DBMS Lab

Name: **Soham Das**

Section: **A1**

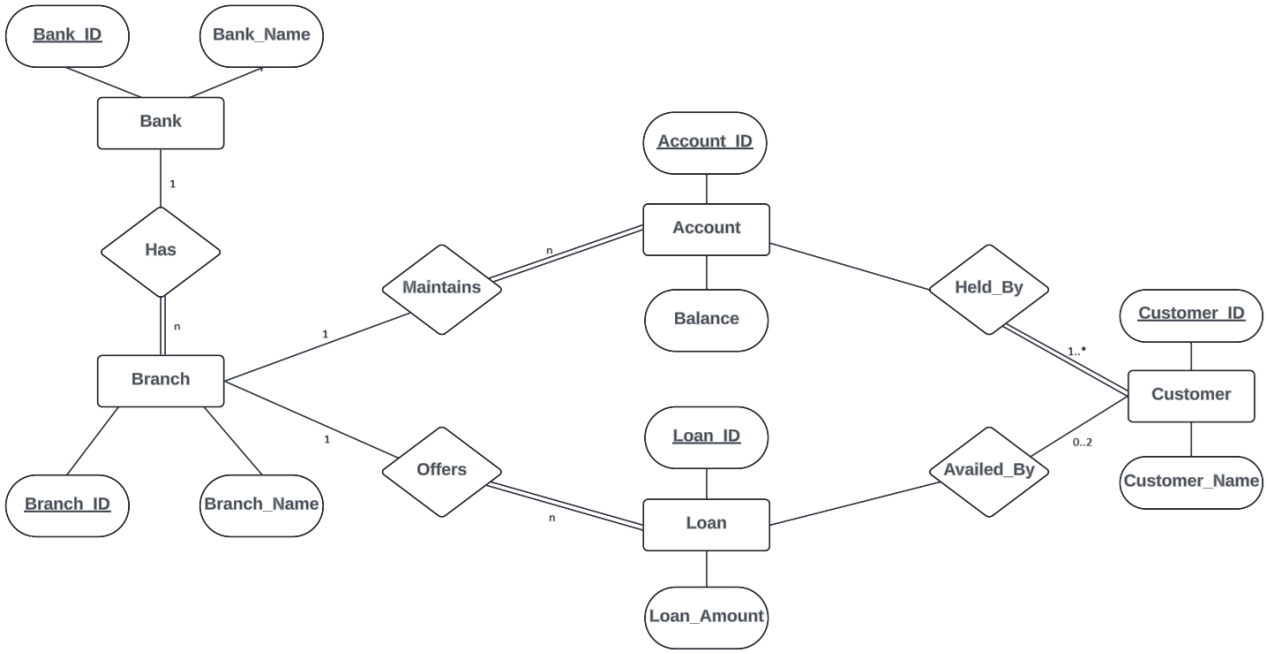
Roll No: **002311001004**

Assignment - 3

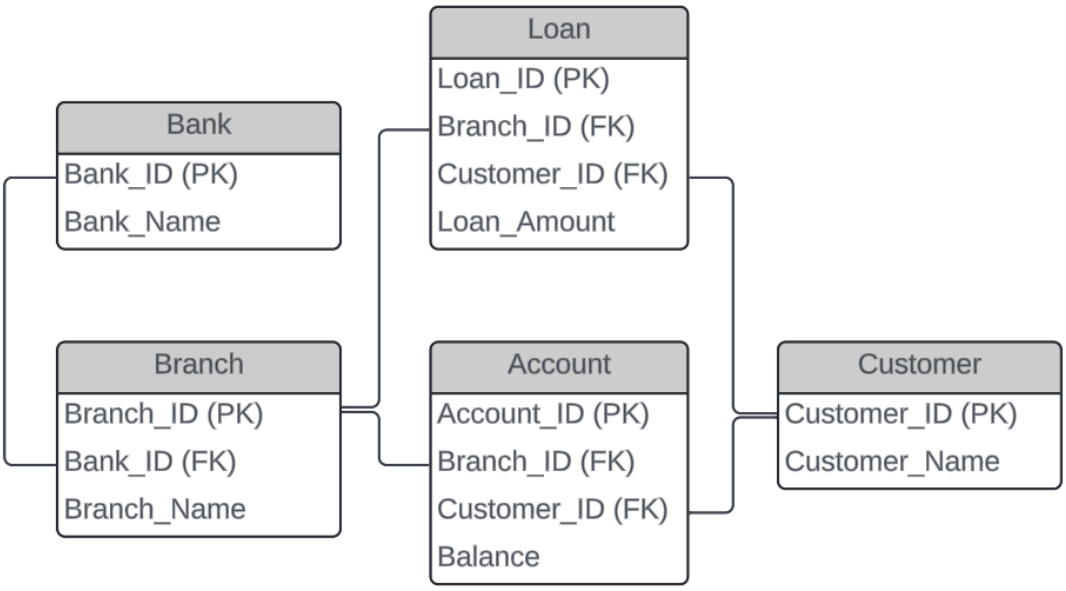
IT-UG2

Question: Consider a BANK database. Each bank can have multiple branches, and each branch can have multiple accounts and loans. Assumptions also can be made. Design an ER diagram and database schema for the system. Specify the primary key, foreign key and other constraints for all required tables. Draw the ER diagram in MS Word.

ER Diagram



Database Schema



1. Insert at least five tuples in each table.

```
--BANK Table
CREATE TABLE Bank (
   Bank_ID NUMBER (2) PRIMARY KEY,
   Bank_Name VARCHAR2(25) NOT NULL);

INSERT INTO Bank VALUES (10, 'SBI');
INSERT INTO Bank VALUES (20, 'PNB');
INSERT INTO Bank VALUES (30, 'HDFC');
INSERT INTO Bank VALUES (40, 'AXIS');
INSERT INTO Bank VALUES (50, 'ICICI');
```

BANK_ID	BANK_NAME
10	SBI
20	PNB
30	HDFC
40	AXIS
50	ICICI

```
--BRANCH Table
CREATE TABLE Branch (
   Branch_ID NUMBER (3) PRIMARY KEY,
   Branch_Name VARCHAR2(40) NOT NULL,
   Bank_ID NUMBER (2),
   FOREIGN KEY (Bank_ID) REFERENCES Bank(Bank_ID));

INSERT INTO Branch VALUES (101, 'Sealdah', 10);
INSERT INTO Branch VALUES (102, 'Salt Lake', 10);
INSERT INTO Branch VALUES (103, 'Park Street', 10);
INSERT INTO Branch VALUES (201, 'Sealdah', 20);
INSERT INTO Branch VALUES (202, 'Behala', 20);
INSERT INTO Branch VALUES (203, 'Teghoria', 20);
INSERT INTO Branch VALUES (301, 'Gariahat', 30);
INSERT INTO Branch VALUES (302, 'Howrah', 30);
INSERT INTO Branch VALUES (303, 'Park Street', 30);
INSERT INTO Branch VALUES (401, 'Rajabazaar', 40);
INSERT INTO Branch VALUES (402, 'College Street', 40);
INSERT INTO Branch VALUES (403, 'Newtown', 40);
INSERT INTO Branch VALUES (501, 'Maidan', 50);
INSERT INTO Branch VALUES (502, 'Ruby', 50);
INSERT INTO Branch VALUES (503, 'Gariahat', 50);
```

BRANCH_ID	BRANCH_NAME	BANK_ID
101	Sealdah	10
102	Salt Lake	10
103	Park Street	10
201	Sealdah	20
202	Behala	20
203	Teghoria	20
301	Gariahat	30
302	Howrah	30
303	Park Street	30
401	Rajabazaar	40
402	College Street	40
403	Newtown	40
501	Maidan	50
502	Ruby	50
503	Gariahat	50

```
--CUSTOMER Table
CREATE TABLE Customer (
   Customer_ID NUMBER (4) PRIMARY KEY,
   Customer_Name VARCHAR(100) NOT NULL);

INSERT INTO Customer VALUES (1001, 'Ramesh');
INSERT INTO Customer VALUES (1002, 'Suresh');
```

```
INSERT INTO Customer VALUES (1003, 'Pakeya');
INSERT INTO Customer VALUES (2001, 'Charan');
INSERT INTO Customer VALUES (2002, 'Ram');
INSERT INTO Customer VALUES (2003, 'Pankaj');
INSERT INTO Customer VALUES (3001, 'Soham');
INSERT INTO Customer VALUES (3002, 'Debodit');
INSERT INTO Customer VALUES (3003, 'Anuska');
INSERT INTO Customer VALUES (4001, 'Prama');
INSERT INTO Customer VALUES (4002, 'Rishika');
INSERT INTO Customer VALUES (4003, 'Sayani');
INSERT INTO Customer VALUES (5001, 'Tanish');
INSERT INTO Customer VALUES (5002, 'Aman');
INSERT INTO Customer VALUES (5003, 'Neel');
```

CUSTOMER_ID	CUSTOMER_NAME
1001	Ramesh
1002	Suresh
1003	Pakeya
2001	Charan
2002	Ram
2003	Pankaj
3001	Soham
3002	Debodit
3003	Anuska
4001	Prama
4002	Rishika
4003	Sayani
5001	Tanish
5002	Aman
5003	Neel

```
--ACCOUNT Table
CREATE TABLE Account (
    Account_ID NUMBER (11) PRIMARY KEY,
    Balance DECIMAL(10, 2),
    Customer_ID NUMBER (4),
    Branch_ID NUMBER (3),
    FOREIGN KEY (Branch_ID) REFERENCES Branch(Branch_ID)
on DELETE CASCADE,
    FOREIGN KEY (Customer_ID) REFERENCES
Customer(Customer_ID) on DELETE CASCADE);

Insert into ACCOUNT values (55690129329, 35486.12, 1001, 101);
Insert into ACCOUNT values (89545237546, 1234.56, 1003, 101);
Insert into ACCOUNT values (54321987654, 98765.43, 1002, 102);
Insert into ACCOUNT values (39800005552, 45678.90, 1003, 103);
Insert into ACCOUNT values (59498307311, 2345.57, 2001, 202);
Insert into ACCOUNT values (65592459895, 67890.12, 2002, 202);
Insert into ACCOUNT values (71875151611, 54321.98, 2001, 201);
Insert into ACCOUNT values (77975635172, 8765.43, 2003, 203);
Insert into ACCOUNT values (84531363466, 34567.89, 3001, 302);
Insert into ACCOUNT values (46372949259, 78901.23, 3002, 301);
Insert into ACCOUNT values (82273579151, 65432.10, 4001, 401);
Insert into ACCOUNT values (57679662463, 4321.98, 4002, 402);
Insert into ACCOUNT values (90680460457, 21098.76, 4003, 403);
Insert into ACCOUNT values (18567978674, 3456.78, 3003, 403);
Insert into ACCOUNT values (94720840720, 67789.01, 5001, 501);
Insert into ACCOUNT values (10346725987, 54321.10, 5002, 503);
Insert into ACCOUNT values (35487204793, 9876.54, 5003, 503);
Insert into ACCOUNT values (28645816408, 123456.67, 5001, 503);
```

ACCOUNT_ID	BALANCE	CUSTOMER_ID	BRANCH_ID
55690129329	35486.12	1001	101
89545237546	1234.56	1003	101
54321987654	98765.43	1002	102
39800005552	45678.9	1003	103
59498307311	2345.57	2001	202
65592459895	67890.12	2002	202
71875151611	54321.98	2001	201
77975635172	8765.43	2003	203
84531363466	34567.89	3001	302
46372949259	78901.23	3002	301
82273579151	65432.1	4001	401
57679662463	4321.98	4002	402
90680460457	21098.76	4003	403
18567978674	3456.78	3003	403
94720840720	67789.01	5001	501
10346725987	54321.1	5002	503
35487204793	9876.54	5003	503
28645816408	123456.67	5001	503

```
--LOAN Table
CREATE TABLE Loan (
   Loan_ID NUMBER (5) PRIMARY KEY,
   Loan_Amount DECIMAL(10, 2),
   Customer_ID NUMBER (4),
   Branch_ID NUMBER (3),
   FOREIGN KEY (Branch_ID) REFERENCES
Branch(Branch_ID) on DELETE CASCADE,
   FOREIGN KEY (Customer_ID) REFERENCES
Customer(Customer_ID) ON DELETE CASCADE);

Insert into LOAN values (10001, 35486.12, 1001, 101);
Insert into LOAN values (10002, 1234.56, 1002, 101);
Insert into LOAN values (10003, 98765.43, 1002, 102);
Insert into LOAN values (20001, 2345.57, 2002, 202);
Insert into LOAN values (20002, 67890.12, 2002, 202);
Insert into LOAN values (20003, 54321.98, 2001, 201);
Insert into LOAN values (30001, 8765.43, 2003, 203);
Insert into LOAN values (30002, 34567.89, 3002, 302);
Insert into LOAN values (30003, 78901.23, 3001, 301);
Insert into LOAN values (40001, 65432.10, 4001, 401);
Insert into LOAN values (40002, 21098.76, 4003, 403);
Insert into LOAN values (40003, 3456.78, 4003, 403);
Insert into LOAN values (50001, 67789.01, 5001, 501);
Insert into LOAN values (50002, 54321.10, 5003, 503);
Insert into LOAN values (50003, 9876.54, 5002, 503);
```

LOAN_ID	LOAN_AMOUNT	CUSTOMER_ID	BRANCH_ID
10001	35486.12	1001	101
10002	1234.56	1002	101
10003	98765.43	1002	102
20001	2345.57	2002	202
20002	67890.12	2002	202
20003	54321.98	2001	201
30001	8765.43	2003	203
30002	34567.89	3002	302
30003	78901.23	3001	301
40001	65432.1	4001	401
40002	21098.76	4003	403
40003	3456.78	4003	403
50001	67789.01	5001	501
50002	54321.1	5003	503
50003	9876.54	5002	503

2. Every customer must have at least one account but is restricted to at most two loans at a time.

```
ALTER table CUSTOMER ADD Account_ID number (11) NOT NULL
ADD CONSTRAINT FK FOREIGN KEY (ACCOUNT_ID) REFERENCES ACCOUNT
(ACCOUNT_ID);

CREATE TRIGGER MaxLoansPerCustomer
BEFORE INSERT ON Loan
FOR EACH ROW
BEGIN
   DECLARE loan_count INT;
   SELECT COUNT(*) INTO loan_count
   FROM Loan
   WHERE CustomerID = NEW.CustomerID;

   IF loan_count >= 2 THEN
      SIGNAL SQLSTATE '45000'
      SET MESSAGE_TEXT = 'A customer can have a maximum of 2 loans.';
   END IF;
END;
```

Triggers

Trigger	Body

MAXLOANSPERCUSTOMER
ENABLED
INSERT
	BEGIN
	DECLARE loan_count INT;
	SELECT COUNT(*) INTO loan_count
	FROM Loan
	WHERE CustomerID = NEW.CustomerID;
	
	IF loan_count >= 2 THEN
	SIGNAL SQLSTATE '45000'
	SET MESSAGE_TEXT = 'A customer can have a maximum of 2 loans.';
	END IF;
	END;

3. Give all the account details of a person who has accounts in SBI.

SELECT * FROM Account
WHERE Branch_ID IN (SELECT Branch_ID FROM Branch WHERE Bank_ID = (SELECT Bank_ID FROM Bank WHERE Bank_Name = 'SBI'));

ACCOUNT_ID	BALANCE	CUSTOMER_ID	BRANCH_ID
55690129329	35486.12	1001	101
89545237546	1234.56	1003	101
54321987654	98765.43	1002	102
39800005552	45678.9	1003	103

4. Find the account holder name who has more than 2 accounts.

SELECT Customer_Name FROM Customer
WHERE Customer_ID IN (SELECT Customer_ID FROM Account GROUP BY Customer_ID HAVING COUNT(Account_ID) > 2);

```
no data found
```

5. Rename the accounts table as account details.

ALTER TABLE Account RENAME TO Account_Details;

```
Table altered.
```

6. Find the loan amount and loan taken from which bank for each account holder.

```
SELECT C.Customer_Name, L.Loan_Amount, B.Bank_Name FROM Loan L
JOIN Customer C ON L.Customer_ID = C.Customer_ID
JOIN Branch Br ON L.Branch_ID = Br.Branch_ID
JOIN Bank B ON Br.Bank_ID = B.Bank_ID;
```

CUSTOMER_NAME	LOAN_AMOUNT	BANK_NAME
Ramesh	35486.12	SBI
Suresh	1234.56	SBI
Suresh	98765.43	SBI
Charan	54321.98	PNB
Ram	2345.57	PNB
Ram	67890.12	PNB
Pankaj	8765.43	PNB
Soham	78901.23	HDFC
Debodit	34567.89	HDFC
Prama	65432.1	AXIS
Sayani	21098.76	AXIS
Sayani	3456.78	AXIS
Tanish	67789.01	ICICI
Aman	9876.54	ICICI
Neel	54321.1	ICICI

7. Find the account no. and account holder name who has not taken any loan.

```
SELECT A.Account_ID, C.Customer_Name FROM Account_Details A
JOIN Customer C ON A.Customer_ID = C.Customer_ID
WHERE A.Customer_ID NOT IN (SELECT Customer_ID FROM Loan);
```

ACCOUNT_ID	CUSTOMER_NAME
18567978674	Anuska
57679662463	Rishika
89545237546	Pakeya
39800005552	Pakeya

8. Delete the account of all the persons who had accounts in PNB, Sealdah branch.

```
DELETE FROM Account_Details WHERE Branch_ID IN (SELECT Branch_ID FROM Branch
WHERE Bank_ID = (SELECT Bank_ID FROM Bank WHERE Bank_Name = 'PNB') AND
Branch_Name = 'Sealdah');
```

```
1 row(s) deleted.
```

9. Update the branch to SBI, Salt Lake branch for all the persons who had a SBI account in Sealdah branch.

```
UPDATE Account_Details
SET Branch_ID = (SELECT Branch_ID FROM Branch WHERE Branch_Name = 'Salt Lake' AND
Bank_ID = (SELECT Bank_ID FROM Bank WHERE Bank_Name = 'SBI'))
WHERE Branch_ID = (SELECT Branch_ID FROM Branch WHERE Branch_Name = 'Sealdah'
AND Bank_ID = (SELECT Bank_ID FROM Bank WHERE Bank_Name = 'SBI'));
```

```
2 row(s) updated.
```

10. Find the maximum account balance of a person with account no 54321987654 among all of his accounts.

```
SELECT MAX(Balance) AS Max_Balance
FROM Account_Details WHERE
Customer_ID = (SELECT Customer_ID FROM Account_Details WHERE Account_ID =
54321987654);
```

MAX_BALANCE
98765.43