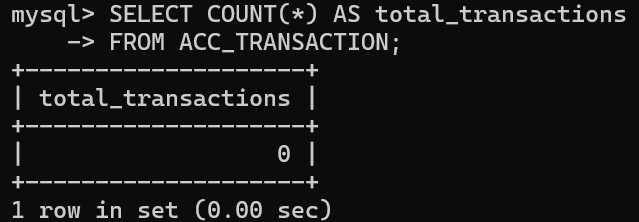
1. SELSCT COUNT(\*) AS total\_transactions

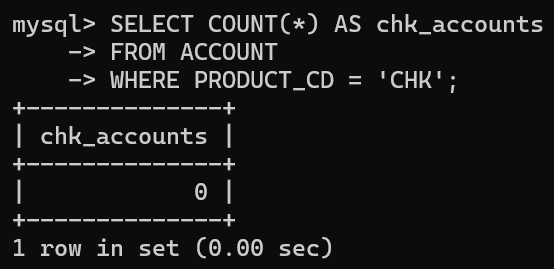
FROM ACC\_TRANSACTION;

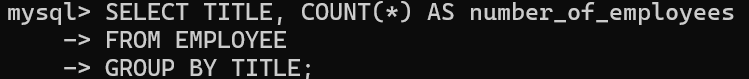
****

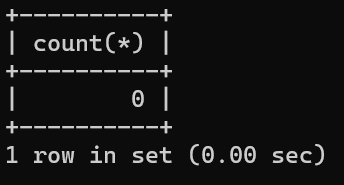
1. SELECT COUNT(\*) AS chk\_accounts

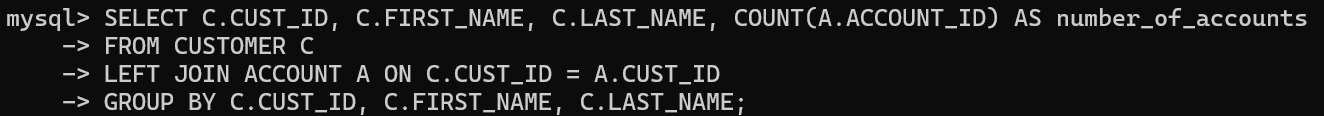
FROM ACCOUNT

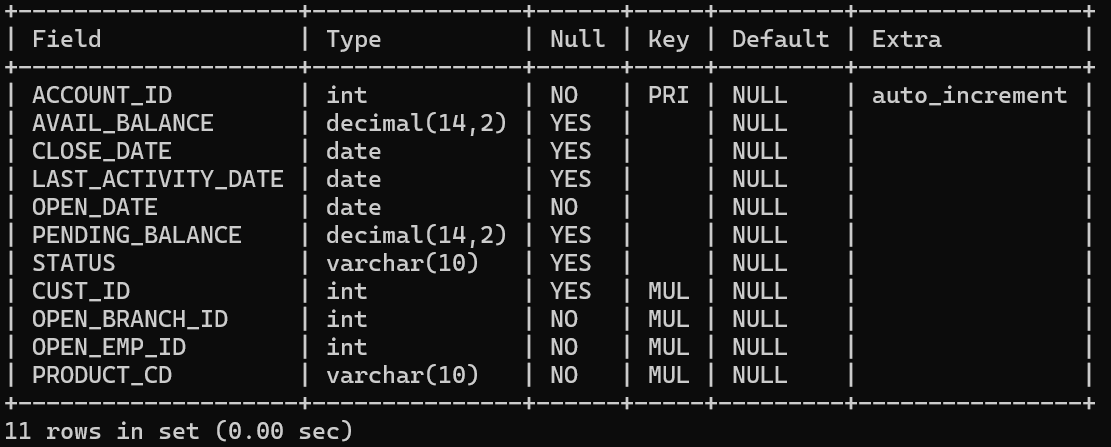
WHERE PRODUCT\_CD = 'CHK';



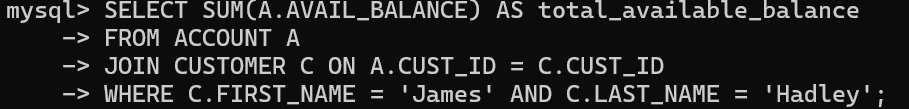
1. 

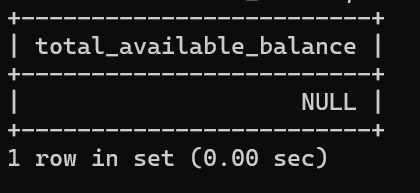


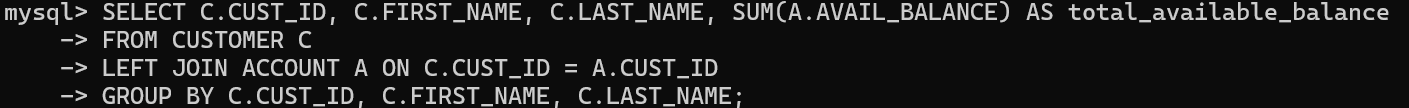
4.

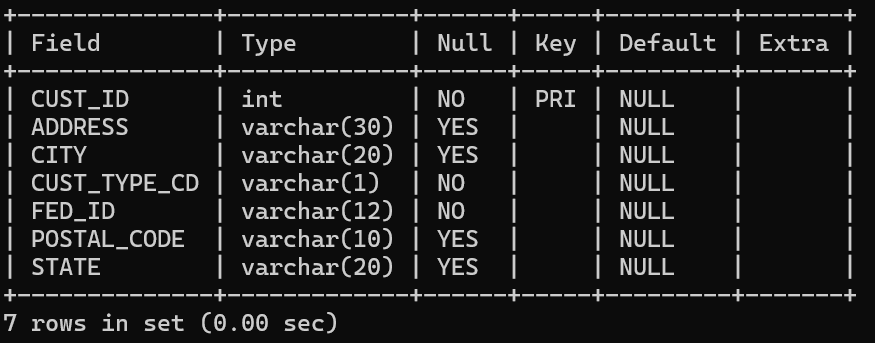


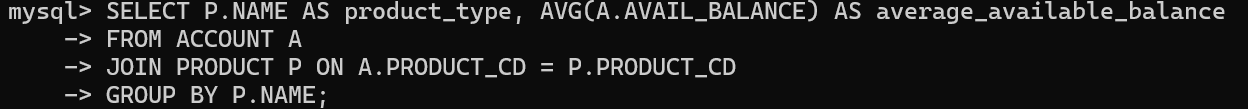
5.

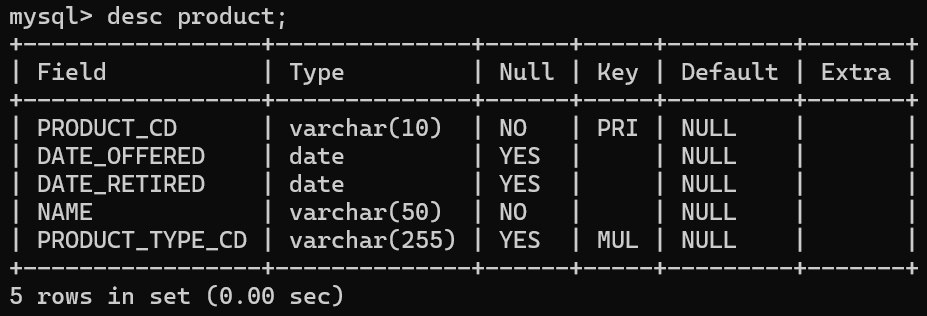




6.



7.



1. Find the total available balance in customers’ accounts where the opening branch was the Woburn Branch.

SELECT SUM(AVAIL\_BALANCE) AS TOTAL\_AVAILABLE\_BALANCE

FROM ACCOUNT

WHERE OPEN\_BRANCH\_ID = 2;

1，1057.75

2，500

3，3000

4，2258.02

5，200

14，5000

1057.75+500+3000+2258.02+200+5000=12015.77

1. Produce a list of account product types and the highest available balance for each.

SELECT PRODUCT\_CD, MAX(AVAIL\_BALANCE) AS Max\_Available\_Balance

FROM ACCOUNT

GROUP BY PRODUCT\_CD;

CHK:23575.12

SAV:767.77

CD:50000

MM:9845.55

BUS:0

1. What is the minimum available balance?

SELECT MIN(AVAIL\_BALANCE) AS Min\_Available\_Balance

FROM ACCOUNT;

0

1. Produce a list of the total available balance per customer. The balance displayed should be rounded down.

CELECT CUST\_ID,

FLOOR（SUM（AVAIL\_BALANCE））AS Total\_Rounded\_Down\_Balance FROM ACCOUNT GROUP BY CUST\_ID;

- Customer 1: 1057.75 + 500 + 3000 = 4557.75 -> 4557

- Customer 2: 2258.02 + 200 = 2458.02 -> 2458

- Customer 3: 1057.75 + 2212.5 = 3270.25 -> 3270

- Customer 4: 2258.02 + 200 + 767.77 + 5487.09 = 8712.88 -> 8712

- Customer 5: 2237.97 -> 2237

- Customer 6: 122.37 + 10000 = 10122.37 -> 10122

- Customer 7: 5000 -> 5000

- Customer 8: 3487.19 + 387.99 = 3875.18 -> 3875

- Customer 9: 125.67 + 9345.55 + 1500 = 10971.22 -> 10971

- Customer 10: 23575.12 + 0 = 23575.12 -> 23575

- Customer 12: 38552.05 -> 38552

- Customer 13: 50000 -> 50000

12. The Output lists of EMPLOYEE details in the following formats (only one column should be output):

a. Employees Name; [LAST\_NAME], [FIRST\_NAME] e.g. Smith, Michael

SELECT LAST\_NAME || ', ' || FIRST\_NAME AS Employee\_Name

FROM EMPLOYEE;

1. Employee Position; [FIRST\_NAME] [LAST\_NAME] Position: [TITLE] e.g. Michael Smith Position:

SELECT FIRST\_NAME || ' ' || LAST\_NAME || ' Position: ' || TITLE AS Employee\_Position

FROM EMPLOYEE;

name:Smith,Michael

position:Michael Smith Position:President

1. Update ‘Fear leads to anger; anger leads to hatred; hatred leads to conflict; conflict leads to suffering’ set ‘Fear leads to panic buying; Panic buying leads to hatred; hatred leads to conflict; conflict leads to suffering.’;
2. UPDATE CUSTOMER

SET FED\_ID = CASE

WHEN FED\_ID LIKE '%-%' AND LENGTH(FED\_ID) = 10 THEN SUBSTRING(FED\_ID, 1, 3) || SUBSTRING(FED\_ID, 4, 7)

WHEN FED\_ID LIKE '%-%' AND LENGTH(FED\_ID) = 9 THEN SUBSTRING(FED\_ID, 1, 2) || SUBSTRING(FED\_ID, 3, 7)

ELSE FED\_ID

END

WHERE FED\_ID LIKE '%-%';

1. SELECT YEAR(TXN\_DATE) AS Year, COUNT(\*) AS Count

FROM ACC\_TRASACTION

GROUP BY YEAR(TXN\_DATE)

ORDER BY Year;

1. UPDATE EMPLOYEE

SET Job\_Title = UPPER(Job\_Title);

SELECT

COUNT(\*) AS Count,

CASE

WHEN Job\_Title LIKE '%TELLER%' THEN 'Cashier'

ELSE 'Other'

END AS JobTitle

FROM EMPLOYEE

GROUP BY JobTitle

ORDER BY JobTitle;

1. SELECT

c.CUST\_ID,

c.ADDRESS,

c.CITY,

SUM(a.AVAL\_BALANCE) AS TOTAL\_AVAILABLE\_BALANCE

FROM

CUSTOMER c

JOIN

ACCOUNTS a ON c.CUST\_ID = a.CUST\_D

GROUP BY

c.CUST\_ID,

c.ADDRESS,

c.CITY

HAVING

SUM(a.AVAL\_BALANCE) < 5000;

1. SELECT

b.BRANCH\_ID,

b.NAME,

COUNT(e.EMP\_ID) AS TOTAL\_EMPLOYEES

FROM

BRANCH b

LEFT JOIN

EMPLOYEE e ON b.BRANCH\_ID = e.ASSIGNED\_BRANCH\_ID

GROUP BY

b.BRANCH\_ID,

b.NAME

ORDER BY

b.BRANCH\_ID;

1. SELECT

COUNT(\*) AS Count,

CASE

WHEN PRODUCT\_CD = 'CHK' THEN 'Checking Account'

WHEN PRODUCT\_CD = 'SAV' THEN 'Savings Account'

END AS Product\_Type

FROM

ACCOUNTS

WHERE

PRODUCT\_CD IN ('CHK', 'SAV')

GROUP BY

PRODUCT\_CD;