Q1. Find the name and loan number of all customers having a loan at the Downtown branch

Command:

```
-- Question [ 01 ]
SELECT c.customer_name, b.loan_number
FROM customer c
JOIN borrower b ON c.customer_id = b.customer_id
JOIN loan l ON b.loan_number = l.loan_number
WHERE l.branch_name = 'Downtown';
```

```
mysql> -- Question [ 01 ]
mysql> SELECT c.customer_name, b.loan_number
    -> FROM customer c
    -> JOIN borrower b ON c.customer_id = b.customer_id
    -> JOIN loan l ON b.loan_number = l.loan_number
    -> WHERE l.branch_name = 'Downtown';
+---------------+-------------+
| customer_name | loan_number |
+---------------+-------------+
| Johnson       | L-14        |
| Jones         | L-17        |
| Williams      | L-17        |
+---------------+-------------+
3 rows in set (0.02 sec)
```

Q2. Find all the possible pairs of customers who are from the same city. show in the format Customer1, Customer2, City.

Command:

```
-- Question [ 02 ]
SELECT
c1.customer_name AS Customer1,
c2.customer_name AS Customer2,
c1.customer_city AS City
FROM customer c1
JOIN customer c2 ON c1.customer_city = c2.customer_city
```

```
WHERE c1.customer_id < c2.customer_id;
```

```
mysql> -- Question [ 02 ]
mysql> SELECT
    -> c1.customer_name AS Customer1,
    -> c2.customer_name AS Customer2,
    -> c1.customer_city AS City
    -> FROM customer c1
    -> JOIN customer c2 ON c1.customer_city = c2.customer_city
    -> WHERE c1.customer_id < c2.customer_id;
+-----------+-----------+-----------+
| Customer1 | Customer2 | City      |
+-----------+-----------+-----------+
| Jones     | Hayes     | Harrison  |
| Smith     | Curry     | Rye       |
| Lindsay   | Adams     | Pittsfield |
| Turner    | Green     | Stamford  |
+-----------+-----------+-----------+
4 rows in set (0.00 sec)
```

Q3. If the bank gives out 4% interest to all accounts, show the total interest across each branch. Print Branch_name, Total_Interest.

Command:

```
-- Question [ 03 ]
SELECT a.branch_name AS Branch_name,
SUM(0.04 * a.balance) AS Total_Interest
FROM account a
GROUP BY a.branch_name;
```

```
mysql> -- Question [ 03 ]
mysql> SELECT a.branch_name AS Branch_name,
    -> SUM(0.04 * a.balance) AS Total_Interest
    -> FROM account a
    -> GROUP BY a.branch_name;
+-----------------+--------------------+
| Branch_name     | Total_Interest     |
+-----------------+--------------------+
| Downtown        |              20.00 |
| Perryridge      |              16.00 |
| Brighton        |              66.00 |
| Mianus          |              28.00 |
| Redwood         |              28.00 |
| Round Hill      |              14.00 |
+-----------------+--------------------+
6 rows in set (0.00 sec)
```

Q4. Find account numbers with the highest balances for each city in the database.

Command:

```sql
-- Question [ 04 ]
SELECT a.account_number, b.branch_city, a.balance
FROM account a
JOIN branch b ON a.branch_name = b.branch_name
WHERE (b.branch_city, a.balance) IN (
    SELECT b.branch_city, MAX(a.balance) AS Highest_Balance
    FROM account a
    JOIN branch b ON a.branch_name = b.branch_name
    GROUP BY b.branch_city
);
```

```
mysql> -- Question [ 04 ]
mysql> SELECT a.account_number, b.branch_city, a.balance
    -> FROM account a
    -> JOIN branch b ON a.branch_name = b.branch_name
    -> WHERE (b.branch_city, a.balance) IN (
    ->     SELECT b.branch_city, MAX(a.balance) AS Highest_Balance
    ->     FROM account a
    ->     JOIN branch b ON a.branch_name = b.branch_name
    ->     GROUP BY b.branch_city
    -> );
+----------------+-------------+---------+
| account_number | branch_city | balance |
+----------------+-------------+---------+
| A-201          | Brooklyn    |     900 |
| A-215          | Horseneck   |     700 |
| A-222          | Palo Alto   |     700 |
+----------------+-------------+---------+
3 rows in set (0.00 sec)
```

Q5. Show the loan number, loan amount, and name of customers who have the top 5 highest loan amounts. The data should be sorted by increasing amounts, then decreasing loan numbers in case of the same loan amount. [Hint for top 5 check the "limit" keyword in mysql]

Command:

```sql
-- Question [ 05 ]
SELECT l.loan_number, l.amount, c.customer_name
FROM loan l
JOIN borrower b ON l.loan_number = b.loan_number
JOIN customer c ON b.customer_id = c.customer_id
ORDER BY l.amount ASC, l.loan_number DESC
LIMIT 5;
```

```
mysql> -- Question [ 05 ]
mysql> SELECT l.loan_number, l.amount, c.customer_name
    -> FROM loan l
    -> JOIN borrower b ON l.loan_number = b.loan_number
    -> JOIN customer c ON b.customer_id = c.customer_id
    -> ORDER BY l.amount ASC, l.loan_number DESC
    -> LIMIT 5;
+-------------+--------+---------------+
| loan_number | amount | customer_name |
+-------------+--------+---------------+
| L-93        |    500 | Curry         |
| L-11        |    900 | Smith         |
| L-17        |   1000 | Jones         |
| L-17        |   1000 | Williams      |
| L-16        |   1300 | Adams         |
+-------------+--------+---------------+
5 rows in set (0.00 sec)
```

Q6. Find the names of customers with an account and also a loan at the Perryridge branch.

Command:

```sql
-- Question [ 06 ]
SELECT c.customer_name
FROM customer c
WHERE EXISTS (
    SELECT 1
    FROM depositor d
    WHERE d.customer_id = c.customer_id
)
AND EXISTS (
    SELECT 1
    FROM borrower b
    JOIN loan l ON b.loan_number = l.loan_number
    WHERE l.branch_name = 'Perryridge'
    AND b.customer_id = c.customer_id
);
```

```
mysql> -- Question [ 06 ]
mysql> SELECT c.customer_name
    -> FROM customer c
    -> WHERE EXISTS (
    ->     SELECT 1
    ->     FROM depositor d
    ->     WHERE d.customer_id = c.customer_id
    -> )
    -> AND EXISTS (
    ->     SELECT 1
    ->     FROM borrower b
    ->     JOIN loan l ON b.loan_number = l.loan_number
    ->     WHERE l.branch_name = 'Perryridge'
    ->     AND b.customer_id = c.customer_id
    -> );
+---------------+
| customer_name |
+---------------+
| Hayes         |
+---------------+
1 row in set (0.00 sec)
```

Q7. Find the total loan amount of all customers having at least 2 loans from the bank. Show in format customer name, total_loan

COMMAND:

```sql
SELECT c.customer_name, SUM(l.amount) AS total_loan
FROM customer c
JOIN borrower b ON c.customer_id = b.customer_id
JOIN loan l ON b.loan_number = l.loan_number
GROUP BY c.customer_id, c.customer_name
HAVING COUNT(b.loan_number) >= 2;
```

```
mysql> SELECT c.customer_name, SUM(l.amount) AS total_loan
    -> FROM customer c
    -> JOIN borrower b ON c.customer_id = b.customer_id
    -> JOIN loan l ON b.loan_number = l.loan_number
    -> GROUP BY c.customer_id, c.customer_name
    -> HAVING COUNT(b.loan_number) ≥ 2;
+---------------+------------+
| customer_name | total_loan |
+---------------+------------+
| Smith         |       2900 |
+---------------+------------+
1 row in set (0.00 sec)
```