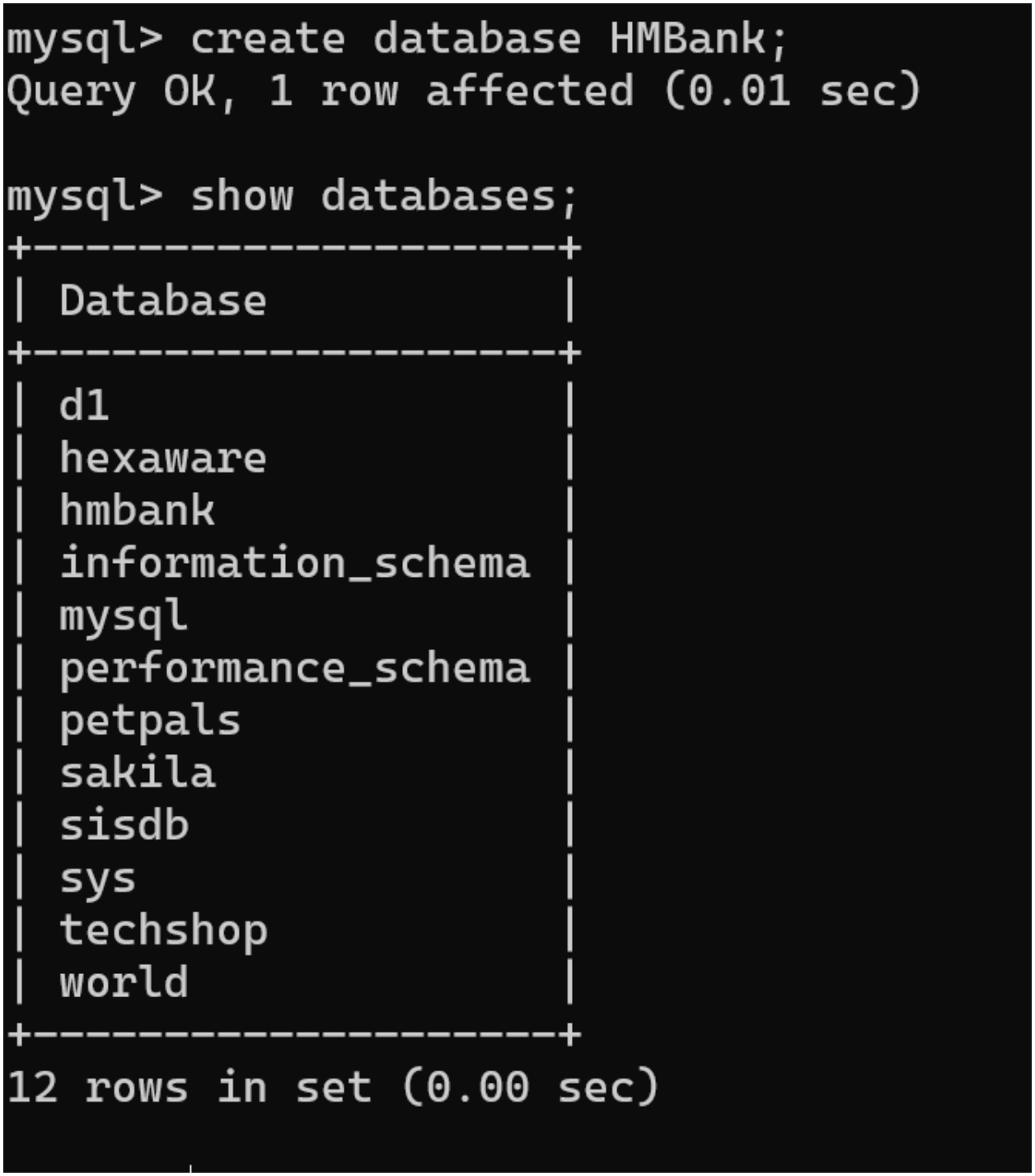
# Assignment sql-3

Task-1:Database creation

1. Create database “HMBank”



2.2. Define the schema for the Customers, Accounts, and Transactions tables based on the provided schema.

1.Customers

```
mysql> create table customers(customer_id int primary key,first_name text,last_name text,DOB date,email varchar(100),phone_number bigint);
Query OK, 0 rows affected (0.05 sec)

mysql> desc customers;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| customer_id  | int          | NO   | PRI | NULL    |       |
| first_name   | text         | YES  |     | NULL    |       |
| last_name    | text         | YES  |     | NULL    |       |
| DOB          | date         | YES  |     | NULL    |       |
| email        | varchar(100) | YES  |     | NULL    |       |
| phone_number | bigint       | YES  |     | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
6 rows in set (0.02 sec)
```

```
mysql> alter table customers add address varchar(200);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc customers;
+---------------+---------------+------+-----+---------+-------+
| Field         | Type          | Null | Key | Default | Extra |
+---------------+---------------+------+-----+---------+-------+
| customer_id   | int           | NO   | PRI | NULL    |       |
| first_name    | text          | YES  |     | NULL    |       |
| last_name     | text          | YES  |     | NULL    |       |
| DOB           | date          | YES  |     | NULL    |       |
| email         | varchar(100)  | YES  |     | NULL    |       |
| phone_number  | bigint        | YES  |     | NULL    |       |
| address       | varchar(200)  | YES  |     | NULL    |       |
+---------------+---------------+------+-----+---------+-------+
7 rows in set (0.00 sec)
```

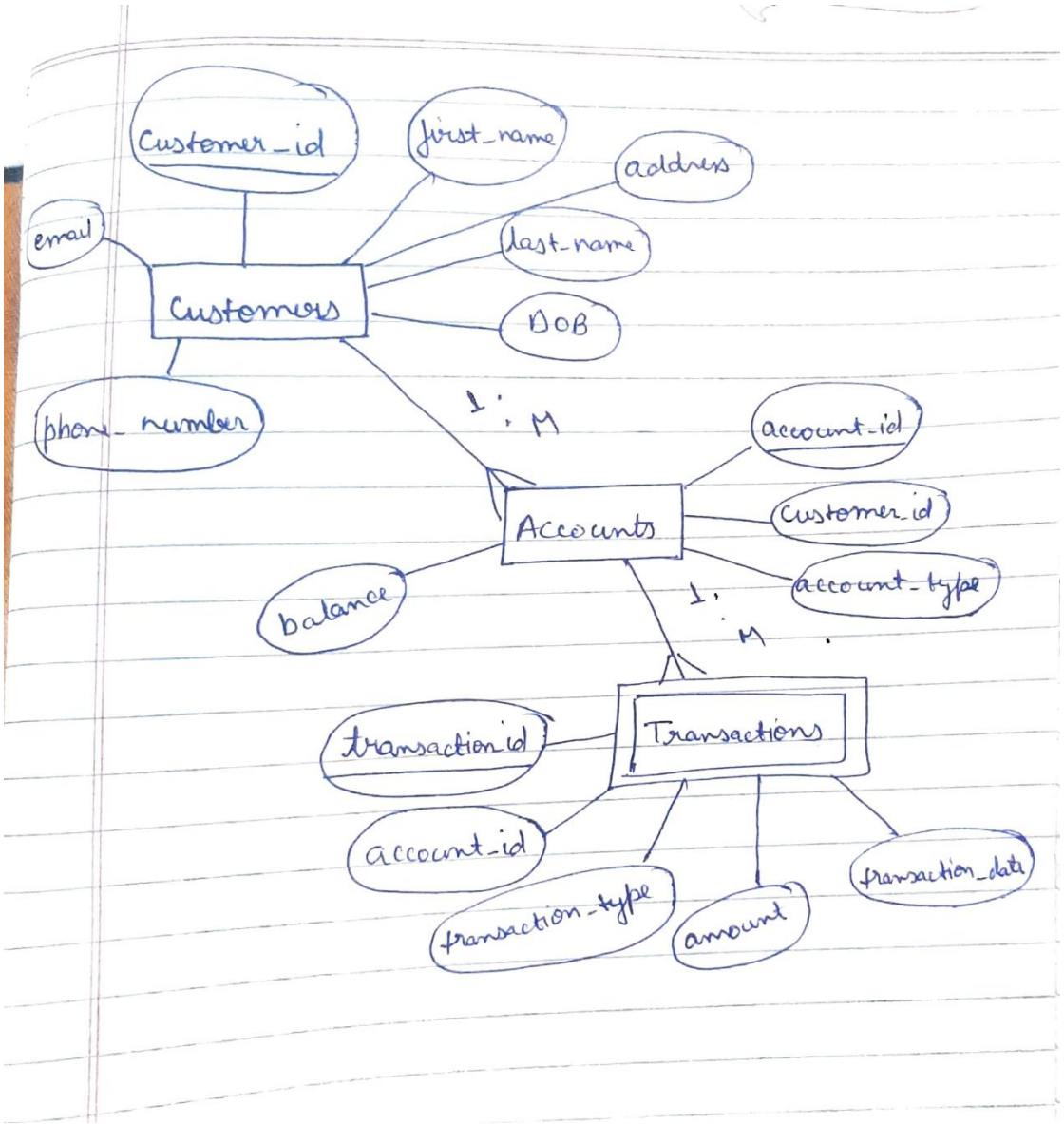
2. Accounts

```
ysql> create table accounts(account_id int primary key,customer_id int,foreign key(customer_id) references customers(customer_id),account_type te
t,balance bigint);
uery OK, 0 rows affected (0.07 sec)

ysql> desc accounts;
+--------------+--------+------+-----+---------+-------+
| Field        | Type   | Null | Key | Default | Extra |
+--------------+--------+------+-----+---------+-------+
| account_id   | int    | NO   | PRI | NULL    |       |
| customer_id  | int    | YES  | MUL | NULL    |       |
| account_type | text   | YES  |     | NULL    |       |
| balance      | bigint | YES  |     | NULL    |       |
+--------------+--------+------+-----+---------+-------+
 rows in set (0.00 sec)
```

3. Transactions

```
mysql> create table transactions(transaction_id int primary key,account_id int,foreign key(account_id) references accounts(account_id),transaction
_type text,amount bigint,transaction_date date);
Query OK, 0 rows affected (0.07 sec)

mysql> desc transactions;
+------------------+--------+------+-----+---------+-------+
| Field            | Type   | Null | Key | Default | Extra |
+------------------+--------+------+-----+---------+-------+
| transaction_id   | int    | NO   | PRI | NULL    |       |
| account_id       | int    | YES  | MUL | NULL    |       |
| transaction_type | text   | YES  |     | NULL    |       |
| amount           | bigint | YES  |     | NULL    |       |
| transaction_date | date   | YES  |     | NULL    |       |
+------------------+--------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

4.create erd for database



5.create primary and foreign key

Already done above

6.write sql script for creating table customers ,amounts and transactions

```
mysql> alter table customers add address varchar(200);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc customers;
+---------------+---------------+------+-----+---------+-------+
| Field         | Type          | Null | Key | Default | Extra |
+---------------+---------------+------+-----+---------+-------+
| customer_id   | int           | NO   | PRI | NULL    |       |
| first_name    | text          | YES  |     | NULL    |       |
| last_name     | text          | YES  |     | NULL    |       |
| DOB           | date          | YES  |     | NULL    |       |
| email         | varchar(100)  | YES  |     | NULL    |       |
| phone_number  | bigint        | YES  |     | NULL    |       |
| address       | varchar(200)  | YES  |     | NULL    |       |
+---------------+---------------+------+-----+---------+-------+
7 rows in set (0.00 sec)
```

```
ysql> create table accounts(account_id int primary key,customer_id int,foreign key(customer_id) references customers(customer_id),account_type te
t,balance bigint);
uery OK, 0 rows affected (0.07 sec)

ysql> desc accounts;
+--------------+--------+------+-----+---------+-------+
| Field        | Type   | Null | Key | Default | Extra |
+--------------+--------+------+-----+---------+-------+
| account_id   | int    | NO   | PRI | NULL    |       |
| customer_id  | int    | YES  | MUL | NULL    |       |
| account_type | text   | YES  |     | NULL    |       |
| balance      | bigint | YES  |     | NULL    |       |
+--------------+--------+------+-----+---------+-------+
 rows in set (0.00 sec)
```

```
mysql> create table transactions(transaction_id int primary key,account_id int,foreign key(account_id) references accounts(account_id),transaction
_type text,amount bigint,transaction_date date);
Query OK, 0 rows affected (0.07 sec)

mysql> desc transactions;
+------------------+--------+------+-----+---------+-------+
| Field            | Type   | Null | Key | Default | Extra |
+------------------+--------+------+-----+---------+-------+
| transaction_id   | int    | NO   | PRI | NULL    |       |
| account_id       | int    | YES  | MUL | NULL    |       |
| transaction_type | text   | YES  |     | NULL    |       |
| amount           | bigint | YES  |     | NULL    |       |
| transaction_date | date   | YES  |     | NULL    |       |
+------------------+--------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

Task-2

1.insert 10 sample data in:

Customers

```
mysql> INSERT INTO customers VALUES
    -> (1, 'John', 'Doe', '1990-05-15', 'john@example.com', '1234567890', '123 Main St, City, Country'),
    -> (2, 'Alice', 'Smith', '1988-08-20', 'alice@example.com', '9876543210', '456 Elm St, City, Country'),
    -> (3, 'Bob', 'Johnson', '1995-02-10', 'bob@example.com', '7890123456', '789 Oak St, City, Country'),
    -> (4, 'Emma', 'Williams', '1992-11-25', 'emma@example.com', '3456789012', '567 Maple St, City, Country'),
    -> (5, 'Michael', 'Brown', '1985-09-30', 'michael@example.com', '2345678901', '890 Cedar St, City, Country'),
    -> (6, 'Sophia', 'Davis', '1998-04-18', 'sophia@example.com', '9012345678', '234 Pine St, City, Country'),
    -> (7, 'Daniel', 'Miller', '1993-07-05', 'daniel@example.com', '4567890123', '678 Birch St, City, Country'),
    -> (8, 'Olivia', 'Garcia', '1997-12-12', 'olivia@example.com', '6789012345', '901 Walnut St, City, Country'),
    -> (9, 'Ethan', 'Wilson', '1996-01-28', 'ethan@example.com', '1231231234', '345 Spruce St, City, Country'),
    -> (10, 'Ava', 'Martinez', '1994-06-08', 'ava@example.com', '8908908901', '789 Oak St, City, Country');
Query OK, 10 rows affected (0.02 sec)
Records: 10  Duplicates: 0  Warnings: 0
```

```
mysql> select * from customers;
+-------------+------------+-----------+------------+--------------------+--------------+------------------------------+
| customer_id | first_name | last_name | DOB        | email              | phone_number | address                      |
+-------------+------------+-----------+------------+--------------------+--------------+------------------------------+
|           1 | John       | Doe       | 1990-05-15 | john@example.com   |   1234567890 | 123 Main St, City, Country   |
|           2 | Alice      | Smith     | 1988-08-20 | alice@example.com  |   9876543210 | 456 Elm St, City, Country    |
|           3 | Bob        | Johnson   | 1995-02-10 | bob@example.com    |   7890123456 | 789 Oak St, City, Country    |
|           4 | Emma       | Williams  | 1992-11-25 | emma@example.com   |   3456789012 | 567 Maple St, City, Country  |
|           5 | Michael    | Brown     | 1985-09-30 | michael@example.com |   2345678901 | 890 Cedar St, City, Country  |
|           6 | Sophia     | Davis     | 1998-04-18 | sophia@example.com |   9012345678 | 234 Pine St, City, Country   |
|           7 | Daniel     | Miller    | 1993-07-05 | daniel@example.com |   4567890123 | 678 Birch St, City, Country  |
|           8 | Olivia     | Garcia    | 1997-12-12 | olivia@example.com |   6789012345 | 901 Walnut St, City, Country |
|           9 | Ethan      | Wilson    | 1996-01-28 | ethan@example.com  |   1231231234 | 345 Spruce St, City, Country |
|          10 | Ava        | Martinez  | 1994-06-08 | ava@example.com    |   8908908901 | 789 Oak St, City, Country    |
+-------------+------------+-----------+------------+--------------------+--------------+------------------------------+
10 rows in set (0.00 sec)
```

Accounts

```
mysql> insert into accounts values(1,1,'Current',20000),(2,1,'Saving',50000),(3,2,'Current',300),(4,3,'Saving',80000),(5,4,'zero_balance',0),(6,5,
'Current',100000),(7,5,'Saving',40000),(8,6,'Current',70000),(9,7,'Saving',90000),(10,7,'Current',30000);
Query OK, 10 rows affected (0.02 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> select * from accounts;
+------------+-------------+--------------+---------+
| account_id | customer_id | account_type | balance |
+------------+-------------+--------------+---------+
|          1 |           1 | Current      |   20000 |
|          2 |           1 | Saving       |   50000 |
|          3 |           2 | Current      |     300 |
|          4 |           3 | Saving       |   80000 |
|          5 |           4 | zero_balance |       0 |
|          6 |           5 | Current      |  100000 |
|          7 |           5 | Saving       |   40000 |
|          8 |           6 | Current      |   70000 |
|          9 |           7 | Saving       |   90000 |
|         10 |           7 | Current      |   30000 |
+------------+-------------+--------------+---------+
10 rows in set (0.00 sec)
```

Transactions

```
mysql> insert into transactions values(1,1,'Transfer',10000,'2023-11-02'),(2,3,'Withdrawal',200,'2022-03-02'),(3,2,'Transfer',20000,'2021-04-04'),
(4,1,'Deposite',10000,'2022-11-12'),(5,5,'Deposite',1000,'2023-02-02'),(6,10,'Withdrawal',2000,'2021-11-01'),(7,8,'Deposite',10000,'2022-10-11'),(
8,7,'Deposite',20000,'2023-10-05'),(9,8,'Transfer',4000,'2023-11-06'),(10,6,'Transfer',5000,'2023-01-01');
Query OK, 10 rows affected (0.01 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> select * from transactions;
+----------------+------------+------------------+--------+------------------+
| transaction_id | account_id | transaction_type | amount | transaction_date |
+----------------+------------+------------------+--------+------------------+
|              1 |          1 | Transfer         |  10000 | 2023-11-02       |
|              2 |          3 | Withdrawal       |    200 | 2022-03-02       |
|              3 |          2 | Transfer         |  20000 | 2021-04-04       |
|              4 |          1 | Deposite         |  10000 | 2022-11-12       |
|              5 |          5 | Deposite         |   1000 | 2023-02-02       |
|              6 |         10 | Withdrawal       |   2000 | 2021-11-01       |
|              7 |          8 | Deposite         |  10000 | 2022-10-11       |
|              8 |          7 | Deposite         |  20000 | 2023-10-05       |
|              9 |          8 | Transfer         |   4000 | 2023-11-06       |
|             10 |          6 | Transfer         |   5000 | 2023-01-01       |
+----------------+------------+------------------+--------+------------------+
10 rows in set (0.00 sec)
```

2.write sql query for following:

1.. Write a SQL query to retrieve the name, account type and email of all customers.

```
ly dependent on columns in GROUP BY clause; this is incompatible with sql_mode=only_fu
mysql> SELECT CONCAT(c.first_name, ' ', c.last_name) AS name, c.email, a.account_type
    -> FROM customers c
    -> LEFT JOIN accounts a ON a.customer_id = c.customer_id;
+---------------+---------------------+--------------+
| name          | email               | account_type |
+---------------+---------------------+--------------+
| John Doe      | john@example.com    | Current      |
| Alice Smith   | alice@example.com   | Current      |
| Bob Johnson   | bob@example.com     | Saving       |
| Emma Williams | emma@example.com    | zero_balance |
| Michael Brown | michael@example.com | Current      |
| Michael Brown | michael@example.com | Saving       |
| Sophia Davis  | sophia@example.com  | Current      |
| Daniel Miller | daniel@example.com  | Saving       |
| Daniel Miller | daniel@example.com  | Current      |
| Olivia Garcia | olivia@example.com  | NULL         |
| Ethan Wilson  | ethan@example.com   | NULL         |
| Ava Martinez  | ava@example.com     | NULL         |
+---------------+---------------------+--------------+
12 rows in set (0.00 sec)
```

2.Write a SQL query to list all transaction corresponding customer.

```
mysql> SELECT c.customer_id, c.first_name, c.last_name, c.email, t.transaction_id, t.transaction_type, t.amount, t.transaction_dat
    -> FROM customers c
    -> JOIN accounts a ON c.customer_id = a.customer_id
    -> JOIN transactions t ON a.account_id = t.account_id;
+-------------+------------+-----------+---------------------+----------------+------------------+--------+------------------+
| customer_id | first_name | last_name | email               | transaction_id | transaction_type | amount | transaction_date |
+-------------+------------+-----------+---------------------+----------------+------------------+--------+------------------+
|           1 | John       | Doe       | john@example.com    |              1 | Transfer         |  10000 | 2023-11-02       |
|           2 | Alice      | Smith     | alice@example.com   |              2 | Withdrawal       |    200 | 2022-03-02       |
|           1 | John       | Doe       | john@example.com    |              4 | Deposite         |  10000 | 2022-11-12       |
|           4 | Emma       | Williams  | emma@example.com    |              5 | Deposite         |   1000 | 2023-02-02       |
|           7 | Daniel     | Miller    | daniel@example.com  |              6 | Withdrawal       |   2000 | 2021-11-01       |
|           6 | Sophia     | Davis     | sophia@example.com  |              7 | Deposite         |  10000 | 2022-10-11       |
|           5 | Michael    | Brown     | michael@example.com |              8 | Deposite         |  20000 | 2023-10-05       |
|           6 | Sophia     | Davis     | sophia@example.com  |              9 | Transfer         |   4000 | 2023-11-06       |
|           5 | Michael    | Brown     | michael@example.com |             10 | Transfer         |   5000 | 2023-01-01       |
+-------------+------------+-----------+---------------------+----------------+------------------+--------+------------------+
9 rows in set (0.00 sec)
```

3.write an sql query to increase the balance of a specific account by a certain amount

```
ERROR 1242 (21000): Subquery returns more than 1 row
mysql> select * from accounts;
+------------+-------------+--------------+---------+
| account_id | customer_id | account_type | balance |
+------------+-------------+--------------+---------+
|          1 |           1 | Current      |   20000 |
|          2 |           1 | Saving       |   50000 |
|          3 |           2 | Current      |     300 |
|          4 |           3 | Saving       |   80000 |
|          5 |           4 | zero_balance |       0 |
|          6 |           5 | Current      |  100000 |
|          7 |           5 | Saving       |   40000 |
|          8 |           6 | Current      |   70000 |
|          9 |           7 | Saving       |   90000 |
|         10 |           7 | Current      |   30000 |
+------------+-------------+--------------+---------+
10 rows in set (0.00 sec)

mysql> update accounts set balance=balance+10000 where account_id=10;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from accounts;
+------------+-------------+--------------+---------+
| account_id | customer_id | account_type | balance |
+------------+-------------+--------------+---------+
|          1 |           1 | Current      |   20000 |
|          2 |           1 | Saving       |   50000 |
|          3 |           2 | Current      |     300 |
|          4 |           3 | Saving       |   80000 |
|          5 |           4 | zero_balance |       0 |
|          6 |           5 | Current      |  100000 |
|          7 |           5 | Saving       |   40000 |
|          8 |           6 | Current      |   70000 |
|          9 |           7 | Saving       |   90000 |
|         10 |           7 | Current      |   40000 |
+------------+-------------+--------------+---------+
10 rows in set (0.00 sec)
```

4.combine first and last name of customer as full_name

```
mysql> select concat(First_Name,' ',Last_Name) as full_name from customers;
+----------------+
| full_name      |
+----------------+
| John Doe       |
| Alice Smith    |
| Bob Johnson    |
| Emma Williams  |
| Michael Brown  |
| Sophia Davis   |
| Daniel Miller  |
| Olivia Garcia  |
| Ethan Wilson   |
| Ava Martinez   |
+----------------+
10 rows in set (0.00 sec)
```

5.remove accounts with balance zero and account type is saving

```
mysql> select * from accounts;
+------------+-------------+--------------+---------+
| account_id | customer_id | account_type | balance |
+------------+-------------+--------------+---------+
|          1 |           1 | Current      |   20000 |
|          2 |           1 | Saving       |       0 |
|          3 |           2 | Current      |     300 |
|          4 |           3 | Saving       |   80000 |
|          5 |           4 | zero_balance |       0 |
|          6 |           5 | Current      |  100000 |
|          7 |           5 | Saving       |   40000 |
|          8 |           6 | Current      |   70000 |
|          9 |           7 | Saving       |   90000 |
|         10 |           7 | Current      |   40000 |
+------------+-------------+--------------+---------+
10 rows in set (0.00 sec)

mysql> delete from accounts where balance=0 and account_type='Saving';
Query OK, 1 row affected (0.04 sec)

mysql> select * from accounts;
+------------+-------------+--------------+---------+
| account_id | customer_id | account_type | balance |
+------------+-------------+--------------+---------+
|          1 |           1 | Current      |   20000 |
|          3 |           2 | Current      |     300 |
|          4 |           3 | Saving       |   80000 |
|          5 |           4 | zero_balance |       0 |
|          6 |           5 | Current      |  100000 |
|          7 |           5 | Saving       |   40000 |
|          8 |           6 | Current      |   70000 |
|          9 |           7 | Saving       |   90000 |
|         10 |           7 | Current      |   40000 |
+------------+-------------+--------------+---------+
9 rows in set (0.00 sec)
```

6.find customers living in a specific city.

```
mysql> select * from customers;
+-------------+------------+-----------+------------+---------------------+--------------+---------+
| customer_id | first_name | last_name | DOB        | email               | phone_number | address |
+-------------+------------+-----------+------------+---------------------+--------------+---------+
|           1 | John       | Doe       | 1990-05-15 | john@example.com    |   1234567890 | Gwalior |
|           2 | Alice      | Smith     | 1988-08-20 | alice@example.com   |   9876543210 | Gwalior |
|           3 | Bob        | Johnson   | 1995-02-10 | bob@example.com     |   7890123456 | Delhi   |
|           4 | Emma       | Williams  | 1992-11-25 | emma@example.com    |   3456789012 | Jaipur  |
|           5 | Michael    | Brown     | 1985-09-30 | michael@example.com |   2345678901 | Delhi   |
|           6 | Sophia     | Davis     | 1998-04-18 | sophia@example.com  |   9012345678 | Gurgaon |
|           7 | Daniel     | Miller    | 1993-07-05 | daniel@example.com  |   4567890123 | Jaipur  |
|           8 | Olivia     | Garcia    | 1997-12-12 | olivia@example.com  |   6789012345 | Morena  |
|           9 | Ethan      | Wilson    | 1996-01-28 | ethan@example.com   |   1231231234 | Manali  |
|          10 | Ava        | Martinez  | 1994-06-08 | ava@example.com     |   8908908901 | Agra    |
+-------------+------------+-----------+------------+---------------------+--------------+---------+
10 rows in set (0.00 sec)

mysql> select * from customers where address='Jaipur';
+-------------+------------+-----------+------------+--------------------+--------------+---------+
| customer_id | first_name | last_name | DOB        | email              | phone_number | address |
+-------------+------------+-----------+------------+--------------------+--------------+---------+
|           4 | Emma       | Williams  | 1992-11-25 | emma@example.com   |   3456789012 | Jaipur  |
|           7 | Daniel     | Miller    | 1993-07-05 | daniel@example.com |   4567890123 | Jaipur  |
+-------------+------------+-----------+------------+--------------------+--------------+---------+
2 rows in set (0.00 sec)
```

7.Get the specific account balance for a specific account.

```
mysql> select * from accounts;
+------------+-------------+--------------+---------+
| account_id | customer_id | account_type | balance |
+------------+-------------+--------------+---------+
|          1 |           1 | Current      |   20000 |
|          3 |           2 | Current      |     300 |
|          4 |           3 | Saving       |   80000 |
|          5 |           4 | zero_balance |       0 |
|          6 |           5 | Current      |  100000 |
|          7 |           5 | Saving       |   40000 |
|          8 |           6 | Current      |   70000 |
|          9 |           7 | Saving       |   90000 |
|         10 |           7 | Current      |   40000 |
+------------+-------------+--------------+---------+
9 rows in set (0.00 sec)

mysql> select balance from accounts where account_id=3;
+---------+
| balance |
+---------+
|     300 |
+---------+
1 row in set (0.00 sec)
```

8.list all accounts with balance > 1000

```
mysql> select * from accounts where balance>1000;
+------------+-------------+--------------+---------+
| account_id | customer_id | account_type | balance |
+------------+-------------+--------------+---------+
|          1 |           1 | Current      |   20000 |
|          4 |           3 | Saving       |   80000 |
|          6 |           5 | Current      |  100000 |
|          7 |           5 | Saving       |   40000 |
|          8 |           6 | Current      |   70000 |
|          9 |           7 | Saving       |   90000 |
|         10 |           7 | Current      |   40000 |
+------------+-------------+--------------+---------+
7 rows in set (0.00 sec)
```

9.get all transactions for a specific account

```
mysql> select * from transactions;
+----------------+------------+------------------+--------+------------------+
| transaction_id | account_id | transaction_type | amount | transaction_date |
+----------------+------------+------------------+--------+------------------+
|              1 |          1 | Transfer         |  10000 | 2023-11-02       |
|              2 |          3 | Withdrawal       |    200 | 2022-03-02       |
|              4 |          1 | Deposite         |  10000 | 2022-11-12       |
|              5 |          5 | Deposite         |   1000 | 2023-02-02       |
|              6 |         10 | Withdrawal       |   2000 | 2021-11-01       |
|              7 |          8 | Deposite         |  10000 | 2022-10-11       |
|              8 |          7 | Deposite         |  20000 | 2023-10-05       |
|              9 |          8 | Transfer         |   4000 | 2023-11-06       |
|             10 |          6 | Transfer         |   5000 | 2023-01-01       |
+----------------+------------+------------------+--------+------------------+
9 rows in set (0.00 sec)

mysql> select * from transactions where account_id=1;
+----------------+------------+------------------+--------+------------------+
| transaction_id | account_id | transaction_type | amount | transaction_date |
+----------------+------------+------------------+--------+------------------+
|              1 |          1 | Transfer         |  10000 | 2023-11-02       |
|              4 |          1 | Deposite         |  10000 | 2022-11-12       |
+----------------+------------+------------------+--------+------------------+
2 rows in set (0.00 sec)
```

10.write a sql query to calculate the interest accrued on savings accounts based on a given interest rate =5%

```
mysql> select account_id ,balance *0.05 as interestaccrued from accounts where account_type='Saving';
+------------+-----------------+
| account_id | interestaccrued |
+------------+-----------------+
|          4 |         4000.00 |
|          7 |         2000.00 |
|          9 |         4500.00 |
+------------+-----------------+
3 rows in set (0.00 sec)
```

11.write a sql query to identify accounts where the balance is less than a specified overdraft limit=1000.

```
mysql> select * from accounts;
+------------+-------------+--------------+---------+
| account_id | customer_id | account_type | balance |
+------------+-------------+--------------+---------+
|          1 |           1 | Current      |   20000 |
|          3 |           2 | Current      |     300 |
|          4 |           3 | Saving       |   80000 |
|          5 |           4 | zero_balance |       0 |
|          6 |           5 | Current      |   -1004 |
|          7 |           5 | Saving       |   40000 |
|          8 |           6 | Current      |   70000 |
|          9 |           7 | Saving       |   90000 |
|         10 |           7 | Current      |   40000 |
+------------+-------------+--------------+---------+
9 rows in set (0.00 sec)

mysql> select * from accounts where balance<-1000;
+------------+-------------+--------------+---------+
| account_id | customer_id | account_type | balance |
+------------+-------------+--------------+---------+
|          6 |           5 | Current      |   -1004 |
+------------+-------------+--------------+---------+
1 row in set (0.00 sec)
```

12. Write a sql query to find the customers not living in a specific city=’Gwalior’

```
mysql> select * from customers;
+-------------+------------+-----------+------------+---------------------+--------------+---------+
| customer_id | first_name | last_name | DOB        | email               | phone_number | address |
+-------------+------------+-----------+------------+---------------------+--------------+---------+
|           1 | John       | Doe       | 1990-05-15 | john@example.com    |   1234567890 | Gwalior |
|           2 | Alice      | Smith     | 1988-08-20 | alice@example.com   |   9876543210 | Gwalior |
|           3 | Bob        | Johnson   | 1995-02-10 | bob@example.com     |   7890123456 | Delhi   |
|           4 | Emma       | Williams  | 1992-11-25 | emma@example.com    |   3456789012 | Jaipur  |
|           5 | Michael    | Brown     | 1985-09-30 | michael@example.com |   2345678901 | Delhi   |
|           6 | Sophia     | Davis     | 1998-04-18 | sophia@example.com  |   9012345678 | Gurgaon |
|           7 | Daniel     | Miller    | 1993-07-05 | daniel@example.com  |   4567890123 | Jaipur  |
|           8 | Olivia     | Garcia    | 1997-12-12 | olivia@example.com  |   6789012345 | Morena  |
|           9 | Ethan      | Wilson    | 1996-01-28 | ethan@example.com   |   1231231234 | Manali  |
|          10 | Ava        | Martinez  | 1994-06-08 | ava@example.com     |   8908908901 | Agra    |
+-------------+------------+-----------+------------+---------------------+--------------+---------+
10 rows in set (0.00 sec)

mysql> select * from customers where address<>'Gwalior';
+-------------+------------+-----------+------------+---------------------+--------------+---------+
| customer_id | first_name | last_name | DOB        | email               | phone_number | address |
+-------------+------------+-----------+------------+---------------------+--------------+---------+
|           3 | Bob        | Johnson   | 1995-02-10 | bob@example.com     |   7890123456 | Delhi   |
|           4 | Emma       | Williams  | 1992-11-25 | emma@example.com    |   3456789012 | Jaipur  |
|           5 | Michael    | Brown     | 1985-09-30 | michael@example.com |   2345678901 | Delhi   |
|           6 | Sophia     | Davis     | 1998-04-18 | sophia@example.com  |   9012345678 | Gurgaon |
|           7 | Daniel     | Miller    | 1993-07-05 | daniel@example.com  |   4567890123 | Jaipur  |
|           8 | Olivia     | Garcia    | 1997-12-12 | olivia@example.com  |   6789012345 | Morena  |
|           9 | Ethan      | Wilson    | 1996-01-28 | ethan@example.com   |   1231231234 | Manali  |
|          10 | Ava        | Martinez  | 1994-06-08 | ava@example.com     |   8908908901 | Agra    |
+-------------+------------+-----------+------------+---------------------+--------------+---------+
8 rows in set (0.00 sec)
```

Task -3

1.Write a SQL query to Find the average account balance for all customers.

```
mysql> SELECT AVG(balance) AS average_balance
    -> FROM accounts;
+-----------------+
| average_balance |
+-----------------+
|      37699.5556 |
+-----------------+
1 row in set (0.00 sec)
```

2.Write a SQL query to Retrieve the top 10 highest account balances.

```
mysql> SELECT account_id, balanc
    -> FROM accounts
    -> ORDER BY balance DESC
    -> LIMIT 10;
+------------+---------+
| account_id | balance |
+------------+---------+
|          9 |   90000 |
|          4 |   80000 |
|          8 |   70000 |
|          7 |   40000 |
|         10 |   40000 |
|          1 |   20000 |
|          3 |     300 |
|          5 |       0 |
|          6 |   -1004 |
+------------+---------+
9 rows in set (0.00 sec)
```

3.Write a SQL query to Calculate Total Deposits for All Customers in specific date

```
mysql> SELECT SUM(amount) AS total_deposits
    -> FROM transactions
    -> WHERE transaction_type = 'Deposite'
    -> AND DATE(transaction_date) = '2023-10-05';
+----------------+
| total_deposits |
+----------------+
|          20000 |
+----------------+
1 row in set (0.01 sec)
```

4.Write a SQL query to Find the Oldest and Newest Customers.

```
mysql> (
    ->     SELECT
    ->         first_name, last_name, DOB
    ->     FROM
    ->         customers
    ->     ORDER BY
    ->         DOB ASC
    ->     LIMIT 1
    -> )
    -> UNION
    -> (
    ->     SELECT
    ->         first_name, last_name, DOB
    ->     FROM
    ->         customers
    ->     ORDER BY
    ->         DOB DESC
    ->     LIMIT 1
    -> );
+------------+-----------+------------+
| first_name | last_name | DOB        |
+------------+-----------+------------+
| Michael    | Brown     | 1985-09-30 |
| Sophia     | Davis     | 1998-04-18 |
+------------+-----------+------------+
2 rows in set (0.01 sec)
```

5.Write a SQL query to Retrieve transaction details along with the account type

```
mysql> SELECT
    ->     t.transaction_id, t.account_id, t.transaction_type, t.amount, t.transaction_date, a.account_type
    -> FROM
    ->     transactions t
    -> JOIN
    ->     accounts a ON t.account_id = a.account_id;
+----------------+------------+------------------+--------+------------------+--------------+
| transaction_id | account_id | transaction_type | amount | transaction_date | account_type |
+----------------+------------+------------------+--------+------------------+--------------+
|              1 |          1 | Transfer         |  10000 | 2023-11-02       | Current      |
|              2 |          3 | Withdrawal       |    200 | 2022-03-02       | Current      |
|              4 |          1 | Deposite         |  10000 | 2022-11-12       | Current      |
|              5 |          5 | Deposite         |   1000 | 2023-02-02       | zero_balance |
|              6 |         10 | Withdrawal       |   2000 | 2021-11-01       | Current      |
|              7 |          8 | Deposite         |  10000 | 2022-10-11       | Current      |
|              8 |          7 | Deposite         |  20000 | 2023-10-05       | Saving       |
|              9 |          8 | Transfer         |   4000 | 2023-11-06       | Current      |
|             10 |          6 | Transfer         |   5000 | 2023-01-01       | Current      |
+----------------+------------+------------------+--------+------------------+--------------+
9 rows in set (0.00 sec)
```

6.Write a SQL query to Get a list of customers along with their account details.

```
mysql> SELECT
    ->     c.customer_id, c.first_name, c.last_name, c.email, c.phone_number, c.address,
    ->     a.account_id, a.account_type, a.balance
    -> FROM
    ->     customers c
    -> JOIN
    ->     accounts a ON c.customer_id = a.customer_id;
+-------------+------------+-----------+--------------------+--------------+---------+------------+--------------+---------+
| customer_id | first_name | last_name | email              | phone_number | address | account_id | account_type | balance |
+-------------+------------+-----------+--------------------+--------------+---------+------------+--------------+---------+
|           1 | John       | Doe       | john@example.com   |   1234567890 | Gwalior |          1 | Current      |   20000 |
|           2 | Alice      | Smith     | alice@example.com  |   9876543210 | Gwalior |          3 | Current      |     300 |
|           3 | Bob        | Johnson   | bob@example.com    |   7890123456 | Delhi   |          4 | Saving       |   80000 |
|           4 | Emma       | Williams  | emma@example.com   |   3456789012 | Jaipur  |          5 | zero_balance |       0 |
|           5 | Michael    | Brown     | michael@example.com |   2345678901 | Delhi   |          6 | Current      |   -1004 |
|           5 | Michael    | Brown     | michael@example.com |   2345678901 | Delhi   |          7 | Saving       |   40000 |
|           6 | Sophia     | Davis     | sophia@example.com |   9012345678 | Gurgaon |          8 | Current      |   70000 |
|           7 | Daniel     | Miller    | daniel@example.com |   4567890123 | Jaipur  |          9 | Saving       |   90000 |
|           7 | Daniel     | Miller    | daniel@example.com |   4567890123 | Jaipur  |         10 | Current      |   40000 |
+-------------+------------+-----------+--------------------+--------------+---------+------------+--------------+---------+
9 rows in set (0.00 sec)
```

7.Write a SQL query to Retrieve transaction details along with customer information for a specific account.

```
mysql> SELECT
    ->     t.transaction_id, t.account_id, t.transaction_type, t.amount, t.transaction_date,
    ->     c.customer_id, c.first_name, c.last_name, c.email, c.phone_number, c.address
    -> FROM
    ->     transactions t
    -> JOIN
    ->     accounts a ON t.account_id = a.account_id
    -> JOIN
    ->     customers c ON a.customer_id = c.customer_id
    -> WHERE
    ->     t.account_id =3;
+----------------+------------+------------------+--------+------------------+-------------+------------+-----------+------------------+--------------+---------+
| transaction_id | account_id | transaction_type | amount | transaction_date | customer_id | first_name | last_name | email            | phone_number | address |
+----------------+------------+------------------+--------+------------------+-------------+------------+-----------+------------------+--------------+---------+
|              2 |          3 | Withdrawal       |    200 | 2022-03-02       |           2 | Alice      | Smith     | alice@example.com |   9876543210 | Gwalior |
+----------------+------------+------------------+--------+------------------+-------------+------------+-----------+------------------+--------------+---------+
1 row in set (0.00 sec)
```

8.Write a SQL query to Identify customers who have more than one account.

```
mysql> SELECT customer_id, COUNT(account_id) AS num_accounts
    -> FROM accounts
    -> GROUP BY customer_id
    -> HAVING COUNT(account_id) > 1;
+-------------+--------------+
| customer_id | num_accounts |
+-------------+--------------+
|           5 |            2 |
|           7 |            2 |
+-------------+--------------+
2 rows in set (0.00 sec)
```

9.Write a SQL query to Calculate the difference in transaction amounts between deposits and withdrawals.

```
mysql> SELECT SUM(CASE WHEN transaction_type = 'Deposite' THEN amount ELSE 0 END) AS total_deposits,
    ->        SUM(CASE WHEN transaction_type = 'Withdrawal' THEN amount ELSE 0 END) AS total_withdrawals,
    ->        SUM(CASE WHEN transaction_type = 'Deposite' THEN amount ELSE 0 END) - SUM(CASE WHEN transaction_type = 'Withdrawal' THEN amount ELSE 0 END) AS difference
    -> FROM transactions;
+----------------+-------------------+------------+
| total_deposits | total_withdrawals | difference |
+----------------+-------------------+------------+
|          41000 |              2200 |      38800 |
+----------------+-------------------+------------+
1 row in set (0.01 sec)
```

10. Write a SQL query to Calculate the average daily balance for each account over a specified period.

```
mysql> SELECT account_id, AVG(daily_balance) AS average_daily_balance
    -> FROM (
    ->     SELECT t.account_id, t.transaction_date,
    ->            SUM(CASE
    ->                    WHEN t.transaction_type = 'Deposite' THEN t.amount
    ->                    WHEN t.transaction_type = 'Transfer' THEN t.amount
    ->                    WHEN t.transaction_type = 'Withdrawal' THEN -t.amount
    ->                    ELSE 0
    ->                END) AS daily_balance
    ->     FROM transactions t
    ->     WHERE t.transaction_date BETWEEN '2023-01-01' AND '2023-12-31'
    ->     GROUP BY t.account_id, t.transaction_date
    -> ) AS daily_balances
    -> GROUP BY account_id;
+------------+-----------------------+
| account_id | average_daily_balance |
+------------+-----------------------+
|          1 |            10000.0000 |
|          5 |             1000.0000 |
|          7 |            20000.0000 |
|          8 |             4000.0000 |
|          6 |             5000.0000 |
+------------+-----------------------+
5 rows in set (0.00 sec)
```

11.Calculate the total balance for each account type.

```
mysql> SELECT account_type, SUM(balance) AS total_balance
    -> FROM accounts
    -> GROUP BY account_type;
+--------------+---------------+
| account_type | total_balance |
+--------------+---------------+
| Current      |        129296 |
| Saving       |        210000 |
| zero_balance |             0 |
+--------------+---------------+
3 rows in set (0.00 sec)
```

12. Identify accounts with the highest number of transactions order by descending order.

```
mysql> SELECT account_id, COUNT(transaction_id) AS transaction_count
    -> FROM transactions
    -> GROUP BY account_id
    -> ORDER BY transaction_count DESC;
+------------+-------------------+
| account_id | transaction_count |
+------------+-------------------+
|          1 |                 2 |
|          8 |                 2 |
|          3 |                 1 |
|          5 |                 1 |
|          6 |                 1 |
|          7 |                 1 |
|         10 |                 1 |
+------------+-------------------+
7 rows in set (0.00 sec)
```

13.List customers with high aggregate account balances, along with their account types.

```
mysql> SELECT c.customer_id, c.first_name, c.last_name, a.account_type, SUM(a.balance) AS total_balance
    -> FROM customers c
    -> JOIN accounts a ON c.customer_id = a.customer_id
    -> GROUP BY c.customer_id, a.account_type
    -> HAVING total_balance > 50000
    -> ORDER BY total_balance DESC;
+-------------+------------+-----------+--------------+---------------+
| customer_id | first_name | last_name | account_type | total_balance |
+-------------+------------+-----------+--------------+---------------+
|           7 | Daniel     | Miller    | Saving       |         90000 |
|           3 | Bob        | Johnson   | Saving       |         80000 |
|           6 | Sophia     | Davis     | Current      |         70000 |
+-------------+------------+-----------+--------------+---------------+
3 rows in set (0.00 sec)

mysql>
```

14.. Identify and list duplicate transactions based on transaction amount, date, and account.

```
mysql> SELECT transaction_id, account_id, amount, transaction_date, COUNT(*) AS duplicate_count
    -> FROM transactions
    -> GROUP BY transaction_id, account_id, amount, transaction_date
    -> HAVING COUNT(*) > 1
    -> ORDER BY duplicate_count DESC;
Empty set (0.00 sec)
```

Task-4

1.Retrieve the customer(s) with the highest account balance

```
mysql> SELECT c.first_name, c.last_name, a.balance
    -> FROM customers c
    -> JOIN accounts a ON c.customer_id = a.customer_id
    -> ORDER BY a.balance DESC
    -> LIMIT 1;
+------------+-----------+---------+
| first_name | last_name | balance |
+------------+-----------+---------+
| Daniel     | Miller    |   90000 |
+------------+-----------+---------+
1 row in set (0.00 sec)
```

2.Calculate the average account balance for customers who have more than one account.

```
mysql> SELECT AVG(subq.balance) AS avg_balance
    -> FROM (
    ->     SELECT customer_id, COUNT(*) as num_accounts, SUM(balance) as balance
    ->     FROM accounts
    ->     GROUP BY customer_id
    ->     HAVING num_accounts > 1
    -> ) AS subq;
+-------------+
| avg_balance |
+-------------+
|  84498.0000 |
+-------------+
1 row in set (0.00 sec)
```

3. Retrieve accounts with transactions whose amounts exceed the average transaction amount.

```
mysql> SELECT DISTINCT t.account_id, t.amount
    -> FROM transactions t
    -> WHERE t.amount > (
    ->     SELECT AVG(amount) AS avg_amount
    ->     FROM transactions
    -> )
    -> ;
+------------+--------+
| account_id | amount |
+------------+--------+
|          1 |  10000 |
|          8 |  10000 |
|          7 |  20000 |
+------------+--------+
3 rows in set (0.00 sec)
```

4.Identify customers who have no recorded transactions.

```
mysql> SELECT c.customer_id, c.first_name, c.last_name
    -> FROM customers c
    -> LEFT JOIN accounts a ON c.customer_id = a.customer_id
    -> LEFT JOIN transactions t ON a.account_id = t.account_id
    -> WHERE t.transaction_id IS NULL
    -> ;
+-------------+------------+-----------+
| customer_id | first_name | last_name |
+-------------+------------+-----------+
|           3 | Bob        | Johnson   |
|           7 | Daniel     | Miller    |
|           8 | Olivia     | Garcia    |
|           9 | Ethan      | Wilson    |
|          10 | Ava        | Martinez  |
+-------------+------------+-----------+
5 rows in set (0.00 sec)
```

5.Calculate the total balance of accounts with no recorded transactions

```
mysql> SELECT a.account_id, a.customer_id, a.account_type, COALESCE(SUM(a.balance), 0) AS total_balance
    -> FROM accounts a
    -> LEFT JOIN transactions t ON a.account_id = t.account_id
    -> GROUP BY a.account_id
    -> HAVING SUM(t.amount) IS NULL;
+------------+-------------+--------------+---------------+
| account_id | customer_id | account_type | total_balance |
+------------+-------------+--------------+---------------+
|          4 |           3 | Saving       |         80000 |
|          9 |           7 | Saving       |         90000 |
+------------+-------------+--------------+---------------+
2 rows in set (0.00 sec)
```

6. Retrieve transactions for accounts with the lowest balance.

```
mysql> SELECT *
    -> FROM transactions
    -> WHERE account_id = (
    ->     SELECT account_id
    ->     FROM accounts
    ->     ORDER BY balance
    ->     LIMIT 1
    -> );
+----------------+------------+------------------+--------+------------------+
| transaction_id | account_id | transaction_type | amount | transaction_date |
+----------------+------------+------------------+--------+------------------+
|             10 |          6 | Transfer         |   5000 | 2023-01-01       |
+----------------+------------+------------------+--------+------------------+
1 row in set (0.01 sec)
```

7.. Identify customers who have accounts of multiple types.

```
mysql> SELECT customer_id
    -> FROM accounts
    -> GROUP BY customer_id
    -> HAVING COUNT(DISTINCT account_type) > 1;
+-------------+
| customer_id |
+-------------+
|           5 |
|           7 |
+-------------+
2 rows in set (0.00 sec)
```

8.Calculate the percentage of each account type out of the total number of accounts.

```
mysql> SELECT account_type, COUNT(*) AS count_accounts,
    ->        ROUND((COUNT(*) * 100.0) / (SELECT COUNT(*) FROM accounts), 2) AS percentage
    -> FROM accounts
    -> GROUP BY account_type;
+--------------+---------------+------------+
| account_type | count_accounts | percentage |
+--------------+---------------+------------+
| Current      |             5 |      55.56 |
| Saving       |             3 |      33.33 |
| zero_balance |             1 |      11.11 |
+--------------+---------------+------------+
3 rows in set (0.02 sec)
```

9.Retrieve all transactions for a customer with a given customer_id.

```
mysql> SELECT *
    -> FROM transactions
    -> WHERE account_id IN (SELECT account_id FROM accounts WHERE customer_id = 1);
+----------------+------------+------------------+--------+------------------+
| transaction_id | account_id | transaction_type | amount | transaction_date |
+----------------+------------+------------------+--------+------------------+
|              1 |          1 | Transfer         |  10000 | 2023-11-02       |
|              4 |          1 | Deposite         |  10000 | 2022-11-12       |
+----------------+------------+------------------+--------+------------------+
2 rows in set (0.00 sec)

mysql>
```

10.Calculate the total balance for each account type, including a subquery within the SELECT clause.

```
mysql> SELECT
    ->     account_type,
    ->     (
    ->         SELECT SUM(balance)
    ->         FROM accounts
    ->         WHERE a.account_type = accounts.account_type
    ->     ) AS total_balance
    -> FROM accounts a
    -> GROUP BY account_type;
+--------------+---------------+
| account_type | total_balance |
+--------------+---------------+
| Current      |        129296 |
| Saving       |        210000 |
| zero_balance |             0 |
+--------------+---------------+
3 rows in set (0.00 sec)
```