**J524 THARUN.D**

**Hexaware Assignment – SQL Banking System**

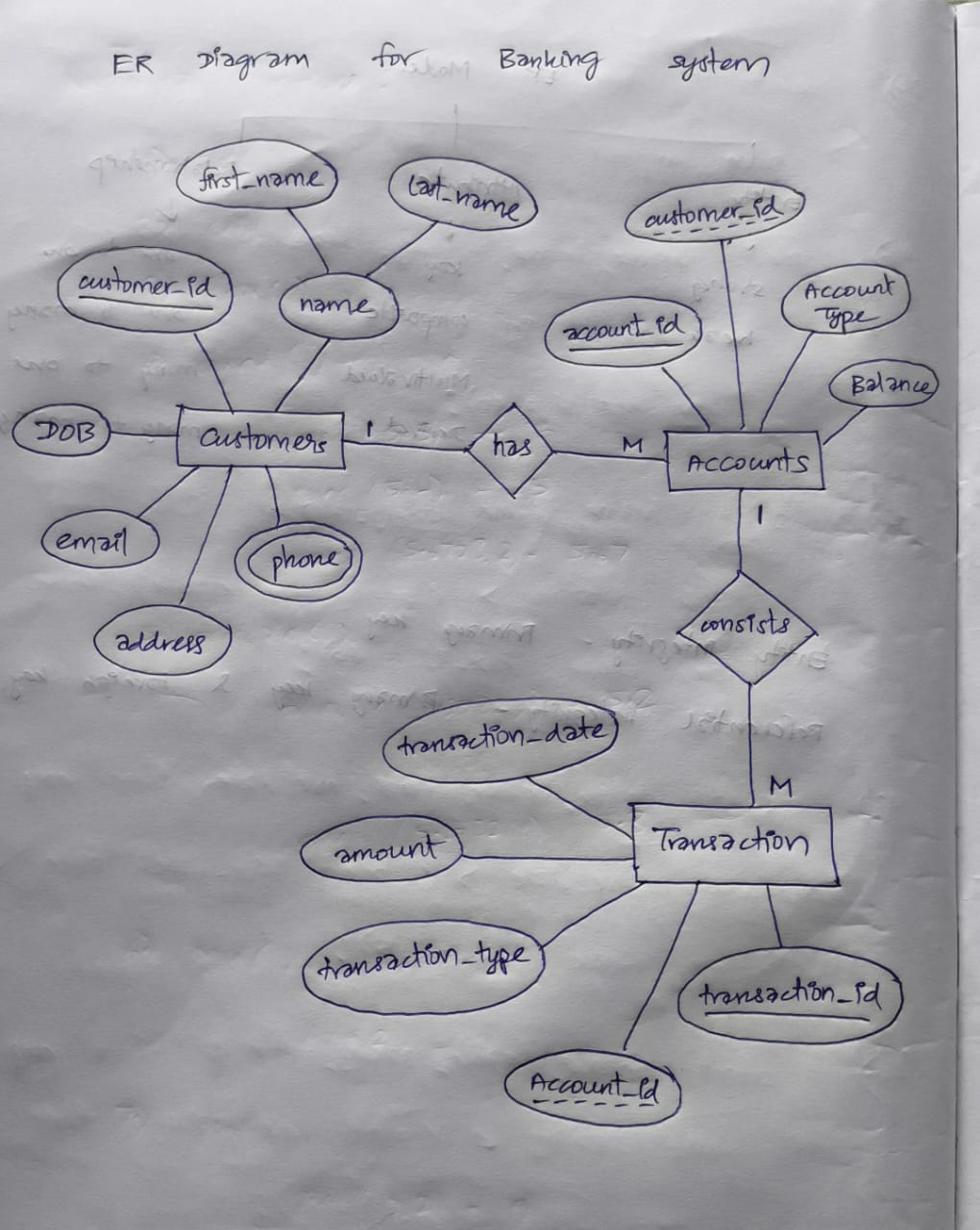
**TASK 1**

1. Create the database named "HMBank"

mysql> create database hmbank;

mysql> use hmbank;

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



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

mysql> create table Customers(cust\_id int primary key, first\_name varchar(20), last\_name varchar(20), dob date, email text, phone\_num varchar(15), address text);

mysql> create table Accounts(acc\_id int primary key, cust\_id int, acc\_type varchar(15), balance decimal(15,2) default 0.00, foreign key(cust\_id) references Customers(cust\_id) on delete cascade);

mysql> create table Transactions(trans\_id int primary key, acc\_id int, trans\_type varchar(15), amount decimal(15,2) check(amount>0), trans\_date date, foreign key(acc\_id) references Accounts(acc\_id) on delete cascade);

**TASK 2**

1. Insert at least 10 sample records into each of the following tables.

mysql> insert into Customers values (1, "Rahul", "Sharma", "1994-10-04", "rahul@gmail.com", "5656565656", "13, nehru street, chennai"), (2, "Akash", "Gupta", "1990-04-10", "akash@yahoo.com", "1234567890", "10, gandhi street, chennai"), (3, "Suresh", "Raina", "1993-10-03", "raina@gmail.com", "3333333333", "22, marina, chennai" ), (4, "Virat", "Kohli", "1985-11-07", "kohli@gmail.com", "4949494949", "18, chinnaswamy, bangalore"), (5, "David", "Miller", "1999-06-01", "miller@yahoo.com", "1010101010", "17, nehru street, chennai"), (6, "Gopal", "Varma", "2000-01-08", "gopal@gmail.com", "2323232323", "23, main street, chennai"), (7, "MS","Dhoni", "1980-07-07", "dhoni@gmail.com", "7777777777", "7, marina, chennai"), (8, "Ravindra", "Jadeja", "1990-11-24","jadeja@gmail.com", "8888888888", "8, marina, chennai"), (9, "DJ", "Bravo","1993-10-12", "bravo@yahoo.com", "9999999999", "9, nehru st, chennai"),(10, "Shivam", "Dube", "1984-02-19", "dube@gmail.com", "2525252525", "58, cross street, chennai");

mysql> insert into accounts (acc\_id, cust\_id, acc\_type, balance) values (1, 5, "current", 5000.00), (2, 3, "zero\_balance", 0.00), (3, 7, "savings", 75000.90), (4, 1, "savings", 15000.50), (5, 10, "zero\_balance", 0.00), (6, 6, "zero\_balance", 0.00), (7, 2, "current", 22000.75), (8, 9, "savings", 9800.10), (9, 4, "savings", 45000.25), (10, 8, "current", 12000.30), (11, 1, "current", 8000.00), (12, 2, "savings", 16000.40);

mysql> insert into transactions (trans\_id, acc\_id, trans\_type, amount, trans\_date) values (1, 1, "deposit", 5000.00, "2024-03-01"), (2, 2, "withdrawal", 2000.00, "2024-03-02"), (3, 3, "deposit", 10000.50, "2024-03-03"), (4, 4, "transfer", 2500.00, "2024-03-04"), (5, 5, "deposit", 7500.00, "2024-03-05"), (6, 6, "withdrawal", 1500.00, "2024-03-06"), (7, 7, "transfer", 3000.00, "2024-03-07"), (8, 8, "deposit", 8000.75, "2024-03-08"), (9, 9, "withdrawal", 4000.00, "2024-03-09"), (10, 10, "deposit", 6000.00, "2024-03-10"), (11, 11, "transfer", 2000.50, "2024-03-11"), (12, 12, "deposit", 9000.00, "2024-03-12");

1. Write SQL queries for the following tasks:
2. Write a SQL query to retrieve the name, account type and email of all customers.

mysql> select concat(first\_name,' ',last\_name) as name, acc\_type, email from Customers, Accounts where Customers.cust\_id = Accounts.cust\_id;

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

mysql> select Customers.cust\_id,first\_name,Accounts.acc\_id,acc\_type,trans\_id,trans\_type,amount,trans\_date from Customers,Accounts,Transactions where Customers.cust\_id=Accounts.cust\_id and Accounts.acc\_id=Transactions.acc\_id;

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

mysql> update accounts set balance=balance+2000 where acc\_id=5;

1. Write a SQL query to Combine first and last names of customers as a full\_name.

mysql> select concat(first\_name," ",last\_name) as full\_name from Customers;

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

mysql> delete from accounts where balance=0 and acc\_type="savings";

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

mysql> select \* from Customers where address like"%chennai%";

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

mysql> select balance from Accounts where acc\_id=5;

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

mysql> select \* from Accounts where acc\_type="current" and balance>1000;

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

mysql> select \* from transactions where acc\_id=5;

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

mysql> select acc\_id,cust\_id,balance,balance\*(0.1) as "interest\_amt" from accounts where acc\_type="savings";

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

mysql> select \* from accounts where balance<2000;

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

mysql> select \* from Customers where address not like "%chennai%";

**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;

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

mysql> select balance from accounts order by balance desc limit 10;

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

mysql> select sum(amount) as "Total Deposits" from transactions where trans\_type="deposit" and trans\_date="2024-03-18";

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

mysql> select c.cust\_id, concat(first\_name,' ',last\_name) as "Oldest Customer" from Customers c join Accounts a on c.cust\_id=a.cust\_id join Transactions t on a.acc\_id=t.acc\_id order by trans\_date desc limit 1;

mysql> select c.cust\_id, concat(first\_name,' ',last\_name) as "Newest Customer" from Customers c join Accounts a on c.cust\_id=a.cust\_id join Transactions t on a.acc\_id=t.acc\_id order by trans\_date limit 1;

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

mysql> select t.\*,a.acc\_type from accounts a right join transactions t on a.acc\_id=t.acc\_id;

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

mysql> select c.cust\_id,first\_name,last\_name,dob,email,phone\_num,address,acc\_id,acc\_type,balance from customers c left join accounts a on c.cust\_id=a.cust\_id;

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

mysql> select t.\*,c.cust\_id,concat(c.first\_name,' ',c.last\_name) as "Cust Name",phone\_num from transactions t left join accounts a on t.acc\_id=a.acc\_id join customers c on a.cust\_id=c.cust\_id where t.acc\_id=5;

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

mysql> select c.cust\_id,c.first\_name,c.last\_name,count(a.acc\_id) from customers c join accounts a on c.cust\_id=a.cust\_id group by c.cust\_id having count(a.acc\_id)>1;

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

mysql> select (sum(case when trans\_type="deposit" then amount end )-sum(case when trans\_type="withdrawal" then amount end)) as "Difference" from transactions;

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

mysql> select acc\_id, sum(amount)/count(distinct trans\_date) from transactions where trans\_date between "2024-03-03" and "2024-03-17" group by acc\_id;

1. Calculate the total balance for each account type.

mysql> select acc\_type, sum(balance) from accounts group by acc\_type;

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

mysql> select acc\_id, count(\*) as "No\_of\_Transactions" from transactions group by acc\_id order by No\_of\_Transactions desc;

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

mysql> select c.cust\_id,c.first\_name,a.acc\_type,sum(a.balance) from customers c join accounts a on c.cust\_id = a.cust\_id group by c.cust\_id,a.acc\_type having sum(balance)>10000 ;

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

mysql> select acc\_id,amount,trans\_date,count(\*) from transactions group by acc\_id,amount,trans\_date having count(\*) >1;

**TASK 4**

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

mysql> select c.cust\_id,c.first\_name,balance from customers c join accounts a on c.cust\_id=a.cust\_id where balance = (select max(balance) from accounts);

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

mysql> select cust\_id, avg(balance) from accounts group by cust\_id having count(acc\_id)>1;

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

mysql> select \* from transactions where amount > (select avg(amount) from transactions);

1. Identify customers who have no recorded transactions.

mysql> select cust\_id from customers where cust\_id not in (select cust\_id from accounts where acc\_id in (select acc\_id from transactions));

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

mysql> select sum(balance) from accounts where acc\_id not in (select acc\_id from transactions);

1. Retrieve transactions for accounts with the lowest balance.

mysql> select \* from transactions where acc\_id in (select acc\_id from accounts where balance in (select min(balance) from accounts));

1. Identify customers who have accounts of multiple types.

mysql> select cust\_id,count(acc\_type) from accounts group by cust\_id having count(acc\_type)>1;

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

mysql> select acc\_type, (count(\*)/(select count(\*) from accounts))\*100.0 from accounts group by acc\_type;

1. Retrieve all transactions for a customer with a given customer\_id.

mysql> select \* from transactions where acc\_id in (select acc\_id from accounts where cust\_id=7);

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

mysql> select distinct acc\_type, (select sum(balance) from accounts a where a.acc\_type = b.acc\_type) as total\_balance from accounts b;