

DBMS – Assignment A2

Name:
PRN:
Roll no.:
Class:

Setting up the database

```
mysql> CREATE DATABASE Database_A2;  
Query OK, 1 row affected (0.01 sec)
```

```
mysql> USE Database_A2;  
Database changed
```

```
mysql> CREATE TABLE Account(accountNum INT, branchName  
VARCHAR(50), balance INT, PRIMARY KEY (accountNum));  
Query OK, 0 rows affected (0.04 sec)
```

```
mysql> CREATE TABLE Branch(branchName VARCHAR(50), branchCity  
VARCHAR(50), assets INT, PRIMARY KEY (branchName));  
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> CREATE TABLE Customer (customerName VARCHAR(50),  
customerStreet VARCHAR(50), customerCity VARCHAR(50), PRIMARY KEY  
(customerName));  
Query OK, 0 rows affected (0.03 sec)
```

```
mysql> CREATE TABLE Depositor (customerName VARCHAR(50),  
accountNum INT);  
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> CREATE TABLE Loan (loanNum INT, branchName VARCHAR(50),  
amount INT, PRIMARY KEY (loanNum));  
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> CREATE TABLE Borrower (customerName VARCHAR(50), loanNum  
INT);  
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> ALTER TABLE Account ADD FOREIGN KEY (branchName) REFERENCES  
Branch(branchName);  
Query OK, 0 rows affected (0.08 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> ALTER TABLE Depositor ADD FOREIGN KEY (customerName)
REFERENCES Customer (customerName);
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> ALTER TABLE Depositor ADD FOREIGN KEY (accountNum)
REFERENCES Account (accountNum);
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> ALTER TABLE Loan ADD FOREIGN KEY (branchName) REFERENCES
Branch (branchName);
Query OK, 0 rows affected (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> ALTER TABLE Borrower ADD FOREIGN KEY (customerName)
REFERENCES Customer (customerName);
Query OK, 0 rows affected (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> ALTER TABLE Borrower ADD FOREIGN KEY (loanNum) REFERENCES
Loan (loanNum);
Query OK, 0 rows affected (0.06 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

Inserting data

```
mysql> INSERT INTO Branch (branchName, branchCity, assets) VALUES
-> ("Dhole Patil", "Kharadi", 50000),
-> ("Nagarwala", "Akurdi", 20000),
-> ("Peachtree", "Wakad", 35000),
-> ("Bishops", "Nigdi", 10000),
-> ("Amanora", "Hadapsar", 60000);
Query OK, 5 rows affected (0.01 sec)
Records: 5 Duplicates: 0 Warnings: 0
```

```
mysql> INSERT INTO Customer (customerName, customerStreet,
customerCity) VALUES
-> ("Kalas", "Airport Road", "Pune"),
-> ("Mehul", "Shahdha", "Nandurbar"),
-> ("Tanmay", "Porwal Road", "Pune"),
-> ("Kshitij", "Hadapasar", "Pune"),
-> ("Aditya", "Mira RD", "Mumbai"),
-> ("Himanshu", "Smart City", "Nandurbar");
Query OK, 6 rows affected (0.01 sec)
Records: 6 Duplicates: 0 Warnings: 0
```

```
mysql> INSERT INTO Account (accountNum, branchName, balance)
VALUES
-> (2501, "Dhole Patil", 5000),
-> (2511, "Nagarwala", 1500),
-> (2521, "Peachtree", 2000),
-> (2512, "Bishops", 5000),
-> (2531, "Amanora", 1000);
Query OK, 5 rows affected (0.01 sec)
Records: 5 Duplicates: 0 Warnings: 0
```

```
mysql> INSERT INTO Loan (loanNum, branchName, amount) VALUES
-> (155, "Dhole Patil", 500),
-> (156, "Nagarwala", 250),
-> (157, "Peachtree", 600),
-> (158, "Bishops", 1400),
-> (159, "Amanora", 25000);
Query OK, 5 rows affected (0.01 sec)
Records: 5 Duplicates: 0 Warnings: 0
```

```
mysql> INSERT INTO Borrower VALUES
-> ("Kalas", 156),
-> ("Mehul", 158),
-> ("Tanmay", 155),
-> ("Kshitij", 157),
-> ("Aditya", 159),
-> ("Himanshu", 158);
Query OK, 6 rows affected (0.00 sec)
Records: 6 Duplicates: 0 Warnings: 0
```

```
mysql> INSERT INTO Depositor VALUES
-> ("Kalas", 2511),
-> ("Mehul", 2512),
-> ("Tanmay", 2501),
-> ("Kshitij", 2521),
-> ("Aditya", 2531),
-> ("Himanshu", 2512);
Query OK, 6 rows affected (0.01 sec)
Records: 6 Duplicates: 0 Warnings: 0
```

Questions

1. Find the names of all branches in loan relation.

```
mysql> SELECT branchName FROM Loan;
+-----+
| branchName |
+-----+
| Amanora    |
| Bishops    |
```

| |
|-------------|
| Dhole Patil |
| Nagarwala |
| Peachtree |

2. Find all loan numbers for loans made at Akurdi Branch with loan amount > 12000.

```
mysql> SELECT loanNum FROM Loan WHERE amount>12000;
```

| loanNum |
|---------|
| 159 |

3. Find all customers who have a loan from bank. Find their names,loan_no and loan amount.

```
mysql> SELECT Borrower.loanNum FROM Borrower INNER JOIN Loan ON
Borrower.loanNum = Loan.loanNum;
```

| loanNum |
|---------|
| 159 |
| 158 |
| 158 |
| 155 |
| 156 |
| 157 |

4. List all customers in alphabetical order who have loan from Akurdi branch.

```
mysql> SELECT customerName FROM Borrower INNER JOIN Loan ON
Borrower.loanNum = Loan.loanNum WHERE branchName = "Akurdi" ORDER
BY customerName;
Empty set (0.00 sec)
```

5. Find all customers who have an account or loan or both at bank.

```
mysql> SELECT customerName FROM Depositor UNION SELECT
customerName FROM Borrower;
```

| customerName |
|--------------|
| Aditya |
| Himanshu |
| Kalas |
| Kshitij |
| Mehul |
| Tanmay |

6. Find all customers who have both account and loan at bank.

```
mysql> SELECT customerName FROM Depositor INTERSECT SELECT  
customerName FROM Borrower;
```

| customerName |
|--------------|
| Aditya |
| Himanshu |
| Kalas |
| Kshitij |
| Mehul |
| Tanmay |

7. Find all customers who have account but no loan at the bank.

```
mysql> SELECT customerName FROM Depositor WHERE customerName NOT  
IN (SELECT customerName FROM Borrower);  
Empty set (0.00 sec)
```

8. Find the average account balance at each branch

```
mysql> SELECT AVG(amount) FROM Loan;
```

| AVG(amount) |
|-------------|
| 5550.0000 |

9. Find no. of depositors at each branch.

```
mysql> SELECT branchName, COUNT(*) AS noOfDepositors FROM Account  
JOIN Depositor ON Account.accountNum = Depositor.accountNum GROUP  
BY branchName;
```

| branchName | noOfDepositors |
|-------------|----------------|
| Amanora | 1 |
| Bishops | 2 |
| Dhole Patil | 1 |
| Nagarwala | 1 |
| Peachtree | 1 |

10. Find name of Customer and city where customer name starts with Letter K.

```
mysql> SELECT DISTINCT branchName, branchCity FROM Branch;
```

| branchName | branchCity |
|------------|------------|
| Amanora | Hadapsar |
| Bishops | Nigdi |

| | |
|-------------|---------|
| Dhole Patil | Kharadi |
| Nagarwala | Akurdi |
| Peachtree | Wakad |

11. Display distinct cities of branch.

```
mysql> SELECT DISTINCT branchName, branchCity FROM Branch;
```

| branchName | branchCity |
|-------------|------------|
| Amanora | Hadapsar |
| Bishops | Nigdi |
| Dhole Patil | Kharadi |
| Nagarwala | Akurdi |
| Peachtree | Wakad |

12. Find the branches where average account balance > 1200

```
mysql> SELECT branchName FROM Account GROUP BY branchName HAVING
AVG(balance) > 1200;
```

| branchName |
|-------------|
| Bishops |
| Dhole Patil |
| Nagarwala |
| Peachtree |

13. Find number of tuples in customer relation.

```
mysql> SELECT COUNT(*) FROM Customer;
```

| COUNT(*) |
|----------|
| 6 |

14. Calculate total loan amount given by bank.

```
mysql> SELECT SUM(amount) AS amount FROM Loan;
```

| amount |
|--------|
| 27750 |

15. Delete all loans with loan amount between 1300 and 1500.

```
mysql> DELETE FROM Borrower WHERE loanNum IN (SELECT loanNum FROM
Loan WHERE amount > 1300 AND amount < 1500);
```

Query OK, 2 rows affected (0.00 sec)

```
mysql> DELETE FROM Loan WHERE amount > 1300 AND amount < 1500;
Query OK, 1 row affected (0.01 sec)
```

```
mysql> SELECT * FROM Borrower;
```

| customerName | loanNum |
|--------------|---------|
| Kalas | 156 |
| Tanmay | 155 |
| Kshitij | 157 |
| Aditya | 159 |

4 rows in set (0.00 sec)

```
mysql> SELECT * FROM Loan;
```

| loanNum | branchName | amount |
|---------|-------------|--------|
| 155 | Dhole Patil | 500 |
| 156 | Nagarwala | 250 |
| 157 | Peachtree | 600 |
| 159 | Amanora | 25000 |

4 rows in set (0.00 sec)

16. Delete all tuples at every branch located in Nigdi.

```
mysql> DELETE FROM Borrower WHERE loanNum IN (SELECT loanNum FROM
Loan WHERE branchName IN (SELECT branchName FROM Branch WHERE
branchCity = "Nigdi"));
```

Query OK, 0 rows affected (0.00 sec)

```
mysql> DELETE FROM Loan WHERE branchName = (SELECT branchName FROM
Branch WHERE branchCity = "Nigdi");
```

Query OK, 0 rows affected (0.00 sec)

```
mysql> DELETE FROM Depositor WHERE accountNum IN (SELECT
accountNum FROM Account WHERE branchName IN (SELECT branchName
FROM Branch WHERE branchCity = "Nigdi"));
```

Query OK, 2 rows affected (0.01 sec)

```
mysql> DELETE FROM Account WHERE branchName = (SELECT branchName
FROM Branch WHERE branchCity = "Nigdi");
```

Query OK, 1 row affected (0.00 sec)

```
mysql> DELETE FROM Branch WHERE branchName = Nigdi;
```

ERROR 1054 (42S22): Unknown column 'Nigdi' in 'where clause'

```
mysql> SELECT * FROM Borrower;
```

| customerName | loanNum |
|--------------|---------|
| Kalas | 156 |
| Tanmay | 155 |
| Kshitij | 157 |
| Aditya | 159 |

```
4 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM Loan;
```

| loanNum | branchName | amount |
|---------|-------------|--------|
| 155 | Dhole Patil | 500 |
| 156 | Nagarwala | 250 |
| 157 | Peachtree | 600 |
| 159 | Amanora | 25000 |

```
4 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM Depositor;
```

| customerName | accountNum |
|--------------|------------|
| Kalas | 2511 |
| Tanmay | 2501 |
| Kshitij | 2521 |
| Aditya | 2531 |

```
4 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM Account;
```

| accountNum | branchName | balance |
|------------|-------------|---------|
| 2501 | Dhole Patil | 5000 |
| 2511 | Nagarwala | 1500 |
| 2521 | Peachtree | 2000 |
| 2531 | Amanora | 1000 |

```
4 rows in set (0.00 sec)
```

--- END OF DOCUMENT ---