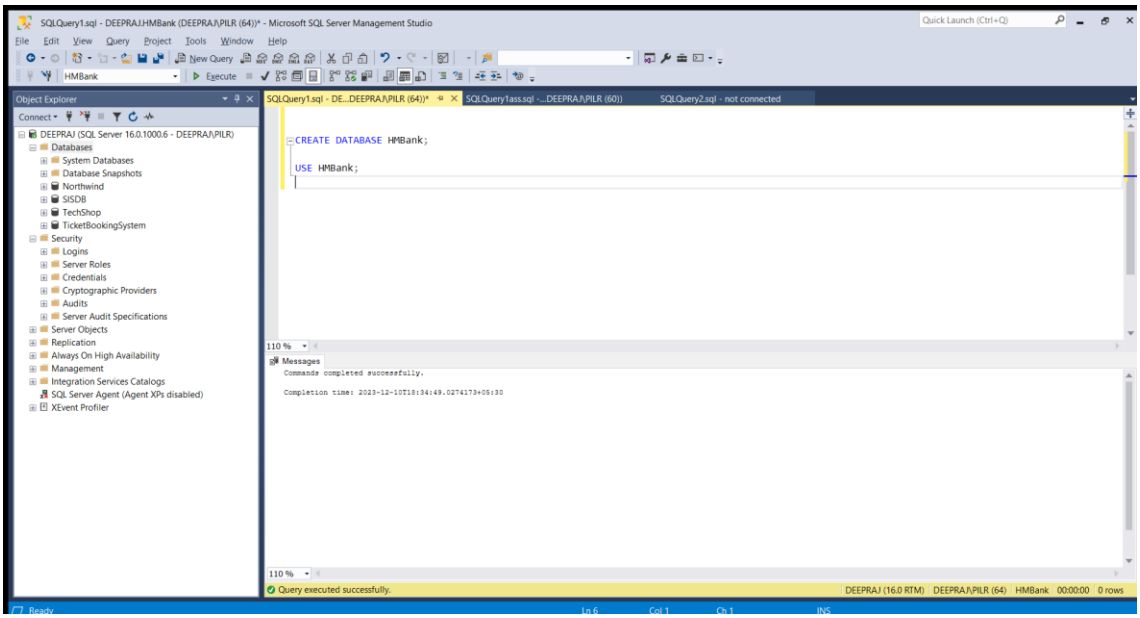
Banking System

Tasks 1: Database Design:

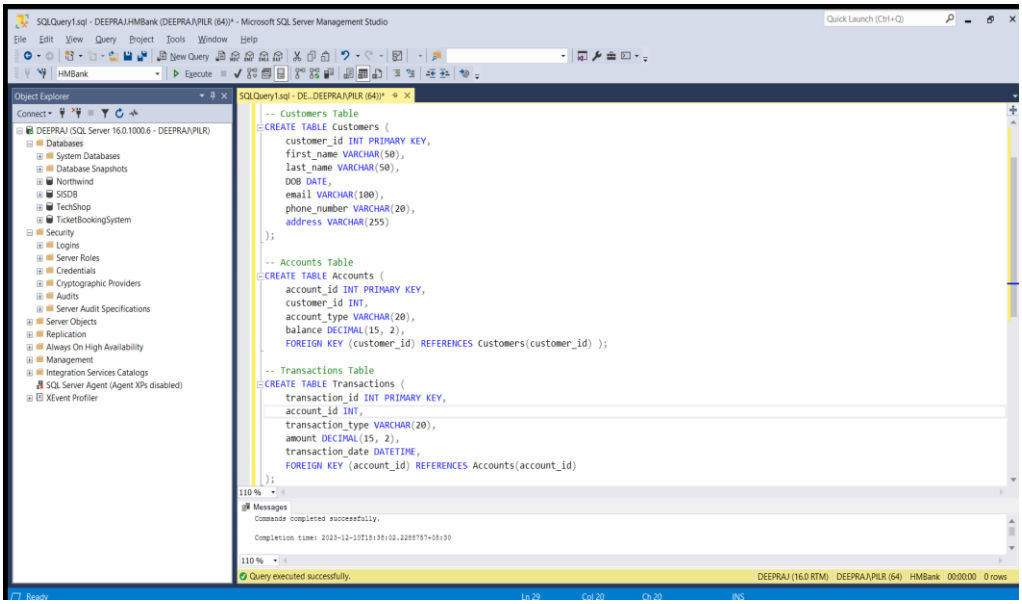
1. Create the database named "HMBank"

Ans)



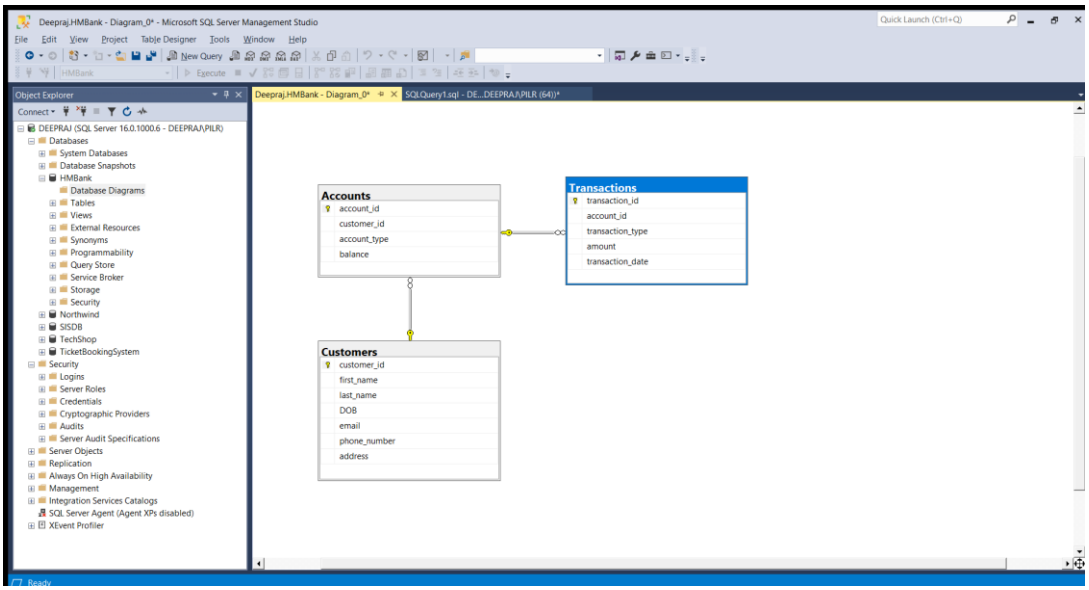
2. Define the schema for the Customers, Accounts, and Transactions tables based on the provided schema.

Ans)



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

Ans)

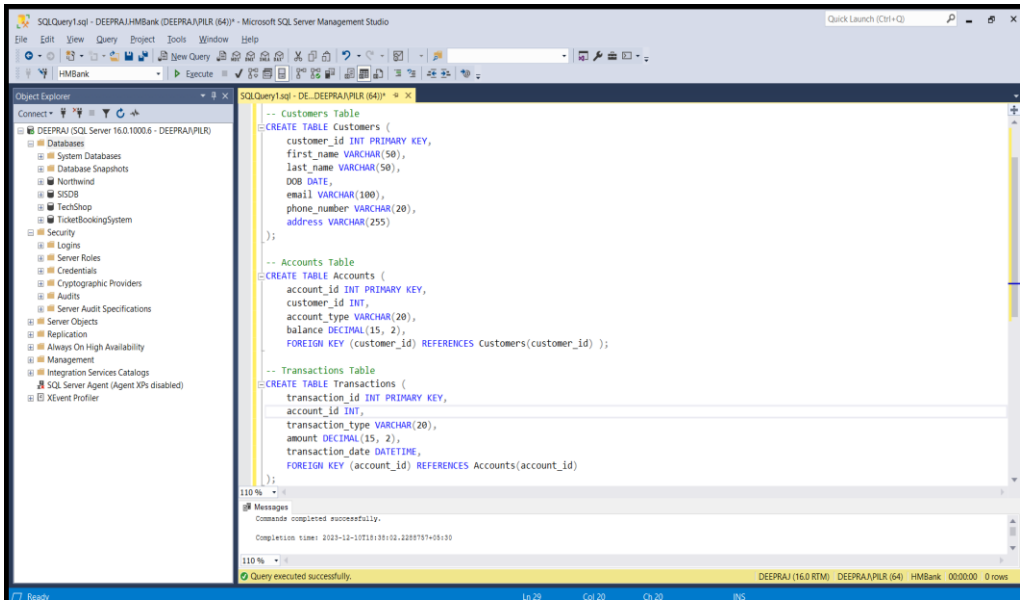


4. Create appropriate Primary Key and Foreign Key constraints for referential integrity.

Ans) Already did in 2 question.

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

Ans)



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

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

Ans)

```sql
-- Insert into Customers table
INSERT INTO Customers (customer_id, first_name, last_name, DOB, email, phone_number, address) VALUES
(1, 'Amit', 'Sharma', '1985-07-12', 'amit.sharma@email.com', '9876543210', '12A MG Road, Mumbai'),
(2, 'Priya', 'Patel', '1990-03-25', 'priya.patel@email.com', '8765432109', '45B Main Street, Delhi'),
(3, 'Rajesh', 'Kumar', '1980-12-05', 'rajesh.kumar@email.com', '7654321098', '78C Residency Road, Bangalore'),
(4, 'Anjali', 'Verma', '1988-06-18', 'anjali.verma@email.com', '6543210987', '34D Park Street, Kolkata'),
(5, 'Sandeep', 'Singh', '1995-02-28', 'sandeep.singh@email.com', '5432109876', '56E Brigade Road, Bangalore'),
(6, 'Neha', 'Gupta', '1987-09-15', 'neha.gupta@email.com', '4321098765', '23F Church Street, Mumbai'),
(7, 'Rahul', 'Saxena', '1992-11-08', 'rahul.saxena@email.com', '3210987654', '67G MG Road, Pune'),
(8, 'Pooja', 'Mishra', '1983-04-02', 'pooja.mishra@email.com', '2109876543', '89H Park Street, Kolkata'),
(9, 'Vikram', 'Yadav', '1989-08-20', 'vikram.yadav@email.com', '1098765432', '78A Brigade Road, Bangalore'),
(10, 'Sarita', 'Malhotra', '1982-01-10', 'sarita.malhotra@email.com', '9876543210', '45B Main Street, Delhi');

-- Insert into Accounts table
INSERT INTO Accounts (account_id, customer_id, account_type, balance) VALUES
(101, 1, 'savings', 15000.00),
(102, 2, 'current', 5000.00),
(103, 3, 'savings', 120000.00),
(104, 4, 'current', 8000.00),
(105, 5, 'savings', 30000.00),
(106, 6, 'current', 10000.00),
(107, 7, 'savings', 50000.00),
(108, 8, 'current', 2000.00),
(109, 9, 'savings', 25000.00),
(110, 10, 'current', 7000.00);
```

110 %

Messages

```
(10 rows affected)

(10 rows affected)

(10 rows affected)

Completion time: 2023-12-10T19:43:51.8215680+05:30
```

```sql
-- Insert into Transactions table
INSERT INTO Transactions (transaction_id, account_id, transaction_type, amount, transaction_date)
VALUES
(1001, 101, 'deposit', 5000.00, '2023-01-10 14:30:00'),
(1002, 102, 'withdrawal', 2000.00, '2023-02-15 09:45:00'),
(1003, 103, 'deposit', 15000.00, '2023-03-20 11:20:00'),
(1004, 104, 'withdrawal', 1000.00, '2023-04-05 16:00:00'),
(1005, 105, 'deposit', 10000.00, '2023-05-12 13:15:00'),
(1006, 106, 'withdrawal', 500.00, '2023-06-18 10:30:00'),
(1007, 107, 'deposit', 20000.00, '2023-07-25 08:00:00'),
(1008, 108, 'withdrawal', 500.00, '2023-08-10 17:45:00'),
(1009, 109, 'deposit', 8000.00, '2023-09-05 14:00:00'),
(1010, 110, 'withdrawal', 1000.00, '2023-10-20 12:30:00');
```

110 %

Messages

```
(10 rows affected)

(10 rows affected)

(10 rows affected)

Completion time: 2023-12-10T19:43:51.8215680+05:30
```

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

   Ans)

```sql
SELECT first_name, last_name, account_type, email FROM Customers
JOIN Accounts ON Customers.customer_id = Accounts.customer_id;
```

110 %

Results | Messages

	first_name	last_name	account_type	email
1	Amit	Sharma	savings	amit.sharma@email.com
2	Priya	Patel	current	priya.patel@email.com
3	Rajesh	Kumar	savings	rajesh.kumar@email.com
4	Anjali	Verma	current	anjali.verma@email.com
5	Sandeep	Singh	savings	sandeep.singh@email.com
6	Neha	Gupta	current	neha.gupta@email.com
7	Rahul	Saxena	savings	rahul.saxena@email.com
8	Pooja	Mishra	current	pooja.mishra@email.com
9	Vikram	Yadav	savings	vikram.yadav@email.com
10	Sarita	Malhotra	current	sarita.malhotra@email.com

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

   Ans)

```
SELECT first_name, last_name, transaction_id, transaction_type, amount, transaction_date
FROM Customers
JOIN Accounts ON Customers.customer_id = Accounts.customer_id
JOIN Transactions ON Accounts.account_id = Transactions.account_id;
```

110 %

Results | Messages

	first_name	last_name	transaction_id	transaction_type	amount	transaction_date
1	Amit	Sharma	1001	deposit	5000.00	2023-01-10 14:30:00.000
2	Priya	Patel	1002	withdrawal	2000.00	2023-02-15 09:45:00.000
3	Rajesh	Kumar	1003	deposit	15000.00	2023-03-20 11:20:00.000
4	Anjali	Verma	1004	withdrawal	1000.00	2023-04-05 16:00:00.000
5	Sandeep	Singh	1005	deposit	10000.00	2023-05-12 13:15:00.000
6	Neha	Gupta	1006	withdrawal	500.00	2023-06-18 10:30:00.000
7	Rahul	Saxena	1007	deposit	20000.00	2023-07-25 08:00:00.000
8	Pooja	Mishra	1008	withdrawal	500.00	2023-08-10 17:45:00.000
9	Vikram	Yadav	1009	deposit	8000.00	2023-09-05 14:00:00.000
10	Sarita	Malhotra	1010	withdrawal	1000.00	2023-10-20 12:30:00.000

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

Ans)

```
UPDATE Accounts
SET balance = balance + 1000.00
WHERE account_id = 101;
```

110 %

Messages

```
(1 row affected)

Completion time: 2023-12-10T19:49:17.6981226+05:30
```

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

Ans)

```
SELECT CONCAT(first_name, ' ', last_name) AS full_name
FROM Customers;
```

110 %

Results | Messages

	full_name
1	Amit Sharma
2	Priya Patel
3	Rajesh Kumar
4	Anjali Verma
5	Sandeep Singh
6	Neha Gupta
7	Rahul Saxena
8	Pooja Mishra
9	Vikram Yadav
10	Sarita Malhotra

Query executed successfully.

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

Ans)

```
DELETE FROM Accounts
WHERE balance = 0 AND account_type = 'savings';
```

110 %

Messages

```
(0 rows affected)

Completion time: 2023-12-10T20:50:35.6579255+05:30
```

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

Ans)

```
SELECT * FROM Customers
WHERE address LIKE '%Delhi%';
```

110 %

Results | Messages

	customer_id	first_name	last_name	DOB	email	phone_number	address
1	2	Priya	Patel	1990-03-25	priya.patel@email.com	8765432109	45B Main Street, Delhi
2	10	Sarita	Malhotra	1982-01-10	sarita.malhotra@email.com	9876543210	45B Main Street, Delhi

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

Ans)

```
SELECT account_id, balance FROM Accounts
WHERE account_id = 101;
```

110 %

Results | Messages

	account_id	balance
1	101	16000.00

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

Ans)

```
SELECT * FROM Accounts
WHERE account_type = 'current' AND balance > 1000.00;
```

110 %

Results | Messages

	account_id	customer_id	account_type	balance
1	102	2	current	5000.00
2	104	4	current	8000.00
3	106	6	current	10000.00
4	108	8	current	2000.00
5	110	10	current	7000.00

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

Ans)

```sql
SELECT * FROM Transactions
WHERE account_id = 102;
```

110 %

Results | Messages

	transaction_id	account_id	transaction_type	amount	transaction_date
1	1002	102	withdrawal	2000.00	2023-02-15 09:45:00.000

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

Ans)

```sql
SELECT account_id, balance * 0.05 AS interest_accrued
FROM Accounts WHERE account_type = 'savings';
```

110 %

Results | Messages

	account_id	interest_accrued
1	101	800.0000
2	103	6000.0000
3	105	1500.0000
4	107	2500.0000
5	109	1250.0000

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

Ans)

```sql
SELECT * FROM Accounts
WHERE balance < -100.00; -- Assuming overdraft limit is -$100.00
```

110 %

Results | Messages

	account_id	customer_id	account_type	balance

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

Ans)

```
SELECT * FROM Customers
WHERE address NOT LIKE '%Delhi%';
```

10 %

Results | Messages

	customer_id	first_name	last_name	DOB	email	phone_number	address
1	1	Amit	Sharma	1985-07-12	amit.sharma@email.com	9876543210	12A MG Road, Mumbai
2	3	Rajesh	Kumar	1980-12-05	rajesh.kumar@email.com	7654321098	78C Residency Road, Bangalore
3	4	Anjali	Verma	1988-06-18	anjali.verma@email.com	6543210987	34D Park Street, Kolkata
4	5	Sandeep	Singh	1995-02-28	sandeep.singh@email.com	5432109876	56E Brigade Road, Bangalore
5	6	Neha	Gupta	1987-09-15	neha.gupta@email.com	4321098765	23F Church Street, Mumbai
6	7	Rahul	Saxena	1992-11-08	rahul.saxena@email.com	3210987654	67G MG Road, Pune
7	8	Pooja	Mishra	1983-04-02	pooja.mishra@email.com	2109876543	89H Park Street, Kolkata
8	9	Vikram	Yadav	1989-08-20	vikram.yadav@email.com	1098765432	78A Brigade Road, Bangalore

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

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

```
SELECT AVG(balance) AS average_balance
FROM Accounts;
```

110 %

Results | Messages

	average_balance
1	27300.000000

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

```
SELECT TOP 10 customer_id, account_id, account_type, balance
FROM Accounts
ORDER BY balance DESC;
```

110 %

Results | Messages

	customer_id	account_id	account_type	balance
1	3	103	savings	120000.00
2	7	107	savings	50000.00
3	5	105	savings	30000.00
4	9	109	savings	25000.00
5	1	101	savings	16000.00
6	6	106	current	10000.00
7	4	104	current	8000.00
8	10	110	current	7000.00
9	2	102	current	5000.00
10	8	108	current	2000.00

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

```
SELECT account_id, SUM(amount) AS total_deposits
FROM Transactions
WHERE transaction_type = 'deposit' AND CAST(transaction_date AS DATE) = '2023-01-10'
GROUP BY account_id;
```

110 %

Results | Messages

	account_id	total_deposits
1	101	5000.00

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

Ans)

```
SELECT MIN(DOB) AS oldest_customer, MAX(DOB) AS newest_customer
FROM Customers;
```

10 %

Results | Messages

	oldest_customer	newest_customer
1	1980-12-05	1995-02-28

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

Ans)

```
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;
```

110 %

Results | Messages

	transaction_id	account_id	transaction_type	amount	transaction_date	account_type
1	1001	101	deposit	5000.00	2023-01-10 14:30:00.000	savings
2	1002	102	withdrawal	2000.00	2023-02-15 09:45:00.000	current
3	1003	103	deposit	15000.00	2023-03-20 11:20:00.000	savings
4	1004	104	withdrawal	1000.00	2023-04-05 16:00:00.000	current
5	1005	105	deposit	10000.00	2023-05-12 13:15:00.000	savings
6	1006	106	withdrawal	500.00	2023-06-18 10:30:00.000	current
7	1007	107	deposit	20000.00	2023-07-25 08:00:00.000	savings
8	1008	108	withdrawal	500.00	2023-08-10 17:45:00.000	current
9	1009	109	deposit	8000.00	2023-09-05 14:00:00.000	savings
10	1010	110	withdrawal	1000.00	2023-10-20 12:30:00.000	current

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

Ans)

```
SELECT c.customer_id, c.first_name, c.last_name, c.DOB, 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;
```

110 %

Results | Messages

	customer_id	first_name	last_name	DOB	email	phone_number	address	account_id	account_type	balance
1	1	Amit	Sharma	1985-07-12	amit.sharma@email.com	9876543210	12A MG Road, Mumbai	101	savings	16000.00
2	2	Priya	Patel	1990-03-25	priya.patel@email.com	8765432109	45B Main Street, Delhi	102	current	5000.00
3	3	Rajesh	Kumar	1980-12-05	rajesh.kumar@email.com	7654321098	78C Residency Road, Bangalore	103	savings	120000.00
4	4	Anjali	Verma	1988-06-18	anjali.verma@email.com	6543210987	34D Park Street, Kolkata	104	current	8000.00
5	5	Sandeep	Singh	1995-02-28	sandeep.singh@email.com	5432109876	56E Brigade Road, Bangalore	105	savings	30000.00
6	6	Neha	Gupta	1987-09-15	neha.gupta@email.com	4321098765	23F Church Street, Mumbai	106	current	10000.00
7	7	Rahul	Saxena	1992-11-08	rahul.saxena@email.com	3210987654	67G MG Road, Pune	107	savings	50000.00
8	8	Pooja	Mishra	1983-04-02	pooja.mishra@email.com	2109876543	89H Park Street, Kolkata	108	current	2000.00
9	9	Vikram	Yadav	1989-08-20	vikram.yadav@email.com	1098765432	78A Brigade Road, Bangalore	109	savings	25000.00
10	10	Sarita	Malhotra	1982-01-10	sarita.malhotra@email.com	9876543210	45B Main Street, Delhi	110	current	7000.00

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

Ans)

```
SELECT c.customer_id, c.first_name, c.last_name, c.email, t.transaction_id, t.transaction_type,
t.amount, t.transaction_date FROM Customers c
JOIN Accounts a ON c.customer_id = a.customer_id
JOIN Transactions t ON a.account_id = t.account_id
WHERE a.account_id = 101; -- Specify the account_id
```

110 %

Results | Messages

	customer_id	first_name	last_name	email	transaction_id	transaction_type	amount	transaction_date
1	1	Amit	Sharma	amit.sharma@email.com	1001	deposit	5000.00	2023-01-10 14:30:00.000

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

Ans)

```sql
SELECT customer_id, COUNT(account_id) AS num_of_accounts
FROM Accounts
GROUP BY customer_id
HAVING COUNT(account_id) > 1;
```

.10 %

Results | Messages

customer_id	num_of_accounts

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

Ans)

```sql
SELECT transaction_type, SUM(CASE WHEN transaction_type = 'deposit' THEN amount ELSE -amount END) AS net_amount
FROM Transactions
GROUP BY transaction_type;
```

110 %

Results | Messages

	transaction_type	net_amount
1	deposit	58000.00
2	withdrawal	-5000.00

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

Ans)

```sql
SELECT a.account_id, AVG(a.balance) AS average_daily_balance
FROM Accounts a JOIN Transactions t ON a.account_id = t.account_id
WHERE CAST(t.transaction_date AS DATE) BETWEEN '2023-01-01' AND '2023-04-10'
GROUP BY a.account_id;
```

110 %

Results | Messages

	account_id	average_daily_balance
1	101	16000.000000
2	102	5000.000000
3	103	120000.000000
4	104	8000.000000

11. Calculate the total balance for each account type.

Ans)

```sql
SELECT account_type, SUM(balance) AS total_balance
FROM Accounts
GROUP BY account_type;
```

110 %

Results | Messages

	account_type	total_balance
1	current	32000.00
2	savings	241000.00

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

Ans)

```
SELECT account_id, COUNT(transaction_id) AS num_of_transactions
FROM Transactions
GROUP BY account_id
ORDER BY num_of_transactions DESC;
```

110 %

Results | Messages

	account_id	num_of_transactions
1	101	1
2	102	1
3	103	1
4	104	1
5	105	1
6	106	1
7	107	1
8	108	1
9	109	1
10	110	1

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

Ans)

```
SELECT c.customer_id, a.account_type, SUM(a.balance) AS aggregate_balance
FROM Customers c
JOIN Accounts a ON c.customer_id = a.customer_id
GROUP BY c.customer_id, a.account_type;
```

110 %

Results | Messages

	customer_id	account_type	aggregate_balance
1	2	current	5000.00
2	4	current	8000.00
3	6	current	10000.00
4	8	current	2000.00
5	10	current	7000.00
6	1	savings	16000.00
7	3	savings	120000.00
8	5	savings	30000.00
9	7	savings	50000.00
10	9	savings	25000.00

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

Ans)

```
SELECT  account_id,amount, transaction_date, COUNT(*) AS duplicate_count
FROM Transactions
GROUP BY amount, transaction_date, account_id
HAVING COUNT(*) > 1;
```

110 %

Results | Messages

account_id	amount	transaction_date	duplicate_count

Tasks 4: Subquery and its type:

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

```
SELECT customer_id, first_name, last_name
FROM Customers
WHERE customer_id = (SELECT TOP 1 customer_id FROM Accounts ORDER BY balance DESC);
```

110 %

Results | Messages

	customer_id	first_name	last_name
1	3	Rajesh	Kumar

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

```
SELECT customer_id, AVG(balance) AS average_balance
FROM Accounts
GROUP BY customer_id
HAVING COUNT(account_id) > 1;
```

110 %

Results | Messages

customer_id	average_balance

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

```
SELECT account_id, transaction_id, amount
FROM Transactions
WHERE amount > (SELECT AVG(amount) FROM Transactions);
```

110 %

Results | Messages

	account_id	transaction_id	amount
1	103	1003	15000.00
2	105	1005	10000.00
3	107	1007	20000.00
4	109	1009	8000.00

4. Identify customers who have no recorded transactions.
   Ans)

```
SELECT customer_id, first_name, last_name
FROM Customers
WHERE customer_id NOT IN (SELECT DISTINCT customer_id FROM Transactions);
```

110 %

Results | Messages

customer_id	first_name	last_name

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

```
SELECT SUM(balance) AS total_balance_no_transactions
FROM Accounts
WHERE account_id NOT IN (SELECT DISTINCT account_id FROM Transactions);
```

110 %

Results | Messages

	total_balance_no_transactions
1	NULL

6. Retrieve transactions for accounts with the lowest balance.
   Ans)

```
SELECT account_id, transaction_id, amount FROM Transactions
WHERE account_id = (SELECT TOP 1 account_id FROM Accounts ORDER BY balance ASC);
```

110 %

Results | Messages

	account_id	transaction_id	amount
1	108	1008	500.00

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

```
SELECT customer_id, first_name, last_name FROM Customers
WHERE customer_id IN (SELECT customer_id FROM Accounts GROUP BY customer_id HAVING COUNT(DISTINCT account_type) > 1);
```

110 %

Results | Messages

customer_id	first_name	last_name

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

```
SELECT account_type, COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Accounts) AS percentage
FROM Accounts GROUP BY account_type;
```

110 %

Results | Messages

	account_type	percentage
1	current	50.000000000000
2	savings	50.000000000000

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

Ans)

```
SELECT * FROM Transactions
WHERE account_id IN
(SELECT account_id FROM Accounts WHERE customer_id = 1); -- Replace 1 with the desired customer_id
```

110 %

Results | Messages

	transaction_id	account_id	transaction_type	amount	transaction_date
1	1001	101	deposit	5000.00	2023-01-10 14:30:00.000

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

Ans)

```
SELECT account_type,
       (SELECT SUM(balance) FROM Accounts WHERE account_type = a.account_type) AS total_balance
FROM Accounts a
GROUP BY account_type;
```

110 %

Results | Messages

	account_type	total_balance
1	current	32000.00
2	savings	241000.00