课程名称:	数据库系统
实验名称:	SQL数据定义和操作
姓 名:	杨吉祥
学 院:	计算机科学与技术学院
系:	竺可桢学院图灵班
专 业:	计算机科学与技术
学 号:	3230106222
指导教师:	陈刚

一.实验目的:

1. 以某个应用场景（如Banking）为例，建立数据库。

2. 数据定义：表的建立、删除；索引的建立、删除；视图的建立、删除。

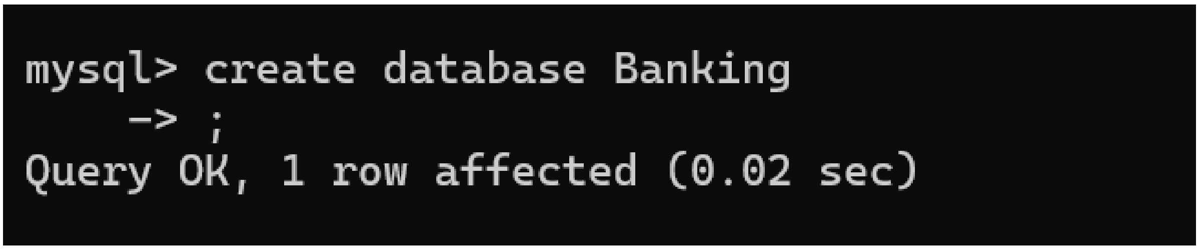
3. 数据更新：用insert/delete/update语句插入/删除/更新表数据。

4. 数据查询：单表查询、多表查询、嵌套子查询等。

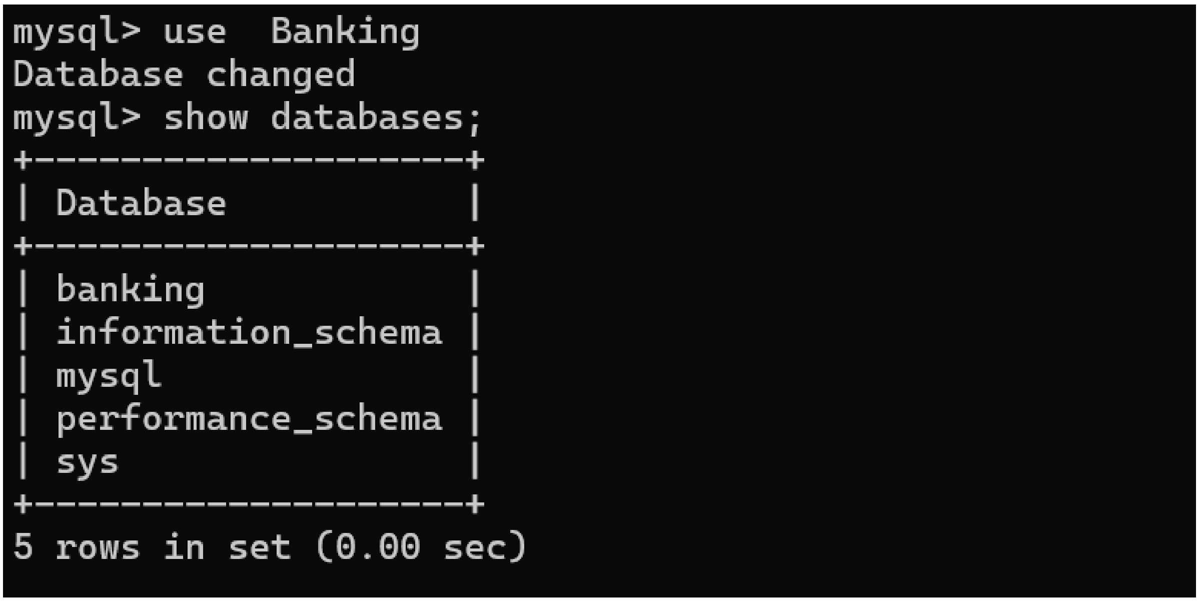
5. 视图操作：通过视图进行数据查询和数据更新。

二.实验内容和要求:

1.建立Banking数据库



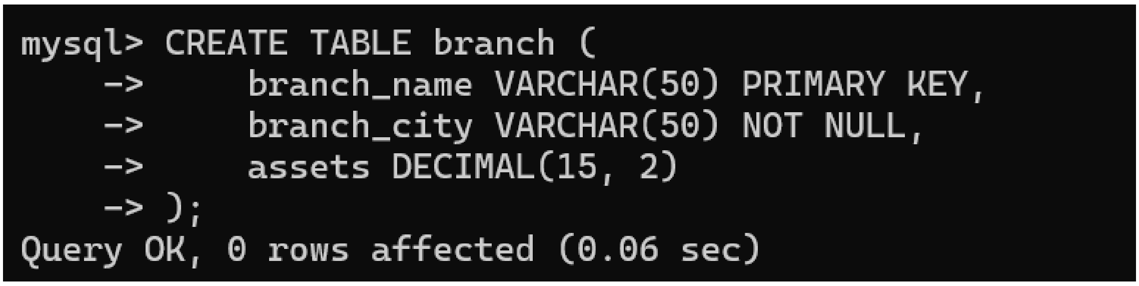
```
mysql> create database Banking
    -> ;
Query OK, 1 row affected (0.02 sec)
```



```
mysql> use  Banking
Database changed
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| banking            |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)
```

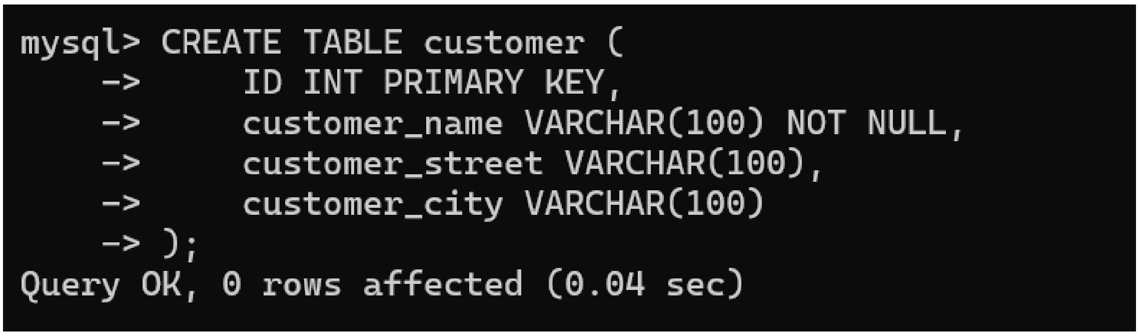
2.数据定义

- 表的建立
  - 建立branch表



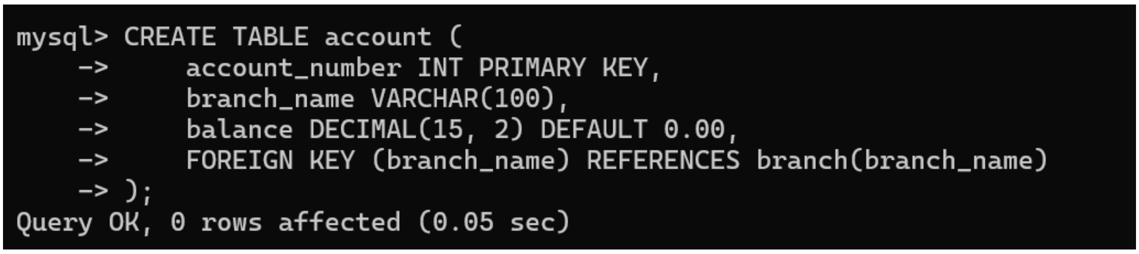
```
mysql> CREATE TABLE branch (
    ->     branch_name VARCHAR(50) PRIMARY KEY,
    ->     branch_city VARCHAR(50) NOT NULL,
    ->     assets DECIMAL(15, 2)
    -> );
Query OK, 0 rows affected (0.06 sec)
```

  - 建立customer表



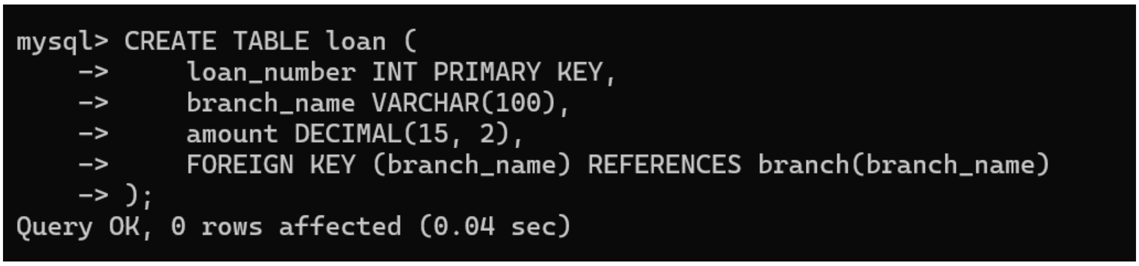
```
mysql> CREATE TABLE customer (
    ->     ID INT PRIMARY KEY,
    ->     customer_name VARCHAR(100) NOT NULL,
    ->     customer_street VARCHAR(100),
    ->     customer_city VARCHAR(100)
    -> );
Query OK, 0 rows affected (0.04 sec)
```

  - 建立account表



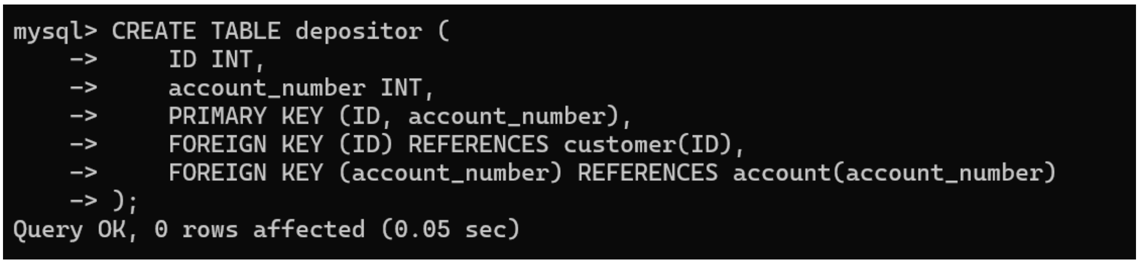
```
mysql> CREATE TABLE account (
    ->     account_number INT PRIMARY KEY,
    ->     branch_name VARCHAR(100),
    ->     balance DECIMAL(15, 2) DEFAULT 0.00,
    ->     FOREIGN KEY (branch_name) REFERENCES branch(branch_name)
    -> );
Query OK, 0 rows affected (0.05 sec)
```

  - 建立loan表



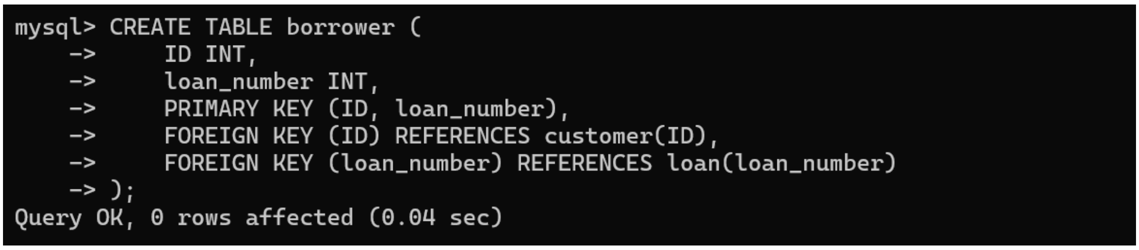
```
mysql> CREATE TABLE loan (
    ->     loan_number INT PRIMARY KEY,
    ->     branch_name VARCHAR(100),
    ->     amount DECIMAL(15, 2),
    ->     FOREIGN KEY (branch_name) REFERENCES branch(branch_name)
    -> );
Query OK, 0 rows affected (0.04 sec)
```

  - 建立depositor表



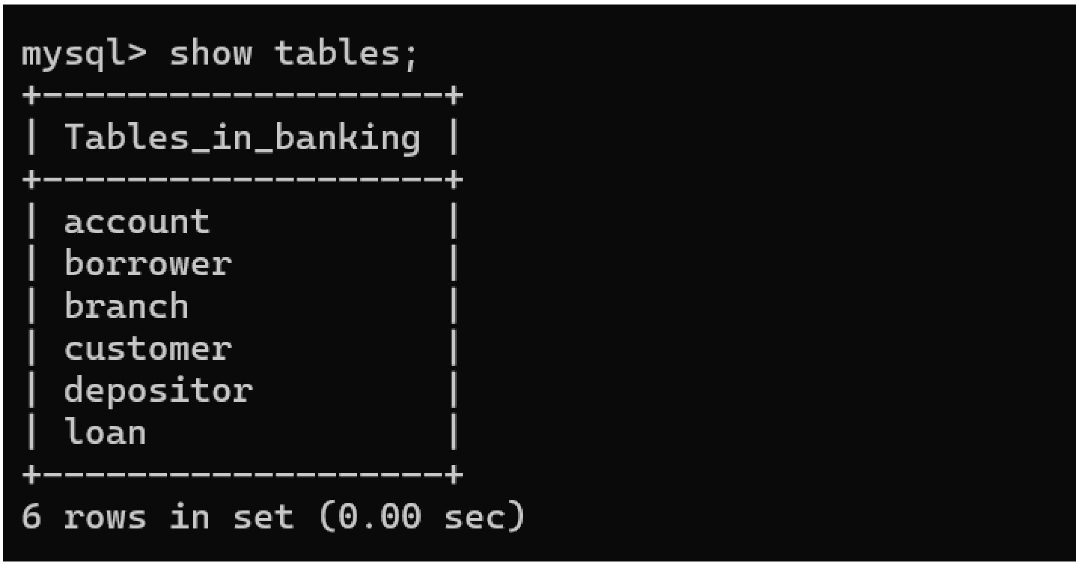
```
mysql> CREATE TABLE depositor (
    ->     ID INT,
    ->     account_number INT,
    ->     PRIMARY KEY (ID, account_number),
    ->     FOREIGN KEY (ID) REFERENCES customer(ID),
    ->     FOREIGN KEY (account_number) REFERENCES account(account_number)
    -> );
Query OK, 0 rows affected (0.05 sec)
```

  - 建立borrower表



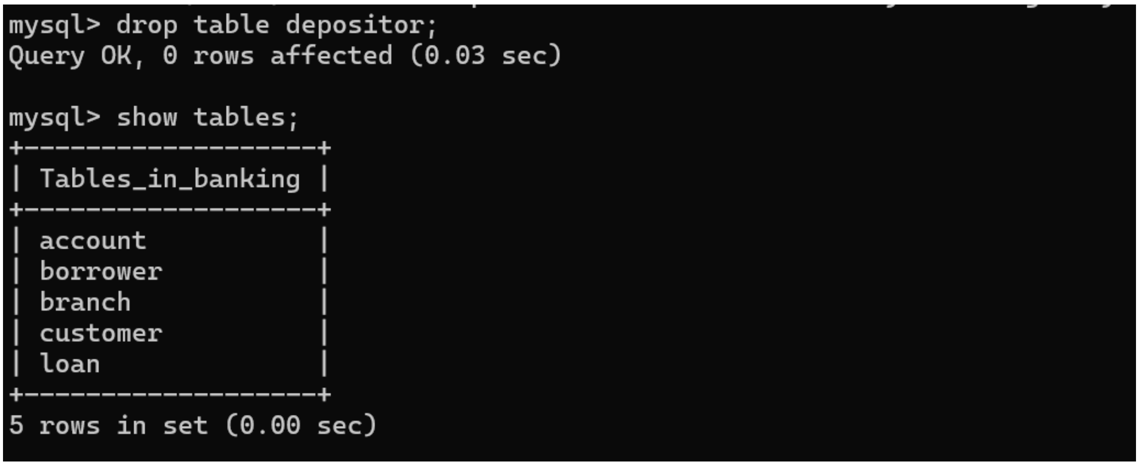
```
mysql> CREATE TABLE borrower (
    ->     ID INT,
    ->     loan_number INT,
    ->     PRIMARY KEY (ID, loan_number),
    ->     FOREIGN KEY (ID) REFERENCES customer(ID),
    ->     FOREIGN KEY (loan_number) REFERENCES loan(loan_number)
    -> );
Query OK, 0 rows affected (0.04 sec)
```

  - 查询表



```
mysql> show tables;
+-------------------+
| Tables_in_banking |
+-------------------+
| account           |
| borrower          |
| branch            |
| customer          |
| depositor         |
| loan              |
+-------------------+
6 rows in set (0.00 sec)
```

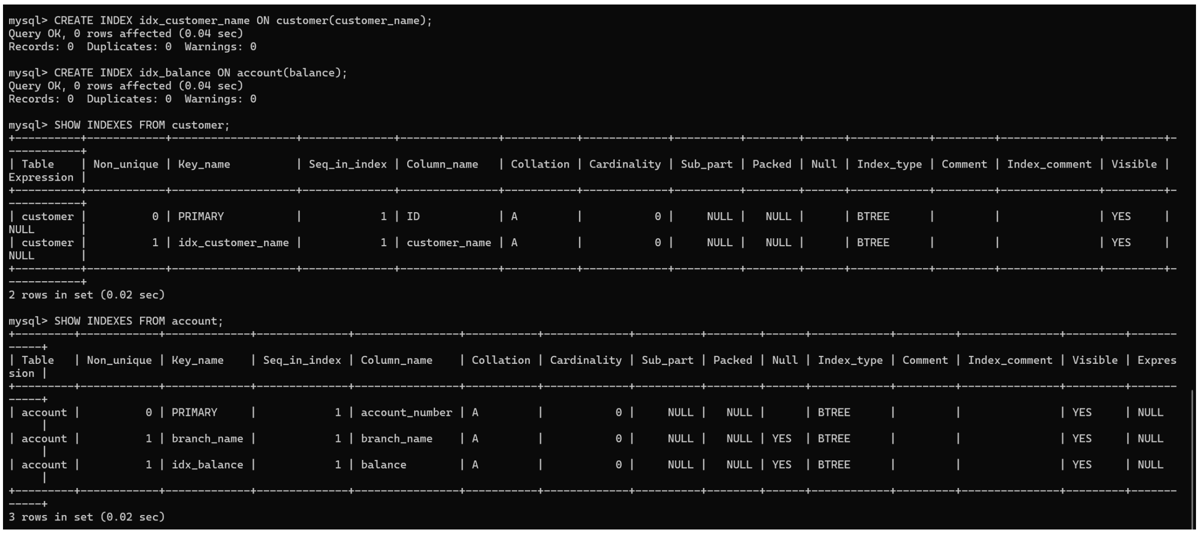
- 表的删除
  - 删除depositor表



```
mysql> drop table depositor;
Query OK, 0 rows affected (0.03 sec)

mysql> show tables;
+-------------------+
| Tables_in_banking |
+-------------------+
| account           |
| borrower          |
| branch            |
| customer          |
| loan              |
+-------------------+
5 rows in set (0.00 sec)
```

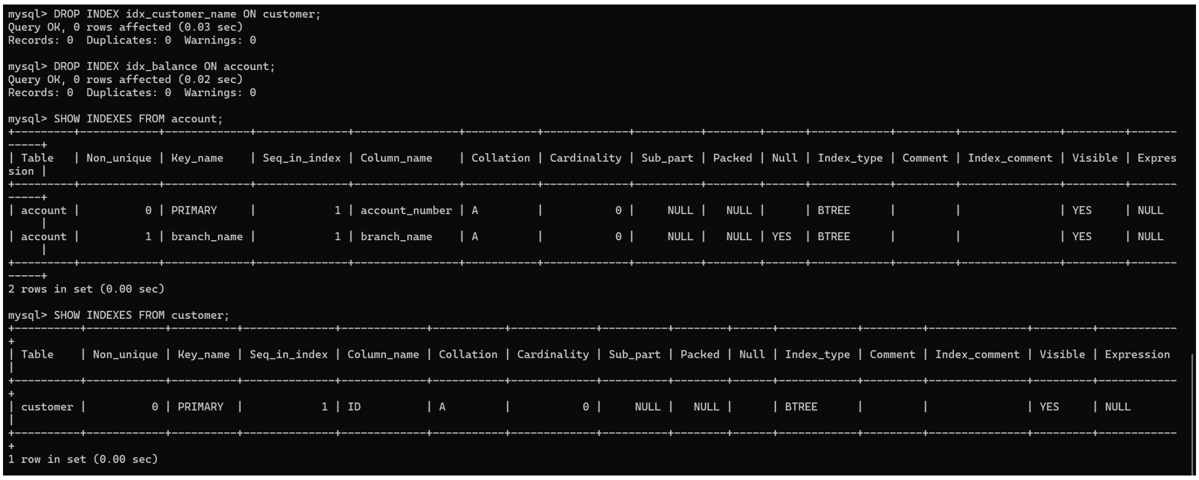
- 索引的建立
    - 在 customer 表的 customer_name 列上创建索引
    - 在 account 表的 balance 列上创建索引

可以看到被创建索引的列的key_name变为索引名称



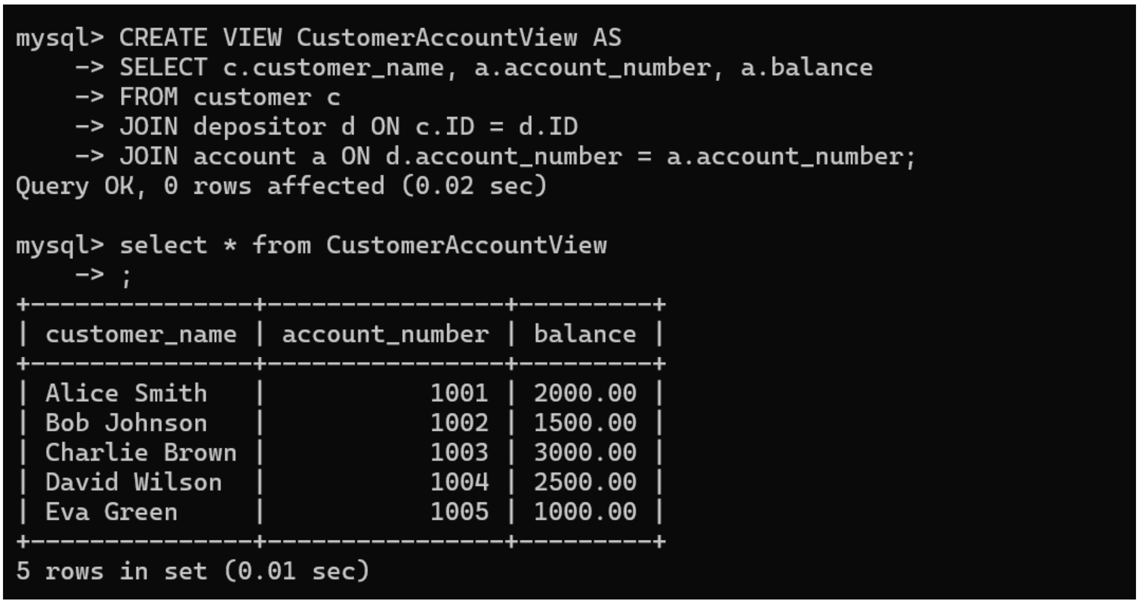
```
mysql> CREATE INDEX idx_customer_name ON customer(customer_name);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> CREATE INDEX idx_balance ON account(balance);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> SHOW INDEXES FROM customer;
+----------+------------+-------------------+--------------+---------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+-
-----------+
| Table    | Non_unique | Key_name          | Seq_in_index | Column_name   | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible |
Expression |
+----------+------------+-------------------+--------------+---------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+-
-----------+
| customer |          0 | PRIMARY           |            1 | ID            | A         |           0 |     NULL |   NULL |      | BTREE      |         |               | YES     |
NULL       |
| customer |          1 | idx_customer_name |            1 | customer_name | A         |           0 |     NULL |   NULL |      | BTREE      |         |               | YES     |
NULL       |
+----------+------------+-------------------+--------------+---------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+-
-----------+
2 rows in set (0.02 sec)

mysql> SHOW INDEXES FROM account;
+---------+------------+-------------+--------------+----------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+-------
-----+
| Table   | Non_unique | Key_name    | Seq_in_index | Column_name    | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expres
sion |
+---------+------------+-------------+--------------+----------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+-------
-----+
| account |          0 | PRIMARY     |            1 | account_number | A         |           0 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL
     |
| account |          1 | branch_name |            1 | branch_name    | A         |           0 |     NULL |   NULL | YES  | BTREE      |         |               | YES     | NULL
     |
| account |          1 | idx_balance |            1 | balance        | A         |           0 |     NULL |   NULL | YES  | BTREE      |         |               | YES     | NULL
     |
+---------+------------+-------------+--------------+----------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+-------
-----+
3 rows in set (0.02 sec)
```

- 索引的删除
    - 删除 customer 表上的 idx_customer_name 索引
    - 删除 account 表上的 idx_balance 索引



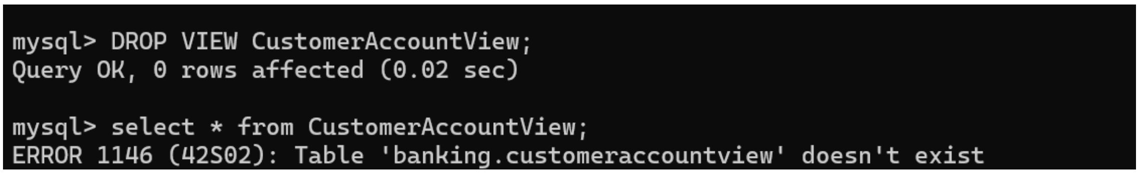
```
mysql> DROP INDEX idx_customer_name ON customer;
Query OK, 0 rows affected (0.03 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> DROP INDEX idx_balance ON account;
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> SHOW INDEXES FROM account;
+---------+------------+-------------+--------------+----------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+-------
-----+
| Table   | Non_unique | Key_name    | Seq_in_index | Column_name    | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expres
sion |
+---------+------------+-------------+--------------+----------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+-------
-----+
| account |          0 | PRIMARY     |            1 | account_number | A         |           0 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL
     |
| account |          1 | branch_name |            1 | branch_name    | A         |           0 |     NULL |   NULL | YES  | BTREE      |         |               | YES     | NULL
     |
+---------+------------+-------------+--------------+----------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+-------
-----+
2 rows in set (0.00 sec)

mysql> SHOW INDEXES FROM customer;
+----------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------
+
| Table    | Non_unique | Key_name | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression
|
+----------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------
+
| customer |          0 | PRIMARY  |            1 | ID          | A         |           0 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL
|
+----------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------
+
1 row in set (0.00 sec)
```

- 视图的建立
    - 创建客户及其账户信息的视图



```
mysql> CREATE VIEW CustomerAccountView AS
    -> SELECT c.customer_name, a.account_number, a.balance
    -> FROM customer c
    -> JOIN depositor d ON c.ID = d.ID
    -> JOIN account a ON d.account_number = a.account_number;
Query OK, 0 rows affected (0.02 sec)

mysql> select * from CustomerAccountView
    -> ;
+---------------+----------------+---------+
| customer_name | account_number | balance |
+---------------+----------------+---------+
| Alice Smith   |           1001 | 2000.00 |
| Bob Johnson   |           1002 | 1500.00 |
| Charlie Brown |           1003 | 3000.00 |
| David Wilson  |           1004 | 2500.00 |
| Eva Green     |           1005 | 1000.00 |
+---------------+----------------+---------+
5 rows in set (0.01 sec)
```

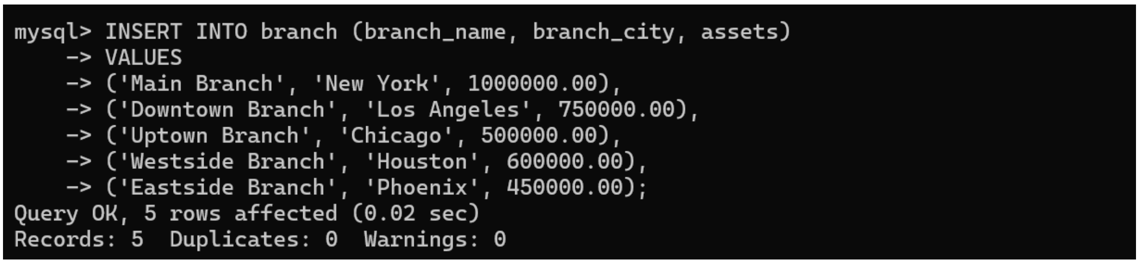
- 视图的删除
  - 删除客户及其账户信息的视图



```
mysql> DROP VIEW CustomerAccountView;
Query OK, 0 rows affected (0.02 sec)

mysql> select * from CustomerAccountView;
ERROR 1146 (42S02): Table 'banking.customeraccountview' doesn't exist
```

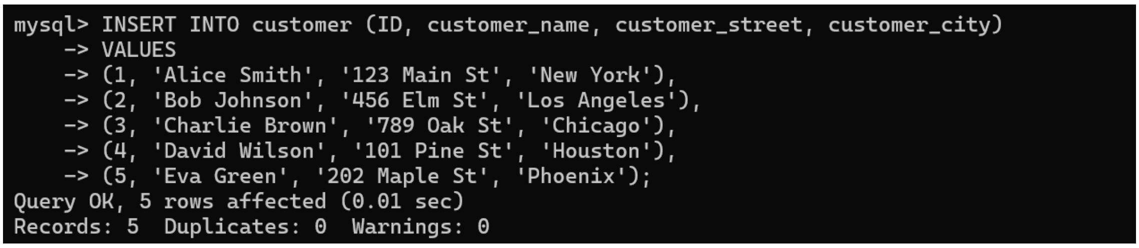
3.数据更新

- 用insert插入表数据
  - 插入 branch 表数据



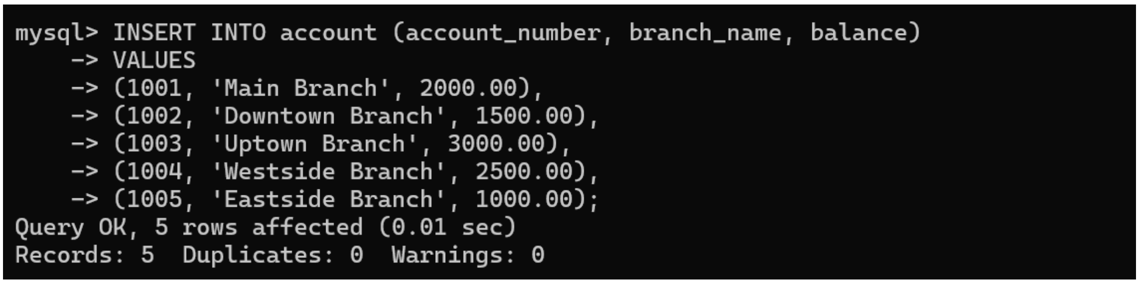
```
mysql> INSERT INTO branch (branch_name, branch_city, assets)
    -> VALUES
    -> ('Main Branch', 'New York', 1000000.00),
    -> ('Downtown Branch', 'Los Angeles', 750000.00),
    -> ('Uptown Branch', 'Chicago', 500000.00),
    -> ('Westside Branch', 'Houston', 600000.00),
    -> ('Eastside Branch', 'Phoenix', 450000.00);
Query OK, 5 rows affected (0.02 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

  - 插入 customer 表数据



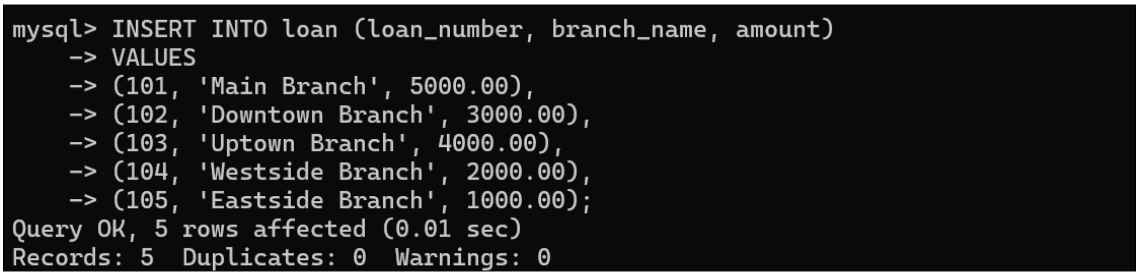
```
mysql> INSERT INTO customer (ID, customer_name, customer_street, customer_city)
    -> VALUES
    -> (1, 'Alice Smith', '123 Main St', 'New York'),
    -> (2, 'Bob Johnson', '456 Elm St', 'Los Angeles'),
    -> (3, 'Charlie Brown', '789 Oak St', 'Chicago'),
    -> (4, 'David Wilson', '101 Pine St', 'Houston'),
    -> (5, 'Eva Green', '202 Maple St', 'Phoenix');
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

  - 插入 account 表数据



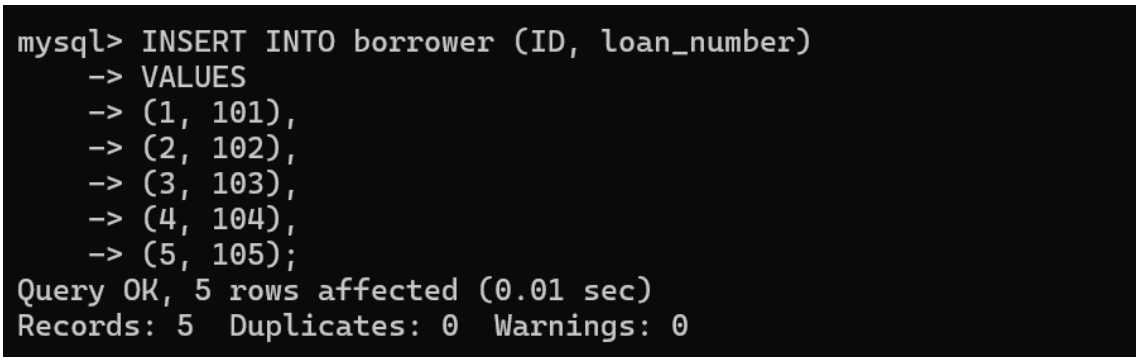
```
mysql> INSERT INTO account (account_number, branch_name, balance)
    -> VALUES
    -> (1001, 'Main Branch', 2000.00),
    -> (1002, 'Downtown Branch', 1500.00),
    -> (1003, 'Uptown Branch', 3000.00),
    -> (1004, 'Westside Branch', 2500.00),
    -> (1005, 'Eastside Branch', 1000.00);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

  - 插入 loan 表数据



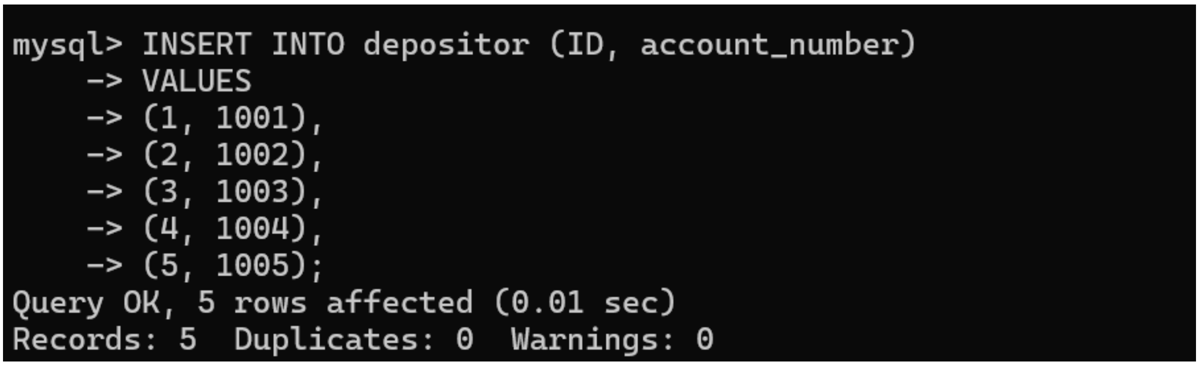
```
mysql> INSERT INTO loan (loan_number, branch_name, amount)
    -> VALUES
    -> (101, 'Main Branch', 5000.00),
    -> (102, 'Downtown Branch', 3000.00),
    -> (103, 'Uptown Branch', 4000.00),
    -> (104, 'Westside Branch', 2000.00),
    -> (105, 'Eastside Branch', 1000.00);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

  - 插入 borrower 表数据



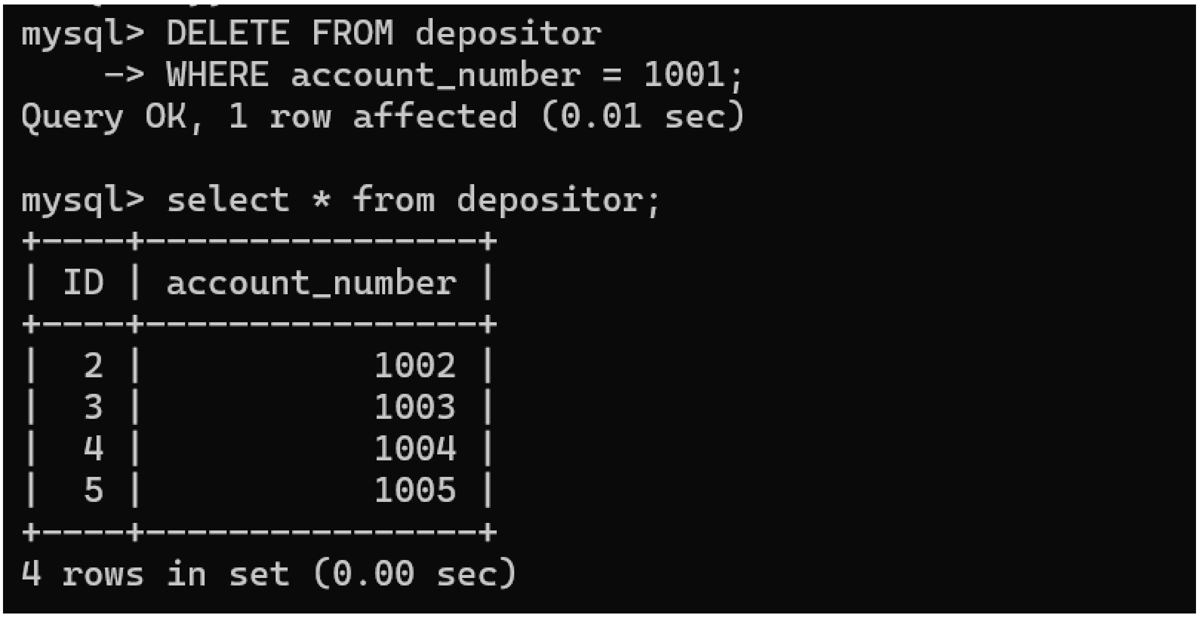
```
mysql> INSERT INTO borrower (ID, loan_number)
    -> VALUES
    -> (1, 101),
    -> (2, 102),
    -> (3, 103),
    -> (4, 104),
    -> (5, 105);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

  - 插入 depositor 表数据



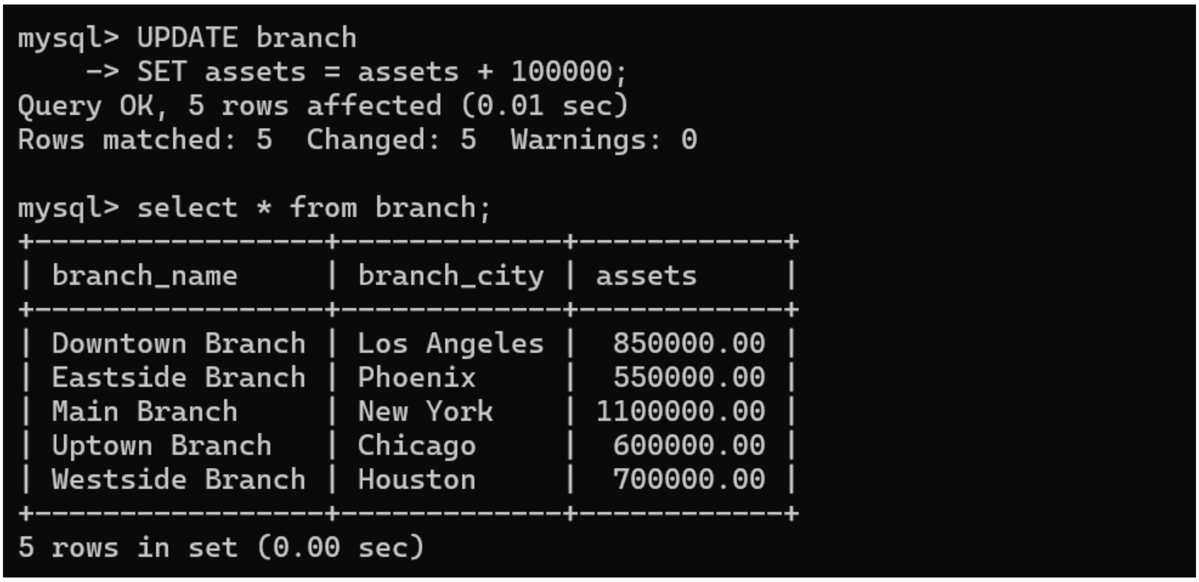
```
mysql> INSERT INTO depositor (ID, account_number)
    -> VALUES
    -> (1, 1001),
    -> (2, 1002),
    -> (3, 1003),
    -> (4, 1004),
    -> (5, 1005);
Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

- 用delete删除表数据
  - 删除 depositor 表中账户编号为 1001 的记录



```
mysql> DELETE FROM depositor
    -> WHERE account_number = 1001;
Query OK, 1 row affected (0.01 sec)

mysql> select * from depositor;
+----+----------------+
| ID | account_number |
+----+----------------+
|  2 |           1002 |
|  3 |           1003 |
|  4 |           1004 |
|  5 |           1005 |
+----+----------------+
4 rows in set (0.00 sec)
```

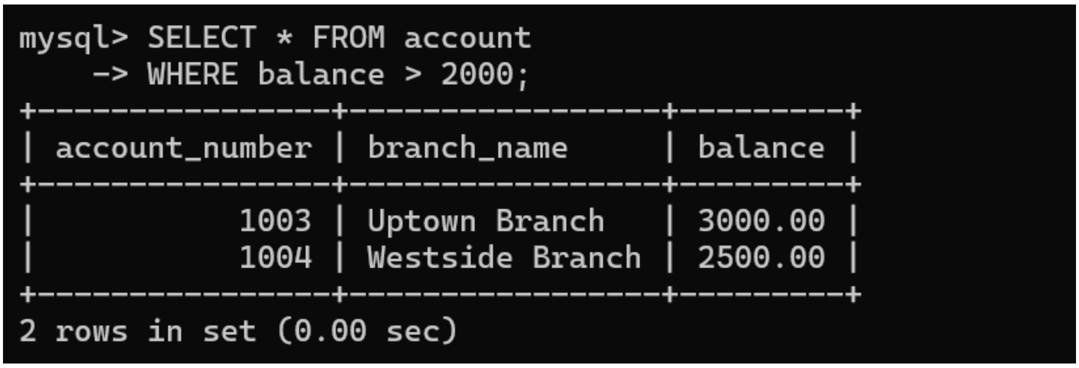
- 用update更新表数据
  - 更新 branch 表中所有分行的资产



```
mysql> UPDATE branch
    -> SET assets = assets + 100000;
Query OK, 5 rows affected (0.01 sec)
Rows matched: 5  Changed: 5  Warnings: 0

mysql> select * from branch;
+------------------+-------------+------------+
| branch_name      | branch_city | assets     |
+------------------+-------------+------------+
| Downtown Branch  | Los Angeles |  850000.00 |
| Eastside Branch  | Phoenix     |  550000.00 |
| Main Branch      | New York    | 1100000.00 |
| Uptown Branch    | Chicago     |  600000.00 |
| Westside Branch  | Houston     |  700000.00 |
+------------------+-------------+------------+
5 rows in set (0.00 sec)
```

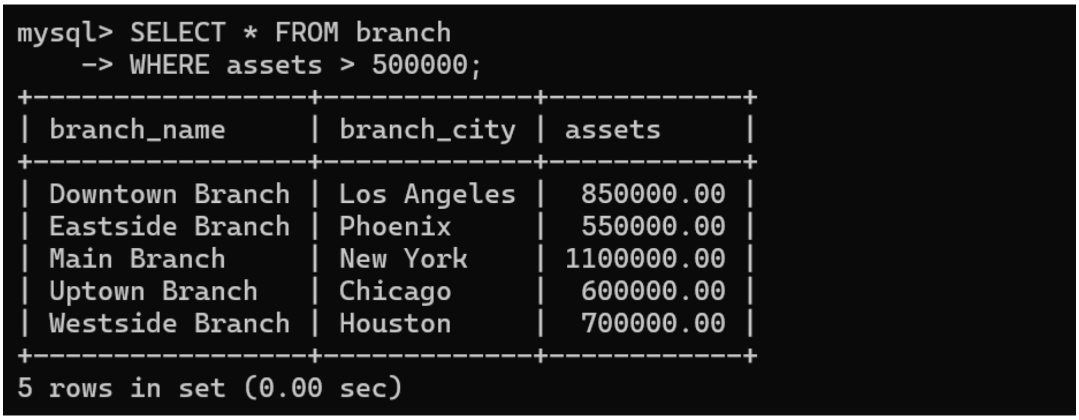
4.数据查询

- 单表查询
  - 查询 account 表中余额大于 2000 的账户



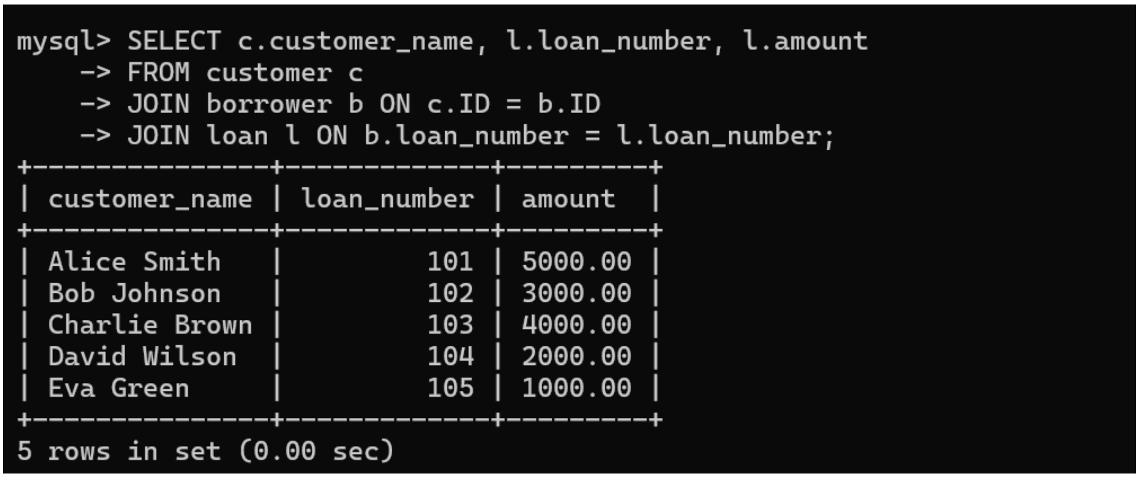
```
mysql> SELECT * FROM account
    -> WHERE balance > 2000;
+----------------+-----------------+---------+
| account_number | branch_name     | balance |
+----------------+-----------------+---------+
|           1003 | Uptown Branch   | 3000.00 |
|           1004 | Westside Branch | 2500.00 |
+----------------+-----------------+---------+
2 rows in set (0.00 sec)
```

  - 查询 branch 表中资产大于 500000 的分行



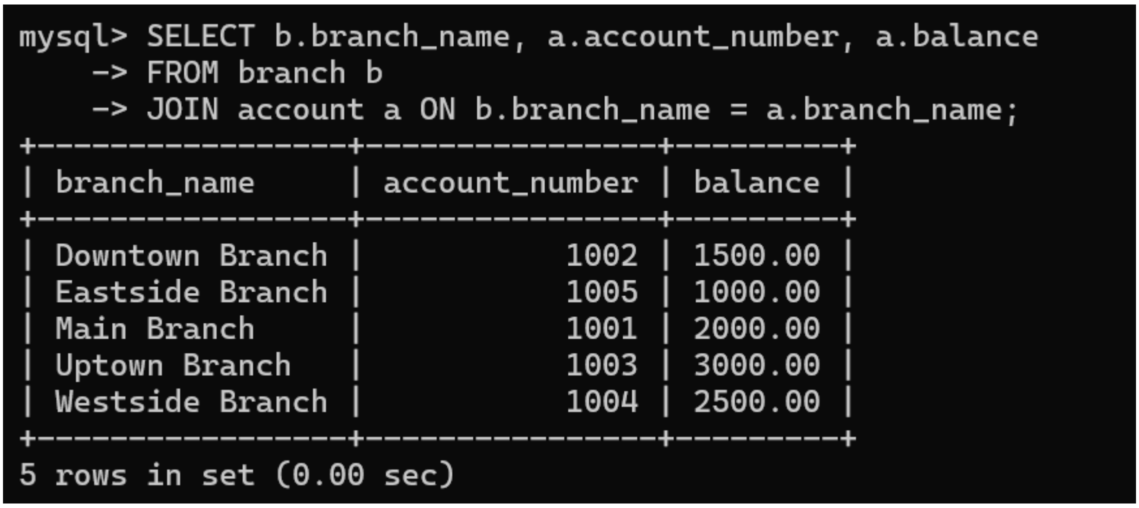
```
mysql> SELECT * FROM branch
    -> WHERE assets > 500000;
+-----------------+-------------+------------+
| branch_name     | branch_city | assets     |
+-----------------+-------------+------------+
| Downtown Branch | Los Angeles |  850000.00 |
| Eastside Branch | Phoenix     |  550000.00 |
| Main Branch     | New York    | 1100000.00 |
| Uptown Branch   | Chicago     |  600000.00 |
| Westside Branch | Houston     |  700000.00 |
+-----------------+-------------+------------+
5 rows in set (0.00 sec)
```

- 多表查询
  - 查询客户及其贷款信息



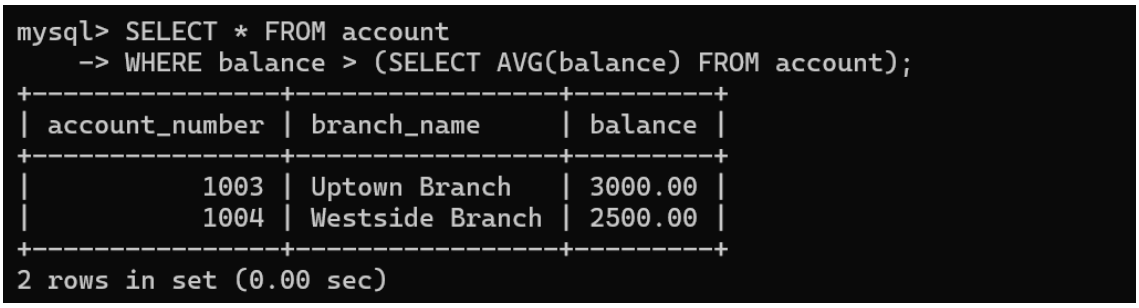
```
mysql> SELECT c.customer_name, l.loan_number, l.amount
    -> FROM customer c
    -> JOIN borrower b ON c.ID = b.ID
    -> JOIN loan l ON b.loan_number = l.loan_number;
+---------------+-------------+---------+
| customer_name | loan_number | amount  |
+---------------+-------------+---------+
| Alice Smith   |         101 | 5000.00 |
| Bob Johnson   |         102 | 3000.00 |
| Charlie Brown |         103 | 4000.00 |
| David Wilson  |         104 | 2000.00 |
| Eva Green     |         105 | 1000.00 |
+---------------+-------------+---------+
5 rows in set (0.00 sec)
```

  - 查询分行及其账户信息



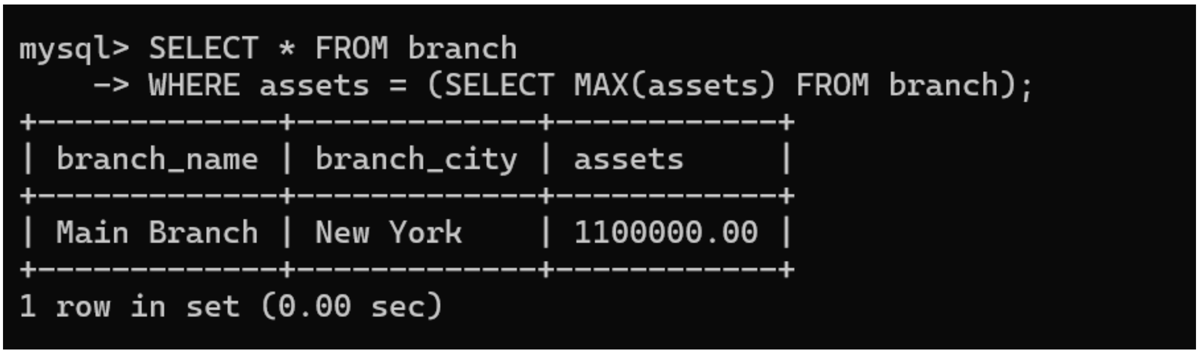
```
mysql> SELECT b.branch_name, a.account_number, a.balance
    -> FROM branch b
    -> JOIN account a ON b.branch_name = a.branch_name;
+-----------------+----------------+---------+
| branch_name     | account_number | balance |
+-----------------+----------------+---------+
| Downtown Branch |           1002 | 1500.00 |
| Eastside Branch |           1005 | 1000.00 |
| Main Branch     |           1001 | 2000.00 |
| Uptown Branch   |           1003 | 3000.00 |
| Westside Branch |           1004 | 2500.00 |
+-----------------+----------------+---------+
5 rows in set (0.00 sec)
```

- 嵌套子查询
  - 查询余额高于平均余额的账户



```
mysql> SELECT * FROM account
    -> WHERE balance > (SELECT AVG(balance) FROM account);
+----------------+-----------------+---------+
| account_number | branch_name     | balance |
+----------------+-----------------+---------+
|           1003 | Uptown Branch   | 3000.00 |
|           1004 | Westside Branch | 2500.00 |
+----------------+-----------------+---------+
2 rows in set (0.00 sec)
```

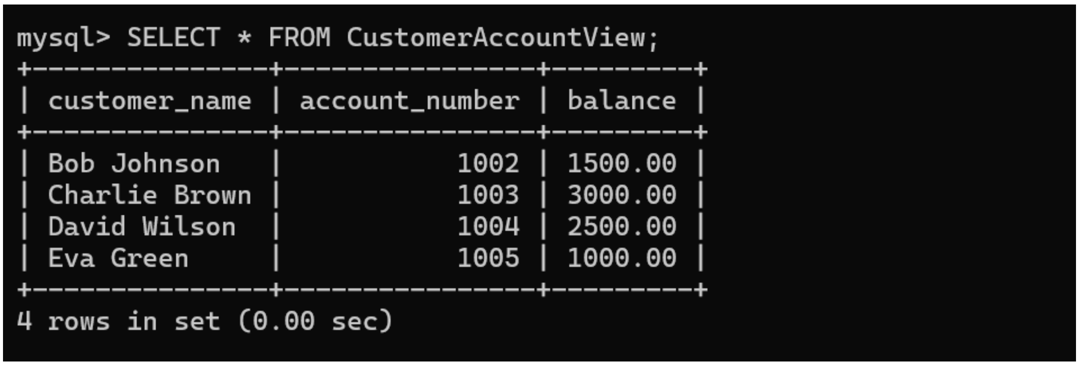
  - 查询资产最高的分行



```
mysql> SELECT * FROM branch
    -> WHERE assets = (SELECT MAX(assets) FROM branch);
+-------------+-------------+------------+
| branch_name | branch_city | assets     |
+-------------+-------------+------------+
| Main Branch | New York    | 1100000.00 |
+-------------+-------------+------------+
1 row in set (0.00 sec)
```

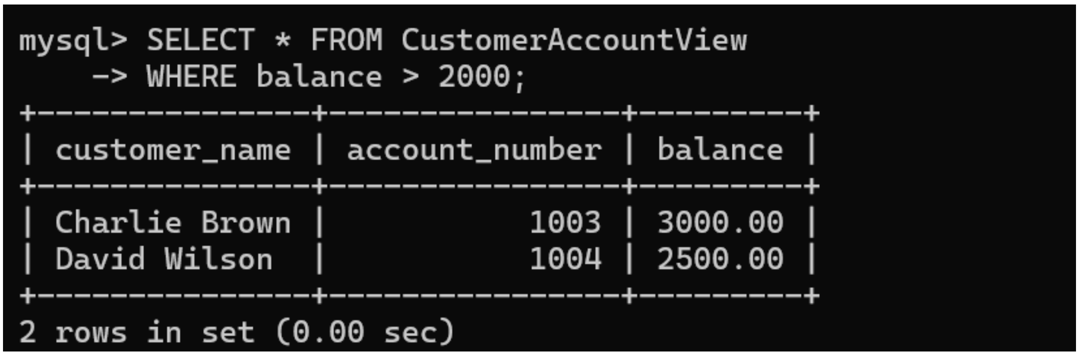
5.视图操作

- 通过视图进行数据查询
  - 假设有一个视图 CustomerAccountView，显示客户及其账户信息：



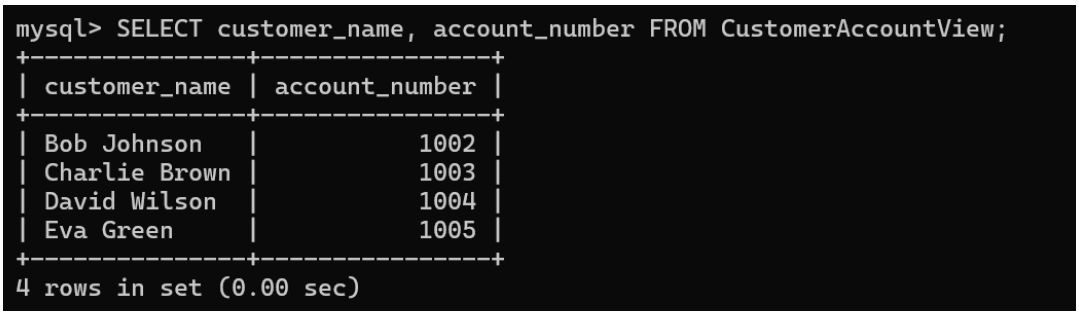
```
mysql> SELECT * FROM CustomerAccountView;
+---------------+----------------+---------+
| customer_name | account_number | balance |
+---------------+----------------+---------+
| Bob Johnson   |           1002 | 1500.00 |
| Charlie Brown |           1003 | 3000.00 |
| David Wilson  |           1004 | 2500.00 |
| Eva Green     |           1005 | 1000.00 |
+---------------+----------------+---------+
4 rows in set (0.00 sec)
```

    - 查询余额大于 2000 的客户账户信息：



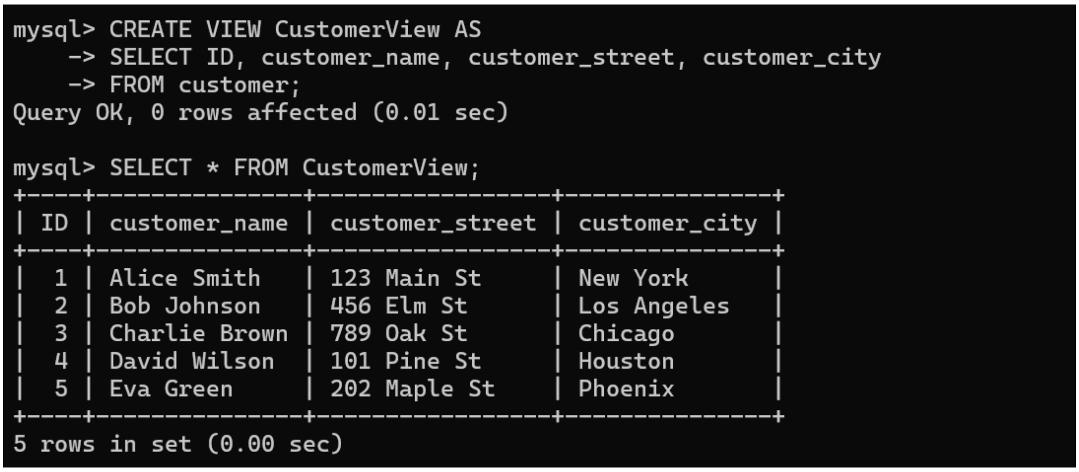
```
mysql> SELECT * FROM CustomerAccountView
    -> WHERE balance > 2000;
+---------------+----------------+---------+
| customer_name | account_number | balance |
+---------------+----------------+---------+
| Charlie Brown |           1003 | 3000.00 |
| David Wilson  |           1004 | 2500.00 |
+---------------+----------------+---------+
2 rows in set (0.00 sec)
```

    - 查询客户姓名和账户编号：



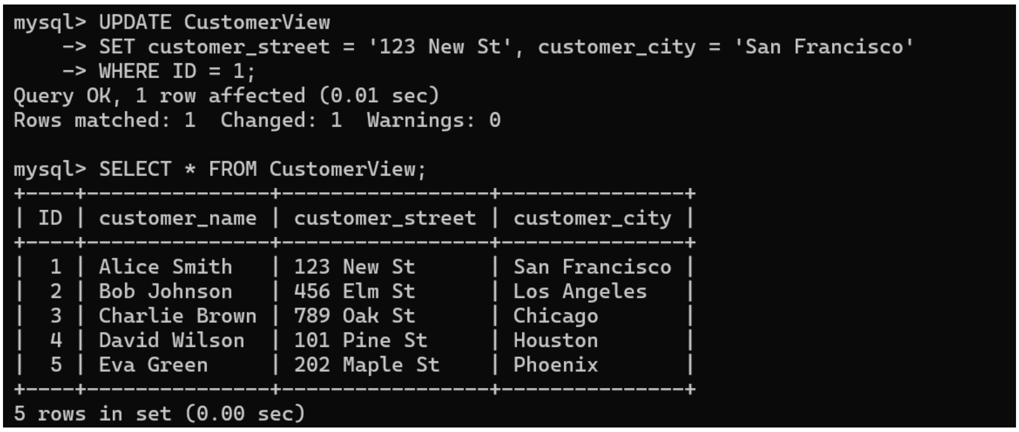
```
mysql> SELECT customer_name, account_number FROM CustomerAccountView;
+---------------+----------------+
| customer_name | account_number |
+---------------+----------------+
| Bob Johnson   |           1002 |
| Charlie Brown |           1003 |
| David Wilson  |           1004 |
| Eva Green     |           1005 |
+---------------+----------------+
4 rows in set (0.00 sec)
```

- 通过视图进行数据更新
  - 假设有一个简单的视图 CustomerView，基于 customer 表：

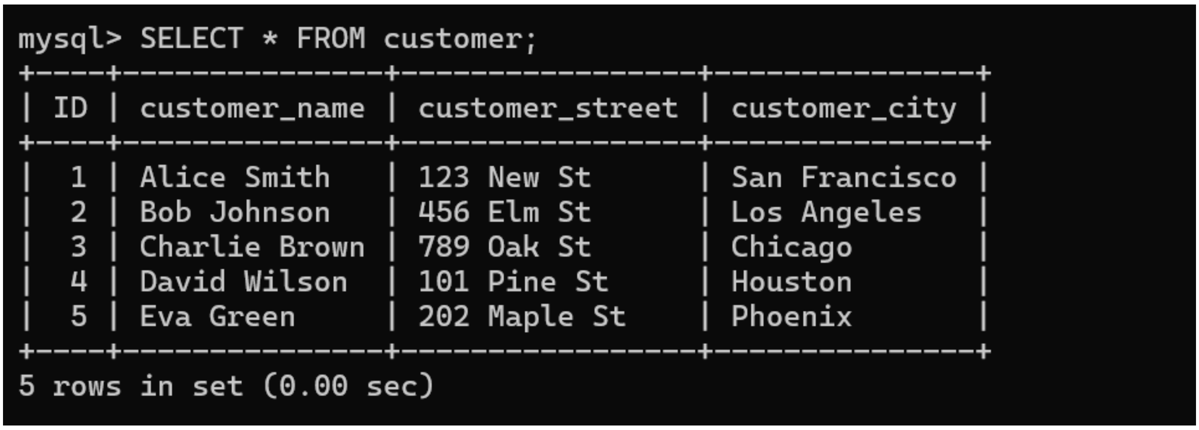


```
mysql> CREATE VIEW CustomerView AS
    -> SELECT ID, customer_name, customer_street, customer_city
    -> FROM customer;
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT * FROM CustomerView;
+----+---------------+-----------------+---------------+
| ID | customer_name | customer_street | customer_city |
+----+---------------+-----------------+---------------+
|  1 | Alice Smith   | 123 Main St     | New York      |
|  2 | Bob Johnson   | 456 Elm St      | Los Angeles   |
|  3 | Charlie Brown | 789 Oak St      | Chicago       |
|  4 | David Wilson  | 101 Pine St     | Houston       |
|  5 | Eva Green     | 202 Maple St    | Phoenix       |
+----+---------------+-----------------+---------------+
5 rows in set (0.00 sec)
```

    - 更新 CustomerView 视图中的客户地址：



```
mysql> UPDATE CustomerView
    -> SET customer_street = '123 New St', customer_city = 'San Francisco'
    -> WHERE ID = 1;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM CustomerView;
+----+---------------+-----------------+---------------+
| ID | customer_name | customer_street | customer_city |
+----+---------------+-----------------+---------------+
|  1 | Alice Smith   | 123 New St      | San Francisco |
|  2 | Bob Johnson   | 456 Elm St      | Los Angeles   |
|  3 | Charlie Brown | 789 Oak St      | Chicago       |
|  4 | David Wilson  | 101 Pine St     | Houston       |
|  5 | Eva Green     | 202 Maple St    | Phoenix       |
+----+---------------+-----------------+---------------+
5 rows in set (0.00 sec)
```



```
mysql> SELECT * FROM customer;
+----+---------------+-----------------+---------------+
| ID | customer_name | customer_street | customer_city |
+----+---------------+-----------------+---------------+
|  1 | Alice Smith   | 123 New St      | San Francisco |
|  2 | Bob Johnson   | 456 Elm St      | Los Angeles   |
|  3 | Charlie Brown | 789 Oak St      | Chicago       |
|  4 | David Wilson  | 101 Pine St     | Houston       |
|  5 | Eva Green     | 202 Maple St    | Phoenix       |
+----+---------------+-----------------+---------------+
5 rows in set (0.00 sec)
```