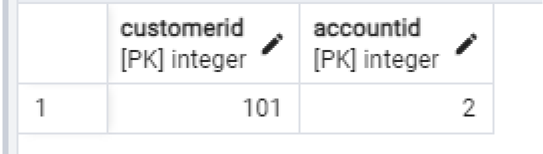
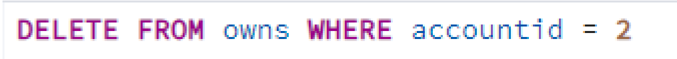
SP:

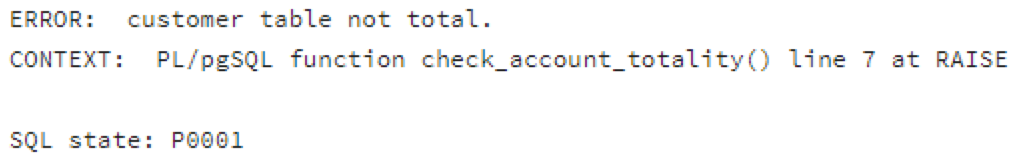
check_account_totality() as used in trigger enforce_account_totality



	customerid [PK] integer	accountid [PK] integer
1	101	2



```
DELETE FROM owns WHERE accountid = 2
```

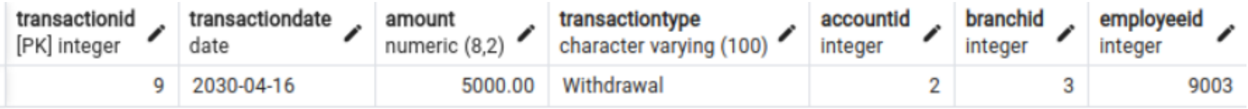


```
ERROR:  customer table not total.
CONTEXT:  PL/pgSQL function check_account_totality() line 7 at RAISE

SQL state: P0001
```

update_cards(accountid, cardnumber, verifycode):

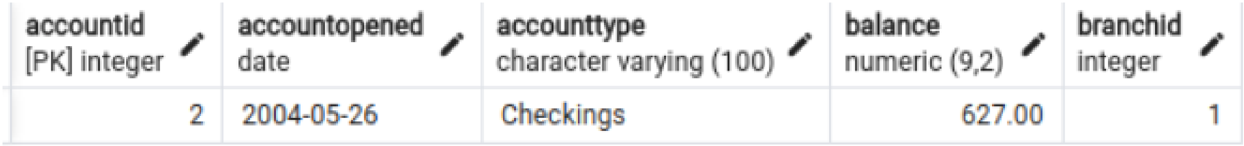
CALL ProcessTransaction(9, 0);

Transaction:



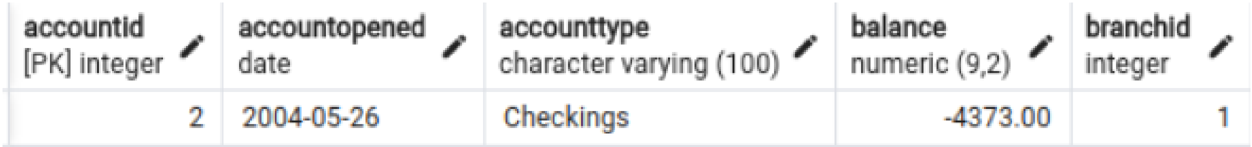
transactionid [PK] integer	transactiondate date	amount numeric (8,2)	transactiontype character varying (100)	accountid integer	branchid integer	employeeid integer
9	2030-04-16	5000.00	Withdrawal	2	3	9003

Account (before):



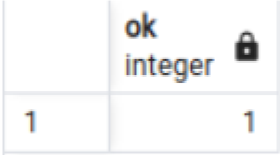
accountid [PK] integer	accountopened date	accounttype character varying (100)	balance numeric (9,2)	branchid integer
2	2004-05-26	Checkings	627.00	1

Account (after):



accountid [PK] integer	accountopened date	accounttype character varying (100)	balance numeric (9,2)	branchid integer
2	2004-05-26	Checkings	-4373.00	1

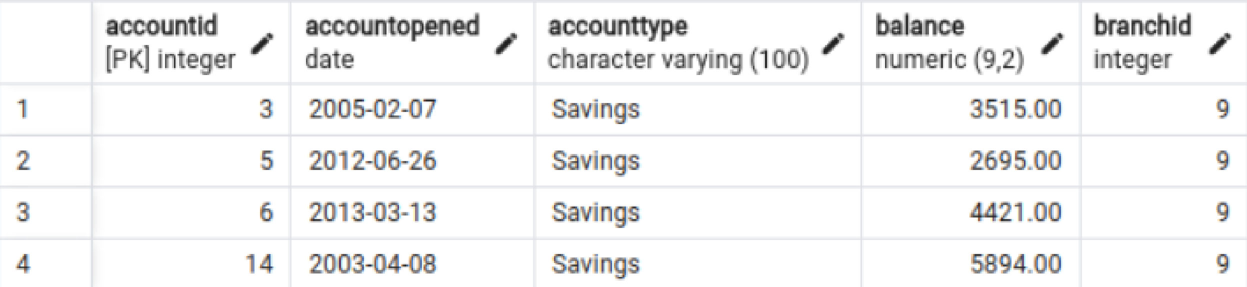
Output of CALL:



	ok integer
1	1

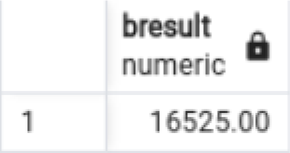
CALL TotalBranchBalance(9, 0);

Accounts with BranchID 9:



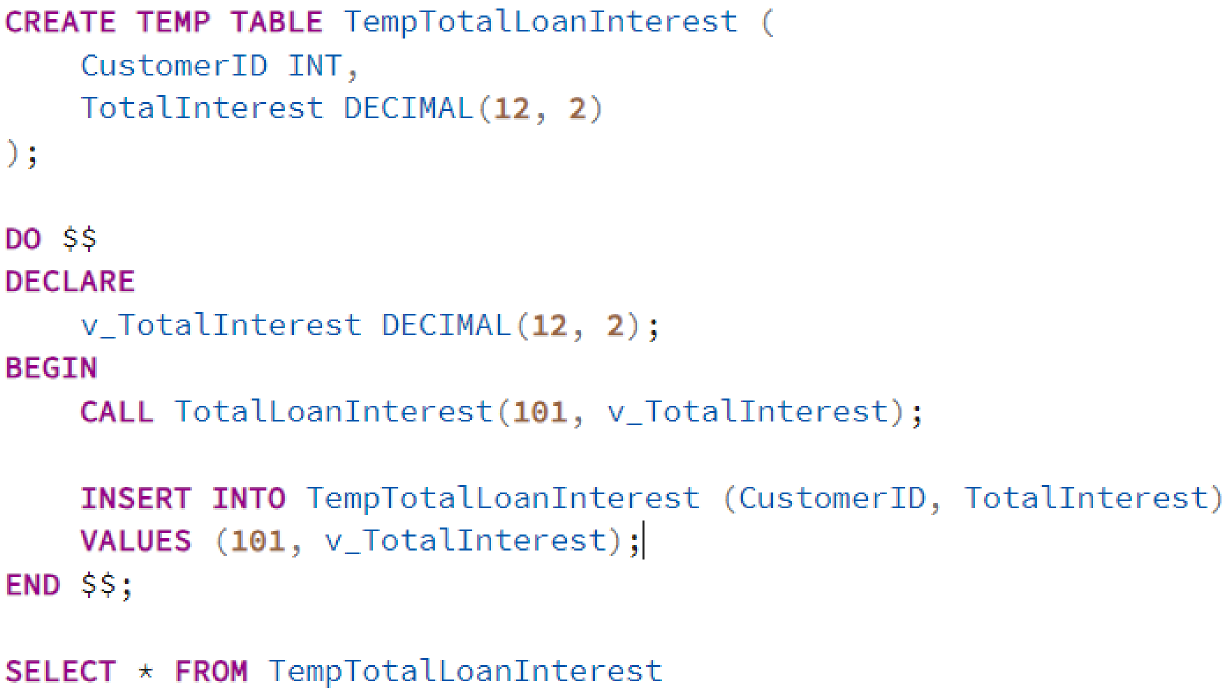
	accountid [PK] integer	accountopened date	accounttype character varying (100)	balance numeric (9,2)	branchid integer
1	3	2005-02-07	Savings	3515.00	9
2	5	2012-06-26	Savings	2695.00	9
3	6	2013-03-13	Savings	4421.00	9
4	14	2003-04-08	Savings	5894.00	9

Output:

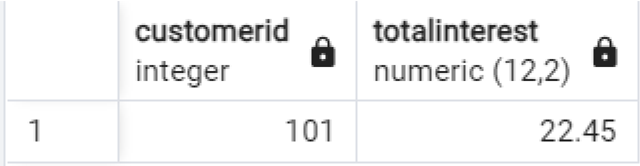


	bresult numeric
1	16525.00

TotalLoanInterest



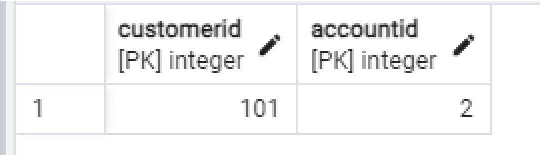
```
CREATE TEMP TABLE TempTotalLoanInterest (
    CustomerID INT,
    TotalInterest DECIMAL(12, 2)
);

DO $$
DECLARE
    v_TotalInterest DECIMAL(12, 2);
BEGIN
    CALL TotalLoanInterest(101, v_TotalInterest);

    INSERT INTO TempTotalLoanInterest (CustomerID, TotalInterest)
    VALUES (101, v_TotalInterest);
END $$;

SELECT * FROM TempTotalLoanInterest
```



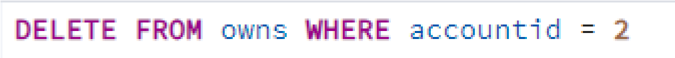
	customerid integer	totalinterest numeric (12,2)
1	101	22.45

TG:

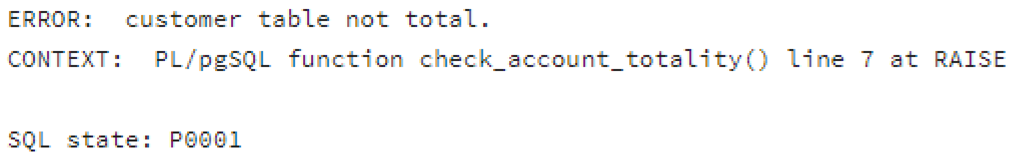
enforce_account_totality



	customerid [PK] integer	accountid [PK] integer
1	101	2

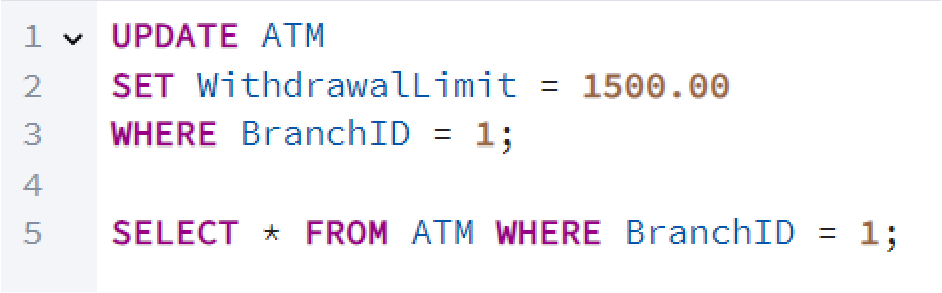


```sql
DELETE FROM owns WHERE accountid = 2
```

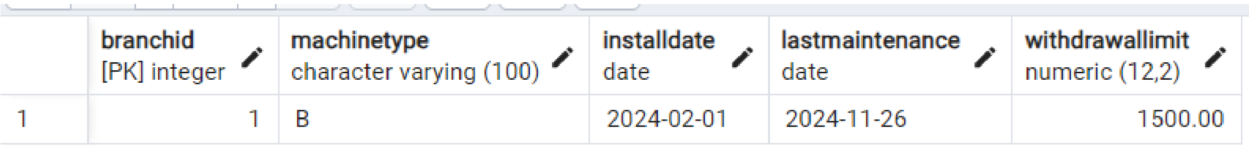


```
ERROR:  customer table not total.
CONTEXT:  PL/pgSQL function check_account_totality() line 7 at RAISE

SQL state: P0001
```

update_last_maintenance()



```sql
UPDATE ATM
SET WithdrawalLimit = 1500.00
WHERE BranchID = 1;

SELECT * FROM ATM WHERE BranchID = 1;
```



	branchid [PK] integer	machinetype character varying (100)	installdate date	lastmaintenance date	withdrawallimit numeric (12,2)
1	1	B	2024-02-01	2024-11-26	1500.00

Transaction_trigger:

Accounts (before trigger):

2	"2004-05-26"	"Checkings"	627.00	1
3	"2005-02-07"	"Savings"	3515.00	9
5	"2012-06-26"	"Savings"	2695.00	9
6	"2013-03-13"	"Savings"	4421.00	9

Insert a transaction doing deposit on account 2:

INSERT INTO Transactions (TransactionID, TransactionDate, Amount, TransactionType, AccountID, BranchID, EmployeeID)
VALUES
(23, '2030-04-22', 6000.00, 'Deposit', 2, 3, 9003);

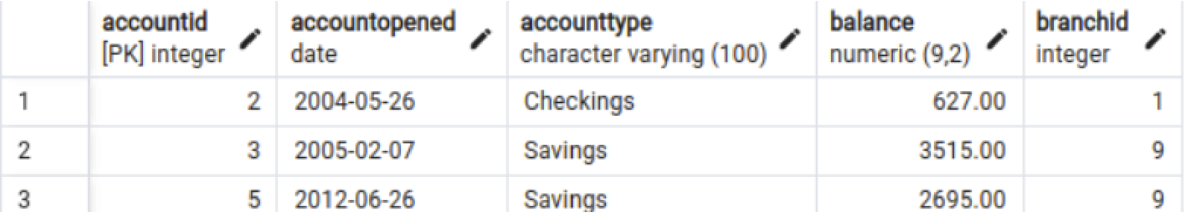
Accounts (after trigger):

17	"2017-12-19"	"Savings"	6529.00	6
18	"2019-06-15"	"Savings"	4866.00	6
19	"2021-08-01"	"Checkings"	2401.00	5
20	"2024-05-22"	"Checkings"	9210.00	2

1	"2003-03-08"	"Checkings"	2609.00	4
4	"2011-12-10"	"Checkings"	4109.00	4
2	"2004-05-26"	"Checkings"	6627.00	1

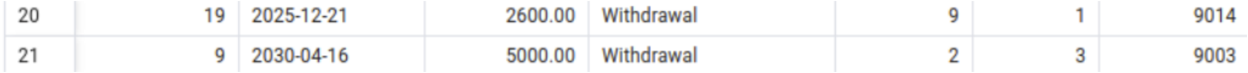
Account 2’s balance has correctly increased by 6000.

check_balance_trigger:
Account 2’s balance:



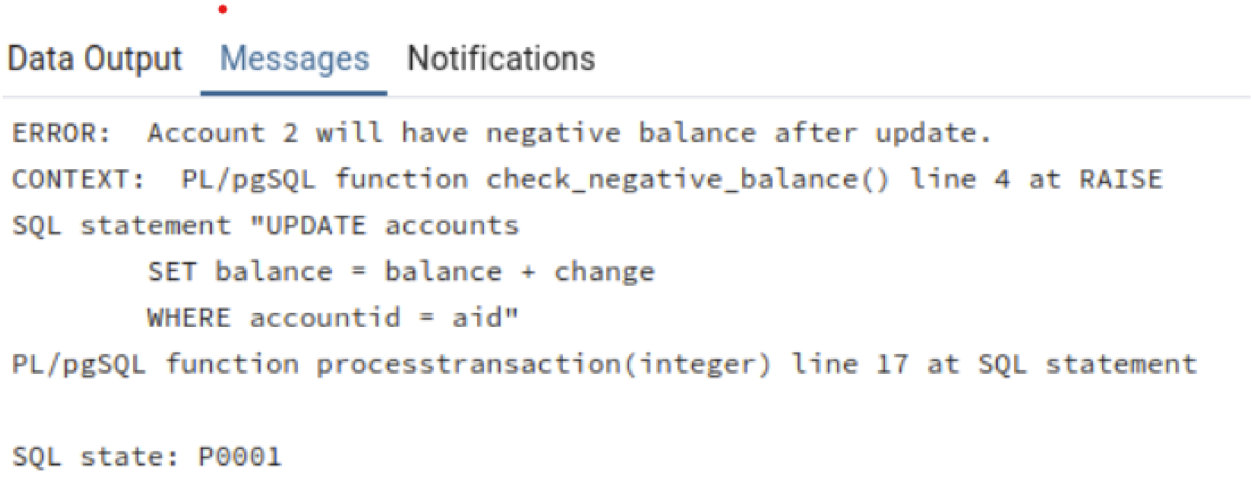
	accountid [PK] integer	accountopened date	accounttype character varying (100)	balance numeric (9,2)	branchid integer
1	2	2004-05-26	Checkings	627.00	1
2	3	2005-02-07	Savings	3515.00	9
3	5	2012-06-26	Savings	2695.00	9

Transaction to be processed (9):



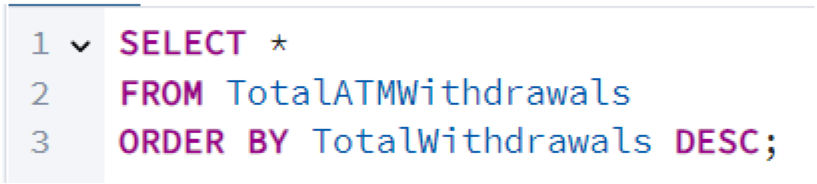
20	19	2025-12-21	2600.00	Withdrawal	9	1	9014
21	9	2030-04-16	5000.00	Withdrawal	2	3	9003

After CALL ProcessTransaction(9,0):

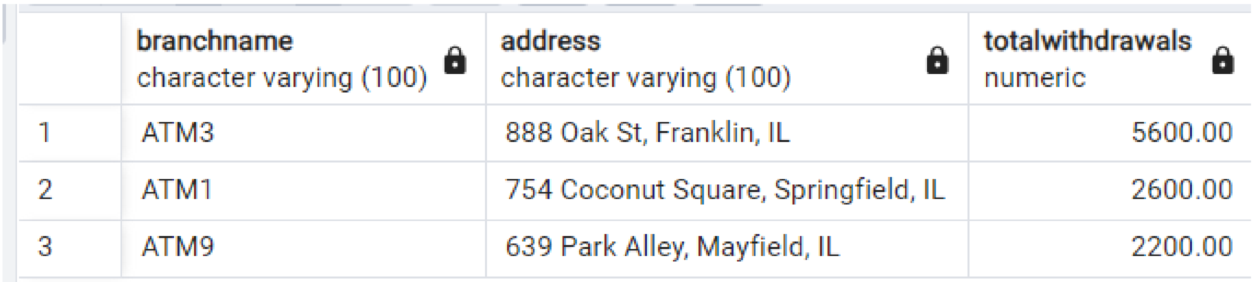


```
ERROR:  Account 2 will have negative balance after update.
CONTEXT:  PL/pgSQL function check_negative_balance() line 4 at RAISE
SQL statement "UPDATE accounts
        SET balance = balance + change
        WHERE accountid = aid"
PL/pgSQL function processtransaction(integer) line 17 at SQL statement

SQL state: P0001
```

Views:
TotalATMWithdrawals

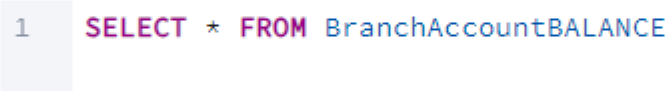


```sql
SELECT *
FROM TotalATMWithdrawals
ORDER BY TotalWithdrawals DESC;
```



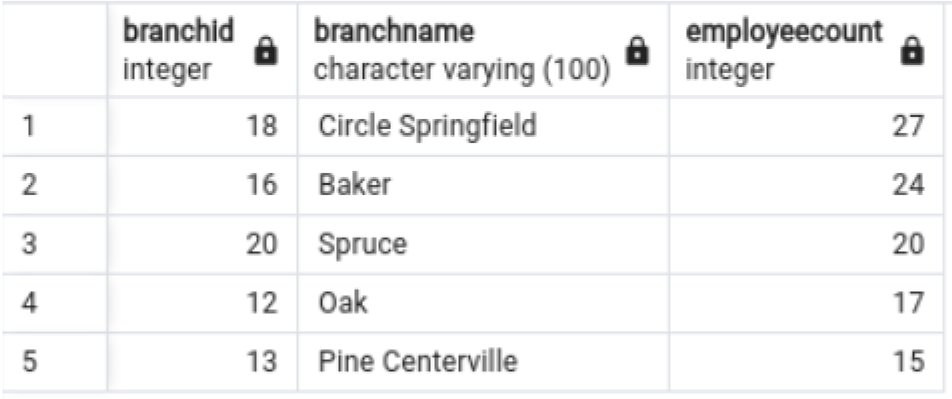
	branchname character varying (100)	address character varying (100)	totalwithdrawals numeric
1	ATM3	888 Oak St, Franklin, IL	5600.00
2	ATM1	754 Coconut Square, Springfield, IL	2600.00
3	ATM9	639 Park Alley, Mayfield, IL	2200.00

BranchAccountBalance



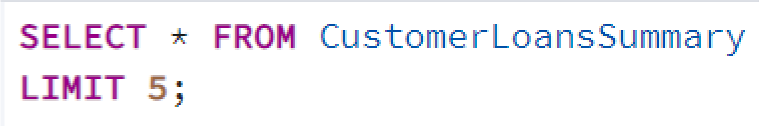
```
SELECT * FROM BranchAccountBALANCE
```

LimitedBranchView:
SELECT * FROM LimitedBranchView
ORDER BY employeecount DESC
LIMIT 5;

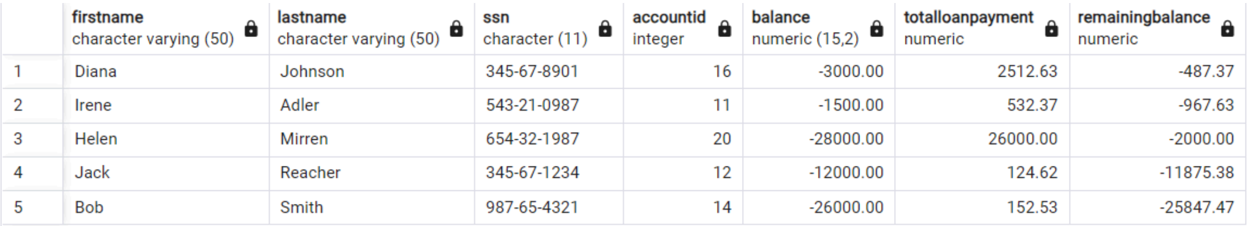


	branchid integer	branchname character varying (100)	employeecount integer
1	18	Circle Springfield	27
2	16	Baker	24
3	20	Spruce	20
4	12	Oak	17
5	13	Pine Centerville	15

CustomerLoansSummary



```
SELECT * FROM CustomerLoansSummary
LIMIT 5;
```



	firstname character varying (50)	lastname character varying (50)	ssn character (11)	accountid integer	balance numeric (15,2)	totalloanpayment numeric	remainingbalance numeric
1	Diana	Johnson	345-67-8901	16	-3000.00	2512.63	-487.37
2	Irene	Adler	543-21-0987	11	-1500.00	532.37	-967.63
3	Helen	Mirren	654-32-1987	20	-28000.00	26000.00	-2000.00
4	Jack	Reacher	345-67-1234	12	-12000.00	124.62	-11875.38
5	Bob	Smith	987-65-4321	14	-26000.00	152.53	-25847.47