from account)

);

--Task 07--

select branch\_city,avg(amount) from loan,branch

where loan.branch\_name=branch.branch\_name

group by branch\_city

having avg(amount)>1500;

--Task 08--

select customer\_name||' '||'ELIGIBLE'

as customer\_name from depositor

where account\_number in

(

    select account\_number from account

    where balance>=

    (

        select sum(amount) from loan

        where loan.branch\_name=account.branch\_name

        and loan.loan\_number in

        (

            select loan\_number from borrower

            where borrower.customer\_name=depositor.customer\_name

        )

    )

);

--Task 09--

select branch\_name || ' Elite' as branch\_name from branch

where branch\_name in

(

    select branch\_name from account

    group by branch\_name

    having sum(balance) >

    (

        select avg(sum\_balance) + 500 from

        (

            select branch\_name, sum(balance) as sum\_balance from account

            group by branch\_name

        )

    )

)

union

select branch\_name || ' Moderate' as branch\_name from branch

where branch\_name in

(

    select branch\_name from account

    group by branch\_name

    having sum(balance) between

    (

        select avg(sum\_balance) + 500 from

        (

            select branch\_name, sum(balance) as sum\_balance from account

            group by branch\_name

        )

    )

    and

    (

        select avg(sum\_balance) - 500 from

        (

            select branch\_name, sum(balance) as sum\_balance from account

            group by branch\_name

        )

    )

)

union

select branch\_name || ' Poor' as branch\_name from branch

where branch\_name in

(

    select branch\_name from account

    group by branch\_name

    having sum(balance) <

    (

        select avg(sum\_balance) - 500 from

        (

            select branch\_name, sum(balance) as sum\_balance from account

            group by branch\_name

        )

    )

);

--Task 10--

select branch\_name, branch\_city from branch

where branch\_city in

(

    select customer\_city from customer

    where customer\_city not in

    (

        select customer\_city from customer

        where customer\_name in

        (

            select customer\_name from depositor

        )

        or

        customer\_name in

        (

            select customer\_name from borrower

        )

    )

)

and branch\_name in

(

    select branch\_name from loan

    group by branch\_name

    having count(\*) > 0

)

and branch\_name in

(

    select branch\_name from account

    group by branch\_name

    having count(\*) > 0

);

--Task 11--

create table customer\_new as

select \* from customer

where customer\_name='SPOON';

--Task 12--

insert into customer\_new

select \* from customer

where customer\_name in

(

    select customer\_name from depositor

)

or

customer\_name in

(

    select customer\_name from borrower

);

--Task 13--

alter table customer\_new

add status varchar2(15);

**Analysis and Explanation**

From this task I learnt some new functionalities and using sub queries properly.

**Difficulties**

I faced a few difficulties during several tasks.