Assignment 3 -SQL & OOPS Banking System

TASK 1: Database Design

1. **Create the database named "HMBank"**

   create database HMBank;

   Output:

```
mysql> create database HMBank;
Query OK, 1 row affected (0.50 sec)
```

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

   Customers Table Schema:

   - **customer_id (Primary Key):** Integer.
   - **first_name:** Varchar (50), Not Null.
   - **last_name:** Varchar (50), Not Null.
   - **DOB (Date of Birth):** Date, Not Null.
   - **email:** Varchar (50), Unique, Not Null.
   - **phone_number:** Varchar (12), Unique, Not Null.
   - **address:** Text, Not Null.

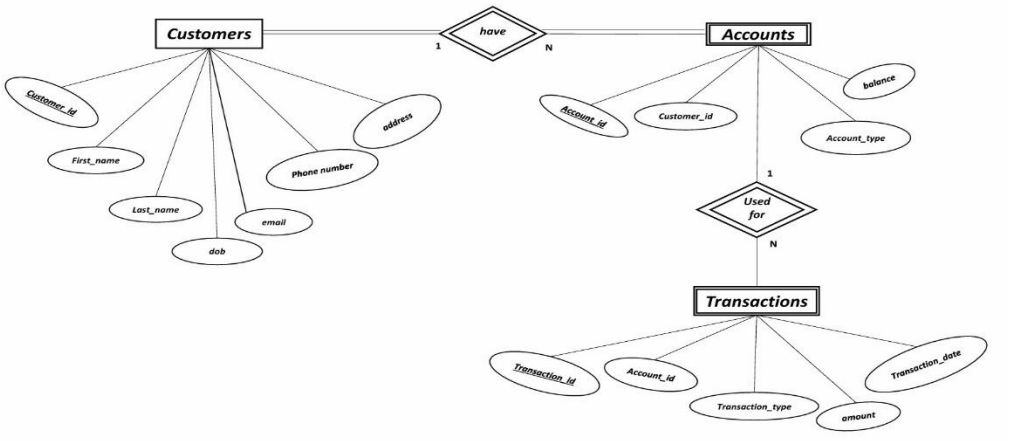
   Accounts Table Schema:

   - **account_id (Primary Key):** Integer.
   - **customer_id (Foreign Key):** Integer, Not Null.
   - **account_type (e.g., savings, current, zero_balance):** Enum, Not Null.
   - **balance:** Decimal (10,2), Not Null.

   Transactions Table Schema:

   - **transaction_id (Primary Key):** Integer.
   - **account_id (Foreign Key):** Integer, Not Null.
   - **transaction_type (e.g., deposit, withdrawal, transfer):** Enum, Not Null.
   - **amount:** Decimal (10,2), Not Null.
   - **transaction_date:** Date, Not Null.

3. **Create an ERD (Entity Relationship Diagram) for the database.**



4. **Create appropriate Primary Key and Foreign Key constraints for referential integrity.**

   Customers Table
   - Primary Key: customer_id

   Accounts Table
   - Primary Key: account_id
   - Foreign Key: customer_id

   Transactions Table
   - Primary Key: transaction_id
   - Foreign Key: account_id

5. **Write SQL scripts to create the mentioned tables with appropriate data types, constraints, and relationships.**
   - **Customers**
   - **Accounts**
   - **Transactions**

Customers Table:

create table customers(customer_id int primary key auto_increment,first_name varchar(50) not null,last_name varchar(50) not null,DOB date not null,email varchar(50) unique not null,phone_number varchar(12) unique not null,address text not null);

Output:

```
mysql> create table customers(customer_id int primary key auto_increment,
    -> first_name varchar(50) not null,
    -> last_name varchar(50) not null,
    -> DOB date not null,
    -> email varchar(50) unique not null,
    -> phone_number varchar(12) unique not null,
    -> address text not null);
Query OK, 0 rows affected (0.07 sec)
```

Accounts Table:

create table accounts (account_id int primary key auto_increment,customer_id int,account_type enum('savings', 'current', 'zero_balance'),balance decimal(10,2) default 0.00,foreign key (customer_id) references customers(customer_id) on delete cascade);

Output:

```
mysql> create table accounts (
    ->     account_id int primary key auto_increment,
    ->     customer_id int,
    ->     account_type enum('savings', 'current', 'zero_balance'),
    ->     balance decimal(10,2) default 0.00,
    ->     foreign key (customer_id) references customers(customer_id) on delete cascade
    -> );
Query OK, 0 rows affected (0.05 sec)
```

Transactions Table:

create table transactions (transaction_id int primary key auto_increment, account_id int, transaction_type enum('deposit', 'withdrawal', 'transfer') not null,amount decimal(10,2) not null,transaction_date date not null,foreign key (account_id) references accounts(account_id) on delete cascade);

Output:

```
mysql> create table transactions (
    ->     transaction_id int primary key auto_increment,
    ->     account_id int,
    ->     transaction_type enum('deposit', 'withdrawal', 'transfer') not null,
    ->     amount decimal(10,2) not null,
    ->     transaction_date date not null,
    ->     foreign key (account_id) references accounts(account_id) on delete cascade
    -> );
Query OK, 0 rows affected (0.05 sec)
```

TASK 2: Select, Where, Between, AND, LIKE

1. **Insert at least 10 sample records into each of the following tables.**
    - **Customers**
    - **Accounts**
    - **Transactions**

Customers Table:

1. insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Sai Vighnessh', 'Balaji', '2003-09-07', 'saivighnessh@gmail.com', 9908287108, '23/15 madipakkam chennai');
2. insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Madhu', 'Kalla', '1995-03-05', 'madhukalla@gmail.com', 9988776655, '1/24 mathura nagar Vijayawada');
3. insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Surya', 'Narayanan', '2003-02-25', 'suryanarayanan@gmail.com', 9876543211, ' F-B,tower apartments Anna nagar chennai');
4. insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Katta', 'Rohit', '2004-04-15', 'rohit77@gmail.com', 9358218982, `31B Whitefield Bengaluru ');
5. insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Kavin', 'Kaarthick', '2003-07-07', 'Kaarthick@gmail.com', 8768975689, ' 98/12H sector B Kolkata');
6. insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Sree', 'Tharan', '2003-09-23', 'sreetharan74@gmail.com', 9056787654, ' 24/87F satara, Pune');
7. insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Sourabh', 'V', '2002-12-11', 'sourabh12@gmail.com', 9756789043, '18C apartment HSR layout Bengaluru');

8. insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Sathesh', 'M', '2002-12-15', 'sathesh45@gmail.com', 9940562789, '51/57 Khadhi colony, Tirupati');
9. insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Subitchan', 'K', '2004-10-10', 'subitchan@gmail.com', 8790365789, '25/11 avadi chennai');
10. insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Surendar', 'P', '2003-06-29', 'surenderp@gmail.com', 9944078956, '17/23 guindy chennai');
11. insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('siva', 'ganesh', '2002-04-12', 'sivaganesh123@gmail.com', 9925677128, '32/16 Guntur Andhrapradesh');

Accounts Table:

1. insert into accounts (customer_id, account_type, balance) values (1, 'savings', 125000.75);
2. insert into accounts (customer_id, account_type, balance) values (2, 'current', 580000.00);
3. insert into accounts (customer_id, account_type, balance) values (3, 'zero_balance', 0.00);
4. insert into accounts (customer_id, account_type, balance) values (4, 'savings', 15770.50);
5. insert into accounts (customer_id, account_type, balance) values (5, 'current', 104000.00);
6. insert into accounts (customer_id, account_type, balance) values (6, 'zero_balance', 0.00);
7. insert into accounts (customer_id, account_type, balance) values (7, 'savings', 27500.25);
8. insert into accounts (customer_id, account_type, balance) values (8, 'current', 185000.90);
9. insert into accounts (customer_id, account_type, balance) values (9, 'zero_balance', 0.00);
10. insert into accounts (customer_id, account_type, balance) values (10, 'savings', 350000.60);
11. insert into accounts (customer_id, account_type, balance) values (11, 'savings', 68760.00);

Transactions Table:

1. insert into transactions (account_id, transaction_type, amount, transaction_date) values (1, 'deposit', 5000.00, '2025-01-05');
2. insert into transactions (account_id, transaction_type, amount, transaction_date) values (2, 'withdrawal', 2000.00, '2025-03-15');
3. insert into transactions (account_id, transaction_type, amount, transaction_date) values (3, 'transfer', 1500.50, '2025-01-25');
4. insert into transactions (account_id, transaction_type, amount, transaction_date) values (4, 'deposit', 8000.75, '2025-02-05');
5. insert into transactions (account_id, transaction_type, amount, transaction_date) values (5, 'withdrawal', 500.00, '2025-03-12');

6. insert into transactions (account_id, transaction_type, amount, transaction_date) values (6, 'transfer', 2500.25, '2025-02-22');
7. insert into transactions (account_id, transaction_type, amount, transaction_date) values (7, 'deposit', 10000.00, '2025-01-03');
8. insert into transactions (account_id, transaction_type, amount, transaction_date) values (8, 'withdrawal', 750.00, '2025-03-10');
9. insert into transactions (account_id, transaction_type, amount, transaction_date) values (9, 'transfer', 3200.80, '2025-02-18');
10. insert into transactions (account_id, transaction_type, amount, transaction_date) values (10, 'deposit', 12000.50, '2025-01-25');

Output:

```
mysql> insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Sai Vighnessh', 'Balaji', '2003-09-07', 'saivighnessh@gmail.com', 9908287108, 'Chennai');
Query OK, 1 row affected (0.01 sec)

mysql> insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Madhu', 'Kalla', '1995-03-05', 'madhukalla@gmail.com', 9988776655, 'Vijayawada');
Query OK, 1 row affected (0.01 sec)

mysql> insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Surya', 'Narayanan', '2003-02-25', 'suryanarayanan@gmail.com', 9876543211, 'Chennai');
Query OK, 1 row affected (0.01 sec)

mysql> insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Katta', 'Rohit', '2004-04-15', 'rohit77@gmail.com', 9358218982, 'Bengaluru');
Query OK, 1 row affected (0.01 sec)

mysql> insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Kavin', 'Kaarthick', '2003-07-07', 'Kaarthick@gmail.com', 8768975689, 'Kolkata');
Query OK, 1 row affected (0.00 sec)

mysql> insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Sree', 'Tharan', '2003-09-23', 'sreetharan74@gmail.com', 9056787654, 'Pune');
Query OK, 1 row affected (0.01 sec)

mysql> insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Sourabh', 'V', '2002-12-11', 'sourabh12@gmail.com', 9756789043, 'Bengaluru');
Query OK, 1 row affected (0.00 sec)

mysql> insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Sathesh', 'M', '2002-12-15', 'sathesh45@gmail.com', 9940562789, 'Tirupati');
Query OK, 1 row affected (0.01 sec)

mysql> insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Subitchan', 'K', '2004-10-10', 'subitchan@gmail.com', 8790365789, 'Chennai');
Query OK, 1 row affected (0.00 sec)

mysql> insert into customers (first_name, last_name, DOB, email, phone_number, address) values ('Surendar', 'P', '2003-06-29', 'surenderp@gmail.com', 9944078956, 'Chennai');
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into accounts (customer_id, account_type, balance)
    -> values (1, 'savings', 125000.75);
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into accounts (customer_id, account_type, balance)
    -> values (2, 'current', 580000.00);
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into accounts (customer_id, account_type, balance)
    -> values (3, 'zero_balance', 0.00);
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into accounts (customer_id, account_type, balance)
    -> values (4, 'savings', 15770.50);
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into accounts (customer_id, account_type, balance)
    -> values (5, 'current', 104000.00);
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into accounts (customer_id, account_type, balance)
    -> values (6, 'zero_balance', 0.00);
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into accounts (customer_id, account_type, balance)
    -> values (7, 'savings', 27500.25);
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into accounts (customer_id, account_type, balance)
    -> values (8, 'current', 185000.90);
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into accounts (customer_id, account_type, balance)
    -> values (9, 'zero_balance', 0.00);
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into accounts (customer_id, account_type, balance)
    -> values (10, 'savings', 350000.60);
Query OK, 1 row affected (0.00 sec)
```

```
mysql>
mysql> insert into transactions (account_id, transaction_type, amount, transaction_date)
    -> values (2, 'withdrawal', 2000.00, '2025-03-15');
Query OK, 1 row affected (0.01 sec)

mysql>
mysql> insert into transactions (account_id, transaction_type, amount, transaction_date)
    -> values (3, 'transfer', 1500.50, '2025-01-25');
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into transactions (account_id, transaction_type, amount, transaction_date)
    -> values (4, 'deposit', 8000.75, '2025-02-05');
Query OK, 1 row affected (0.01 sec)

mysql>
mysql> insert into transactions (account_id, transaction_type, amount, transaction_date)
    -> values (5, 'withdrawal', 500.00, '2025-03-12');
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into transactions (account_id, transaction_type, amount, transaction_date)
    -> values (6, 'transfer', 2500.25, '2025-02-22');
Query OK, 1 row affected (0.01 sec)

mysql>
mysql> insert into transactions (account_id, transaction_type, amount, transaction_date)
    -> values (7, 'deposit', 10000.00, '2025-01-03');
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into transactions (account_id, transaction_type, amount, transaction_date)
    -> values (8, 'withdrawal', 750.00, '2025-03-10');
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into transactions (account_id, transaction_type, amount, transaction_date)
    -> values (9, 'transfer', 3200.80, '2025-02-18');
Query OK, 1 row affected (0.00 sec)

mysql>
mysql> insert into transactions (account_id, transaction_type, amount, transaction_date)
    -> values (10, 'deposit', 12000.50, '2025-01-25');
Query OK, 1 row affected (0.00 sec)
```

2. Write SQL queries for the following tasks:

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

select concat(first_name,' ',last_name) as name,email,accounts.account_type from customers inner join accounts on customers.customer_id=accounts.customer_id;

```
mysql> select concat(first_name,' ',last_name) as name,email,accounts.account_type from customers inner join accounts on customers.customer_id=accounts.customer_id;
+---------------------+-------------------------+--------------+
| name                | email                   | account_type |
+---------------------+-------------------------+--------------+
| Sai Vighnessh Balaji | saivighnessh@gmail.com  | savings      |
| Madhu Kalla         | madhukalla@gmail.com    | current      |
| Surya Narayanan     | suryanarayanan@gmail.com | zero_balance |
| Katta Rohit         | rohit77@gmail.com       | savings      |
| Kavin Kaarthick     | Kaarthick@gmail.com     | current      |
| Sree Tharan         | sreetharan74@gmail.com  | zero_balance |
| Sourabh V           | sourabh12@gmail.com     | savings      |
| Sathesh M           | sathesh45@gmail.com     | current      |
| Subitchan K         | subitchan@gmail.com     | zero_balance |
| Surendar P          | surenderp@gmail.com     | savings      |
+---------------------+-------------------------+--------------+
10 rows in set (0.00 sec)
```

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

select customers.customer_id,concat(first_name,' ',last_name) as name,transactions.account_id,accounts.account_type,transaction_id,transaction_type,amount,transaction_date from customers inner join accounts on customers.customer_id=accounts.customer_id inner join transactions on accounts.account_id=transactions.account_id order by transaction_date;

```
mysql>
mysql> select customers.customer_id,concat(first_name,' ',last_name) as name,transactions.account_id,accounts.account_type,transaction_id,transaction_type,amount,transaction_date from cust
omers inner join accounts on customers.customer_id=accounts.customer_id inner join transactions on accounts.account_id=transactions.account_id order by transaction_date;
+-------------+---------------------+------------+--------------+----------------+------------------+----------+------------------+
| customer_id | name                | account_id | account_type | transaction_id | transaction_type | amount   | transaction_date |
+-------------+---------------------+------------+--------------+----------------+------------------+----------+------------------+
|           7 | Sourabh V           |          7 | savings      |              7 | deposit          | 10000.00 | 2025-01-03       |
|           1 | Sai Vighnessh Balaji |          1 | savings      |              1 | deposit          |  5000.00 | 2025-01-05       |
|           3 | Surya Narayanan     |          3 | zero_balance |              3 | transfer         |  1500.50 | 2025-01-25       |
|          10 | Surendar P          |         10 | savings      |             10 | deposit          | 12000.50 | 2025-01-25       |
|           4 | Katta Rohit         |          4 | savings      |              4 | deposit          |  8000.75 | 2025-02-05       |
|           9 | Subitchan K         |          9 | zero_balance |              9 | transfer         |  3200.80 | 2025-02-18       |
|           6 | Sree Tharan         |          6 | zero_balance |              6 | transfer         |  2500.25 | 2025-02-22       |
|           8 | Sathesh M           |          8 | current      |              8 | withdrawal       |   750.00 | 2025-03-10       |
|           5 | Kavin Kaarthick     |          5 | current      |              5 | withdrawal       |   500.00 | 2025-03-12       |
|           2 | Madhu Kalla         |          2 | current      |              2 | withdrawal       |  2000.00 | 2025-03-15       |
+-------------+---------------------+------------+--------------+----------------+------------------+----------+------------------+
10 rows in set (0.00 sec)
```

iii. **Write a SQL query to increase the balance of a specific account by a certain amount.**

update accounts set balance=balance+2750 where account_id=7;

```
mysql> update accounts set balance=balance+2750 where account_id=7;
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 | savings      | 125000.75 |
|          2 |           2 | current      | 580000.00 |
|          3 |           3 | zero_balance |      0.00 |
|          4 |           4 | savings      |  15770.50 |
|          5 |           5 | current      | 104000.00 |
|          6 |           6 | zero_balance |      0.00 |
|          7 |           7 | savings      |  30250.25 |
|          8 |           8 | current      | 185000.90 |
|          9 |           9 | zero_balance |      0.00 |
|         10 |          10 | savings      | 350000.60 |
+------------+-------------+--------------+-----------+
10 rows in set (0.00 sec)
```

iv. **Write a SQL query to Combine first and last names of customers as a full_name.**

select concat (first_name,' ',last_name) as full_name from customers;

```
mysql> select concat(first_name,' ',last_name) as full_name from customers;
+---------------------+
| full_name           |
+---------------------+
| Sai Vighnessh Balaji |
| Madhu Kalla         |
| Surya Narayanan     |
| Katta Rohit         |
| Kavin Kaarthick     |
| Sree Tharan         |
| Sourabh V           |
| Sathesh M           |
| Subitchan K         |
| Surendar P          |
+---------------------+
10 rows in set (0.00 sec)
```

v. **Write a SQL query to remove accounts with a balance of zero where the account type is savings.**

delete from accounts where account_type='savings' and balance=0;

```
mysql> delete from accounts where account_type='savings' and balance=0;
Query OK, 1 row affected (0.34 sec)
```

vi. **Write a SQL query to Find customers living in a specific city.**

select customer_id,concat(first_name,' ',last_name),phone_number,email from customers where address like '%chennai%';

```
mysql> select customer_id,concat(first_name,' ',last_name),phone_number,email from customers where address like '%chennai%';
+-------------+-----------------------------------+--------------+-------------------------+
| customer_id | concat(first_name,' ',last_name) | phone_number | email                   |
+-------------+-----------------------------------+--------------+-------------------------+
|           1 | Sai Vighnessh Balaji              | 9908287108   | saivighnessh@gmail.com  |
|           3 | Surya Narayanan                   | 9876543211   | suryanarayanan@gmail.com |
|           9 | Subitchan K                       | 8790365789   | subitchan@gmail.com     |
|          10 | Surendar P                        | 9944078956   | surenderp@gmail.com     |
+-------------+-----------------------------------+--------------+-------------------------+
```

vii. Write a SQL query to Get the account balance for a specific account.

select balance from accounts where account_id=2;

```
mysql> select balance from accounts where account_id=2;
+------------+
| balance    |
+------------+
| 580000.00 |
+------------+
1 row in set (0.00 sec)
```

viii. Write a SQL query to Find customers not living in a specific city.

select customer_id,concat(first_name,' ',last_name),phone_number,email from customers where address not like '%bengaluru%';

```
mysql> select customer_id,concat(first_name,' ',last_name),phone_number,email from customers where address not like '%bengaluru%';
+-------------+-----------------------------------+--------------+-------------------------+
| customer_id | concat(first_name,' ',last_name) | phone_number | email                   |
+-------------+-----------------------------------+--------------+-------------------------+
|           1 | Sai Vighnessh Balaji              | 9908287108   | saivighnessh@gmail.com  |
|           2 | Madhu Kalla                       | 9988776655   | madhukalla@gmail.com    |
|           3 | Surya Narayanan                   | 9876543211   | suryanarayanan@gmail.com |
|           5 | Kavin Kaarthick                   | 8768975689   | Kaarthick@gmail.com     |
|           6 | Sree Tharan                       | 9056787654   | sreetharan74@gmail.com  |
|           8 | Sathesh M                         | 9940562789   | sathesh45@gmail.com     |
|           9 | Subitchan K                       | 8790365789   | subitchan@gmail.com     |
|          10 | Surendar P                        | 9944078956   | surenderp@gmail.com     |
+-------------+-----------------------------------+--------------+-------------------------+
8 rows in set (0.00 sec)
```

ix. Write a SQL query to List all current accounts with a balance greater than $1,000.

select * from accounts where account_type='current' and balance>86352.75;

```
mysql> select * from accounts where account_type='current' and balance>86352.75;
+------------+-------------+--------------+------------+
| account_id | customer_id | account_type | balance    |
+------------+-------------+--------------+------------+
|          2 |           2 | current      | 580000.00 |
|          5 |           5 | current      | 104000.00 |
|          8 |           8 | current      | 185000.90 |
+------------+-------------+--------------+------------+
3 rows in set (0.00 sec)
```

x. Write a SQL query to Retrieve all transactions for a specific account.

select concat(customers.first_name,' ',customers.last_name)as name,transactions.* from transactions inner join accounts on accounts.account_id=transactions.account_id inner join customers on customers.customer_id=accounts.customer_id where transactions.account_id=4;

```
mysql> select concat(customers.first_name,' ',customers.last_name)as name,transactions.* from transactions inner join accounts on accounts.account_id=transactions.account_id inner join cus
tomers on customers.customer_id=accounts.customer_id where transactions.account_id=4;
+------------+----------------+------------+------------------+--------+------------------+
| name       | transaction_id | account_id | transaction_type | amount | transaction_date |
+------------+----------------+------------+------------------+--------+------------------+
| Katta Rohit |              4 |          4 | deposit          | 8000.75 | 2025-02-05       |
+------------+----------------+------------+------------------+--------+------------------+
1 row in set (0.00 sec)
```

xi. Write a SQL query to Identify accounts where the balance is less than a specified overdraft limit.

select * from accounts where balance<-20000;

```
mysql> select * from accounts where balance<-20000;
+------------+-------------+--------------+------------+
| account_id | customer_id | account_type | balance    |
+------------+-------------+--------------+------------+
|         13 |           2 | savings      | -25000.00 |
+------------+-------------+--------------+------------+
1 row in set (0.00 sec)
```

xii. Write a SQL query to Calculate the interest accrued on savings accounts based on a given interest rate.

select account_id,customer_id,account_type,balance,balance*(3/100) as interest_accrued from accounts where account_type='savings';

```
mysql> select account_id,customer_id,account_type,balance,balance*(3/100) as interest_accrued from accounts where account_type='savings';
+------------+-------------+--------------+----------+------------------+
| account_id | customer_id | account_type | balance  | interest_accrued |
+------------+-------------+--------------+----------+------------------+
|          1 |           1 | savings      | 125000.75 |      3750.022500 |
|          4 |           4 | savings      |  15770.50 |       473.115000 |
|          7 |           7 | savings      |  30250.25 |       907.507500 |
|         10 |          10 | savings      | 350000.60 |     10500.018000 |
+------------+-------------+--------------+----------+------------------+
4 rows in set (0.00 sec)
```

TASK 3: Aggregate functions, Having, Order By, GroupBy and Joins:

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

   select concat(first_name,' ',last_name) as name,avg(accounts.balance) from customers inner join accounts on customers.customer_id=accounts.customer_id group by customers.customer_id;

```
mysql> select concat(first_name,' ',last_name) as name,avg(accounts.balance) from customers inner join accounts on customers.customer_id=accounts.customer_id group by customers.customer_id
;
+--------------------+----------------------+
| name               | avg(accounts.balance) |
+--------------------+----------------------+
| Sai Vighnessh Balaji |        125000.750000 |
| Madhu Kalla        |        580000.000000 |
| Surya Narayanan    |             0.000000 |
| Katta Rohit        |         15770.500000 |
| Kavin Kaarthick    |        104000.000000 |
| Sree Tharan        |             0.000000 |
| Sourabh V          |         30250.250000 |
| Sathesh M          |        185000.900000 |
| Subitchan K        |             0.000000 |
| Surendar P         |        350000.600000 |
+--------------------+----------------------+
10 rows in set (0.00 sec)
```

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

   select account_id,customer_id,balance from accounts order by balance desc;

```
mysql> select account_id,customer_id,balance from accounts order by balance desc;
+------------+-------------+------------+
| account_id | customer_id | balance    |
+------------+-------------+------------+
|          2 |           2 | 580000.00 |
|         10 |          10 | 350000.60 |
|          8 |           8 | 185000.90 |
|          1 |           1 | 125000.75 |
|          5 |           5 | 104000.00 |
|          7 |           7 |  30250.25 |
|          4 |           4 |  15770.50 |
|          3 |           3 |      0.00 |
|          6 |           6 |      0.00 |
|          9 |           9 |      0.00 |
+------------+-------------+------------+
10 rows in set (0.00 sec)
```

3. **Write a SQL query to Calculate Total Deposits for All Customers in specific date.**
   select account_id,sum(amount) as deposit from transactions where transaction_type ='deposit'and transaction_date='2025-01-25' group by account_id ;

```
mysql> select account_id,sum(amount) as deposit from transactions where transaction_type ='deposit'and transaction_date='2025-01-25' group by account_id ;
+------------+----------+
| account_id | deposit  |
+------------+----------+
|         10 | 12000.50 |
+------------+----------+
1 row in set (0.00 sec)
```

4. **Write a SQL query to Find the Oldest and Newest Customers.**
   select account_id,amount,transaction_date from transactions order by transaction_date;

```
mysql> select account_id,amount,transaction_date from transactions order by transaction_date;
+------------+-----------+------------------+
| account_id | amount    | transaction_date |
+------------+-----------+------------------+
|          7 | 10000.00  | 2025-01-03       |
|          1 |  5000.00  | 2025-01-05       |
|          3 |  1500.50  | 2025-01-25       |
|         10 | 12000.50  | 2025-01-25       |
|          4 |  8000.75  | 2025-02-05       |
|          9 |  3200.80  | 2025-02-18       |
|          6 |  2500.25  | 2025-02-22       |
|          8 |   750.00  | 2025-03-10       |
|          5 |   500.00  | 2025-03-12       |
|          2 |  2000.00  | 2025-03-15       |
+------------+-----------+------------------+
10 rows in set (0.00 sec)
```

5. **Write a SQL query to Retrieve transaction details along with the account type.**
   select transactions.*, accounts.account_type from transactions inner join accounts on accounts.account_id=transactions.account_id;

```
mysql> select transactions.*, accounts.account_type from transactions inner join accounts on accounts.account_id=transactions.account_id;
+----------------+------------+------------------+----------+------------------+--------------+
| transaction_id | account_id | transaction_type | amount   | transaction_date | account_type |
+----------------+------------+------------------+----------+------------------+--------------+
|              1 |          1 | deposit          |  5000.00 | 2025-01-05       | savings      |
|              2 |          2 | withdrawal       |  2000.00 | 2025-03-15       | current      |
|              3 |          3 | transfer         |  1500.50 | 2025-01-25       | zero_balance |
|              4 |          4 | deposit          |  8000.75 | 2025-02-05       | savings      |
|              5 |          5 | withdrawal       |   500.00 | 2025-03-12       | current      |
|              6 |          6 | transfer         |  2500.25 | 2025-02-22       | zero_balance |
|              7 |          7 | deposit          | 10000.00 | 2025-01-03       | savings      |
|              8 |          8 | withdrawal       |   750.00 | 2025-03-10       | current      |
|              9 |          9 | transfer         |  3200.80 | 2025-02-18       | zero_balance |
|             10 |         10 | deposit          | 12000.50 | 2025-01-25       | savings      |
+----------------+------------+------------------+----------+------------------+--------------+
10 rows in set (0.00 sec)
```

6. **Write a SQL query to Get a list of customers along with their account details.**
   select concat (first_name,' ',last_name),accounts.* from accounts inner join customers on accounts.customer_id=customers.customer_id;

```
mysql> select concat(first_name,' ',last_name),accounts.* from accounts inner join customers on accounts.customer_id=customers.customer_id;
+---------------------------------+------------+-------------+--------------+-----------+
| concat(first_name,' ',last_name) | account_id | customer_id | account_type | balance   |
+---------------------------------+------------+-------------+--------------+-----------+
| Sai Vighnessh Balaji            |          1 |           1 | savings      | 125000.75 |
| Madhu Kalla                     |          2 |           2 | current      | 580000.00 |
| Surya Narayanan                 |          3 |           3 | zero_balance |      0.00 |
| Katta Rohit                     |          4 |           4 | savings      |  15770.50 |
| Kavin Kaarthick                 |          5 |           5 | current      | 104000.00 |
| Sree Tharan                     |          6 |           6 | zero_balance |      0.00 |
| Sourabh V                       |          7 |           7 | savings      |  30250.25 |
| Sathesh M                       |          8 |           8 | current      | 185000.90 |
| Subitchan K                     |          9 |           9 | zero_balance |      0.00 |
| Surendar P                      |         10 |          10 | savings      | 350000.60 |
+---------------------------------+------------+-------------+--------------+-----------+
10 rows in set (0.00 sec)
```

7. **Write a SQL query to Retrieve transaction details along with customer information for a specific account.**
   select concat(customers.first_name,' ',customers.last_name)as name,transactions.* from transactions inner join accounts on accounts.account_id=transactions.account_id inner join customers on customers.customer_id=accounts.customer_id where transactions.account_id=2;

```
mysql> select concat(customers.first_name,' ',customers.last_name)as name,transactions.* from transactions inner join accounts on accounts.account_id=transactions.account_id inner join cus
tomers on customers.customer_id=accounts.customer_id where transactions.account_id=2;
+------------+----------------+------------+------------------+--------+------------------+
| name       | transaction_id | account_id | transaction_type | amount | transaction_date |
+------------+----------------+------------+------------------+--------+------------------+
| Madhu Kalla |              2 |          2 | withdrawal       | 2000.00 | 2025-03-15       |
+------------+----------------+------------+------------------+--------+------------------+
1 row in set (0.00 sec)
```

8. **Write a SQL query to Identify customers who have more than one account.**
   select customer_id, count(account_id) as account_count from accounts group by customer_id having count(account_id) > 1;

```
mysql> select customer_id, count(account_id) as account_count from accounts group by customer_id having count(account_id) > 1;
+-------------+---------------+
| customer_id | account_count |
+-------------+---------------+
|           2 |             2 |
+-------------+---------------+
1 row in set (0.00 sec)
```

9. **Write a SQL query to Calculate the difference in transaction amounts between deposits and withdrawals.**
   select (select sum(amount) from transactions where transaction_type='deposit')-(select sum(amount) from transactions where transaction_type='withdraw
   al') as difference;

```
mysql> select (select sum(amount) from transactions where transaction_type='deposit')-(select sum(amount) from transactions where transaction_type='withdraw
al') as difference;
+------------+
| difference |
+------------+
|   31751.25 |
+------------+
1 row in set (0.00 sec)
```

10. **Write a SQL query to Calculate the average daily balance for each account over a specified period.**
    select transactions.account_id, avg(accounts.balance) as average from transactions inner join accounts on transactions.account_id=accounts.account_id where transaction_date between '2025-01-03' and '2025-02-22'group by account_id;

```
mysql> select transactions.account_id, avg(accounts.balance) as average from transactions inner join accounts on transactions.account_id=accounts.account_id where transaction_date between
'2025-01-03' and '2025-02-22'group by account_id;
+------------+--------------+
| account_id | average      |
+------------+--------------+
|          1 | 125000.750000 |
|          3 |      0.000000 |
|          4 |  15770.500000 |
|          6 |      0.000000 |
|          7 |  30250.250000 |
|          9 |      0.000000 |
|         10 | 350000.600000 |
+------------+--------------+
7 rows in set (0.00 sec)
```

11. **Calculate the total balance for each account type.**
    select account_type,sum(balance) from accounts group by account_type;

```
mysql> select account_type,sum(balance) from accounts group by account_type;
+--------------+--------------+
| account_type | sum(balance) |
+--------------+--------------+
| savings      |    521022.10 |
| current      |    869000.90 |
| zero_balance |         0.00 |
+--------------+--------------+
3 rows in set (0.00 sec)
```

12. **Identify accounts with the highest number of transactions order by descending order.**
    select account_id, count(transaction_id) as count from transactions group by account_id order by count desc;

```
mysql> select account_id, count(transaction_id) as count from transactions group by account_id order by count desc;
+------------+-------+
| account_id | count |
+------------+-------+
|          1 |     1 |
|          2 |     1 |
|          3 |     1 |
|          4 |     1 |
|          5 |     1 |
|          6 |     1 |
|          7 |     1 |
|          8 |     1 |
|          9 |     1 |
|         10 |     1 |
+------------+-------+
10 rows in set (0.00 sec)
```

13. **List customers with high aggregate account balances, along with their account types.**
    select concat(first_name,' ',last_name) as name,account_type,sum(accounts.balance) as balance from accounts inner join customers on accounts.customer_id=customers.customer_id group by account_type,customers.customer_id;

```
mysql> select concat(first_name,' ',last_name) as name,account_type,sum(accounts.balance) as balance from accounts inner join customers on accounts.customer_id=customers.customer_id group
by account_type,customers.customer_id;
+---------------------+--------------+-----------+
| name                | account_type | balance   |
+---------------------+--------------+-----------+
| Sai Vighnessh Balaji | savings      | 125000.75 |
| Madhu Kalla         | current      | 580000.00 |
| Surya Narayanan     | zero_balance |      0.00 |
| Katta Rohit         | savings      |  15770.50 |
| Kavin Kaarthick     | current      | 104000.00 |
| Sree Tharan         | zero_balance |      0.00 |
| Sourabh V           | savings      |  30250.25 |
| Sathesh M           | current      | 185000.90 |
| Subitchan K         | zero_balance |      0.00 |
| Surendar P          | savings      | 350000.60 |
+---------------------+--------------+-----------+
10 rows in set (0.00 sec)
```

TASK 4: Subquery and its type:

1. **Retrieve the customer(s) with the highest account balance.**
   select customers.customer_id,customers.first_name from accounts inner join customers on customers.customer_id=accounts.customer_id where balance=(select max(balance) from accounts);

```
mysql> select customers.customer_id,customers.first_name from accounts inner join customers on customers.customer_id=accounts.customer_id where balance=(select max(balance) from accounts);
+-------------+------------+
| customer_id | first_name |
+-------------+------------+
|           2 | Madhu      |
+-------------+------------+
1 row in set (0.00 sec)
```

2. **Calculate the average account balance for customers who have more than one account.**
   select avg(balance) as avg_balance from accounts where customer_id in (select customer_id from accounts group by customer_id having count(account_id) > 1);

```
mysql> select avg(balance) as avg_balance from accounts where customer_id in (select customer_id from accounts group by customer_id having count(account_id)
 > 1);
+---------------+
| avg_balance   |
+---------------+
| 185000.000000 |
+---------------+
1 row in set (0.00 sec)
```

3. **Retrieve accounts with transactions whose amounts exceed the average transaction amount.**
   select transaction_id,account_id from transactions where amount>(select avg(amount) from transactions);

```
mysql> select transaction_id,account_id from transactions where amount>(select avg(amount) from transactions);
+----------------+------------+
| transaction_id | account_id |
+----------------+------------+
|              1 |          1 |
|              4 |          4 |
|              7 |          7 |
|             10 |         10 |
+----------------+------------+
4 rows in set (0.00 sec)
```

4. **Identify customers who have no recorded transactions.**
   select customer_id from accounts where account_id not in (select account_id from transactions);

```
mysql> select customer_id from accounts where account_id not in (select account_id from transactions);
+--------------+
| customer_id |
+--------------+
|           11 |
+--------------+
1 row in set (0.00 sec)
```

5. **Calculate the total balance of accounts with no recorded transactions.**
   select sum(balance) as total_balance from accounts where account_id not in (select account_id from transactions);

```
mysql> select sum(balance) as total_balance from accounts where account_id not in (select account_id from transactions);
+---------------+
| total_balance |
+---------------+
|      68760.00 |
+---------------+
1 row in set (0.00 sec)
```

6. **Retrieve transactions for accounts with the lowest balance.**
   select transactions.transaction_id,accounts.account_id from accounts inner join transactions on transactions.account_id=accounts.account_id where balance=(select min(balance) from accounts);

```
mysql> select transactions.transaction_id,accounts.account_id from accounts inner join transactions on transactions.account_id=accounts.account_id where balance=(select min(balance) from a
ccounts);
+----------------+------------+
| transaction_id | account_id |
+----------------+------------+
|              3 |          3 |
|              6 |          6 |
|              9 |          9 |
+----------------+------------+
3 rows in set (0.00 sec)
```

7. **Retrieve all transactions for a customer with a given customer_id.**
   select customers.customer_id,customers.first_name,transaction_id, transaction_date,transaction_type,amount from transactions inner join accounts on accounts.account_id=transactions.account_id inner join customers on customers.customer_id=accounts.customer_id where first_name=(select first_name from customers where customer_id=2);

```
mysql> select customers.customer_id,customers.first_name,transaction_id,transaction_date,transaction_type,amount from transactions inner join accounts on ac
counts.account_id=transactions.account_id inner join customers on customers.customer_id=accounts.customer_id where first_name=(select first_name from custom
ers where customer_id=2);
+-------------+------------+----------------+------------------+------------------+---------+
| customer_id | first_name | transaction_id | transaction_date | transaction_type | amount  |
+-------------+------------+----------------+------------------+------------------+---------+
|           2 | Madhu      |              2 | 2025-03-15       | withdrawal       | 2000.00 |
+-------------+------------+----------------+------------------+------------------+---------+
1 row in set (0.00 sec)
```

8. **Identify customers who have accounts of multiple types.**
   select customer_id from accounts where customer_id in (select customer_id from accounts group by customer_id having count(account_type) > 1);

```
mysql> select customer_id from accounts where customer_id in (select customer_id from accounts group by customer_id having count(account_type) > 1);
+-------------+
| customer_id |
+-------------+
|           2 |
|           2 |
|           2 |
+-------------+
3 rows in set (0.00 sec)
```

9. **Calculate the total balance for each account type, including a subquery within the SELECT clause.**
   select account_type,sum(balance) from accounts where account_id in (select account_id from accounts where balance>=0) group by account_type;

```
mysql> select account_type,sum(balance) from accounts where account_id in (select account_id from accounts where balance>=0) group by account_type;
+--------------+--------------+
| account_type | sum(balance) |
+--------------+--------------+
| savings      |    521022.10 |
| current      |    869000.90 |
| zero_balance |         0.00 |
+--------------+--------------+
3 rows in set (0.00 sec)
```

10. **Calculate the percentage of each account type out of the total number of accounts.**
    select account_type, ((count(account_id))/(select count(account_id) from accounts))*100 as percentage from accounts group by account_type;

```
mysql> select account_type,((count(account_id))/(select count(account_id) from accounts))*100 as percentage from accounts group by account_type;
+--------------+------------+
| account_type | percentage |
+--------------+------------+
| savings      |    53.8462 |
| current      |    23.0769 |
| zero_balance |    23.0769 |
+--------------+------------+
3 rows in set (0.00 sec)
```