

Consider a banking database and perform actions from the relation below and display the output along with the SQL query.

Customer (customer\_id, customer\_name, address, phone, email)

Borrows (borrow\_id, customer\_id, loan\_number)

Loan (loan\_number, loan\_type, amount)

- a. Display the names of all customers who live in “Lalitpur” in ascending order of name.
- b. Count the total number of customers having loan at the bank.
- c. Find the name of those customers who have loan amount greater than or equal to 500000.
- d. Find the average loan amount of each type.

### **Creation of Tables:**

#### ***SQL Query:***

Create table customer (

customer\_id INT PRIMARY KEY,

customer\_name VARCHAR(40),

address VARCHAR(40),

phone VARCHAR(40),

email VARCHAR(100) UNIQUE

);

Create table loan (

loan\_number INT PRIMARY KEY,

loan\_type VARCHAR(50),

amount INT

);

Create table borrows (

borrow\_id INT PRIMARY KEY,

```
customer_id INT,  
loan_number INT,  
FOREIGN KEY (customer_id) REFERENCES customer (customer_id)  
FOREIGN KEY (loan_number) REFERENCES loan (loan_number)  
);
```

### **Insertion of Data:**

#### ***SQL Query:***

Insert into customer (customer\_id, customer\_name, address, phone, email)

Values

```
(7901, "Hari Bahadur", "Lalitpur", "9876543210", "hari123@gmail.com"),  
(7902, "Madan Shrestha", "Kathmandu", "9812345678", "sthamadan@gmail.com"),  
(7903, "Rajesh Hamal", "Lalitpur", "9845612370", "rajesh123@gmail.com");
```

Insert into loan (loan\_number, loan\_type, amount)

Values

```
(101, "Home Loan", 750000),  
(102, "Personal Loan", 300000),  
(103, "Education Loan", 600000),  
(104, "Business Loan", 800000);
```

Insert into borrows (borrow\_id, customer\_id, loan\_number)

Values

```
(301, 7901, 101),  
(302, 7901, 102),  
(303, 7902, 103);  
(304, 7903, 104),  
(305, 7903, 103);
```

### *Table after Insertion of Data:*

```
mysql> select * from customer;
+-----+-----+-----+-----+-----+
| customer_id | customer_name | address | phone | email |
+-----+-----+-----+-----+-----+
| 7901 | Hari Bahadur | Lalitpur | 9876543210 | hari123@gmail.com |
| 7902 | Madan Shrestha | Kathmandu | 9812345678 | sthamadan@gmail.com |
| 7903 | Rajesh Hamal | Lalitpur | 9845612370 | rajesh123@gmail.com |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

*Customer Table*

```
mysql> select * from loan;
+-----+-----+-----+
| loan_number | loan_type | amount |
+-----+-----+-----+
| 101 | Home Loan | 750000 |
| 102 | Personal Loan | 300000 |
| 103 | Education Loan | 600000 |
| 104 | Business Loan | 800000 |
+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> select * from borrows;
+-----+-----+-----+
| borrow_id | customer_id | loan_number |
+-----+-----+-----+
| 301 | 7901 | 101 |
| 302 | 7901 | 102 |
| 303 | 7902 | 103 |
| 304 | 7903 | 104 |
| 305 | 7903 | 103 |
+-----+-----+-----+
5 rows in set (0.00 sec)
```

*Loan Table*

*Borrow Table*

a. Display the names of all customers who live in “Lalitpur” in ascending order of name.

#### *SQL Query:*

Select customer\_name from customer where address = “Lalitpur”

Order by customer\_name asc;

#### *Table:*

Here, the table shows the name of the customer (customer\_name) by ascending order who are residing in “Lalitpur”.

```
mysql> select customer_name from customer where address = "Lalitpur"
       → order by customer_name asc;
+-----+
| customer_name |
+-----+
| Hari Bahadur |
| Rajesh Hamal |
+-----+
2 rows in set (0.00 sec)
```

**b. Count the total number of customers having loan at the bank.**

***SQL Query:***

```
Select count (distinct customer_id) as Total_Customers_with_Loan  
From borrows;
```

***Table:***

Here, the table shows the number of customers who have taken at least one loan. It ensures that each customer is counted only once, even if they have multiple loans.

```
mysql> select count(distinct customer_id) as Total_Customers_with_Loan  
→ from borrows;  
+-----+  
| Total_Customers_with_Loan |  
+-----+  
| 3 |  
+-----+  
1 row in set (0.00 sec)
```

**c. Find the name of those customers who have loan amount greater than or equal to 500000.**

***SQL Query:***

```
Select distinct c.customer_name  
From customer as c  
Join borrows as b on c.customer_id = b.customer_id  
Join loan as l on l.loan_number = b.loan_number  
Where l.amount >= 500000;
```

***Table:***

Here, the table shows the names of customers who have at least one loan with an amount of 5,00,000 or more.

```
mysql> select distinct c.customer_name
      → from customer as c
      → join borrows as b on c.customer_id = b.customer_id
      → join loan as l on l.loan_number = b.loan_number
      → where l.amount ≥ 500000;
+-----+
| customer_name |
+-----+
| Hari Bahadur  |
| Madan Shrestha |
| Rajesh Hamal   |
+-----+
3 rows in set (0.00 sec)
```

**d. Find the average loan amount of each type**

***SQL Query:***

Select loan\_type, avg(amount) as Average\_Loan

From loan

Group by loan\_type;

***Table:***

Here, the table shows the average loan amount for each type of loan.

```
mysql> select loan_type, avg(amount) as Average_Loan
      → from loan
      → group by loan_type;
+-----+-----+
| loan_type | Average_Loan |
+-----+-----+
| Business Loan | 800000.0000 |
| Education Loan | 600000.0000 |
| Home Loan | 750000.0000 |
| Personal Loan | 300000.0000 |
+-----+-----+
4 rows in set (0.00 sec)
```