RETAIL BANKING

 # Create Account Table

```
create table Account (
        accountNumber varchar(20) PRIMARY KEY,
        accountBalance FLOAT(100),
        branchID varchar(30)
 )
```

 # Create Customer table

```
create table Customer (
        customerID varchar(20) PRIMARY KEY,
        customerName varchar(30),
        customerEmail varchar(30),
        customerContact varchar(30),
        customerAddress varchar(100),
        accountNumber varchar(20),
        foreign key (accountNumber) references Account
 )
```

 # Create Branch Table

```
create table Branch (
        branchID varchar(20) PRIMARY KEY,
        branchName FLOAT(30),
        branchAddress varchar(100),
        branchContact varchar(20),
        foreign key (branchID) references Account

 )
```

 # Create table for LOAN

```
create table LOAN (
        loanID varchar(20),
        customerID varchar(20),
        loanAmount FLOAT(100),
        numberOfInstallments INTEGER,
        installmentsLeft INTEGER,
        foreign key (customerID) references Customer
)
```

 # INSERTING THE VALUES INTO ACCOUNT TABLE

```
insert into Account
values ("ICI19087", 199980.90, "JK110")

insert into Account
values ("ICI18067", 13499.90, "KJ110")

insert into Account
values ("ICI80988", 18790.90, "JK110")
```

 # INSERTING THE VALUES INTO CUSTOMER TABLE

INSERT into Customer
values ("I0191019", "Neeraj Pandey", "neeraj.dummy@gmail.com", "987878660", "BS-4 Gyan Apartments, New Delhi", "ICI19087")

INSERT into Customer
values ("I0189019", "Tanuj Sood", "tanuj.dummy@gmail.com", "987878630", "LS-4 Gyan Apartments, New Delhi", "ICI18067")

INSERT into Customer
values ("I0989019", "Atishay Khanna", "khanna.dummy@gmail.com", "987828610", "KK-4 Gyan Apartments, New Delhi", "ICI80988")

INSERTING THE VALUES INTO BRANCH TABLE

INSERT INTO Branch
values ("JK110", "IC Bank, Janak Puri", "BC Block, Janak Puri, New Delhi", "+9111-98989898")

INSERT INTO Branch
values ("KJ110", "IC Bank, Kalkaji", "BB Block, Kalkaji, New Delhi", "+9111-78787878")

INSERTING THE VALUES INTO LOAN TABLE

insert into LOAN
values ("ICLOAN1769", "I0191019", 18000, 36, 13)

insert into LOAN
values ("ICLOAN9779", "I0189019", 12000, 20, 18)

insert into LOAN
values ("ICLOAN8729", "I0989019", 30000, 48, 25)

5 Meaningful Questions and Queries and Relational Algebra

1) Identify the customer names, account number whose bank balance is more than 100980 dollars.

Relational:

π customerName, accountNumber(σ accountBalance > 100980 (Customer $\triangleleft \triangleright$ Account))

SQL:
SELECT Customer.customerName, Customer.accountNumber
FROM Customer
INNER JOIN Account ON Customer.accountNumber=Account.accountNumber
where Account.accountBalance > 100980;

2) Identify customer names who has taken a loan for more than 15000 dollars

Relational: **π customerName (σ loanAmount > 15000 (Customer ◁▷ LOAN))**

SQL:
**SELECT Customer.customerName
FROM Customer
INNER JOIN LOAN ON Customer.customerID=LOAN.customerID
where LOAN.loanAmount > 15000;**

3) Identify the customer name, customer account number whose contact number is "987878630".

Relational: **π passengerName (σ customerContact= "987878630" (Customer))**

SQL:
**SELECT customerName
FROM Customer
WHERE Customer.customerContact = "987878630"**

3) Identify customer names, account number who are left with more than 20 loan instalments.

Relational: **π customerName, accountNumber (σ installmentsLeft > 20 (Customer ◁▷ LOAN))**

SQL:
**SELECT Customer.customerName, Customer.accountNumber
FROM Customer
INNER JOIN LOAN ON Customer.customerID=LOAN.customerID
where LOAN. installmentsLeft > 20;**

5) Identify the customer name, customer account number whose branch ID is "JK110".

Relational: **π customerName, customerAccount (σ branchID = "JK110" (Customer ◁▷ Account ◁▷ Branch))**

SQL:

**SELECT Customer.customerName, Customer.accountNumber
FROM Customer
INNER JOIN Account ON Customer.accountNumber=Account.accountNumber
INNER JOIN Branch ON Account.branchID = Branch.branchID
WHERE branchID = "JK110"**