LAB REPORT

COURSE TITLE	:	Database Systems Lab
COURSE CODE	:	CSE 208
LAB REPORT NO.	:	02
SUBMISSION DATE	:	06-08-2025

SUBMITTED TO

NAME	:	Farha Akhter Munmun
DEPT. OF	:	Computer Science and Engineering (CSE) Bangladesh University of Business & Technology (BUBT)

SUBMITTED BY

NAME	:	Shadman Shahriar
ID NO.	:	20245103408
INTAKE	:	53
SECTION	:	1
PROGRAM	:	B.Sc. Engg. in CSE

Database Systems Lab

Prerequisite

Creating a new database:

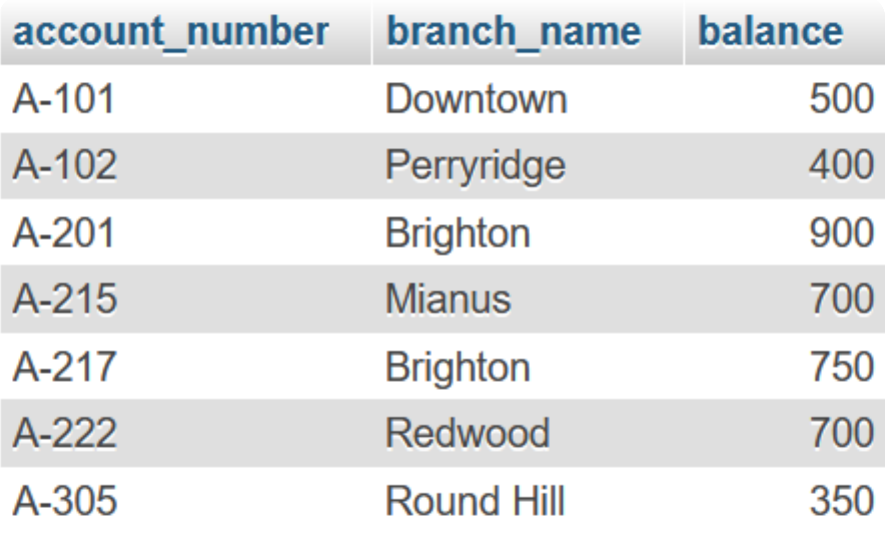
```sql
CREATE DATABASE bank_408;
```

Creating six tables: `account` , `branch` , `depositor` , `customer` , `loan` and `borrower`

```sql
CREATE TABLE account (account_number char(5), branch_name
varchar(32), balance int(7));
CREATE TABLE branch (branch_name varchar(32), branch_city
varchar(32), assets int(8));
CREATE TABLE depositor (customer_name varchar(32),
account_number char(5));
CREATE TABLE customer (customer_name varchar(32),
customer_street varchar(32), customer_city varchar(32));
CREATE TABLE loan (loan_number char(5), branch_name
varchar(32), amount int(8));
CREATE TABLE borrower (customer_name varchar(32),
loan_number char(8));
```

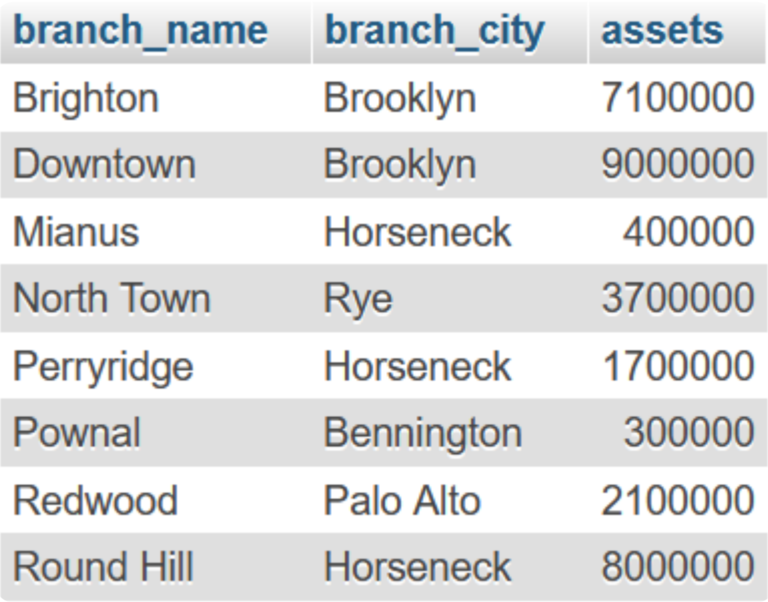
Inserting data to the tables:

```sql
-- Inserting data into `account`
INSERT INTO account VALUES("A-101", "Downtown", 500);
INSERT INTO account VALUES("A-102", "Perryridge", 400);
INSERT INTO account VALUES("A-201", "Brighton", 900);
INSERT INTO account VALUES("A-215", "Mianus", 700);
INSERT INTO account VALUES("A-217", "Brighton", 750);
INSERT INTO account VALUES("A-222", "Redwood", 700);
INSERT INTO account VALUES("A-305", "Round Hill", 350);

-- Inserting data into `branch`
INSERT INTO branch VALUES("Brighton", "Brooklyn", 7100000);
INSERT INTO branch VALUES("Downtown", "Brooklyn", 9000000);
INSERT INTO branch VALUES("Mianus", "Horseneck", 400000);
INSERT INTO branch VALUES("North Town", "Rye", 3700000);
INSERT INTO branch VALUES("Perryridge", "Horseneck",
1700000);
INSERT INTO branch VALUES("Pownal", "Bennington", 300000);
INSERT INTO branch VALUES("Redwood", "Palo Alto", 2100000);
INSERT INTO branch VALUES("Round Hill", "Horseneck",
8000000);

-- Inserting data into `depositor`
INSERT INTO depositor VALUES("Hayes", "A-102");
INSERT INTO depositor VALUES("Johnson", "A-101");
INSERT INTO depositor VALUES("Johnson", "A-201");
INSERT INTO depositor VALUES("Jones", "A-217");
INSERT INTO depositor VALUES("Lindsay", "A-222");
INSERT INTO depositor VALUES("Smith", "A-215");
INSERT INTO depositor VALUES("Turner", "A-305");

-- Inserting data into `customer`
INSERT INTO customer VALUES("Adams", "Spring",
"Pittsfield");
INSERT INTO customer VALUES("Brooks", "Senator",
"Brooklyn");
INSERT INTO customer VALUES("Curry", "North", "Rye");
INSERT INTO customer VALUES("Glenn", "Sand Hill",
"Woodside");
INSERT INTO customer VALUES("Green", "Walnut", "Stamford");
INSERT INTO customer VALUES("Hayes", "Main", "Harrison");
INSERT INTO customer VALUES("Johnson", "Alma", "Palo
Alto");
INSERT INTO customer VALUES("Jones", "Main", "Harrison");
INSERT INTO customer VALUES("Lindsay", "Park",
"Pittsfield");
INSERT INTO customer VALUES("Smith", "North", "Rye");
INSERT INTO customer VALUES("Turner", "Putnam",
"Stamford");
INSERT INTO customer VALUES("Williams", "Nassau",
"Princeton");

-- Inserting data into `loan`
INSERT INTO loan VALUES("L-11", "Round Hill", 900);
INSERT INTO loan VALUES("L-14", "Downtown", 1500);
INSERT INTO loan VALUES("L-15", "Perryridge", 1500);
INSERT INTO loan VALUES("L-16", "Perryridge", 1300);
INSERT INTO loan VALUES("L-17", "Downtown", 1000);
INSERT INTO loan VALUES("L-23", "Redwoord", 2000);
INSERT INTO loan VALUES("L-93", "Mianus", 500);

-- Inserting data into `borrower`
INSERT INTO borrower VALUES("Adams", "L-16");
INSERT INTO borrower VALUES("Curry", "L-93");
INSERT INTO borrower VALUES("Hayes", "L-15");
INSERT INTO borrower VALUES("Johnson", "L-14");
INSERT INTO borrower VALUES("Jones", "L-17");
INSERT INTO borrower VALUES("Smith", "L-11");
INSERT INTO borrower VALUES("Smith", "L-23");
INSERT INTO borrower VALUES("Williams", "L-17");
```



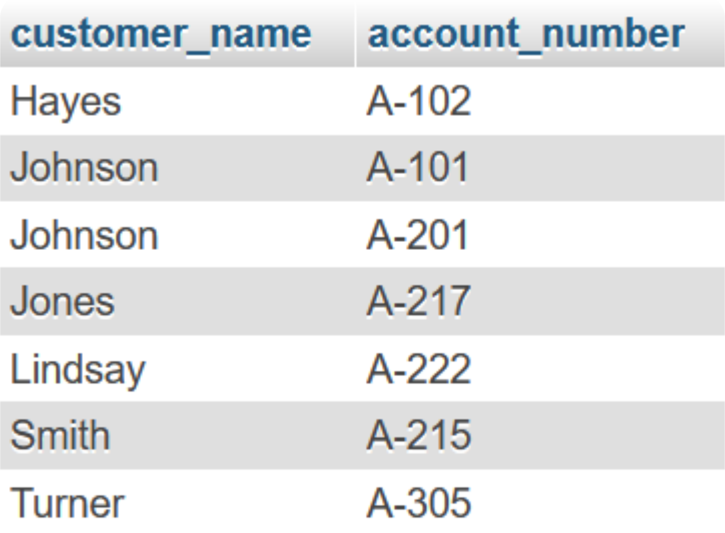
account_number	branch_name	balance
A-101	Downtown	500
A-102	Perryridge	400
A-201	Brighton	900
A-215	Mianus	700
A-217	Brighton	750
A-222	Redwood	700
A-305	Round Hill	350

Figure - 1.1. account relation



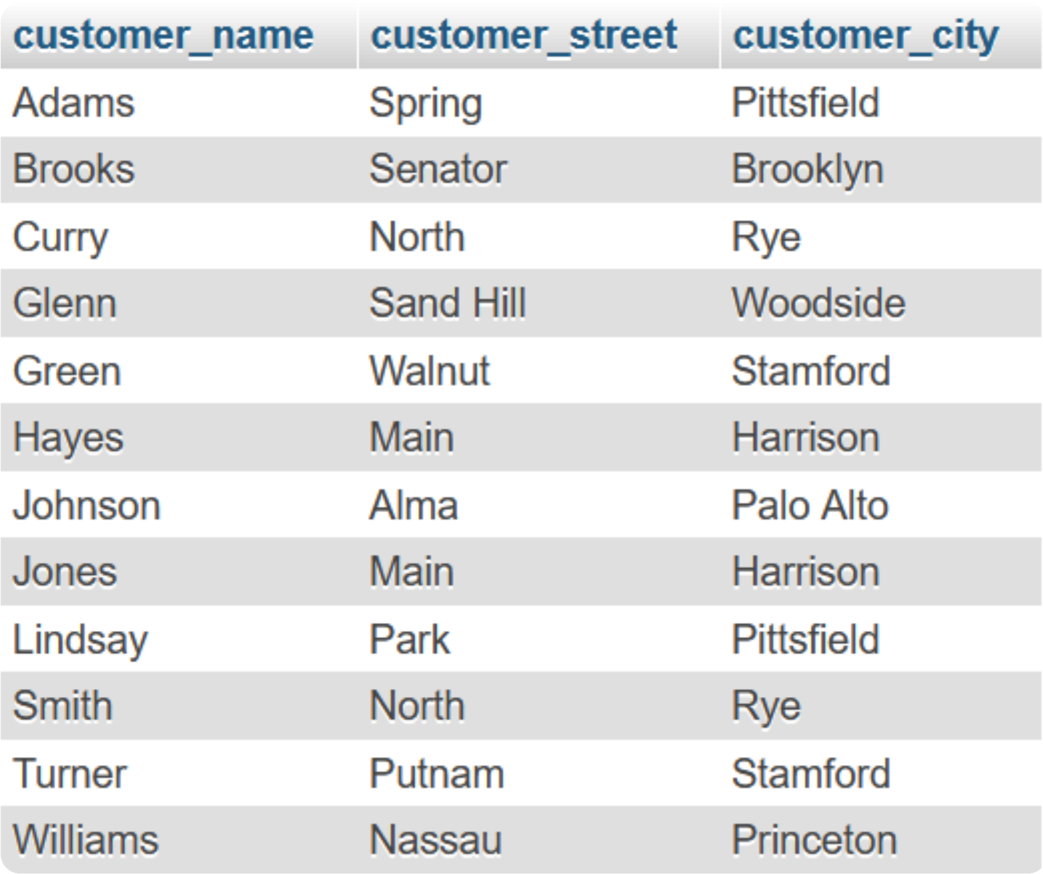
branch_name	branch_city	assets
Brighton	Brooklyn	7100000
Downtown	Brooklyn	9000000
Mianus	Horseneck	400000
North Town	Rye	3700000
Perryridge	Horseneck	1700000
Pownal	Bennington	300000
Redwood	Palo Alto	2100000
Round Hill	Horseneck	8000000

Figure - 1.2. branch relation



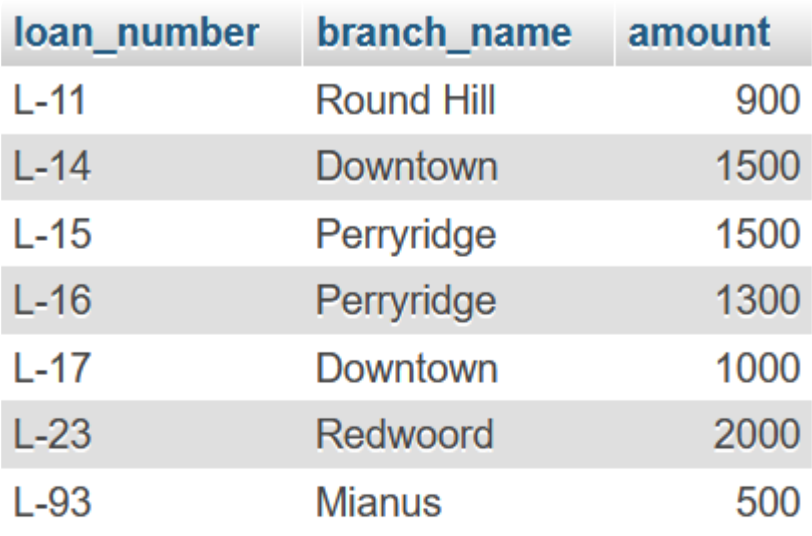
customer_name	account_number
Hayes	A-102
Johnson	A-101
Johnson	A-201
Jones	A-217
Lindsay	A-222
Smith	A-215
Turner	A-305

Figure - 1.3. depositor relation



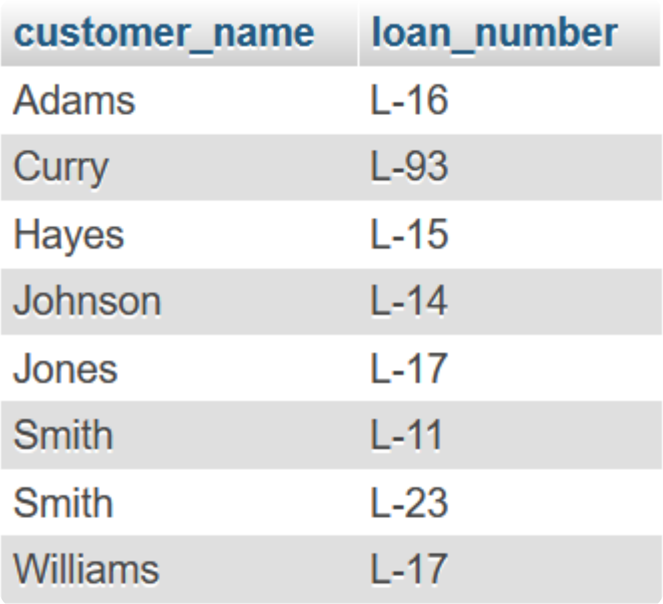
customer_name	customer_street	customer_city
Adams	Spring	Pittsfield
Brooks	Senator	Brooklyn
Curry	North	Rye
Glenn	Sand Hill	Woodside
Green	Walnut	Stamford
Hayes	Main	Harrison
Johnson	Alma	Palo Alto
Jones	Main	Harrison
Lindsay	Park	Pittsfield
Smith	North	Rye
Turner	Putnam	Stamford
Williams	Nassau	Princeton

Figure - 1.4. customer relation



loan_number	branch_name	amount
L-11	Round Hill	900
L-14	Downtown	1500
L-15	Perryridge	1500
L-16	Perryridge	1300
L-17	Downtown	1000
L-23	Redwoord	2000
L-93	Mianus	500

Figure - 1.5. loan relation



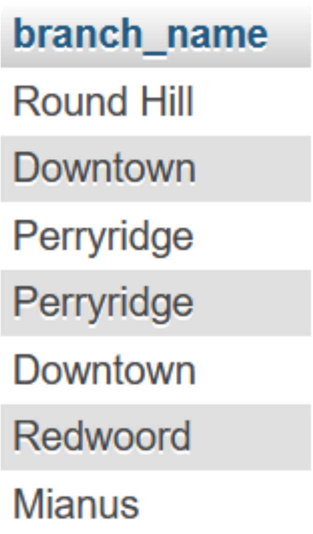
customer_name	loan_number
Adams	L-16
Curry	L-93
Hayes	L-15
Johnson	L-14
Jones	L-17
Smith	L-11
Smith	L-23
Williams	L-17

Figure - 1.6. borrower relation

Lab Tasks

1. Find the names of all branches in the `loan` relation

```
SELECT branch_name FROM loan;
```

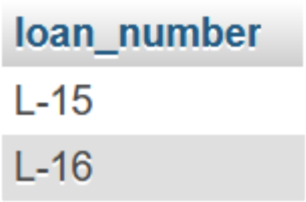


branch_name
Round Hill
Downtown
Perryridge
Perryridge
Downtown
Redwoord
Mianus

Figure - 1.7. Task 1

2. Find all loan numbers for loans made at the `Perryridge` branch with loan amounts greater than 300

```
SELECT loan_number FROM loan WHERE branch_name="Perryridge"
AND amount > 300;
```

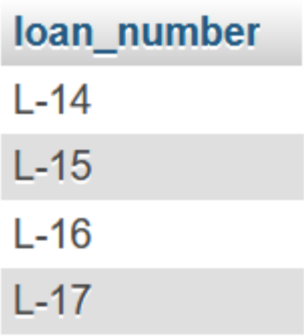


loan_number
L-15
L-16

Figure - 1.8. Task 2

3. Find all the loan numbers of the customers who has loan either `Perryridge` branch or `Downtown` branch

```
SELECT loan_number FROM loan WHERE branch_name IN("Perryridge
"Downtown");
```

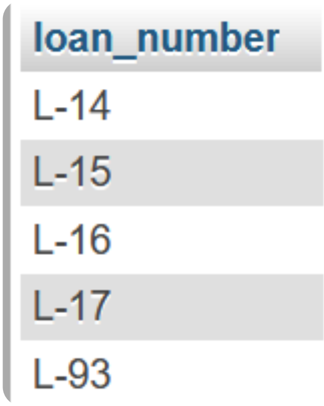


loan_number
L-14
L-15
L-16
L-17

Figure - 1.9. Task 3

4. Find all the loan numbers of the customers who has loan either `Perryridge` branch or `Downtown` branch or `Mianus` branch

```
SELECT loan_number FROM loan WHERE branch_name IN("Perryridge
"Downtown", "Mianus");
```

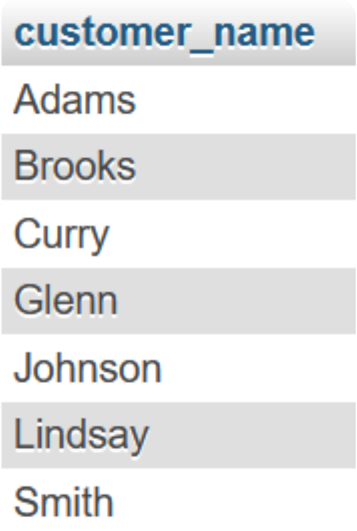


loan_number
L-14
L-15
L-16
L-17
L-93

Figure - 1.10. Task 4

5. Find the names of all customers who are not from `Stamford` or `Princeton` or `Harrison` city

```
SELECT customer_name FROM customer WHERE customer_city NOT
IN("Stamford", "Princeton", "Harrison");
```

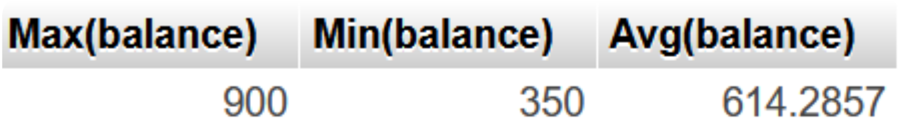


customer_name
Adams
Brooks
Curry
Glenn
Johnson
Lindsay
Smith

Figure - 1.11. Task 5

6. Find the largest, minimum, and average account balance in the `account` relation

```
SELECT Max(balance), Min(balance), Avg(balance) FROM
account;
```

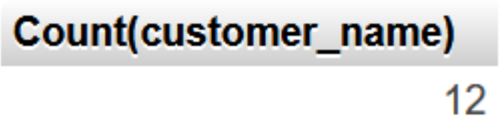


Max(balance)	Min(balance)	Avg(balance)
900	350	614.2857

Figure - 1.12. Task 6

7. Find the total number of customer from `customer` relation

```
SELECT Count(customer_name) FROM customer;
```

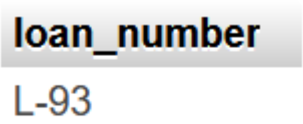


Count(customer_name)
12

Figure - 1.13. Task 7

8. Find the loan number of those loans with loan amounts between 400 and 800

```
SELECT loan_number FROM loan WHERE amount > 400 AND amount
< 800;
```

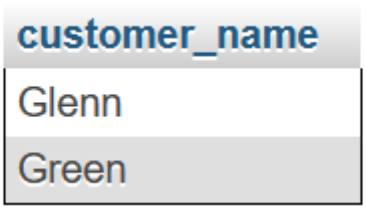


loan_number
L-93

Figure - 1.14. Task 8

9. Find the names of all customers whose name start with `G`

```
SELECT customer_name FROM customer WHERE customer_name LIKE
"G%";
```

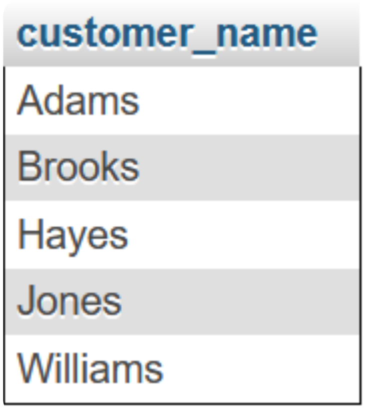


customer_name
Glenn
Green

Figure - 1.15. Task 9

10. Find the names of all customers whose name ends with `s`

```
SELECT customer_name FROM customer WHERE customer_name LIKE
"%s";
```

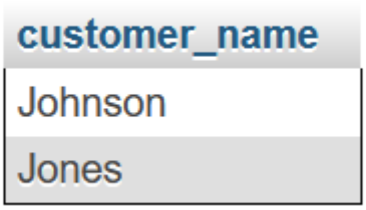


customer_name
Adams
Brooks
Hayes
Jones
Williams

Figure - 1.16. Task 10

11. Find the names of all customers whose name has a `o` in 2nd position

```
SELECT customer_name FROM customer WHERE customer_name LIKE
"_o%";
```

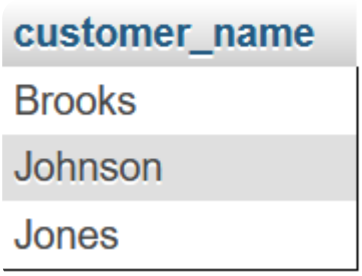


customer_name
Johnson
Jones

Figure - 1.17. Task 11

12. Find the names of all customers whose name has a `o` in any position except 1st and last letter

```
SELECT customer_name FROM customer WHERE customer_name LIKE
"%o%";
```

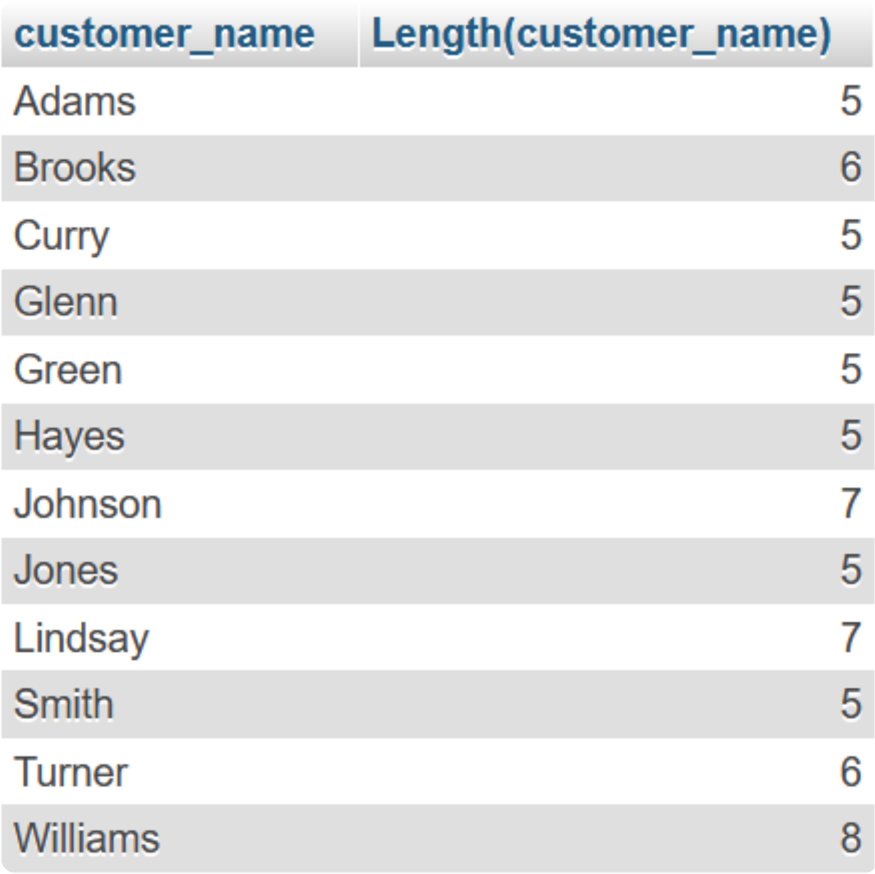


customer_name
Brooks
Johnson
Jones

Figure - 1.18. Task 12

13. Find the length of the name of all customers from `Customer` relation

```
SELECT customer_name, Length(customer_name) FROM customer;
```

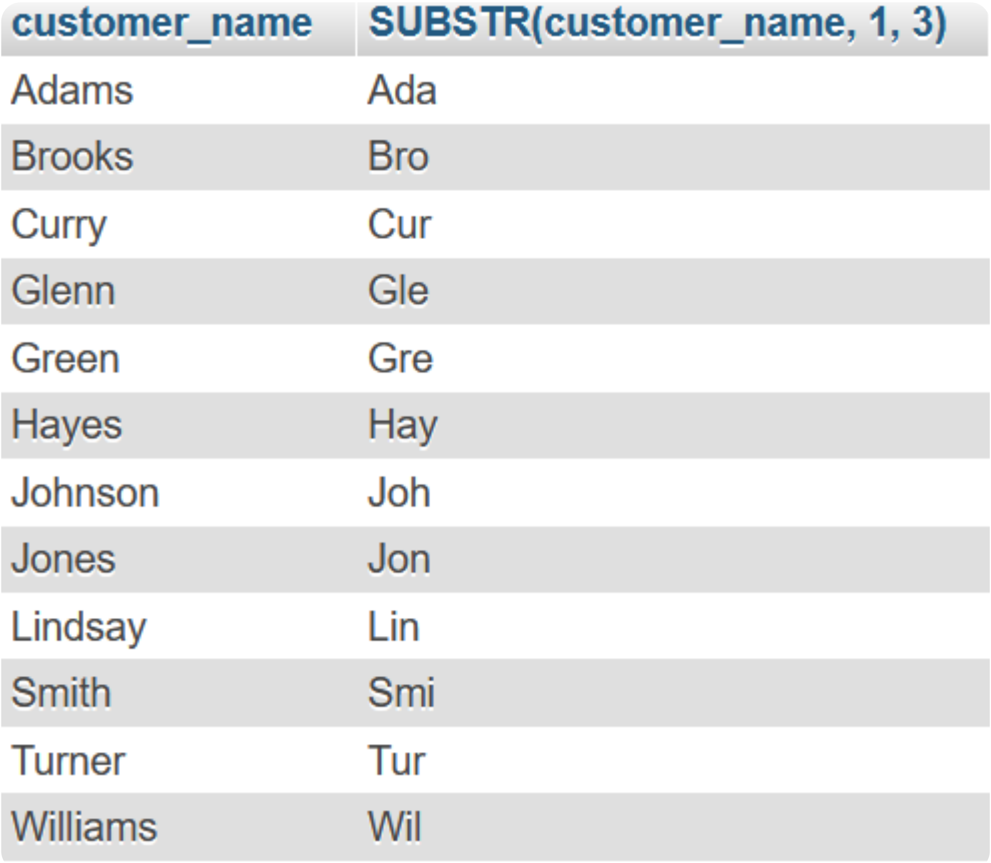


customer_name	Length(customer_name)
Adams	5
Brooks	6
Curry	5
Glenn	5
Green	5
Hayes	5
Johnson	7
Jones	5
Lindsay	7
Smith	5
Turner	6
Williams	8

Figure - 1.19. Task 13

14. Find 1st three characters of each customer name from `customer` relation

```
SELECT customer_name, SUBSTR(customer_name, 1, 3) FROM
customer;
```



customer_name	SUBSTR(customer_name, 1, 3)
Adams	Ada
Brooks	Bro
Curry	Cur
Glenn	Gle
Green	Gre
Hayes	Hay
Johnson	Joh
Jones	Jon
Lindsay	Lin
Smith	Smi
Turner	Tur
Williams	Wil

Figure - 1.20. Task 14