DBMS LAB-5

NAME:- ABHISHIKTH BODA

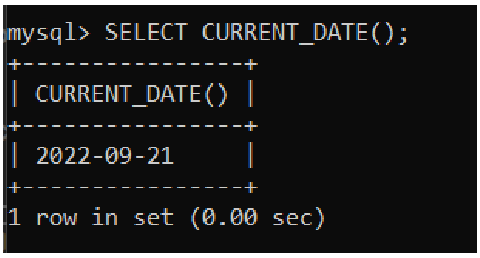
ROLL NUMBER:- S20210010044

DATE:- 21-09-2022

EXERCISE1:-

QUESTION1:-

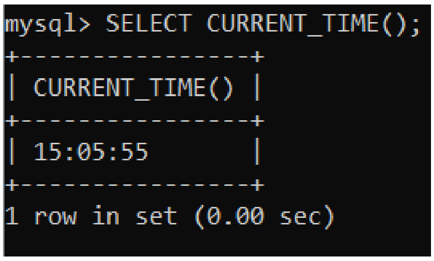
1. CURRENT_DATE:-



```
mysql> SELECT CURRENT_DATE();
+----------------+
| CURRENT_DATE() |
+----------------+
| 2022-09-21     |
+----------------+
1 row in set (0.00 sec)
```

QUESTION2:-

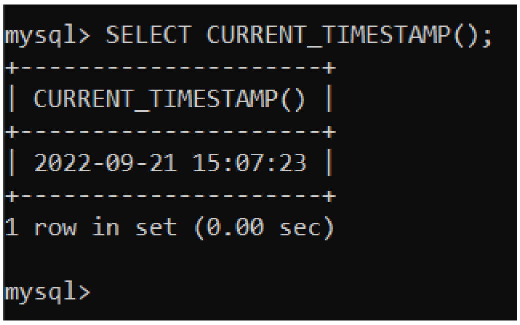
2.CURRENT_TIME:-



```
mysql> SELECT CURRENT_TIME();
+----------------+
| CURRENT_TIME() |
+----------------+
| 15:05:55       |
+----------------+
1 row in set (0.00 sec)
```

QUESTION3:-

3. CURRENT_TIMESTAMP:-

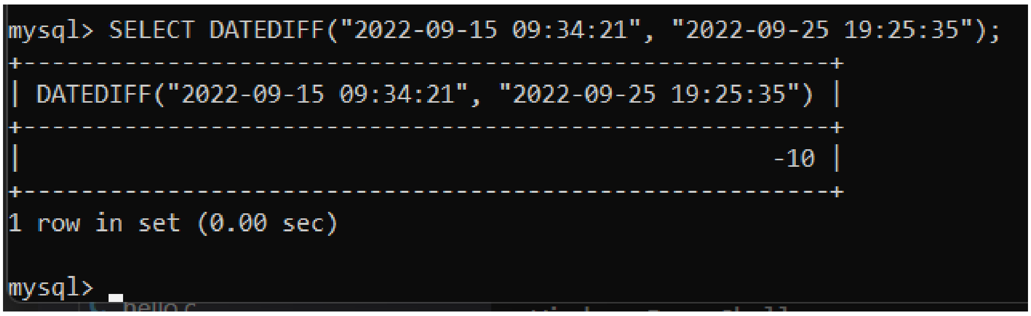


```
mysql> SELECT CURRENT_TIMESTAMP();
+---------------------+
| CURRENT_TIMESTAMP() |
+---------------------+
| 2022-09-21 15:07:23 |
+---------------------+
1 row in set (0.00 sec)

mysql>
```

QUESTION4:-

4. DATEDIFF:-

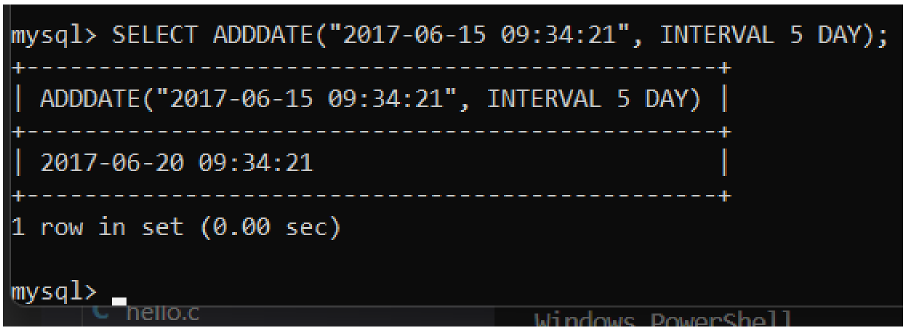


```
mysql> SELECT DATEDIFF("2022-09-15 09:34:21", "2022-09-25 19:25:35");
+--------------------------------------------------------+
| DATEDIFF("2022-09-15 09:34:21", "2022-09-25 19:25:35") |
+--------------------------------------------------------+
|                                                    -10 |
+--------------------------------------------------------+
1 row in set (0.00 sec)

mysql> _
```

QUESTION5:-

5. ADDDATE:-

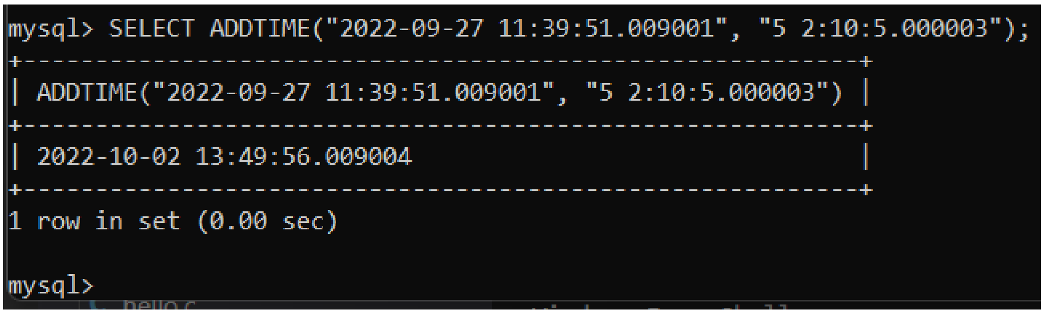


```
mysql> SELECT ADDDATE("2017-06-15 09:34:21", INTERVAL 5 DAY);
+------------------------------------------------+
| ADDDATE("2017-06-15 09:34:21", INTERVAL 5 DAY) |
+------------------------------------------------+
| 2017-06-20 09:34:21                            |
+------------------------------------------------+
1 row in set (0.00 sec)

mysql> _
```

QUESTION6:-

6. ADDTIME:-

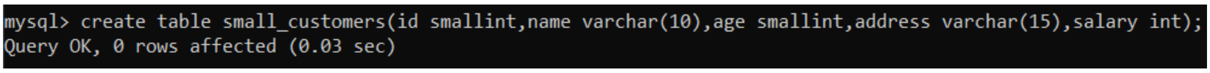


```
mysql> SELECT ADDTIME("2022-09-27 11:39:51.009001", "5 2:10:5.000003");
+----------------------------------------------------------+
| ADDTIME("2022-09-27 11:39:51.009001", "5 2:10:5.000003") |
+----------------------------------------------------------+
| 2022-10-02 13:49:56.009004                               |
+----------------------------------------------------------+
1 row in set (0.00 sec)

mysql>
```

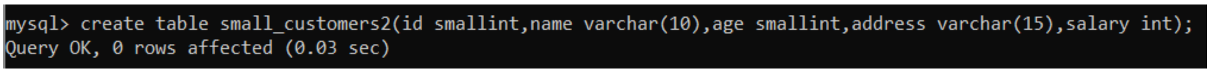
EXERCISE 2:-

1. create table small_customers(id smallint,name varchar(10),age smallint,address varchar(15),salary int);



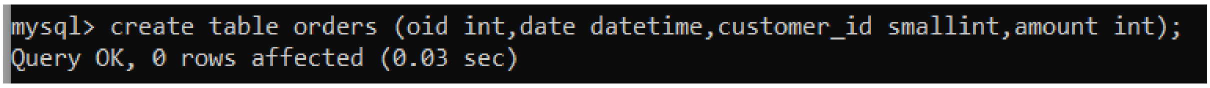
```
mysql> create table small_customers(id smallint,name varchar(10),age smallint,address varchar(15),salary int);
Query OK, 0 rows affected (0.03 sec)
```

2. create table small_customers2(id smallint,name varchar(10),age smallint,address varchar(15),salary int);



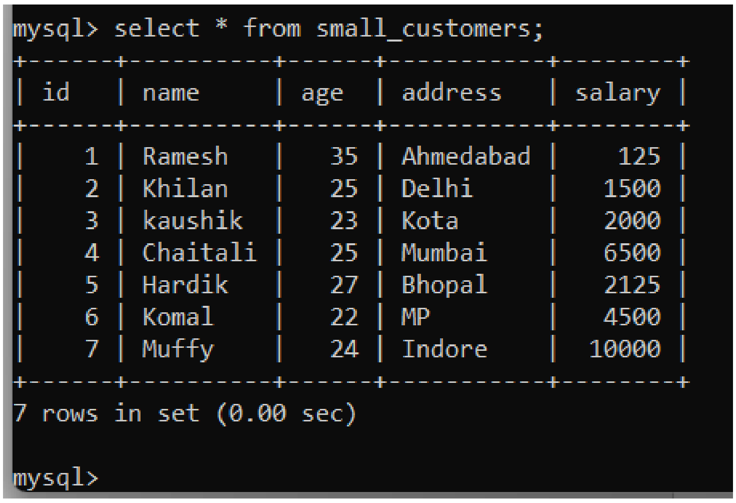
```
mysql> create table small_customers2(id smallint,name varchar(10),age smallint,address varchar(15),salary int);
Query OK, 0 rows affected (0.03 sec)
```

3. create table orders (oid int,date datetime,customer_id smallint,amount int);



```
mysql> create table orders (oid int,date datetime,customer_id smallint,amount int);
Query OK, 0 rows affected (0.03 sec)
```

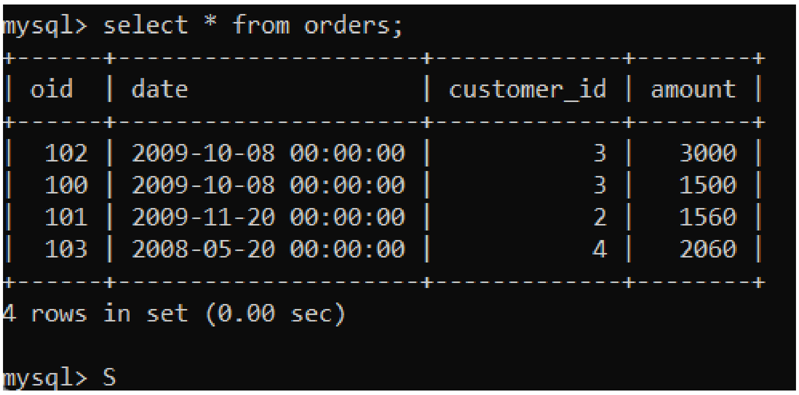
4. LOAD DATA LOCAL INFILE 'small_customers.csv' INTO table small_customers COLUMNS TERMINATED BY ',';



```
mysql> select * from small_customers;
+------+----------+------+-----------+--------+
| id   | name     | age  | address   | salary |
+------+----------+------+-----------+--------+
|    1 | Ramesh   |   35 | Ahmedabad |    125 |
|    2 | Khilan   |   25 | Delhi     |   1500 |
|    3 | kaushik  |   23 | Kota      |   2000 |
|    4 | Chaitali |   25 | Mumbai    |   6500 |
|    5 | Hardik   |   27 | Bhopal    |   2125 |
|    6 | Komal    |   22 | MP        |   4500 |
|    7 | Muffy    |   24 | Indore    |  10000 |
+------+----------+------+-----------+--------+
7 rows in set (0.00 sec)

mysql>
```

5. LOAD DATA LOCAL INFILE 'orders.csv' INTO table orders COLUMNS TERMINATED BY ',';



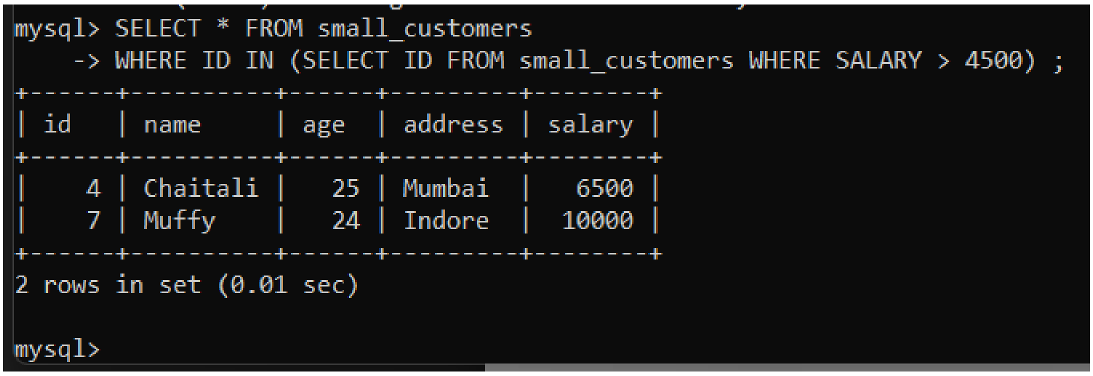
```
mysql> select * from orders;
+------+---------------------+-------------+--------+
| oid  | date                | customer_id | amount |
+------+---------------------+-------------+--------+
|  102 | 2009-10-08 00:00:00 |           3 |   3000 |
|  100 | 2009-10-08 00:00:00 |           3 |   1500 |
|  101 | 2009-11-20 00:00:00 |           2 |   1560 |
|  103 | 2008-05-20 00:00:00 |           4 |   2060 |
+------+---------------------+-------------+--------+
4 rows in set (0.00 sec)

mysql> S
```

SUB QUERIES:-

QUESTION1:-

With select:-

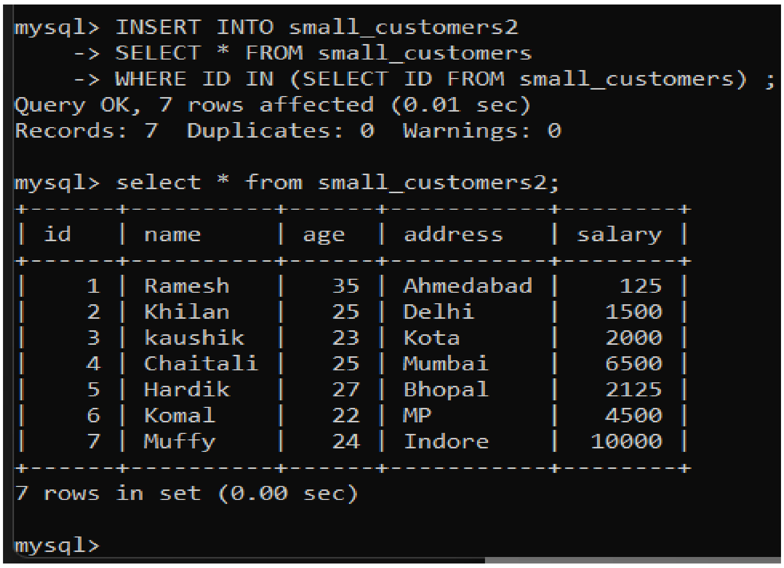


```
mysql> SELECT * FROM small_customers
    -> WHERE ID IN (SELECT ID FROM small_customers WHERE SALARY > 4500) ;
+------+----------+------+---------+--------+
| id   | name     | age  | address | salary |
+------+----------+------+---------+--------+
|    4 | Chaitali |   25 | Mumbai  |   6500 |
|    7 | Muffy    |   24 | Indore  |  10000 |
+------+----------+------+---------+--------+
2 rows in set (0.01 sec)

mysql>
```

QUESTION2:-

With insert:-



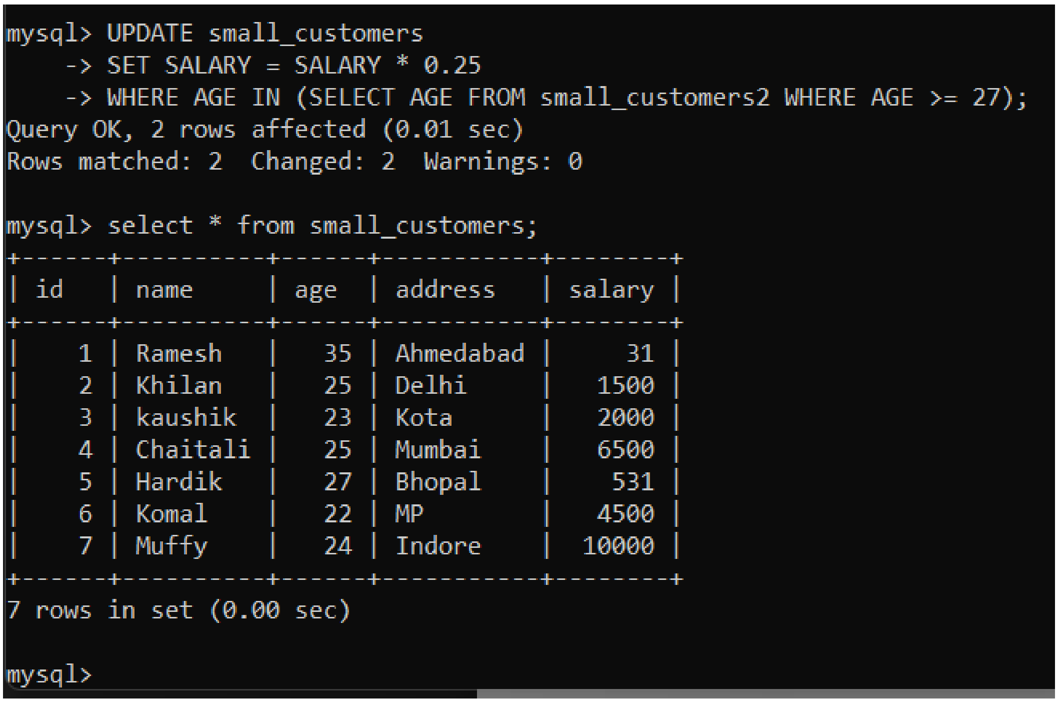
```
mysql> INSERT INTO small_customers2
    -> SELECT * FROM small_customers
    -> WHERE ID IN (SELECT ID FROM small_customers) ;
Query OK, 7 rows affected (0.01 sec)
Records: 7  Duplicates: 0  Warnings: 0

mysql> select * from small_customers2;
+------+----------+------+-----------+--------+
| id   | name     | age  | address   | salary |
+------+----------+------+-----------+--------+
|    1 | Ramesh   |   35 | Ahmedabad |    125 |
|    2 | Khilan   |   25 | Delhi     |   1500 |
|    3 | kaushik  |   23 | Kota      |   2000 |
|    4 | Chaitali |   25 | Mumbai    |   6500 |
|    5 | Hardik   |   27 | Bhopal    |   2125 |
|    6 | Komal    |   22 | MP        |   4500 |
|    7 | Muffy    |   24 | Indore    |  10000 |
+------+----------+------+-----------+--------+
7 rows in set (0.00 sec)

mysql>
```

QUESTION3:-

With update:-



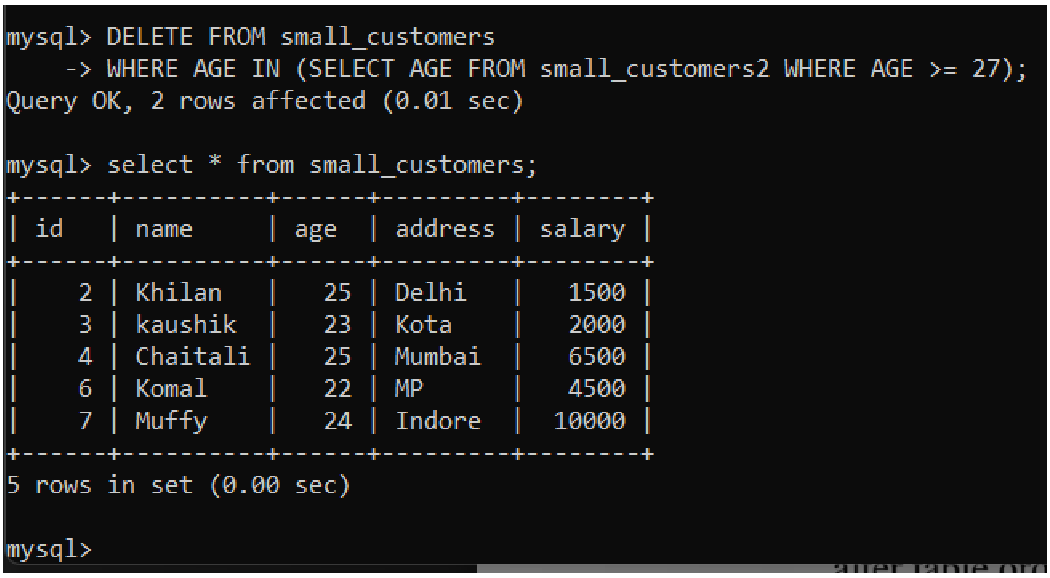
```
mysql> UPDATE small_customers
    -> SET SALARY = SALARY * 0.25
    -> WHERE AGE IN (SELECT AGE FROM small_customers2 WHERE AGE >= 27);
Query OK, 2 rows affected (0.01 sec)
Rows matched: 2  Changed: 2  Warnings: 0

mysql> select * from small_customers;
+------+----------+------+-----------+--------+
| id   | name     | age  | address   | salary |
+------+----------+------+-----------+--------+
|    1 | Ramesh   |   35 | Ahmedabad |     31 |
|    2 | Khilan   |   25 | Delhi     |   1500 |
|    3 | kaushik  |   23 | Kota      |   2000 |
|    4 | Chaitali |   25 | Mumbai    |   6500 |
|    5 | Hardik   |   27 | Bhopal    |    531 |
|    6 | Komal    |   22 | MP        |   4500 |
|    7 | Muffy    |   24 | Indore    |  10000 |
+------+----------+------+-----------+--------+
7 rows in set (0.00 sec)

mysql>
```

QUESTION4:-

With delete:-



```
mysql> DELETE FROM small_customers
    -> WHERE AGE IN (SELECT AGE FROM small_customers2 WHERE AGE >= 27);
Query OK, 2 rows affected (0.01 sec)

mysql> select * from small_customers;
+------+----------+------+---------+--------+
| id   | name     | age  | address | salary |
+------+----------+------+---------+--------+
|    2 | Khilan   |   25 | Delhi   |   1500 |
|    3 | kaushik  |   23 | Kota    |   2000 |
|    4 | Chaitali |   25 | Mumbai  |   6500 |
|    6 | Komal    |   22 | MP      |   4500 |
|    7 | Muffy    |   24 | Indore  |  10000 |
+------+----------+------+---------+--------+
5 rows in set (0.00 sec)

mysql>
```

EXERCISE3:-

Use Bank Database Schema:-

branch (branch_name, branch_city, assets)

customer (customer_name, customer_street, customer_city)

loan (loan_number, branch_name, amount)

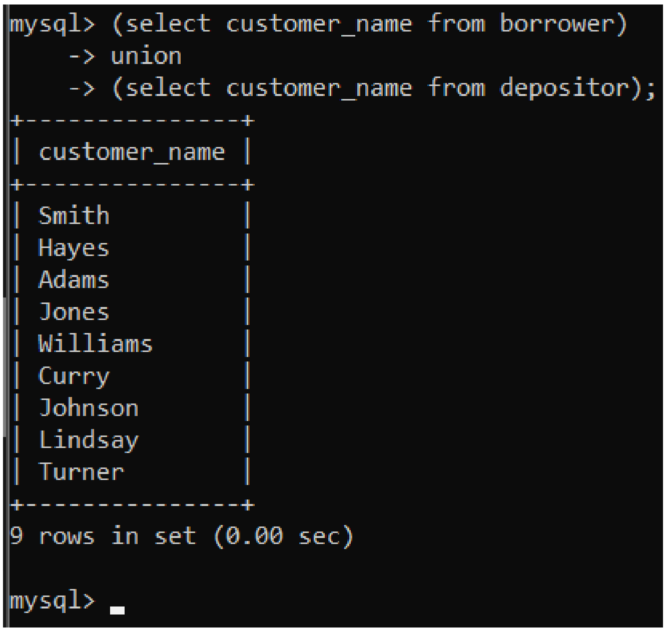
borrower (customer_name, loan_number)

account (account_number, branch_name, balance)

depositor (customer_name, account_number)

QUESTION1:-

Find all the bank customers having a loan, an account or both at the bank.

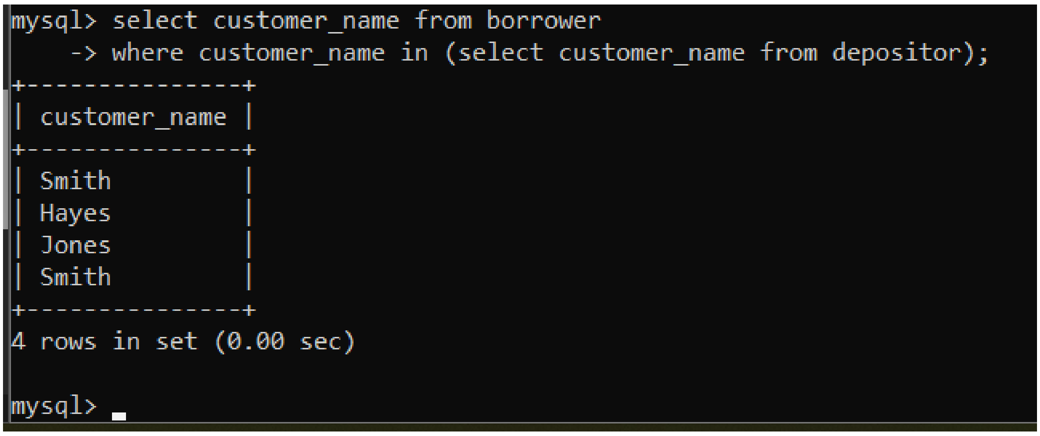


```
mysql> (select customer_name from borrower)
    -> union
    -> (select customer_name from depositor);
+---------------+
| customer_name |
+---------------+
| Smith         |
| Hayes         |
| Adams         |
| Jones         |
| Williams      |
| Curry         |
| Johnson       |
| Lindsay       |
| Turner        |
+---------------+
9 rows in set (0.00 sec)

mysql> _
```

QUESTION2:-

Find those customers who are borrowers from the bank and who appear in the list of account holders (i.e present in depositor table).

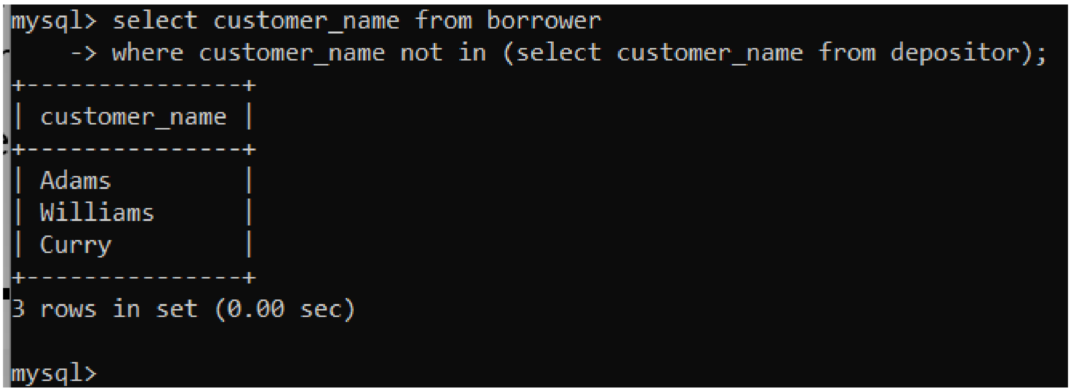


```
mysql> select customer_name from borrower
    -> where customer_name in (select customer_name from depositor);
+---------------+
| customer_name |
+---------------+
| Smith         |
| Hayes         |
| Jones         |
| Smith         |
+---------------+
4 rows in set (0.00 sec)

mysql> _
```

QUESTION3:-

Find all the customers who have loans at the bank ,but do not have an account at the bank.

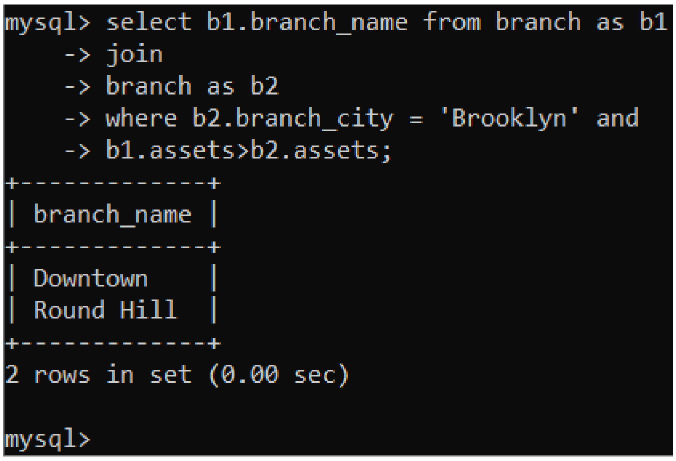


```
mysql> select customer_name from borrower
    -> where customer_name not in (select customer_name from depositor);
+---------------+
| customer_name |
+---------------+
| Adams         |
| Williams      |
| Curry         |
+---------------+
3 rows in set (0.00 sec)

mysql>
```

QUESTION4:-

Find the names of all branches that have assets greater than those of at least one branch located in Brooklyn (without using subquery).

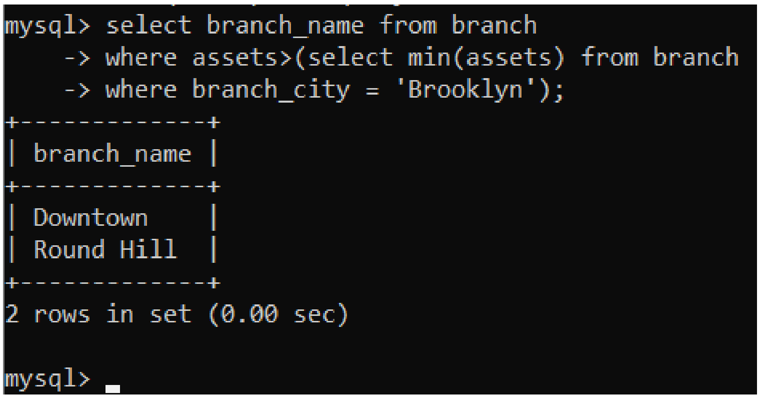


```
mysql> select b1.branch_name from branch as b1
    -> join
    -> branch as b2
    -> where b2.branch_city = 'Brooklyn' and
    -> b1.assets>b2.assets;
+-------------+
| branch_name |
+-------------+
| Downtown    |
| Round Hill  |
+-------------+
2 rows in set (0.00 sec)

mysql>
```

QUESTION 5:-

Find the names of all branches that have assets greater than those of at least one branch located in Brooklyn (using subquery).

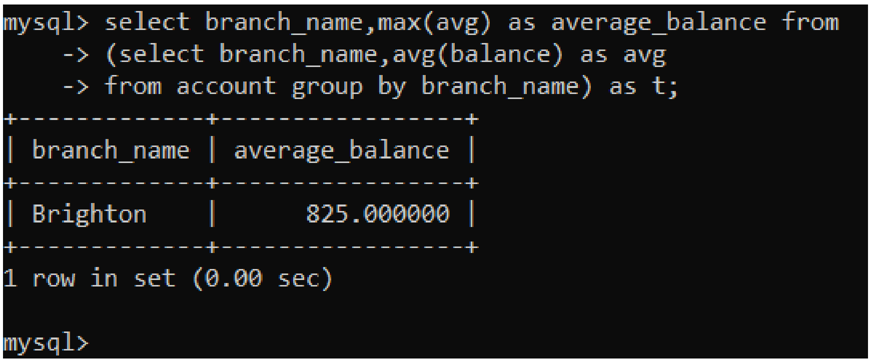


```
mysql> select branch_name from branch
    -> where assets>(select min(assets) from branch
    -> where branch_city = 'Brooklyn');
+-------------+
| branch_name |
+-------------+
| Downtown    |
| Round Hill  |
+-------------+
2 rows in set (0.00 sec)

mysql> _
```

QUESTION 6:-

Find the branch that has the highest average balance.

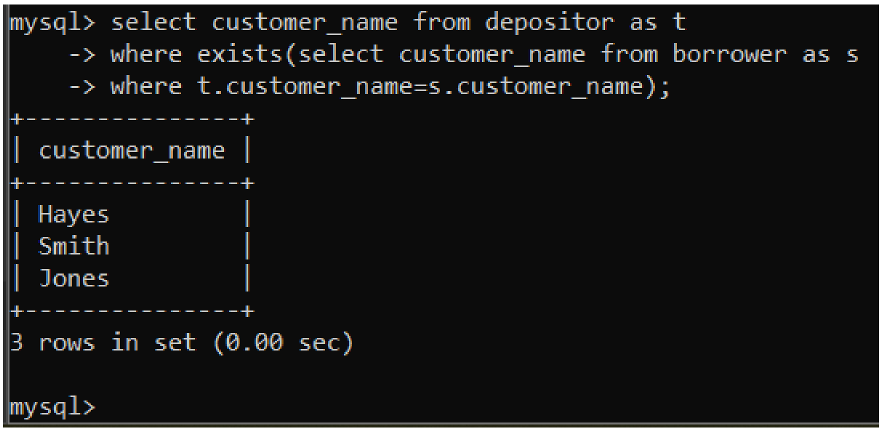


```
mysql> select branch_name,max(avg) as average_balance from
    -> (select branch_name,avg(balance) as avg
    -> from account group by branch_name) as t;
+-------------+-----------------+
| branch_name | average_balance |
+-------------+-----------------+
| Brighton    |      825.000000 |
+-------------+-----------------+
1 row in set (0.00 sec)

mysql>
```

QUESTION 7:-

Find all the customers who have both an account and a loan at the bank,by a subquery using “exists” keyword.

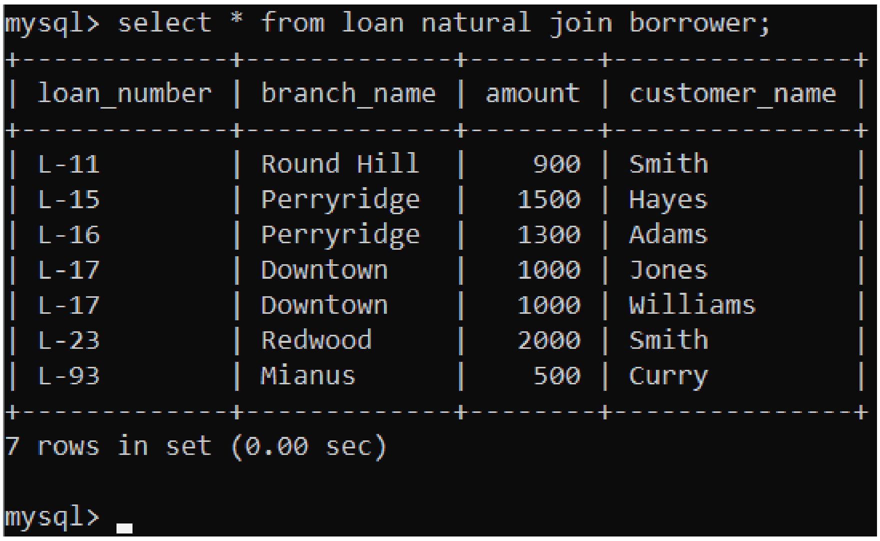


```
mysql> select customer_name from depositor as t
    -> where exists(select customer_name from borrower as s
    -> where t.customer_name=s.customer_name);
+---------------+
| customer_name |
+---------------+
| Hayes         |
| Smith         |
| Jones         |
+---------------+
3 rows in set (0.00 sec)

mysql>
```

QUESTION 8:-

Perform natural join between tables loan and borrower.

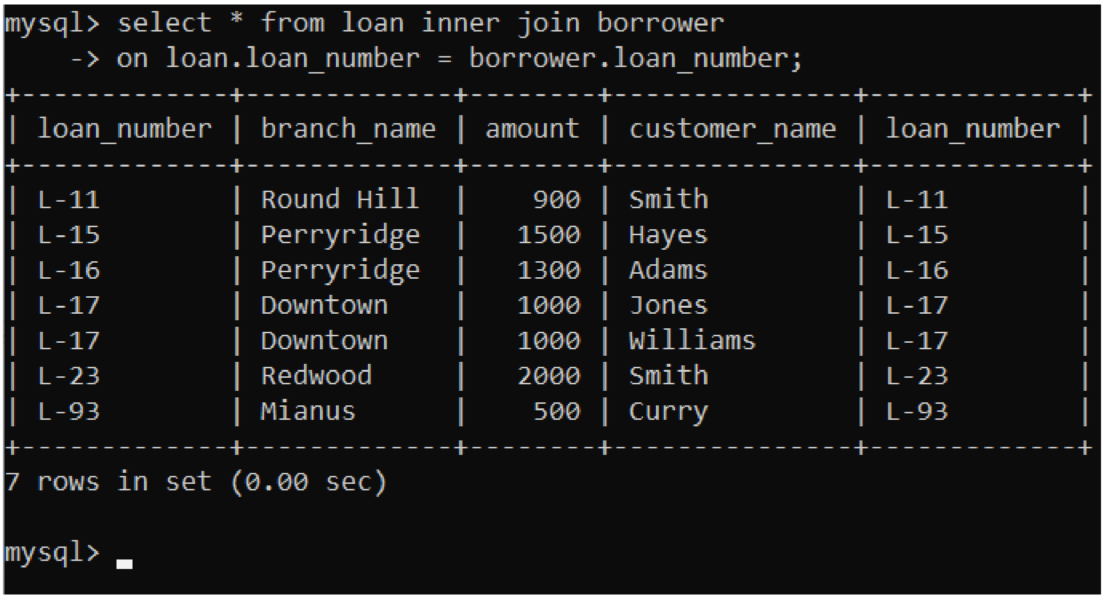


```
mysql> select * from loan natural join borrower;
+-------------+-------------+--------+---------------+
| loan_number | branch_name | amount | customer_name |
+-------------+-------------+--------+---------------+
| L-11        | Round Hill  |    900 | Smith         |
| L-15        | Perryridge  |   1500 | Hayes         |
| L-16        | Perryridge  |   1300 | Adams         |
| L-17        | Downtown    |   1000 | Jones         |
| L-17        | Downtown    |   1000 | Williams      |
| L-23        | Redwood     |   2000 | Smith         |
| L-93        | Mianus      |    500 | Curry         |
+-------------+-------------+--------+---------------+
7 rows in set (0.00 sec)

mysql> _
```

QUESTION 9:-

Perform inner join between tables loan and borrower,with loan_number as joining condition.

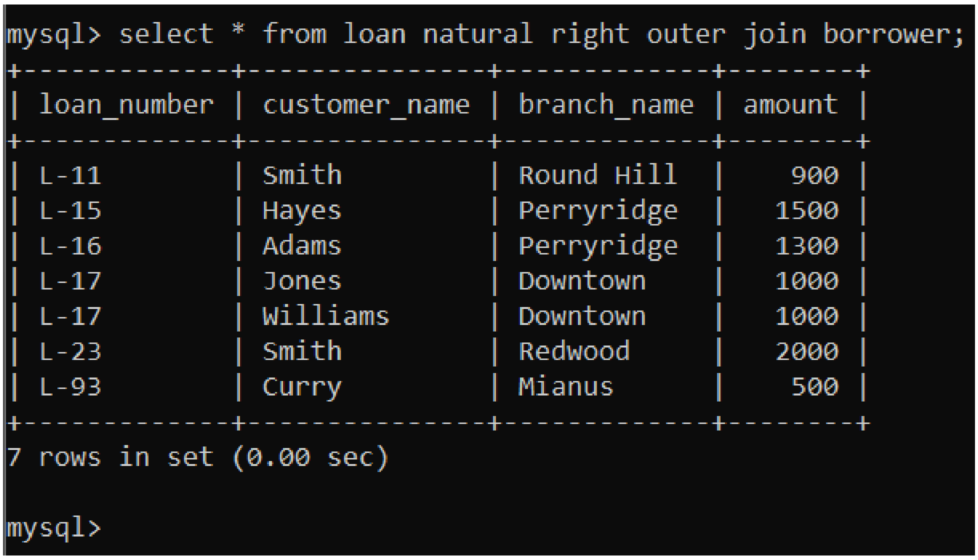


```
mysql> select * from loan inner join borrower
    -> on loan.loan_number = borrower.loan_number;
+-------------+-------------+--------+---------------+-------------+
| loan_number | branch_name | amount | customer_name | loan_number |
+-------------+-------------+--------+---------------+-------------+
| L-11        | Round Hill  |    900 | Smith         | L-11        |
| L-15        | Perryridge  |   1500 | Hayes         | L-15        |
| L-16        | Perryridge  |   1300 | Adams         | L-16        |
| L-17        | Downtown    |   1000 | Jones         | L-17        |
| L-17        | Downtown    |   1000 | Williams      | L-17        |
| L-23        | Redwood     |   2000 | Smith         | L-23        |
| L-93        | Mianus      |    500 | Curry         | L-93        |
+-------------+-------------+--------+---------------+-------------+
7 rows in set (0.00 sec)

mysql> _
```

QUESTION 10:-

Perform natural right outer join between tables loan and borrower.

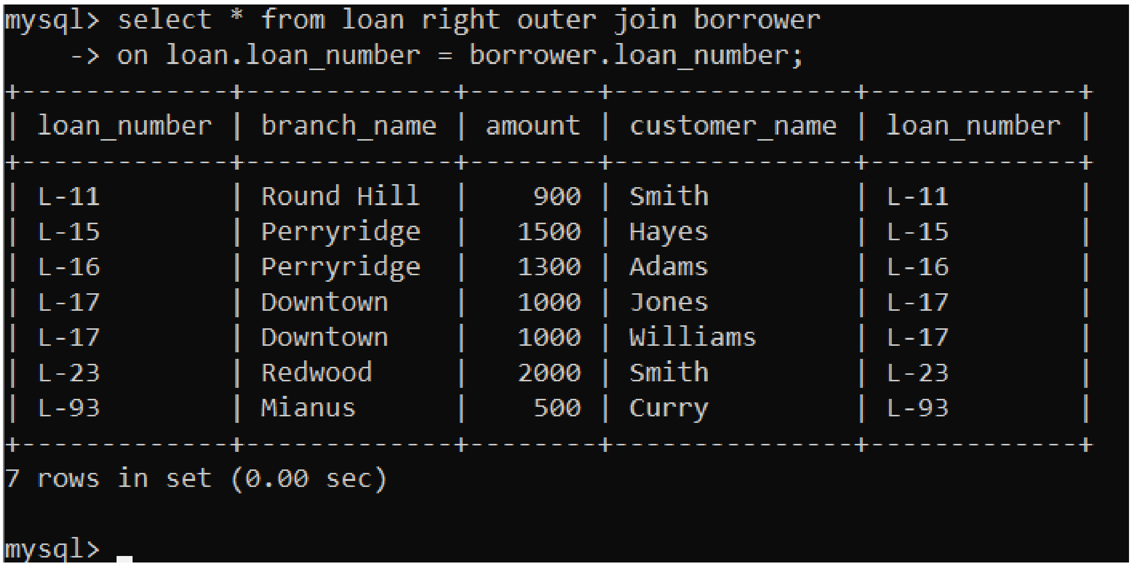


```
mysql> select * from loan natural right outer join borrower;
+-------------+---------------+------------+--------+
| loan_number | customer_name | branch_name | amount |
+-------------+---------------+------------+--------+
| L-11        | Smith         | Round Hill |    900 |
| L-15        | Hayes         | Perryridge |   1500 |
| L-16        | Adams         | Perryridge |   1300 |
| L-17        | Jones         | Downtown   |   1000 |
| L-17        | Williams      | Downtown   |   1000 |
| L-23        | Smith         | Redwood    |   2000 |
| L-93        | Curry         | Mianus     |    500 |
+-------------+---------------+------------+--------+
7 rows in set (0.00 sec)

mysql>
```

QUESTION 11:-

Perform right outer join between tables loan and borrower, with loan_number as joining condition.

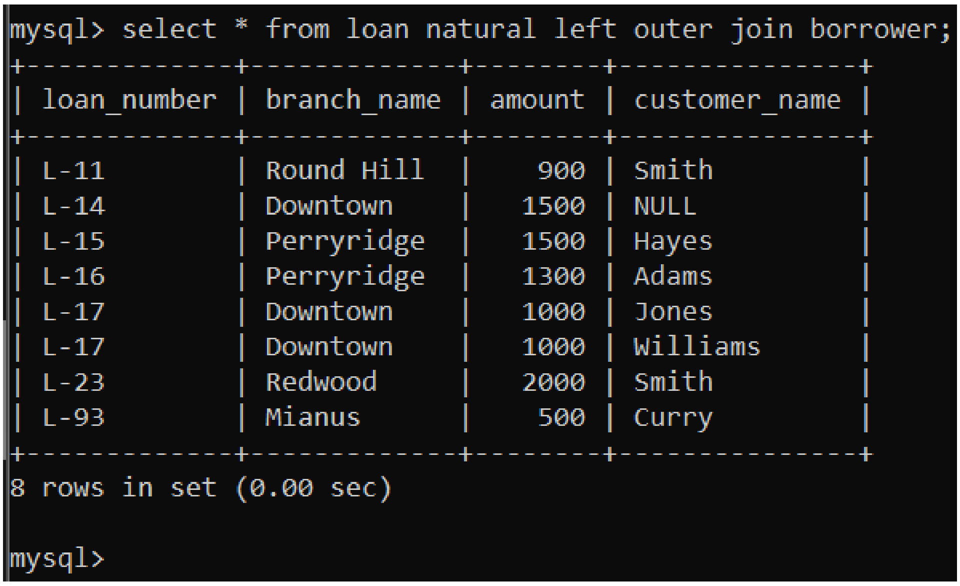


```
mysql> select * from loan right outer join borrower
    -> on loan.loan_number = borrower.loan_number;
+-------------+-------------+--------+---------------+-------------+
| loan_number | branch_name | amount | customer_name | loan_number |
+-------------+-------------+--------+---------------+-------------+
| L-11        | Round Hill  |    900 | Smith         | L-11        |
| L-15        | Perryridge  |   1500 | Hayes         | L-15        |
| L-16        | Perryridge  |   1300 | Adams         | L-16        |
| L-17        | Downtown    |   1000 | Jones         | L-17        |
| L-17        | Downtown    |   1000 | Williams      | L-17        |
| L-23        | Redwood     |   2000 | Smith         | L-23        |
| L-93        | Mianus      |    500 | Curry         | L-93        |
+-------------+-------------+--------+---------------+-------------+
7 rows in set (0.00 sec)

mysql> _
```

QUESTION 12:-

Perform natural left outer join between tables loan and borrower.

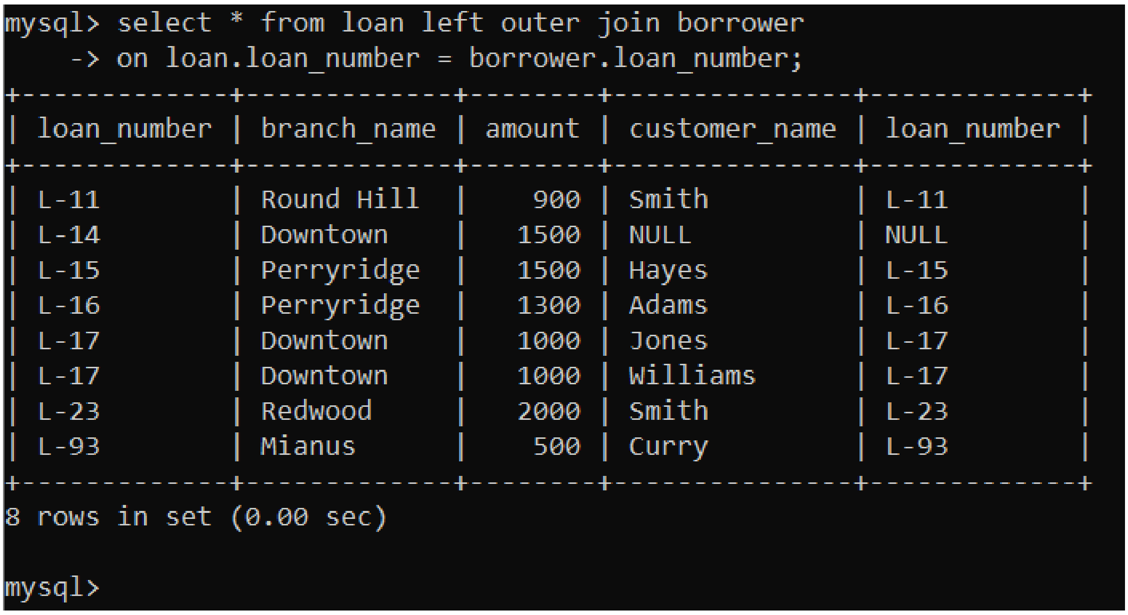


```
mysql> select * from loan natural left outer join borrower;
+-------------+-------------+--------+---------------+
| loan_number | branch_name | amount | customer_name |
+-------------+-------------+--------+---------------+
| L-11        | Round Hill  |    900 | Smith         |
| L-14        | Downtown    |   1500 | NULL          |
| L-15        | Perryridge  |   1500 | Hayes         |
| L-16        | Perryridge  |   1300 | Adams         |
| L-17        | Downtown    |   1000 | Jones         |
| L-17        | Downtown    |   1000 | Williams      |
| L-23        | Redwood     |   2000 | Smith         |
| L-93        | Mianus      |    500 | Curry         |
+-------------+-------------+--------+---------------+
8 rows in set (0.00 sec)

mysql>
```

QUESTION 13:-

Perform left outer join between tables loan and borrower,with loan_number as joining condition.

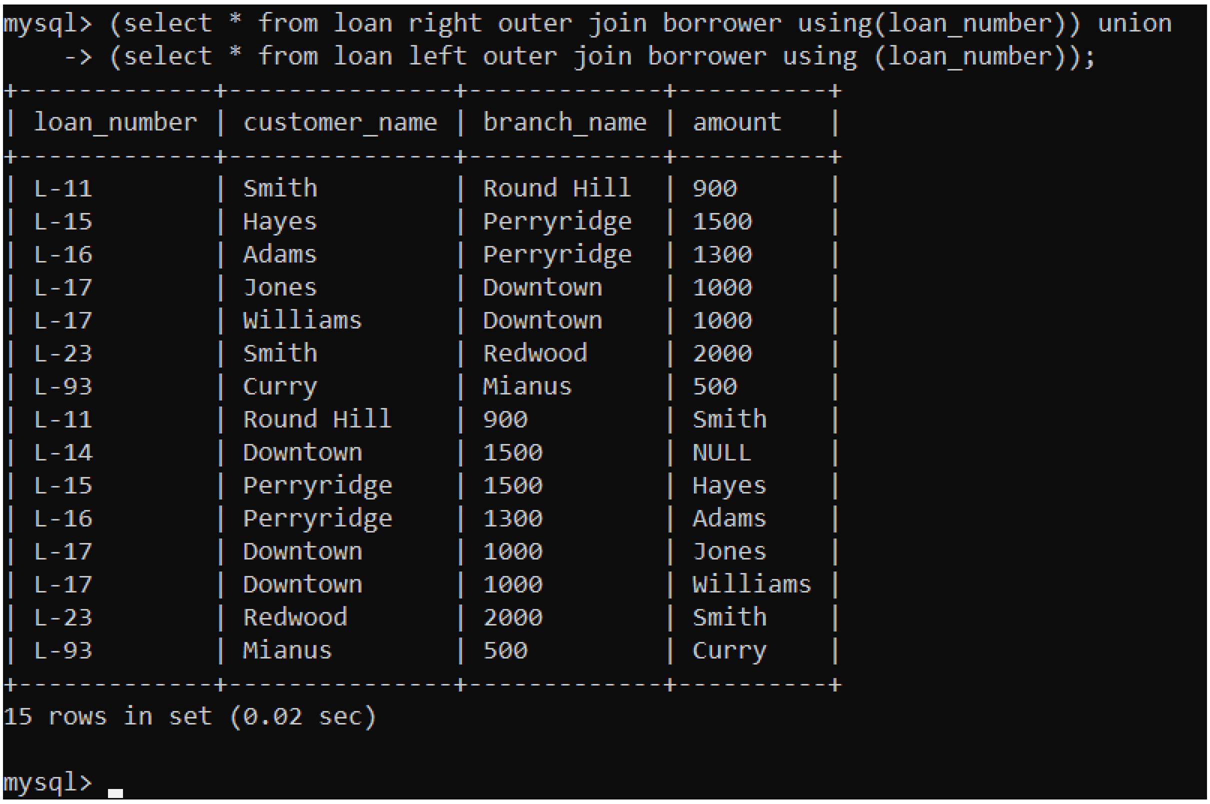


```
mysql> select * from loan left outer join borrower
    -> on loan.loan_number = borrower.loan_number;
+-------------+-------------+--------+---------------+-------------+
| loan_number | branch_name | amount | customer_name | loan_number |
+-------------+-------------+--------+---------------+-------------+
| L-11        | Round Hill  |    900 | Smith         | L-11        |
| L-14        | Downtown    |   1500 | NULL          | NULL        |
| L-15        | Perryridge  |   1500 | Hayes         | L-15        |
| L-16        | Perryridge  |   1300 | Adams         | L-16        |
| L-17        | Downtown    |   1000 | Jones         | L-17        |
| L-17        | Downtown    |   1000 | Williams      | L-17        |
| L-23        | Redwood     |   2000 | Smith         | L-23        |
| L-93        | Mianus      |    500 | Curry         | L-93        |
+-------------+-------------+--------+---------------+-------------+
8 rows in set (0.00 sec)

mysql>
```

QUESTION 14:-

Perform full outer join between tables loan and borrower



```
mysql> (select * from loan right outer join borrower using(loan_number)) union
    -> (select * from loan left outer join borrower using (loan_number));
+-------------+---------------+-------------+----------+
| loan_number | customer_name | branch_name | amount   |
+-------------+---------------+-------------+----------+
| L-11        | Smith         | Round Hill  | 900      |
| L-15        | Hayes         | Perryridge  | 1500     |
| L-16        | Adams         | Perryridge  | 1300     |
| L-17        | Jones         | Downtown    | 1000     |
| L-17        | Williams      | Downtown    | 1000     |
| L-23        | Smith         | Redwood     | 2000     |
| L-93        | Curry         | Mianus      | 500      |
| L-11        | Round Hill    | 900         | Smith    |
| L-14        | Downtown      | 1500        | NULL     |
| L-15        | Perryridge    | 1500        | Hayes    |
| L-16        | Perryridge    | 1300        | Adams    |
| L-17        | Downtown      | 1000        | Jones    |
| L-17        | Downtown      | 1000        | Williams |
| L-23        | Redwood       | 2000        | Smith    |
| L-93        | Mianus        | 500         | Curry    |
+-------------+---------------+-------------+----------+
15 rows in set (0.02 sec)

mysql> _
```

←THE END→