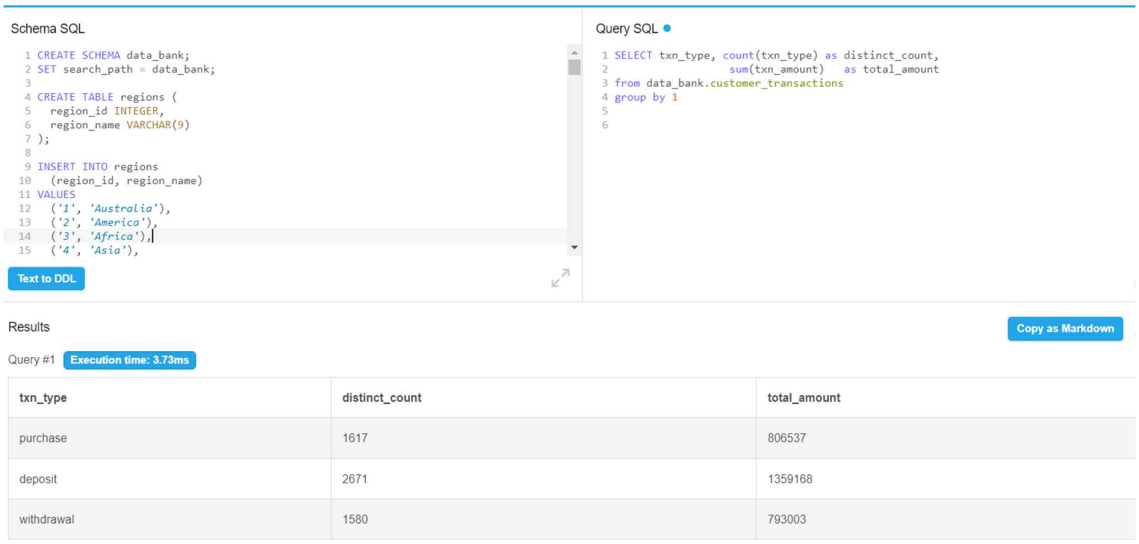
B. Customer Transactions

1. **What is the unique count and total amount for each transaction type?**

Query SQL

```sql
SELECT txn_type, count(txn_type) as distinct_count,
                 sum(txn_amount)   as total_amount
from data_bank.customer_transactions
group by 1
```



Schema SQL

```sql
CREATE SCHEMA data_bank;
SET search_path = data_bank;

CREATE TABLE regions (
  region_id INTEGER,
  region_name VARCHAR(9)
);

INSERT INTO regions
  (region_id, region_name)
VALUES
  ('1', 'Australia'),
  ('2', 'America'),
  ('3', 'Africa'),
  ('4', 'Asia'),
```

Text to DDL

Query SQL

```sql
SELECT txn_type, count(txn_type) as distinct_count,
                 sum(txn_amount)   as total_amount
from data_bank.customer_transactions
group by 1
```

Results

Copy as Markdown

Query #1 Execution time: 3.73ms

txn_type	distinct_count	total_amount
purchase	1617	806537
deposit	2671	1359168
withdrawal	1580	793003

- **txn_type**: Groups the data by each transaction type (e.g., deposit, withdrawal, purchase).
- **COUNT(DISTINCT txn_type)**: Counts the number of unique transactions for each type.
- **SUM(txn_amount):** Calculates the total transaction amount for each transaction type.
- **GROUP BY txn_type**: Ensures the results are grouped by the different transaction types.

- There were more deposits (2671) followed by purchases(1617)and then withdrawals (1580).

2.What is the average total historical deposit counts and amounts for all customers?

Query SQL

```sql
SELECT
    AVG(deposit_count) AS avg_deposit_count,
    AVG(deposit_amount) AS avg_deposit_amount
FROM (
    SELECT
        customer_id,
        COUNT(*) AS deposit_count,
        SUM(txn_amount) AS deposit_amount
    FROM data_bank.customer_transactions
    WHERE txn_type = 'deposit'
    GROUP BY customer_id
) AS customer_deposits;
```

Schema SQL

```sql
CREATE SCHEMA data_bank;
SET search_path = data_bank;

CREATE TABLE regions (
  region_id INTEGER,
  region_name VARCHAR(9)
);

INSERT INTO regions
  (region_id, region_name)
VALUES
  ('1', 'Australia'),
  ('2', 'America'),
  ('3', 'Africa'),
  ('4', 'Asia'),
```

Text to DDL

Query SQL

```sql
SELECT
    AVG(deposit_count) AS avg_deposit_count,
    AVG(deposit_amount) AS avg_deposit_amount
FROM (
    SELECT
        customer_id,
        COUNT(*) AS deposit_count,
        SUM(txn_amount) AS deposit_amount
    FROM data_bank.customer_transactions
    WHERE txn_type = 'deposit'
    GROUP BY customer_id
) AS customer_deposits;
```

Results

Query #1 Execution time: 3.51ms

avg_deposit_count	avg_deposit_amount
5.3420000000000000	2718.3360000000000000

Subquery (customer_deposits):

- The subquery filters for only deposit transactions (WHERE txn_type = 'deposit').
- It groups by customer_id to calculate:
    - COUNT(*) AS deposit_count: The total number of deposits per customer.

- SUM(txn_amount) AS deposit_amount: The total deposit amount per customer.

Main query:

- AVG(deposit_count): Calculates the average deposit count across all customers.
- AVG(deposit_amount): Calculates the average total deposit amount across all customers.

The average deposit count for a customer is 5 and the average deposit amount for a customer is 2,718.

3.For each month - how many Data Bank customers make more than 1 deposit and either 1 purchase or 1 withdrawal in a single month?

Query SQL

```
1
2   with  unique_count as (
3   SELECT
4         customer_id,
5         extract(month from txn_date) AS txn_month,
6         SUM(CASE WHEN txn_type = 'deposit' THEN 1 ELSE 0 END) AS deposit_count,
7         SUM(CASE WHEN txn_type = 'purchase' THEN 1 ELSE 0 END) AS purchase_count,
8         SUM(CASE WHEN txn_type = 'withdrawal' THEN 1 ELSE 0 END) AS withdrawal_count
9     FROM data_bank.customer_transactions
10    GROUP BY customer_id, txn_month)
11
12
13 select  txn_month, count(customer_id)
14 from unique_count
15 where deposit_count>1 and purchase_count >=1 or withdrawal_count >= 1
16 group by txn_month
17
```

Schema SQL

```
CREATE SCHEMA data_bank;
SET search_path = data_bank;

CREATE TABLE regions (
  region_id INTEGER,
  region_name VARCHAR(9)
);

INSERT INTO regions
  (region_id, region_name)
VALUES
  ('1', 'Australia'),
  ('2', 'America'),
  ('3', 'Africa'),
  ('4', 'Asia'),
```

Text to DDL

Query SQL

```
 with  unique_count as (
 SELECT
       customer_id,
       extract(month from txn_date) AS txn_month,
       SUM(CASE WHEN txn_type = 'deposit' THEN 1 ELSE 0 END) AS deposit_count,
       SUM(CASE WHEN txn_type = 'purchase' THEN 1 ELSE 0 END) AS purchase_count,
       SUM(CASE WHEN txn_type = 'withdrawal' THEN 1 ELSE 0 END) AS
withdrawal_count
   FROM data_bank.customer_transactions
   GROUP BY customer_id, txn_month)


select  txn_month, count(customer_id)
from unique_count
```

Results

Copy as Markdown

Query #1 Execution time: 13.9ms

txn_month	count
3	354
4	171
1	246
2	337

March had the highest number of customers (354) who had made more than 1 deposit and either 1 withdrawal or 1 deposit while April had the least number of such customers (171).

4.What is the closing balance for each customer at the end of the month?

Query SQL

```
WITH monthly_transactions AS (
    SELECT
        customer_id,
        DATE_TRUNC('month', txn_date) AS txn_month,
  SUM(CASE WHEN txn_type = 'deposit' THEN txn_amount
           WHEN txn_type IN ('withdrawal', 'purchase') THEN -txn_amount
           ELSE 0 END) AS monthly_balance_change
    FROM data_bank.customer_transactions
    GROUP BY customer_id, txn_month
)
SELECT
    customer_id,
    txn_month,
    sum(monthly_balance_change)over(partition by customer_id order by txn_month)
FROM monthly_transactions
```

Schema SQL

```sql
CREATE SCHEMA data_bank;
SET search_path = data_bank;

CREATE TABLE regions (
  region_id INTEGER,
  region_name VARCHAR(9)
);

INSERT INTO regions
  (region_id, region_name)
VALUES
  ('1', 'Australia'),
  ('2', 'America'),
  ('3', 'Africa'),
  ('4', 'Asia'),
```

Text to DDL

Query SQL

```sql
WITH monthly_transactions AS (
    SELECT
        customer_id,
        DATE_TRUNC('month', txn_date) AS txn_month,
  SUM(CASE WHEN txn_type = 'deposit' THEN txn_amount
           WHEN txn_type IN ('withdrawal', 'purchase') THEN -txn_amount
           ELSE 0 END) AS monthly_balance_change
    FROM data_bank.customer_transactions
    GROUP BY customer_id, txn_month
)
SELECT
    customer_id,
    txn_month,
    sum(monthly_balance_change)over(partition by customer_id order by txn_month)
FROM monthly_transactions
```

Results

Copy as Markdown

Query #1 Execution time: 18.71ms

customer_id	txn_month	sum
1	2020-01-01T00:00:00.000Z	312
1	2020-03-01T00:00:00.000Z	-640
2	2020-01-01T00:00:00.000Z	549
2	2020-03-01T00:00:00.000Z	610

Monthly_Transactions CTE:

- **DATE_TRUNC('month', txn_date)**: Extracts the month from the transaction date to group transactions by month.
- **SUM(CASE...END)**: Calculates the net balance change for each customer in each month. Deposits increase the balance (txn_amount is added), while withdrawals and purchases decrease it (txn_amount is subtracted).

Main query:

- **SUM(...) OVER (PARTITION BY customer_id ORDER BY txn_month)**: This uses a window function to calculate the cumulative sum of the monthly balance changes for each customer. It tracks the running total (closing balance) at the end of each month.

Retrieves the **closing balance** for each customer at the end of each month, ordered by customer and month.