

Data bank

REVOLUTIONIZING BANKING WITH SECURE DATA STORAGE INTEGRATION

Prepared by TAHIRA MUQADAS

introduction

- ▶ Welcome to data bank, the forefront of innovation at the intersection of neo-banking, cryptocurrency, and secure data storage. Founded by danny with a vision to redefine the banking landscape, data bank introduces a revolutionary concept where digital banking seamlessly integrates with cutting-edge data management solutions. At data bank, we transcend the limitations of traditional banking by offering not only financial services but also the world's most secure distributed data storage platform. Our unique approach links customers' cloud storage limits directly to their account balances, providing a dynamic and personalized storage solution. Driven by the mission to empower our customers, data bank faces the challenge of expanding our customer base while accurately forecasting their data storage needs. This case study delves into the metrics, growth strategies, and data analysis techniques essential for guiding our future developments and ensuring continued success.

Table: region: **region_id**: unique identifier for the region, **region name**: name of the region.

Table: customer nodes: **node_id**: unique identifier for the node, **start_date**: start date of node allocation, **end_date**: end date of node allocation, **customer_id**: unique identifier for the customer, **region_id**: identifier for the region associated with the customer. Table:

Transaction: **customer_id**: unique identifier for the customer, **transaction_date**: date of the transaction, **transaction_amount**: amount of the transaction.

```
select *from customer;
```

```
select *from region;
```

```
select *from transaction;
```

1. How many unique nodes are there on the Data Bank system? ;

```
select count(distinct node_id) as unique_idfrom customer;
```

Unique _ id = 5

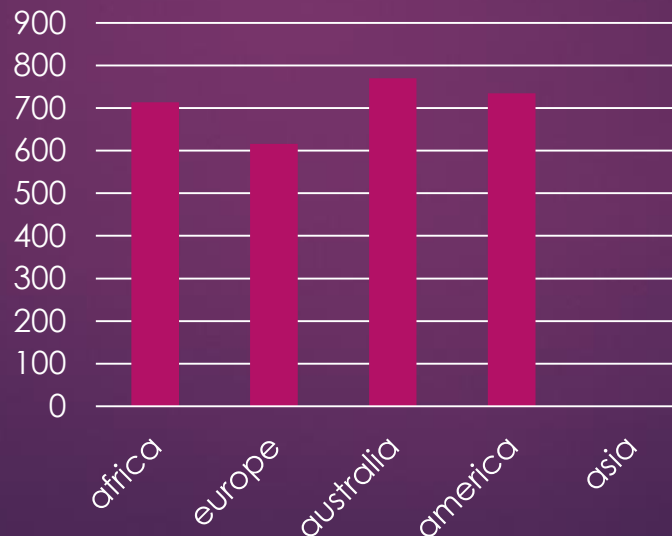
In Australia number of nodes are more then other countries.This can lead to a self reinforcing cycle of growth and engagement.

2. What is the number of nodes per region? ;

```
select region.region_name, count(customer.node_id) as number_of_nodefrom region join customer on region.region_id = customer.region_idgroup by
```

```
region.region_name;
```

region_name	Number of nodes
Africa	714
Europe	616
Australia	770
America	735
Asia	665



- ▶ 3. How many customers are allocated to each region? ;

- ▶

```
SELECT  r.region_id, r.region_name, COUNT(DISTINCT c.customer_id) AS num_customers
FROM customer c JOIN region r ON c.region_id = r.region_id
GROUP BY r.region_id, r.region_name;
```

region_id	region_name	num_customers
1	Australia	110
2	America	105
3	Africa	102
4	Asia	95
5	Europe	88

Number of customers per region



4. How many days on average are customers reallocated to a different node? ;

```
select round(avg(datediff(end_date, start_id))) average_days_allocation
from custome
rwhere end_date is not null and year(end_date) <> 9999;
```

Aveage_days_allocation = 15

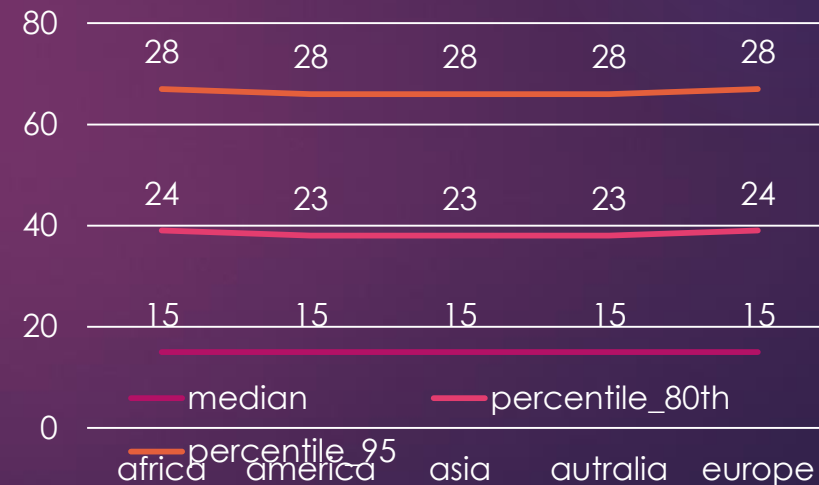
A higher number of customers in Australia signifies a lucrative market opportunity for the data bank. By capitalizing on this larger customer base, the data bank can enhance revenue streams, deepen data insights, and strengthen its market position, driving sustained growth and success.

5. What is the median, 80th and 95th percentile for this same reallocation days metric for each region?

- ▶ rows_ as (select c.customer_id,r.region_name, DATEDIFF(c.end_date, c.start_id) AS days_difference,row_number() over (partition by r.region_name order by DATEDIFF(c.end_date, c.start_id)) AS rows_number,COUNT(*) over (partition by r.region_name) as total_rows
from customer c JOIN region r ON c.region_id = r.region_id where c.end_date not like '%9999%')SELECT region_name,
- ▶ ROUND(AVG(CASE WHEN rows_number between (total_rows/2) and ((total_rows/2)+1) THEN days_difference END), 0) AS Median,
- ▶ MAX(CASE WHEN rows_number = round((0.80 * total_rows),0) THEN days_difference END) AS Percentile_80th,
- ▶ MAX(CASE WHEN rows_number = round((0.95 * total_rows),0) THEN days_difference END) AS Percentile_95th
- ▶ from rows_group by region_name;

By identifying the discrepancy in the 80th percentile across countries despite consistent median days, the data bank can refine its targeting and engagement strategies to address regional variations, optimizing user experience and goal attainment.

region_name	median	Percentile_80th	Percentile_95th
Africa	15	24	28
America	15	23	28
Asia	15	23	28
Australia	15	23	28
Europe	15	24	28

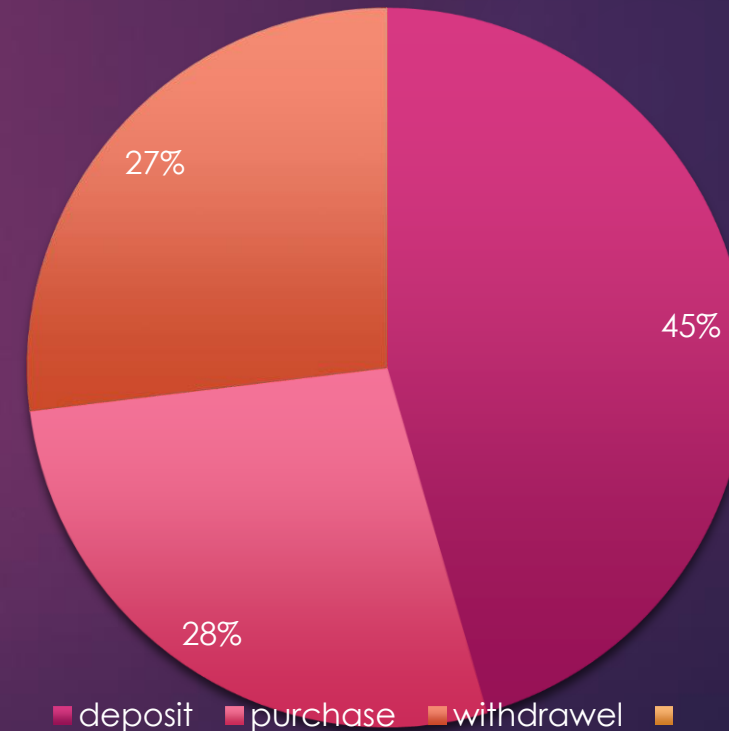


- ▶ B. Customer Transactions1.
- ▶ 1.What is the unique count and total amount for each transaction type? ;
- ▶ Select txn_type,Count(txn_type), Sum(txn_amount)
- ▶ From transaction
- ▶ group by txn_type;

By prioritizing deposit-focused products and services, the data bank can capitalize on higher transaction volumes, boosting revenue and customer retention while aligning with strategic growth objectives.

txn_type	Count(txn_type)	Sum(txn_amount)
deposit	2671	1359168
purchase	1617	806537
withdrawal	1580	793003

Transaction type per sum of transaction amount



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

```
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 transaction
WHERE txn_type = 'deposit' GROUP BY customer_id) AS Customer_Deposits;
```

Avg_Deposit_Count	Avg_Deposit_Amount
5.342	2718.336

The data bank can leverage insights from average deposit amounts and account numbers to tailor product offerings, optimize marketing strategies, and drive revenue growth.

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?;

```
SELECT YEAR(txn_date) AS year, MONTH(txn_date) AS month, customer_id, COUNT(CASE WHEN txn_type = 'deposit' THEN 1
END) AS deposit_count, COUNT(CASE WHEN txn_type = 'purchase' THEN 1 END) AS purchase_count, COUNT(CASE WHEN txn_type
= 'withdrawal' THEN 1 END) AS withdrawal_countFROM transactionGROUP BY YEAR(txn_date), MONTH(txn_date),
customer_idHAVING deposit_count > 1 AND (purchase_count = 1 OR withdrawal_count = 1);Select * from transaction;
```

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

```
SELECT customer_id, EXTRACT(YEAR FROM txn_date) AS year, EXTRACT(MONTH FROM txn_date) AS month, SUM(CASE WHEN
txn_type = 'deposit' THEN txn_amount ELSE 0 END) - SUM(CASE WHEN txn_type IN ('purchase', 'withdrawal') THEN txn_amount ELSE 0
END) AS closing_balanceFROM transactionGROUP BY customer_id, EXTRACT(YEAR FROM txn_date), EXTRACT(MONTH FROM
txn_date)ORDER BY customer_id, year, month;
```

5. What is the percentage of customers who increase their closing balance by more than 5%? ;

```
WITH CustomerBalances AS ( SELECT      customer_id,      SUM(CASE WHEN txn_type = 'deposit' THEN txn_amount ELSE 0 END) AS TotalDeposits,      SUM(CASE WHEN txn_type = 'purchase' THEN txn_amount ELSE 0 END) AS TotalPurchases FROM      transaction GROUP BY      customer_id),BalanceChanges AS ( SELECT      customer_id,      TotalDeposits,      TotalPurchases,      (TotalDeposits - TotalPurchases) AS NetBalanceChange,      ((TotalDeposits - TotalPurchases) / NULLIF(TotalDeposits, 0)) * 100 AS PercentageIncrease FROM      CustomerBalances)SELECT  COUNT(*) AS TotalCustomers,  SUM(CASE WHEN PercentageIncrease > 5 THEN 1 ELSE 0 END) AS CustomersIncreasedMoreThan5Percent,  (SUM(CASE WHEN PercentageIncrease > 5 THEN 1 ELSE 0 END) * 1.0 / COUNT(*)) * 100 AS PercentageOfCustomersIncreasedMoreThan5PercentFROM  BalanceChanges;
```

TotalCustomers	CustomersIncreasedMoreThan5Percent	PercentageOfCustomersIncreasedMoreThan5Percent
500	376	75.2

C. .Data Allocation Challenge;

1.Customer Balance at the End of Each Month:

```
select *from transaction;customer_running_balance;select customer_id, txn_amount, sum(txn_amount) over (partition by customer_id order by txn_date)as customer_balancefrom transaction;
```

2● customer balance at the end of each month;

```
select customer_id, extract(month from txn_date) as month,  sum(txn_amount) over (partition by customer_id order by extract(month from txn_date)) as customer_balance  from transaction  order by customer_id, month;
```


3● minimum, average and maximum values of the running balance for each customer ;

```
with customer_balance as ( select customer_id, txn_date, sum(txn_amount) over ( partition by customer_id order by
txn_date) as running_balance from transaction ) select customer_id, min(running_balance) as minimum_balance,
avg(running_balance) as averag_balance, max(running_balance) as maximun_balance from customer_balance
group by customer_id;
```

Implement Advanced Analytics and AI:

- Leverage artificial intelligence and machine learning to enhance fraud detection and customer service, enabling real-time insights and more personalized banking experiences
- Foster Innovation through Partnerships: Collaborate with fintech startups and other technology companies to integrate innovative services and features that can enhance user experience and security.
- Strengthen Regulatory Compliance: Ensure adherence to global and regional data protection regulations by investing in compliance programs and technologies, thereby building trust and safeguarding against legal and financial risks.