Department of Computer Science and Engineering
Course Title: Database Systems
Course code: CSE302

Lab no: 03

Submitted by:
Student name: Suddip Paul Arnab
ID: 2022-1-60-356

Submitted To:
Dr. Mohammad Rezwanul Huq
Associate Professor
Department of CSE

Date of Submission: 25/02/2024

1) Find all branch names and cities with assets more than 1000000. (on single table)

```
SELECT branch_name,branch_city FROM Branch WHERE assets>1000000;
```

BRANCH_NAME	BRANCH_CITY
Redwood	Palo Alto
Perryridge	Horseneck
Round Hill	Horseneck
North Town	Rye
Brighton	Brooklyn

5 rows returned in 0.01 seconds Download

2) Find all account numbers and their balance which are opened in ‘Downtown’ branch or which have balance in between 600 and 750. (on single table)

```
SELECT account_number,balance FROM Account WHERE branch_name='Downtown'OR (balance>=600 AND balance<=750);
```

ACCOUNT_NUMBER	BALANCE
A-101	500
A-215	700
A-222	700
A-217	750
A-444	625

5 rows returned in 0.01 seconds Download

3) Find all account numbers which are opened in a branch located in ‘Rye’ city. (multiple tables)

```
SELECT account_number FROM Account NATURAL JOIN Branch WHERE Branch.branch_city='Rye';
```

ACCOUNT_NUMBER
A-333
A-444

2 rows returned in 0.04 seconds Download

4) Find all loan numbers which have amount greater than or equal to 1000 and their customers are living in ‘Harrison’ city. (multiple tables)

```
SELECT loan_number  FROM Customer NATURAL JOIN Loan WHERE Loan.amount>=1000 AND Customer.customer_city='Harrison';
```

LOAN_NUMBER
L-17
L-23
L-15
L-14
L-16
L-20
L-17
L-23
L-15
L-14
L-16
L-20

12 rows returned in 0.01 seconds Download

5) Display the account related information based on the descending order of the balance. (order by clause)

```
SELECT * FROM Account ORDER BY balance DESC;
```

ACCOUNT_NUMBER	BRANCH_NAME	BALANCE
A-201	Perryridge	900
A-333	Central	850
A-217	Brighton	750
A-215	Mianus	700
A-222	Redwood	700
A-444	North Town	625
A-101	Downtown	500
A-102	Perryridge	400
A-305	Round Hill	350

6) Display the customer related information in alphabetic order of customer cities. (order by clause)

```
SELECT * FROM Customer ORDER BY customer_city;
```

CUSTOMER_NAME	CUSTOMER_STREET	CUSTOMER_CITY
Brooks	Senator	Brooklyn
Hayes	Main	Harrison
Jones	Main	Harrison
Johnson	Alma	Palo Alto
Adams	Spring	Pittsfield
Lindsay	Park	Pittsfield
Williams	Nassau	Princeton
Curry	North	Rye
McBride	Safety	Rye
Smith	Main	Rye
Majeris	First	Rye
Jackson	University	Salt Lake
Green	Walnut	Stamford
Turner	Putnam	Stamford
Glenn	Sand Hill	Woodside

15 rows returned in 0.03 seconds Download

7) Find all customer names who have an account as well as a loan. (intersect)

```
SELECT customer_name FROM Depositor INTERSECT SELECT customer_name FROM Borrower;
```

CUSTOMER_NAME
Hayes
Jones
Smith

3 rows returned in 0.03 seconds Download

8) Find all customer related information who have an account or a loan. (union)

```
SELECT Customer_name,Customer_street,Customer_city FROM Customer NATURAL JOIN Depositor
UNION
SELECT Customer_name,Customer_street,Customer_city FROM Customer NATURAL JOIN Borrower;
```

CUSTOMER_NAME	CUSTOMER_STREET	CUSTOMER_CITY
Adams	Spring	Pittsfield
Curry	North	Rye
Hayes	Main	Harrison
Jackson	University	Salt Lake
Johnson	Alma	Palo Alto
Jones	Main	Harrison
Lindsay	Park	Pittsfield
Majeris	First	Rye
McBride	Safety	Rye
Smith	Main	Rye
Turner	Putnam	Stamford
Williams	Nassau	Princeton

12 rows returned in 0.01 seconds Download

9) Find all customer names and their cities who have a loan but not an account. (minus)

```
SELECT Customer_name,Customer_city FROM Customer NATURAL JOIN Borrower
MINUS
SELECT Customer_name,Customer_city FROM Customer NATURAL JOIN Depositor;
```

CUSTOMER_NAME	CUSTOMER_CITY
Adams	Pittsfield
Curry	Rye
Jackson	Salt Lake
McBride	Rye
Williams	Princeton

5 rows returned in 0.01 seconds Download

10) Find the total assets of all branches. (aggregate function)

```
SELECT SUM(assets) AS Total_Assets FROM Branch;
```

TOTAL_ASSETS
24600480

1 rows returned in 0.03 seconds Download

11) Find the average balance of accounts at each branch. (aggregate function)

```
SELECT branch_name,ROUND(AVG(balance),2) AS Average_balance FROM Branch NATURAL JOIN Account GROUP BY branch_name;
```

BRANCH_NAME	AVERAGE_BALANCE
Central	850
Downtown	500
Perryridge	650
Mianus	700
North Town	625
Round Hill	350
Redwood	700
Brighton	750

8 rows returned in 0.01 seconds Download

12) Find the average balance of accounts at each branch city. (aggregate function)

```
SELECT branch_city,ROUND(AVG(balance),2) AS Average_balance FROM Branch NATURAL JOIN Account GROUP BY branch_city;
```

BRANCH_CITY	AVERAGE_BALANCE
Palo Alto	700
Brooklyn	625
Horseneck	587.5
Rye	737.5

4 rows returned in 0.02 seconds Download

13) Find the lowest amount of loan at each branch. (aggregate function)

```
SELECT branch_name,MIN(amount) AS Lowest_Loan FROM Branch NATURAL JOIN Loan GROUP BY branch_name;
```

BRANCH_NAME	LOWEST_LOAN
Central	570
Downtown	1000
Perryridge	1300
Mianus	500
North Town	7500
Round Hill	900
Redwood	2000

7 rows returned in 0.03 seconds Download

14) Find the total number of loans at each branch. (aggregate function)

```
SELECT branch_name,COUNT(*) AS Loan_Numbers FROM Branch NATURAL JOIN Loan GROUP BY branch_name;
```

BRANCH_NAME	LOAN_NUMBERS
Central	1
Downtown	2
Perryridge	2
Mianus	1
North Town	1
Round Hill	1
Redwood	1

7 rows returned in 0.01 seconds Download

15) Find the customer name and account number of the account which has the highest balance.(aggregate function)

```
SELECT customer_name,account_number FROM Depositor NATURAL JOIN Account WHERE balance=(SELECT MAX(balance) FROM Account);
```

CUSTOMER_NAME	ACCOUNT_NUMBER
Johnson	A-201

1 rows returned in 0.02 seconds Download