PROGRAM 2: BANKING ENTERPRISE DATABASE

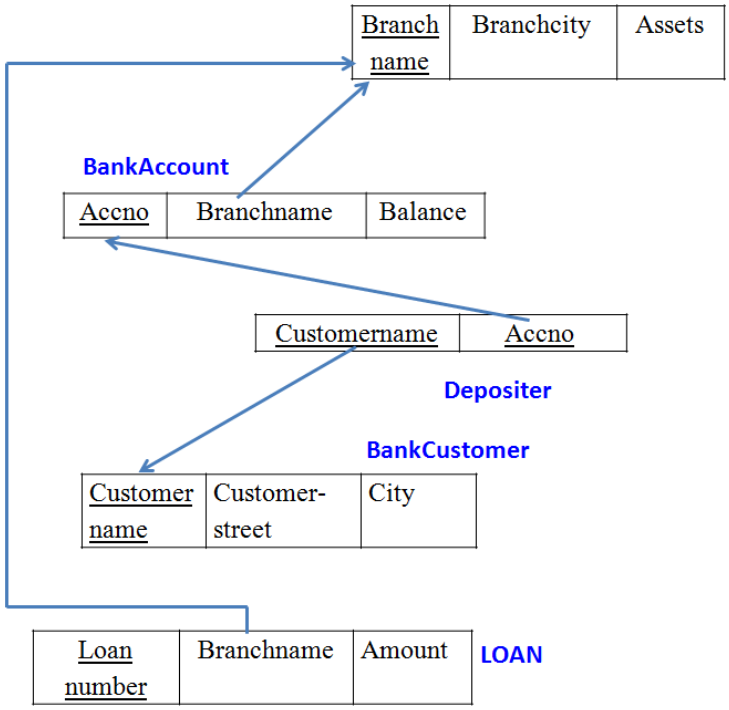
Consider the following database for a banking enterprise.

Branch (branch-name: String, branch-city: String, assets: real)
BankAccount(accno: int, branch-name: String, balance: real)
BankCustomer (customer-name: String, customer-street: String, customer-city: String)
Depositer(customer-name: String, accno: int)
Loan (loan-number: int, branch-name: String, amount: real)

i. Create the above tables by properly specifying the primary keys and the foreign keys.
ii. Enter at least five tuples for each relation.
iii. Find all the customers who have at least two accounts at the *Main* branch (ex. SBI_ResidencyRoad).
iv. Find all the customers who have an account at *all* the branches located in a specific city (Ex. Delhi).
v. Demonstrate how you delete all account tuples at every branch located in a specific city (Ex. Bombay).

INTRODUCTION: This database is developed for supporting banking facilities. Details of the branch along with the accounts and loans handled by them are recorded. Also details of the depositors of the corresponding branches are maintained.

Schema Diagram



Sample Table data

Branch

BRANCHNAME	BRANCHCITY	ASSESTS
SBI_Chamrajpet	Bangalore	50000
SBI_ResidencyRoad	Bangalore	10000
SBI_ShivajiRoad	Bombay	20000
SBI_ParlimentRoad	Delhi	10000
SBI_Jantarmantar	Delhi	20000

BankAccount

ACCNO	BRANCHNAME	BALANCE
1	SBI_Chamrajpet	2000
2	SBI_ResidencyRoad	5000
3	SBI_ShivajiRoad	6000
4	SBI_ParlimentRoad	9000
5	SBI_Jantarmantar	8000
6	SBI_ShivajiRoad	4000
8	SBI_ResidencyRoad	4000
9	SBI_ParlimentRoad	3000
10	SBI_ResidencyRoad	5000
11	SBI_Jantarmantar	2000

BankCustomer

CUSTOMERNAME	CUSTOMERSTREET	CUSTOMERCITY
Avinash	Bull_Temple_Road	Bangalore
Dinesh	Bannergatta_Road	Bangalore
Mohan	NationalCollege_Road	Bangalore
Nikil	Akbar_Road	Delhi
Ravi	Prithviraj_Road	Delhi

Depositer

CUSTOMERNAME	ACCNO
Avinash	1
Dinesh	2
Nikil	4
Ravi	5
Avinash	8
Nikil	9
Dinesh	10
Nikil	11

Loan

LOANNUMBER	BRANCHNAME	AMOUNT
1	SBI_Chamrajpet	1000
2	SBI_ResidencyRoad	2000
3	SBI_ShivajiRoad	3000
4	SBI_ParlimentRoad	4000
5	SBI_Jantarmantar	5000

QUERY 1: Create the above tables by properly specifying the primary keys and the foreign keys.

```
SQL> create table Branch(branchname varchar(30),branchcity varchar(30),assests r
eal, primary key(branchname));
SQL> desc Branch
 Name                                      Null?    Type
 ----------------------------------------- -------- -------------

 BRANCHNAME                                NOT NULL VARCHAR2(30)
 BRANCHCITY                                         VARCHAR2(30)
 ASSESTS                                            FLOAT(63)
```

```
SQL> create table BankAccount(accno integer,branchname varchar(30), balance real
,primary key (accno),foreign key (branchname) references Branch(branchname));
SQL> desc BankAccount
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------

 ACCNO                                     NOT NULL NUMBER(38)
 BRANCHNAME                                         VARCHAR2(30)
 BALANCE                                            FLOAT(63)

SQL> create table BankCustomer(customername varchar(30),customerstreet varchar(3
),customercity varchar(30),primary key(customername));

Table created.

SQL> desc BankCustomer
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------

 CUSTOMERNAME                              NOT NULL VARCHAR2(30)
 CUSTOMERSTREET                                     VARCHAR2(3)
 CUSTOMERCITY                                       VARCHAR2(30)
SQL> create table Depositer(customername varchar(30),accno integer,primary key(c
ustomername,accno),foreign key(customername) references BankCustomer(customernam
e), foreign key(accno) references BankAccount(accno));

Table created.

SQL> desc Depositer;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------

 CUSTOMERNAME                              NOT NULL VARCHAR2(30)
 ACCNO                                     NOT NULL NUMBER(38)

SQL> create table Loan (loannumber int,branchname varchar(30),amount real,primar
y key (loannumber), foreign key (branchname) references Branch(branchname));
```

QUERY 2: Enter at least five tuples for each relation

```
SQL> insert into Branch values('SBI_Chamrajpet','Bangalore',50000);

1 row created.

SQL> insert into Branch values('SBI_ResidencyRoad','Bangalore',10000);

1 row created.

SQL> insert into Branch values('SBI_ShivajiRoad','Bombay',20000);

1 row created.

SQL> insert into Branch values('SBI_ParlimentRoad','Delhi',10000);

1 row created.

SQL> insert into Branch values('SBI_Jantarmantar','Delhi',20000);

1 row created.
SQL> select * from Branch;
```

BRANCHNAME	BRANCHCITY	ASSESTS
SBI_Chamrajpet	Bangalore	50000
SBI_ResidencyRoad	Bangalore	10000
SBI_ShivajiRoad	Bombay	20000
SBI_ParlimentRoad	Delhi	10000
SBI_Jantarmantar	Delhi	20000

Insert records for Loan

```
SQL> insert into Loan values(2,'SBI_ResidencyRoad',2000);
SQL> insert into Loan values(1,'SBI_Chamrajpet',1000);
SQL> insert into Loan values(3,'SBI_ShivajiRoad',3000);
SQL> insert into Loan values(4,'SBI_ParlimentRoad',4000);
SQL> insert into Loan values(5,'SBI_Jantarmantar',5000);

SQL> select * from Loan;
```

LOANNUMBER	BRANCHNAME	AMOUNT
1	SBI_Chamrajpet	1000
2	SBI_ResidencyRoad	2000
3	SBI_ShivajiRoad	3000
4	SBI_ParlimentRoad	4000
5	SBI_Jantarmantar	5000

Similarly insert records for BankAccount, Depositer and BankCustomer

```
SQL> insert into BanKAccount values(11,'SBI_Jantarmantar',2000);

1 row created.

SQL> commit;

Commit complete.

SQL> select * from BankAccount;

     ACCNO BRANCHNAME                         BALANCE
---------- ------------------------------ ----------
         1 SBI_Chamrajpet                       2000
         2 SBI_ResidencyRoad                    5000
         3 SBI_ShivajiRoad                      6000
         4 SBI_ParlimentRoad                    9000
         5 SBI_Jantarmantar                     8000
         6 SBI_ShivajiRoad                      4000
         8 SBI_ResidencyRoad                    4000
         9 SBI_ParlimentRoad                    3000
        10 SBI_ResidencyRoad                    5000
        11 SBI_Jantarmantar                     2000
```

Branch

BRANCHNAME	BRANCHCITY	ASSESTS
SBI_Chamrajpet	Bangalore	50000
SBI_ResidencyRoad	Bangalore	10000
SBI_ShivajiRoad	Bombay	20000
SBI_ParlimentRoad	Delhi	10000
SBI_Jantarmantar	Delhi	20000

BankAccount

ACCNO	BRANCHNAME	BALANCE
1	SBI_Chamrajpet	2000
2	SBI_ResidencyRoad	5000
3	SBI_ShivajiRoad	6000
4	SBI_ParlimentRoad	9000
5	SBI_Jantarmantar	8000
6	SBI_ShivajiRoad	4000
8	SBI_ResidencyRoad	4000
9	SBI_ParlimentRoad	3000
10	SBI_ResidencyRoad	5000
11	SBI_Jantarmantar	2000

BankCustomer

CUSTOMERNAME	CUSTOMERSTREET	CUSTOMERCITY
Avinash	Bull_Temple_Road	Bangalore
Dinesh	Bannergatta_Road	Bangalore
Mohan	NationalCollege_Road	Bangalore
Nikil	Akbar_Road	Delhi
Ravi	Prithviraj_Road	Delhi

Depositer

CUSTOMERNAME	ACCNO
Avinash	1
Dinesh	2
Nikil	4
Ravi	5
Avinash	8
Nikil	9
Dinesh	10
Nikil	11

Loan

LOANNUMBER	BRANCHNAME	AMOUNT
1	SBI_Chamrajpet	1000
2	SBI_ResidencyRoad	2000
3	SBI_ShivajiRoad	3000
4	SBI_ParlimentRoad	4000
5	SBI_Jantarmantar	5000

SQL> commit;

Commit complete.

QUERY 3: Find Find all the customers who have at least two deposits at the same branch (Ex. 'SBI_ResidencyRoad').

```
select  C.customername
from BankCustomer C
where exists (
        select D.customername, count(D.customername)
        from depositer D, BankAccount BA
        where
            D.accno = BA.accno AND
            C.customername = D.customername AND
             BA.branchname = 'SBI_ResidencyRoad'
       group by D. customername
       having count(D.customername)>=2;
            );
```

QUERY 4: Find all the customers who have an account at *all* the branches located in a specific city (Ex. Delhi).

```
select BC.customername
from BankCustomer BC
where not exists  (
                    select bracnhname from Branch where branchcity='Delhi'
                    minus
                    (select BA.branchname from Depositer D, BankAccount BA
                      where D.accno=BA.accno and BC.customername=D.customername)
                   );
```

QUERY 5: Demonstrate how you delete all account tuples at every branch located in a specific city (Ex. Bomay).

```
delete  from BankAccount
where branchname IN (
                    select branchname
                    from Branch
                    where branchcity='BOMBAY'
                      );
```

ADDITIONAL QUERIES:

1. **LIST THE ENTIRE LOAN RELATION IN THE DESCENDING ORDER OF AMOUNT.**

SQL> SELECT * FROM LOAN ORDER BY AMOUNT DESC;

2. **FIND ALL CUSTOMERS HAVING A LAON, AN ACCOUNT OR BOTH AT THE BANK**

SQL> (SELECT CUSTOMER_NAME FROM DEPOSITOR) UNION (SELECT CUSTOMER_NAME FROM BORROWER);

3. **CREATE A VIEW WHICH GIVES EACH BRANCH THE SUM OF THE AMOUNT OF ALL THE LOANS AT THE BRANCH.**

SQL> CREATE VIEW BRANCH_TOTAL_LOAN (BRANCH_NAME, TOTAL_LOAN) AS SELECT BRANCH_NAME, SUM(AMOUNT) FROM LOAN GROUP BY BRANCH_NAME;

4. **THE ANNUAL INTEREST PAYMENTS ARE MADE AND ALL BRANCHES ARE TO BE INCREASED BY 5%.**

SQL> UPDATE ACCOUNT SET BALANCE=BALANCE *1.05;