

Challenge 4 - Finance Analysis



Intro

You are a Finance Analyst working for 'The Big Bank'

You have been tasked with finding out about your customers and their banking behaviour. Examine the accounts they hold and the type of transactions they make to develop greater insight into your customers.

Tables

Here are the tables you will be using

Customers

CustomerID	FirstName	LastName	City	State
1	John	Doe	New York	NY
2	Jane	Doe	New York	NY
3	Bob	Smith	San Francisco	CA
4	Alice	Johnson	San Francisco	CA
5	Michael	Lee	Los Angeles	CA
6	Jennifer	Wang	Los Angeles	CA

Accounts

AccountID	CustomerID	BranchID	AccountType	Balance
1	1	5	Checking	1000
2	1	5	Savings	5000
3	2	1	Checking	2500
4	2	1	Savings	####
5	3	2	Checking	7500
6	3	2	Savings	####
7	4	8	Checking	5000
8	4	8	Savings	####
9	5	14	Checking	####
10	5	14	Savings	####
11	6	2	Checking	5000
12	6	2	Savings	####
13	1	5	Credit Card	-500
14	2	1	Credit Card	-1000
15	3	2	Credit Card	-2000

Transactions

TransactionID	AccountID	TransactionDate	Amount
1	1	2022-01-01	-500
2	1	2022-01-02	-250
3	2	2022-01-03	1000
4	3	2022-01-04	-1000
5	3	2022-01-05	500
6	4	2022-01-06	1000
7	4	2022-01-07	-500
8	5	2022-01-08	-2500
9	6	2022-01-09	500
10	6	2022-01-10	-1000
11	7	2022-01-11	-500
12	7	2022-01-12	-250
13	8	2022-01-13	1000
14	8	2022-01-14	-1000
15	9	2022-01-15	500

Branches

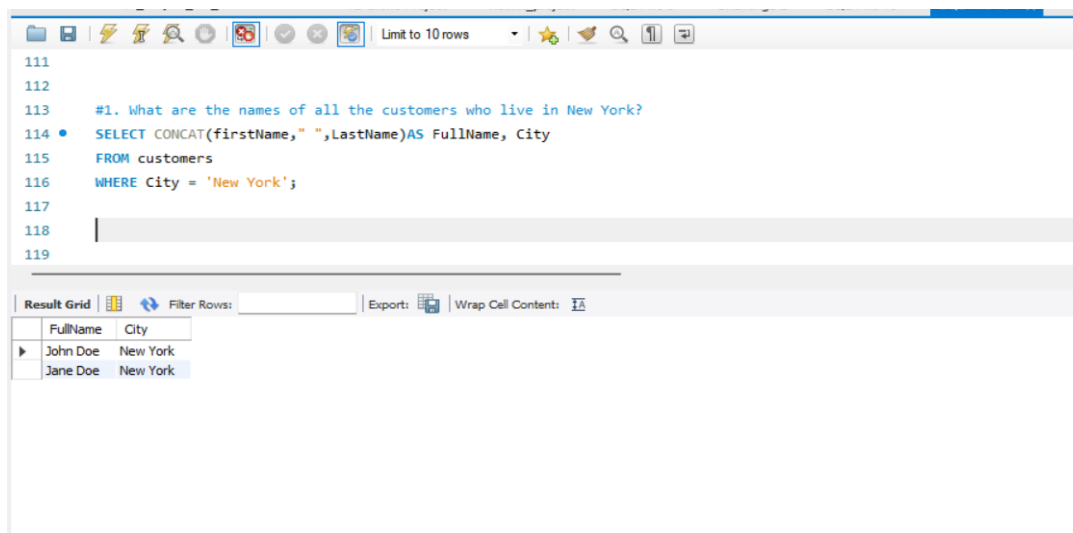
BranchID	BranchName	City	State
1	Main	New York	NY
2	Downtown	San Francisco	CA
3	West LA	Los Angeles	CA
4	East LA	Los Angeles	CA
5	Uptown	New York	NY
6	Financial District	San Francisco	CA
7	Midtown	New York	NY
8	South Bay	San Francisco	CA
9	Downtown	Los Angeles	CA
10	Chinatown	New York	NY
11	Marina	San Francisco	CA
12	Beverly Hills	Los Angeles	CA
13	Brooklyn	New York	NY
14	North Beach	San Francisco	CA
15	Pasadena	Los Angeles	CA

Questions

Answer the following questions

Then write a LinkedIn post saying what you have learnt or enjoyed

1. What are the names of all the customers who live in New York?



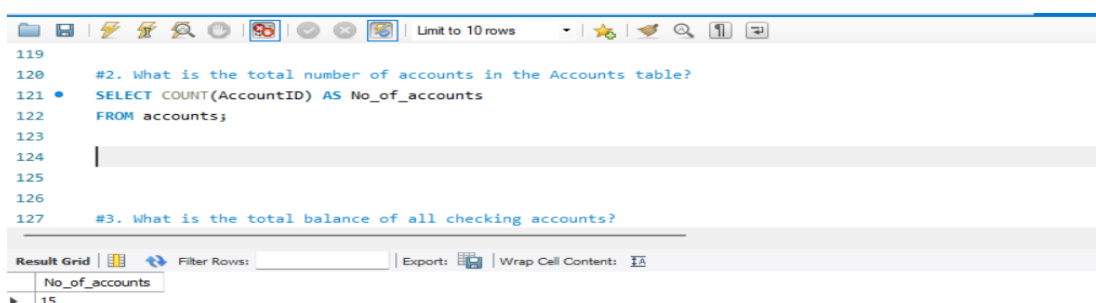
The screenshot shows a SQL query editor with a toolbar at the top. The query is as follows:

```
111
112
113 #1. What are the names of all the customers who live in New York?
114 • SELECT CONCAT(firstName, " ", LastName) AS FullName, City
115 FROM customers
116 WHERE City = 'New York';
117
118
119
```

Below the query, the 'Result Grid' is displayed with the following data:

FullName	City
John Doe	New York
Jane Doe	New York

2. What is the total number of accounts in the Accounts table?



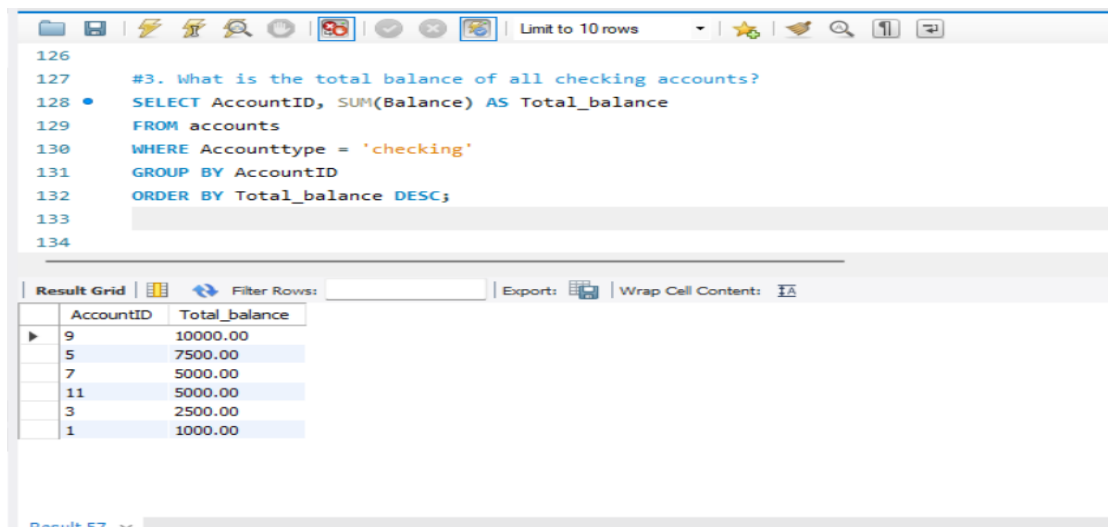
The screenshot shows a SQL query editor with a toolbar at the top. The query is as follows:

```
119
120 #2. What is the total number of accounts in the Accounts table?
121 • SELECT COUNT(AccountID) AS No_of_accounts
122 FROM accounts;
123
124
125
126
127 #3. What is the total balance of all checking accounts?
```

Below the query, the 'Result Grid' is displayed with the following data:

No_of_accounts
15

3. What is the total balance of all checking accounts?



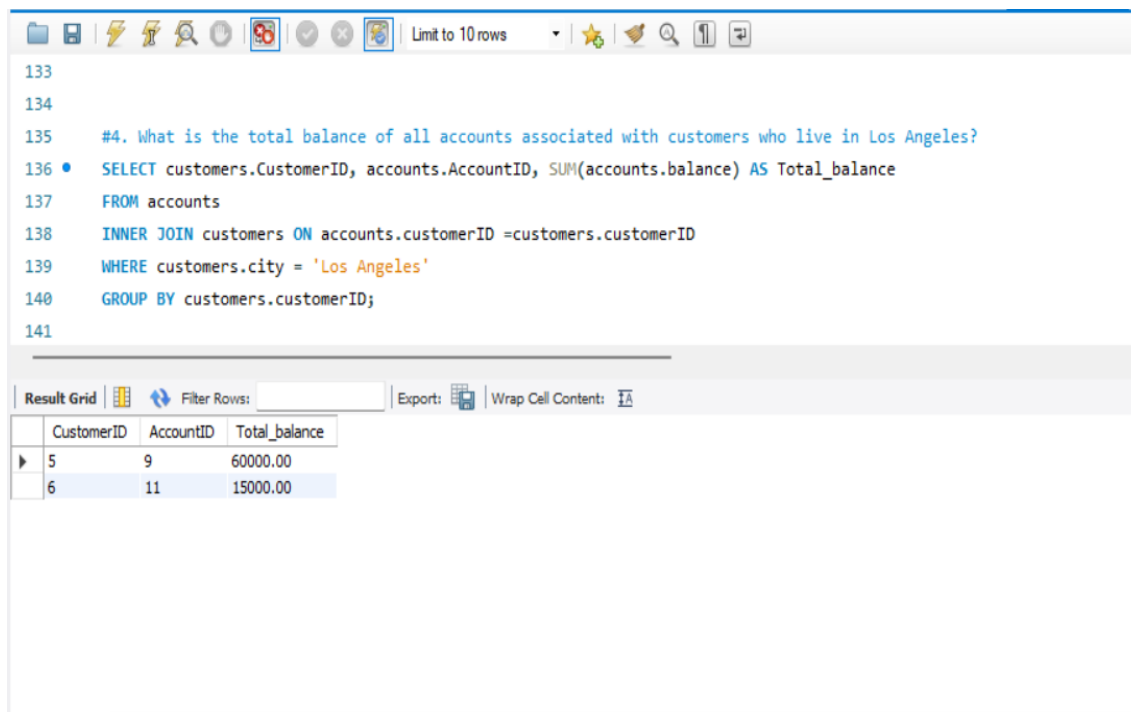
The screenshot shows a SQL IDE interface. The query editor contains the following SQL code:

```
126
127 #3. What is the total balance of all checking accounts?
128 • SELECT AccountID, SUM(Balance) AS Total_balance
129 FROM accounts
130 WHERE Accounttype = 'checking'
131 GROUP BY AccountID
132 ORDER BY Total_balance DESC;
133
134
```

Below the query editor is a 'Result Grid' tab. It shows a table with two columns: 'AccountID' and 'Total_balance'. The data is sorted in descending order of total balance.

AccountID	Total_balance
9	10000.00
5	7500.00
7	5000.00
11	5000.00
3	2500.00
1	1000.00

4. What is the total balance of all accounts associated with customers who live in Los Angeles?



The screenshot shows a SQL IDE interface. The query editor contains the following SQL code:

```
133
134
135 #4. What is the total balance of all accounts associated with customers who live in Los Angeles?
136 • SELECT customers.CustomerID, accounts.AccountID, SUM(accounts.balance) AS Total_balance
137 FROM accounts
138 INNER JOIN customers ON accounts.customerID =customers.customerID
139 WHERE customers.city = 'Los Angeles'
140 GROUP BY customers.customerID;
141
```

Below the query editor is a 'Result Grid' tab. It shows a table with three columns: 'CustomerID', 'AccountID', and 'Total_balance'. The data is sorted in descending order of total balance.

CustomerID	AccountID	Total_balance
5	9	60000.00
6	11	15000.00

5. Which branch has the highest average account balance?

The screenshot shows a SQL IDE with a query editor and a result grid. The query is as follows:

```
143  
144 #5. Which branch has the highest average account balance?  
145 • SELECT branches.branchname, ROUND(AVG(accounts.balance),0) AS Avg_balance  
146 FROM branches  
147 INNER JOIN accounts ON branches.branchID = accounts.branchID  
148 GROUP BY branches.branchName  
149 ORDER BY Avg_balance DESC;  
150  
151
```

The result grid displays the following data:

branchname	Avg_balance
North Beach	30000
South Bay	12500
Downtown	7100
Main	3833
Uptown	1833

6. Which customer has the highest current balance in their accounts?

The screenshot shows a SQL IDE with a query editor and a result grid. The query is as follows:

```
151  
152 #6. Which customer has the highest current balance in their accounts?  
153 • SELECT customers.CustomerID, CONCAT(customers.FirstName, " ", customers.LastName) AS FullName,  
154 accounts.AccountID, Max(accounts.balance) AS Highest_Balance  
155 FROM customers INNER JOIN accounts ON customers.CustomerID = accounts.customerID;  
156  
157  
158  
159
```

The result grid displays the following data:

CustomerID	FullName	AccountID	Highest_Balance
1	John Doe	1	50000.00

7. Which customer has made the most transactions in the Transactions table?

The screenshot shows a SQL IDE with a query editor and a result grid. The query is as follows:

```
160 #7. Which customer has made the most transactions in the Transactions table?
161 • SELECT customers.customerID, accounts.AccountID, MAX(transactions.transactionID) AS No_of_transaction
162 FROM customers INNER JOIN accounts ON customers.customerID = accounts.customerID
163 INNER JOIN transactions ON accounts.AccountID = transactions.AccountID
164 GROUP BY customers.customerID
165 ORDER BY No_of_transaction DESC;
```

The result grid displays the following data:

	customerID	AccountID	No_of_transaction
▶	5	9	15
	4	7	14
	3	5	10
	2	3	7
	1	1	3

8. Which branch has the highest total balance across all of its accounts?

The screenshot shows a SQL IDE with a query editor and a result grid. The query is as follows:

```
169 #8. Which branch has the highest total balance across all of its accounts?
170 • SELECT branches.BranchName, SUM(accounts.balance) AS Total_balance
171 FROM branches INNER JOIN accounts ON branches.BranchID = accounts.BranchID
172 GROUP BY branches.BranchName
173 ORDER BY Total_balance DESC
174 LIMIT 1;
```

The result grid displays the following data:

	BranchName	Total_balance
▶	North Beach	60000.00

9. Which customer has the highest total balance across all of their accounts, including savings and checking accounts?

The screenshot shows a SQL IDE with a query editor and a result grid. The query is as follows:

```
176
177
178 #9. Which customer has the highest total balance across all of their accounts, including savings and checking accounts?
179 • SELECT customers.customerID, CONCAT(customers.FirstName, " ", customers.FirstName) AS FullName, SUM(accounts.balance) AS Total_balance
180 FROM customers INNER JOIN accounts ON customers.customerID = accounts.customerID
181 GROUP BY customers.customerID
182 ORDER BY Total_balance DESC
183 LIMIT 1;
184
```

The result grid shows the following data:

customerID	FullName	Total_balance
5	Michael Michael	60000.00

10. Which branch has the highest number of transactions in the Transactions table?

The screenshot shows a SQL IDE with a query editor and a result grid. The query is as follows:

```
186
187 #10. Which branch has the highest number of transactions in the Transactions table?
188 • SELECT branches.branchID, branches.branchName, SUM(transactions.transactionID) AS No_of_transactions
189 FROM branches INNER JOIN accounts ON branches.branchID = accounts.branchID
190 INNER JOIN transactions ON accounts.accountID = transactions.AccountID
191 GROUP BY branches.branchID
192 ORDER BY No_of_transactions DESC;
193
```

The result grid shows the following data:

branchID	branchName	No_of_transactions
8	South Bay	50
2	Downtown	27
1	Main	22
14	North Beach	15
5	Uptown	6