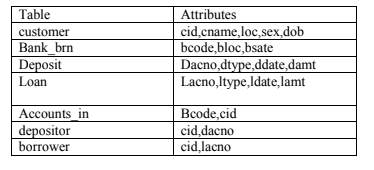
**LAB CYCLE-4**

QUESTION SET 1

Consider the database for a banking enterprise. Write the queries for the below

questions.

1. **Create the following tables**



**QUERY**

CREATE TABLE CUSTOMERS

(

CID INT PRIMARY KEY,

CNAME VARCHAR(25)NOT NULL,

LOC VARCHAR(25),

SEX VARCHAR(25),

DOB DATE

);

CREATE TABLE BBANK\_BRN

(

BCODE INT PRIMARY KEY,

BLOC VARCHAR(25) NOT NULL,

BSTATE VARCHAR(25) NOT NULL

);

CREATE TABLE DEPOSITS

(

DACNO INT PRIMARY KEY,

DDATE DATE NOT NULL,

DAMT INT NOT NULL,

DTYPE VARCHAR(25) NOT NULL

);

CREATE TABLE LOANS

(

LACNO INT PRIMARY KEY,

LDATE DATE NOT NULL,

LAMT INT NOT NULL,

LTYPE VARCHAR(25) NOT NULL

);

CREATE TABLE AACCOUNTS\_IN

(

CID INT NOT NULL,

BCODE INT NOT NULL,

FOREIGN KEY(BCODE) REFERENCES BBANK\_BRN(BCODE),

FOREIGN KEY (CID) REFERENCES CUSTOMERS(CID)

);

CREATE TABLE DEPOSITORS

(

CID INT NOT NULL,

DACNO INT NOT NULL,

FOREIGN KEY(DACNO) REFERENCES DEPOSITS(DACNO),

FOREIGN KEY (CID) REFERENCES CUSTOMERS(CID)

);

CREATE TABLE BORROWERS

(

CID INT NOT NULL,

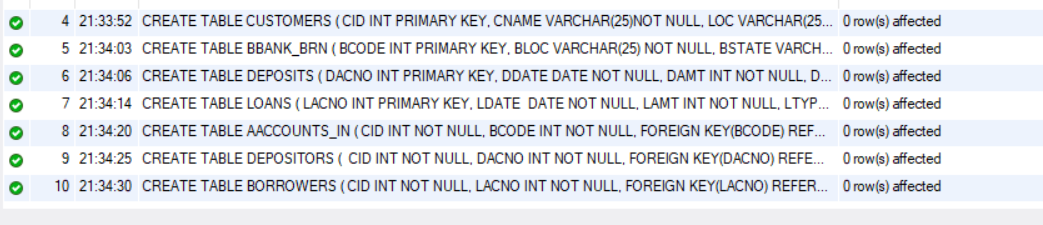
LACNO INT NOT NULL,

FOREIGN KEY(LACNO) REFERENCES LOANS(LACNO),

FOREIGN KEY(CID) REFERENCES CUSTOMERS(CID)

);

**OUTPUT**

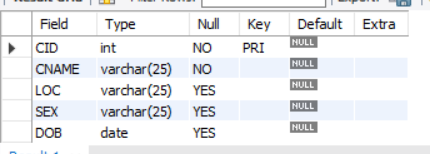


**(ii)Include necessary constraints.**

**QUERY**

DESCRIBE CUSTOMERS;

**OUTPUT**

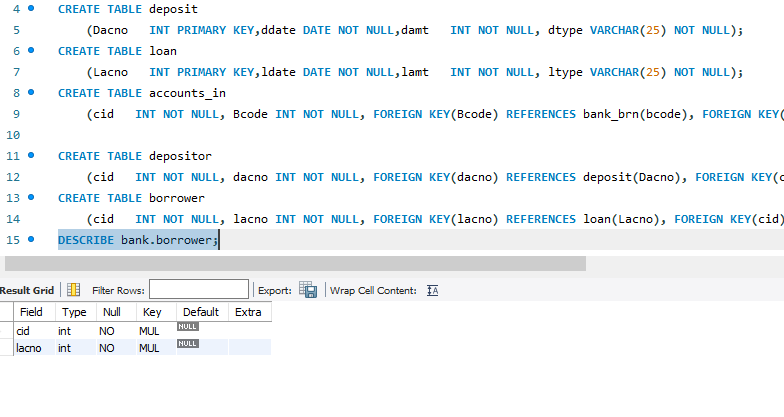


**(iii)Tables are created under the database ‘bank’**

**QUERY**

DESCRIBE BANK.BORROWERS;

**OUTPUT**

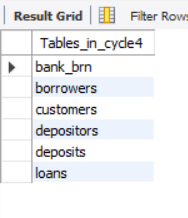


**(iv) Display all the tables in bank database**

**QUERY**

SHOW TABLES;

**OUTPUT**

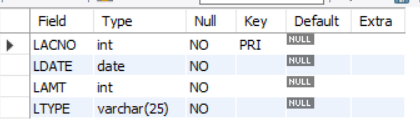


**(v) Describe the structure of all tables**

**QUERY**

DESCRIBE LOANS;

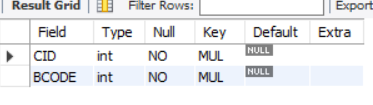
**OUTPUT**



**QUERY**

DESCRIBE AACCOUNTS\_IN;

**OUTPUT**



**(vi) Delete tables.**

**QUERY**

DROP TABLE CUSTOMERS,BORROWERS,AACCOUNTS\_IN,DEPOSITORS;

**OUTPUT**



QUESTION SET 2

Consider the following database for a banking enterprise.

BRANCH (bid:int, branch-name: String, branch-city: String, assets: int)

**QUERY**

CREATE TABLE BRANCH

(

BID INT,

BRANCH\_NAME VARCHAR(25),

BRANCH\_CITY VARCHAR(25),

ASSETS INT

);

INSERT INTO BRANCH(BID,BRANCH\_NAME,BRANCH\_CITY,ASSETS) VALUES

(1,'MAIN','C1',10000),

(2,'B2','C2',20000),

(3,'B3','C3',30000),

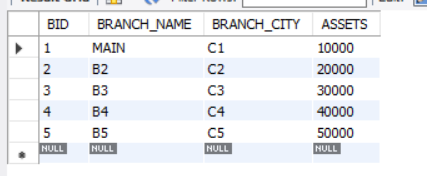
(4,'B4','C4',40000),

(5,'B5','C5',50000);

SELECT \* FROM BRANCH;

**OUTPUT**





ACCOUNTS (accno: int, bid:int, balance: int)

**QUERY**

CREATE TABLE ACCOUNTS

(

ACCNO INT,

BID INT,

BALANCE INT

);

INSERT INTO ACCOUNTS(ACCNO,BID,BALANCE) VALUES

(101,'1',1000),

(102,'1',2000),

(103,'1',3000),

(104,'2',4000),

(105,'2',5000),

(106,'3',4000),

(107,'3',3000),

(108,'4',2000),

(109,'4',3000),

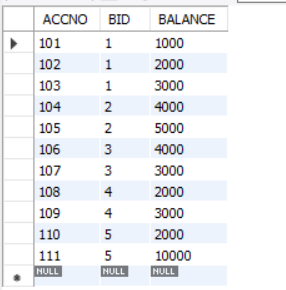
(110,'5',2000),

(111,'5',10000);

SELECT \* FROM ACCOUNTS;

**OUTPUT**





DEPOSITOR (cid:int, accno: int)

**QUERY**

CREATE TABLE DEPOSITOR

(

CID INT,

ACCNO INT

);

INSERT INTO DEPOSITOR(CID,ACCNO) VALUES

(11,101),

(22,102),

(11,103),

(33,104),

(11,105),

(44,106),

(11,107),

(55,108),

(11,109),

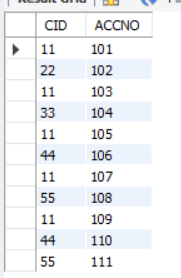
(44,110),

(55,111);

SELECT \* FROM DEPOSITOR;

**OUTPUT**





CUSTOMER(cid:int, customer-name:String,customer-street:String,customer-city: String)

**QUERY**

CREATE TABLE CUSTOMER

(

CID INT,

CUSTOMER\_NAME VARCHAR(25),

CUSTOMER\_STREET VARCHAR(25),

CUSTOMER\_CITY VARCHAR(25)

);

INSERT INTO CUSTOMER(CID,CUSTOMER\_NAME,CUSTOMER\_STREET,CUSTOMER\_CITY) VALUES

(11,'CUST1','S1','CITY1'),

(22,'CUST2','S2','CITY2'),

(33,'CUST3','S3','CITY3'),

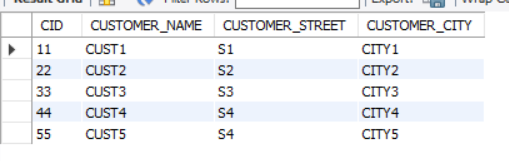
(44,'CUST4','S4','CITY4'),

(55,'CUST5','S4','CITY5');

SELECT \* FROM CUSTOMER;

**OUTPUT**





Set primary key and foreign keys and insert valid records based on questions.

Write SQL queries to

**QUERY**

ALTER TABLE BRANCH ADD PRIMARY KEY (BID);

ALTER TABLE ACCOUNTS ADD PRIMARY KEY (ACCNO);

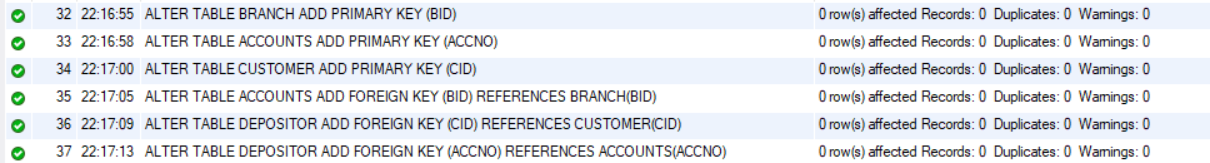
ALTER TABLE CUSTOMER ADD PRIMARY KEY (CID);

ALTER TABLE ACCOUNTS ADD FOREIGN KEY (BID) REFERENCES BRANCH(BID);

ALTER TABLE DEPOSITOR ADD FOREIGN KEY (CID) REFERENCES CUSTOMER(CID);

ALTER TABLE DEPOSITOR ADD FOREIGN KEY (ACCNO) REFERENCES ACCOUNTS(ACCNO);

**OUTPUT**



1. **Find all the customers who have at least two accounts at the Main branch.**

**QUERY**

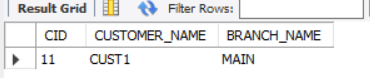
SELECT CUSTOMER.CID,CUSTOMER\_NAME,BRANCH\_NAME FROM

CUSTOMER.CID=DEPOSITOR.CID AND DEPOSITOR.ACCNO=ACCOUNTS.ACCNO

AND ACCOUNTS.BID=1 GROUP BY ACCOUNTS.BID HAVING

COUNT(ACCOUNTS.BID)>=2;

**OUTPUT**



**2. Find all the customers who have an account at all the branches located in a specific city.**

**QUERY**

SELECT CUSTOMER.CID,CUSTOMER\_NAME,ACCOUNTS.ACCNO,

BRANCH\_CITY FROM ACCOUNTS,BRANCH,DEPOSITOR,CUSTOMER WHERE

BRANCH.BID=ACCOUNTS.BID AND ACCOUNTS.ACCNO=DEPOSITOR.ACCNO

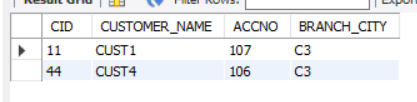
AND DEPOSITOR.CID=CUSTOMER.CID AND BRANCH.BRANCH\_CITY='C3'

GROUP BY DEPOSITOR.CID HAVING COUNT(DISTINCT

BRANCH.BID)=(SELECT COUNT(BID) FROM BRANCH WHERE

BRANCH\_CITY='C3');

**OUTPUT**

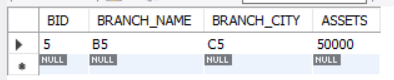


**3. Find the branch with greatest asset.**

**QUERY**

SELECT \* FROM BRANCH WHERE ASSETS=(SELECT MAX(ASSETS) FROM BRANCH);

**OUTPUT**



**4. Find the customer with highest balance.**

**QUERY**

SELECT CUSTOMER.CID,CUSTOMER\_NAME,ACCOUNTS.ACCNO,

BALANCE FROM CUSTOMER,DEPOSITOR,ACCOUNTS WHERE

CUSTOMER.CID=DEPOSITOR.CID AND

DEPOSITOR.ACCNO=ACCOUNTS.ACCNO

AND BALANCE=(SELECT MAX(BALANCE) FROM ACCOUNTS);

**OUTPUT**

