CSE 4308: Database Management Systems Lab

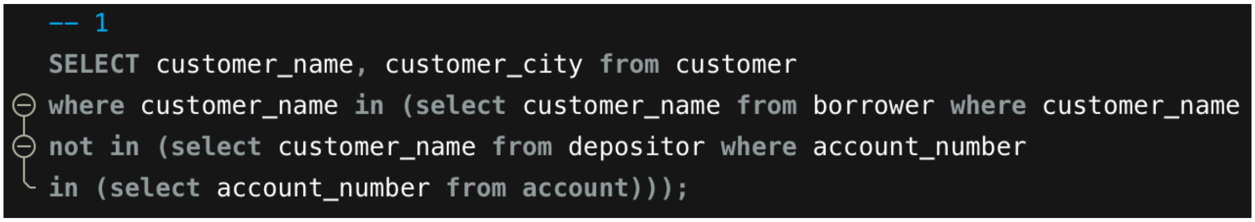
Lab-03

Group-B

Mohammod Mahbub Ur Rahman
220042148
BSc.in Software Engineering

Tasks

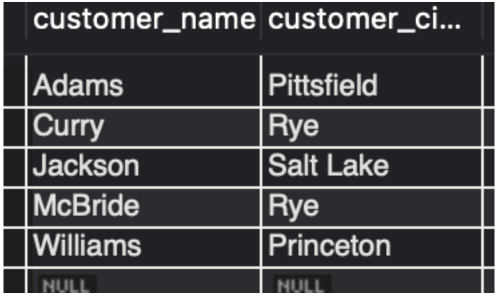
1.



```
-- 1
SELECT customer_name, customer_city from customer
where customer_name in (select customer_name from borrower where customer_name
not in (select customer_name from depositor where account_number
in (select account_number from account)));
```

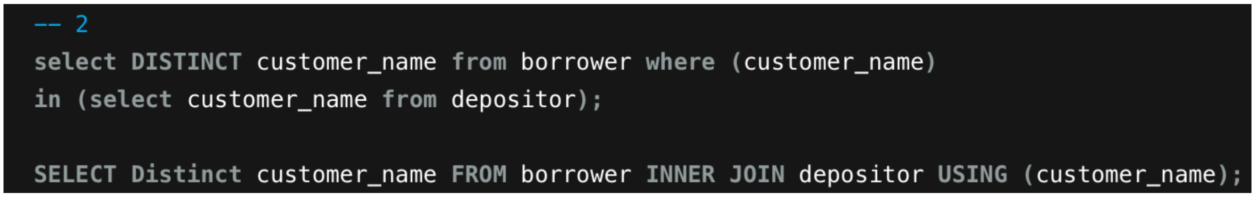
Here I am selecting customer name from borrower whose name is not in the account table

Output:



customer_name	customer_ci...
Adams	Pittsfield
Curry	Rye
Jackson	Salt Lake
McBride	Rye
Williams	Princeton
NULL	NULL

2.

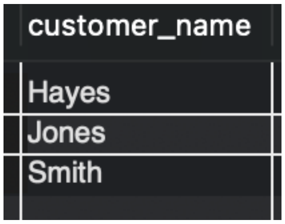


```
-- 2
select DISTINCT customer_name from borrower where (customer_name)
in (select customer_name from depositor);

SELECT Distinct customer_name FROM borrower INNER JOIN depositor USING (customer_name);
```

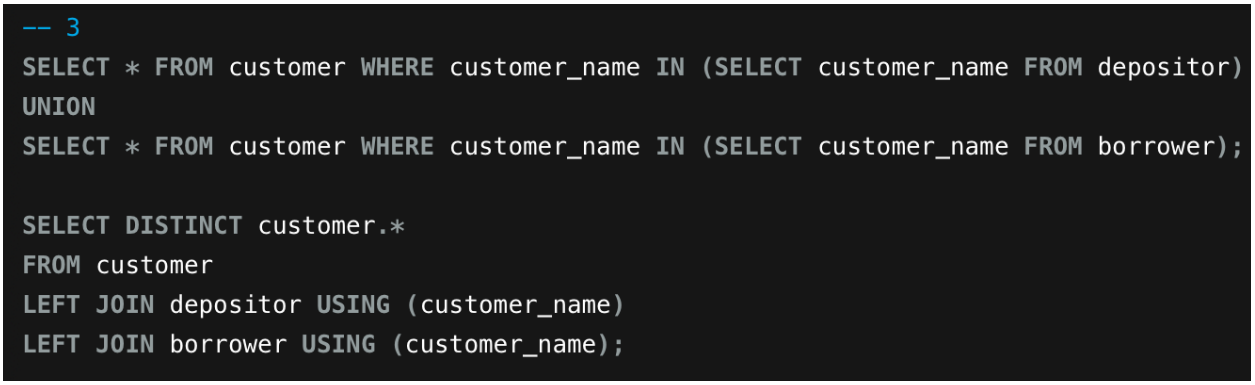
Selecting distinct customer name from borrower table where the customer’s name is also in depositor table

Using set: by inner joining I am selecting the common entry from both table who have the same name.



customer_name
Hayes
Jones
Smith

3.

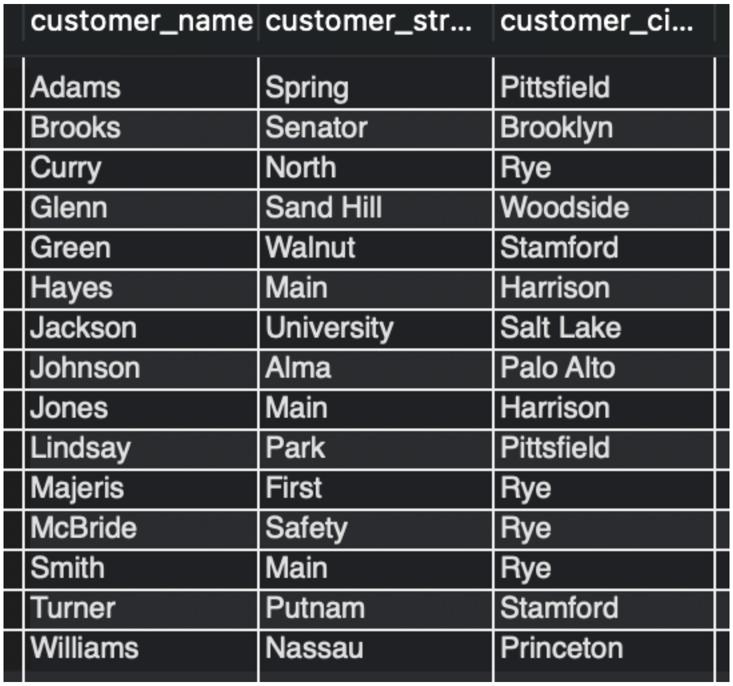


```
-- 3
SELECT * FROM customer WHERE customer_name IN (SELECT customer_name FROM depositor)
UNION
SELECT * FROM customer WHERE customer_name IN (SELECT customer_name FROM borrower);

SELECT DISTINCT customer.*
FROM customer
LEFT JOIN depositor USING (customer_name)
LEFT JOIN borrower USING (customer_name);
```

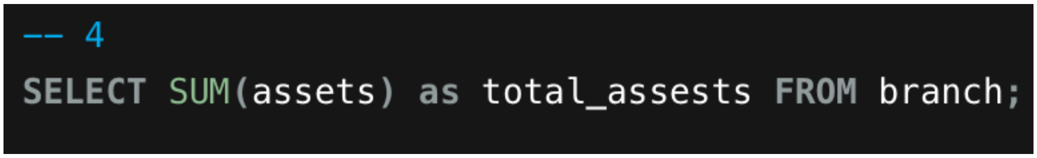
I am doing union operation in both table depositor and borrower. Where they have the name in customer table.

Left join: it joins the common entry (here customer name) from the depositor and borrower and adds it in the customer table.



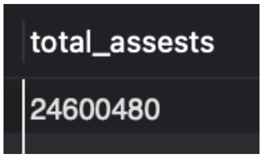
customer_name	customer_str...	customer_ci...
Adams	Spring	Pittsfield
Brooks	Senator	Brooklyn
Curry	North	Rye
Glenn	Sand Hill	Woodside
Green	Walnut	Stamford
Hayes	Main	Harrison
Jackson	University	Salt Lake
Johnson	Alma	Palo Alto
Jones	Main	Harrison
Lindsay	Park	Pittsfield
Majeris	First	Rye
McBride	Safety	Rye
Smith	Main	Rye
Turner	Putnam	Stamford
Williams	Nassau	Princeton

4.



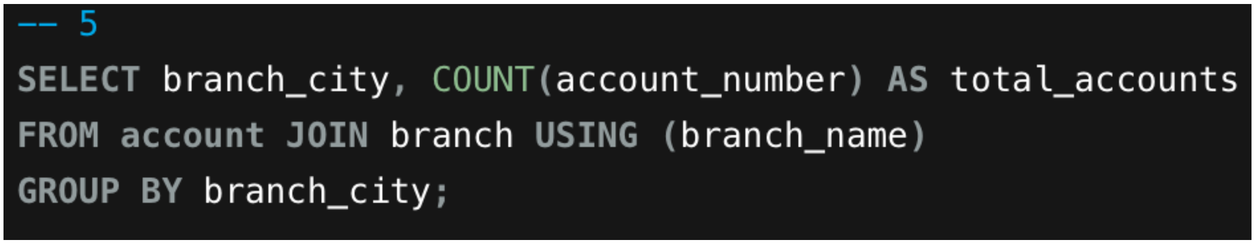
```
-- 4
SELECT SUM(assets) as total_assests FROM branch;
```

It sums the assets of all branches.



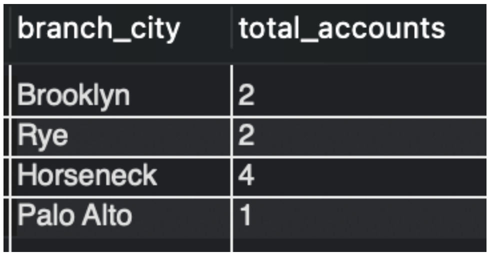
total_assests
24600480

5.



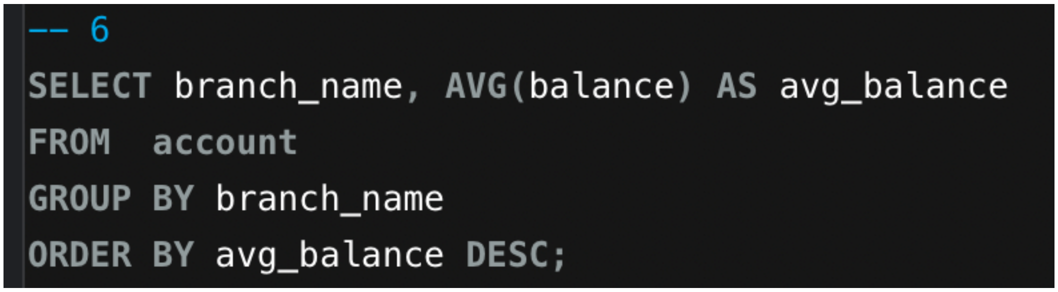
```
-- 5
SELECT branch_city, COUNT(account_number) AS total_accounts
FROM account JOIN branch USING (branch_name)
GROUP BY branch_city;
```

Here I joined account table with branch table using branch name and then counted the account numbers



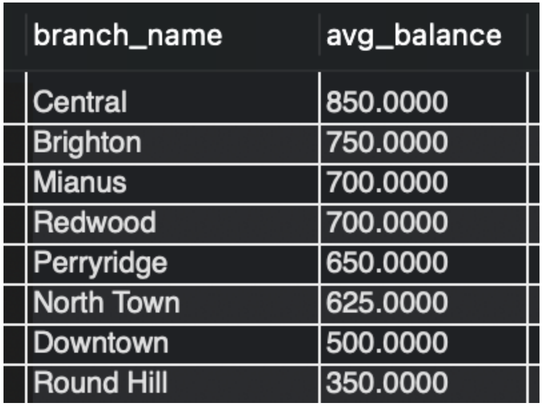
branch_city	total_accounts
Brooklyn	2
Rye	2
Horseneck	4
Palo Alto	1

6.



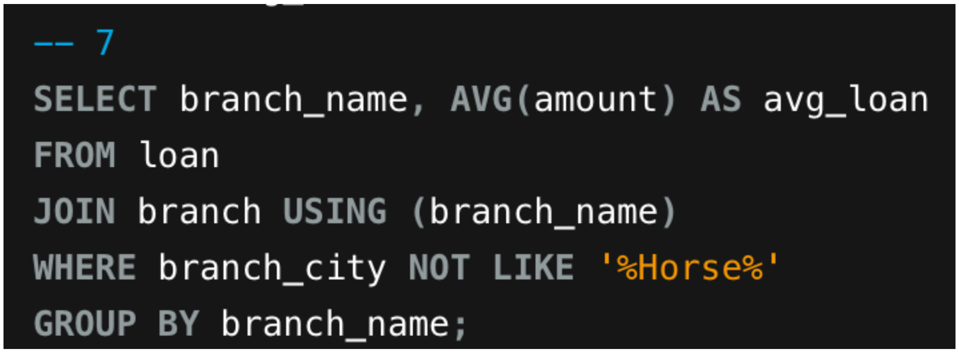
```
-- 6
SELECT branch_name, AVG(balance) AS avg_balance
FROM  account
GROUP BY branch_name
ORDER BY avg_balance DESC;
```

Here I calculated average balance from account and grouped by branch name and ordered by avg_balance in descending order.



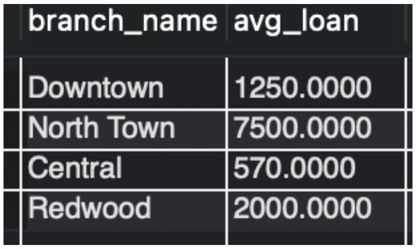
branch_name	avg_balance
Central	850.0000
Brighton	750.0000
Mianus	700.0000
Redwood	700.0000
Perryridge	650.0000
North Town	625.0000
Downtown	500.0000
Round Hill	350.0000

7.



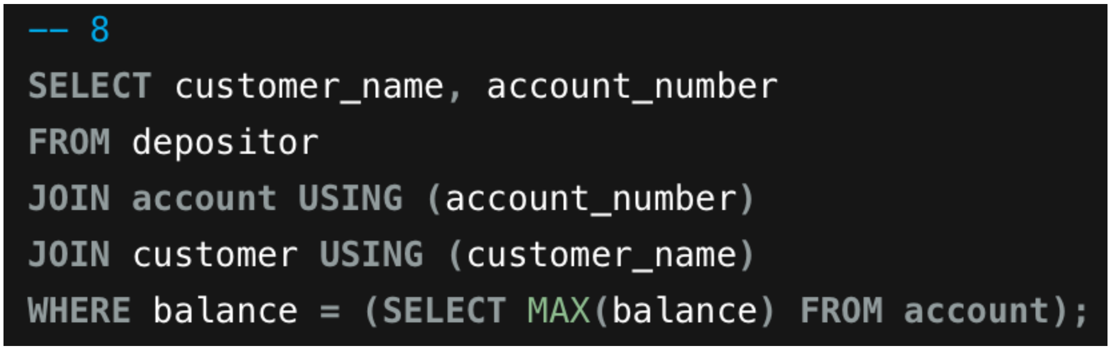
```
-- 7
SELECT branch_name, AVG(amount) AS avg_loan
FROM loan
JOIN branch USING (branch_name)
WHERE branch_city NOT LIKE '%Horse%'
GROUP BY branch_name;
```

Here I am taking branch name and average amount of loan in each branch. I am joining branch with loan using branch name and checking if the city name has horse in it. if not then display it



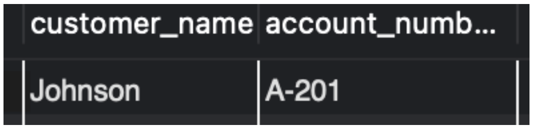
branch_name	avg_loan
Downtown	1250.0000
North Town	7500.0000
Central	570.0000
Redwood	2000.0000

8.



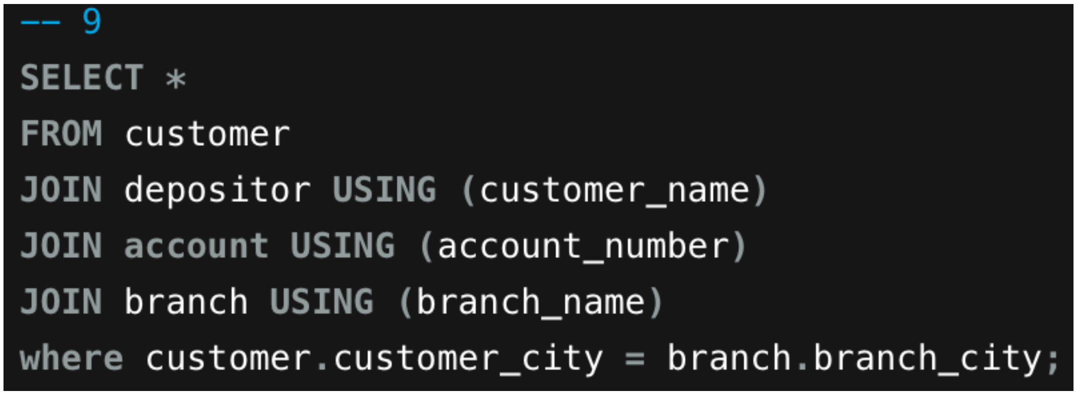
```sql
-- 8
SELECT customer_name, account_number
FROM depositor
JOIN account USING (account_number)
JOIN customer USING (customer_name)
WHERE balance = (SELECT MAX(balance) FROM account);
```

Here I am joining the table account and customer and checking whose balance is max in the account table and returning it.



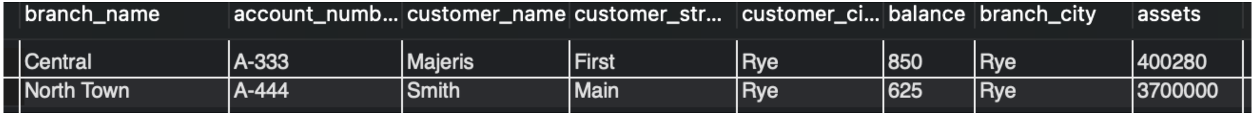
customer_name	account_numb...
Johnson	A-201

9.



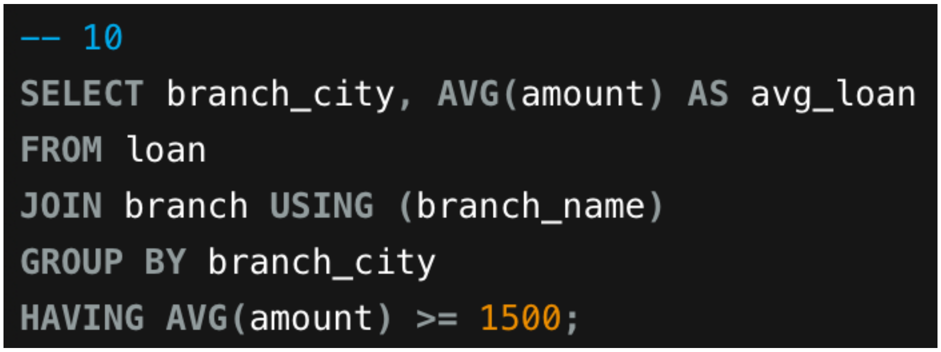
```sql
-- 9
SELECT *
FROM customer
JOIN depositor USING (customer_name)
JOIN account USING (account_number)
JOIN branch USING (branch_name)
where customer.customer_city = branch.branch_city;
```

Here I am joining depositor, account, branch with customer and taking the entry where customer city is equal to the branch city



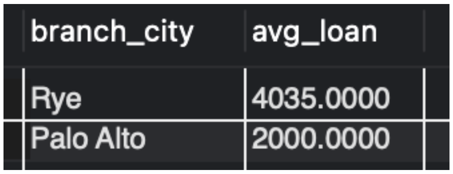
branch_name	account_numb...	customer_name	customer_str...	customer_ci...	balance	branch_city	assets
Central	A-333	Majeris	First	Rye	850	Rye	400280
North Town	A-444	Smith	Main	Rye	625	Rye	3700000

10.



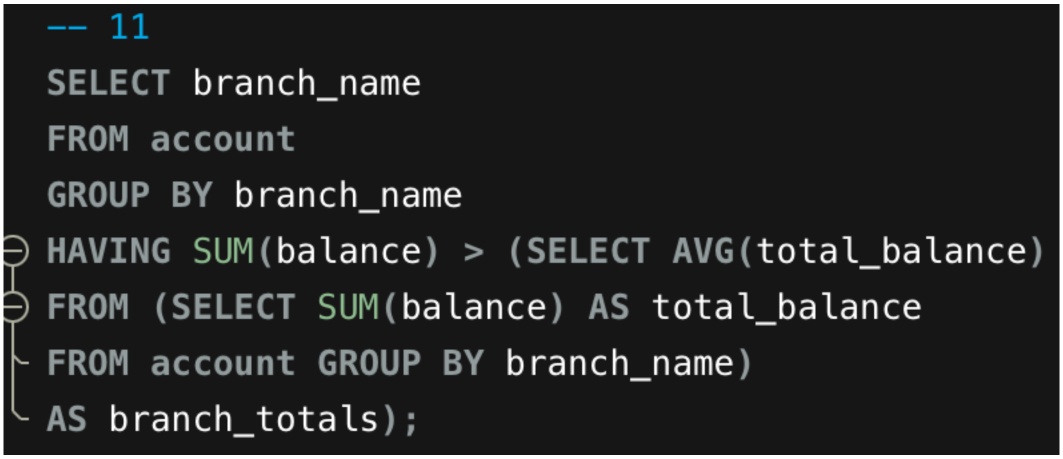
```
-- 10
SELECT branch_city, AVG(amount) AS avg_loan
FROM loan
JOIN branch USING (branch_name)
GROUP BY branch_city
HAVING AVG(amount) >= 1500;
```

I am joining branch with loan and displaying the branch city having average loan greater than 1500.



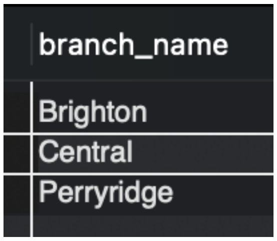
branch_city	avg_loan
Rye	4035.0000
Palo Alto	2000.0000

11.



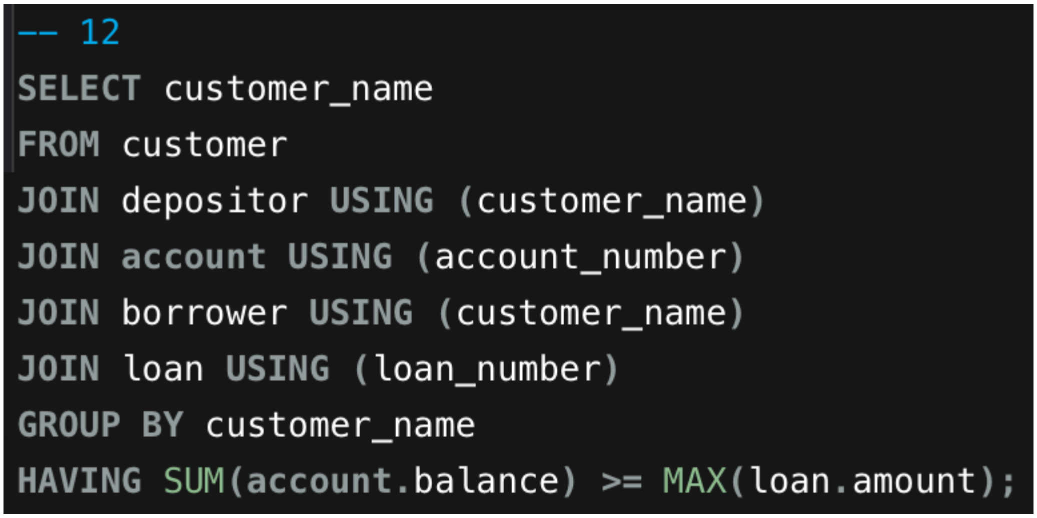
```
-- 11
SELECT branch_name
FROM account
GROUP BY branch_name
HAVING SUM(balance) > (SELECT AVG(total_balance)
FROM (SELECT SUM(balance) AS total_balance
FROM account GROUP BY branch_name)
AS branch_totals);
```

I am selecting branch name whose sum of balance of all account is greater than the average of other branches.



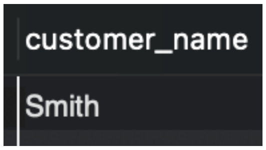
branch_name
Brighton
Central
Perryridge

12.



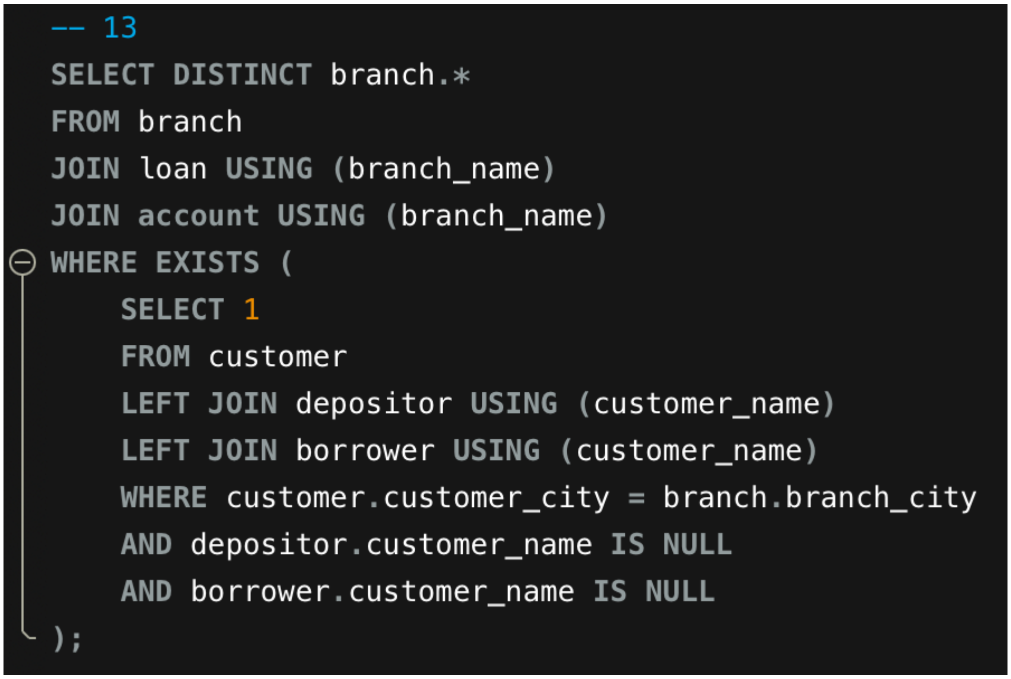
```
-- 12
SELECT customer_name
FROM customer
JOIN depositor USING (customer_name)
JOIN account USING (account_number)
JOIN borrower USING (customer_name)
JOIN loan USING (loan_number)
GROUP BY customer_name
HAVING SUM(account.balance) >= MAX(loan.amount);
```

I am joining depositor, account, borrower, loan with customer and displaying the name who has balance greater than or equal of his loan amount.



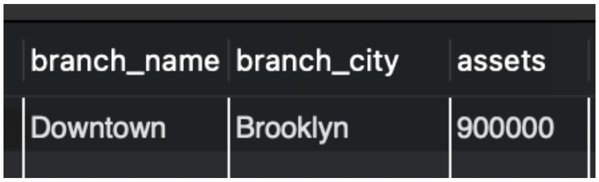
customer_name
Smith

13.



```
-- 13
SELECT DISTINCT branch.*
FROM branch
JOIN loan USING (branch_name)
JOIN account USING (branch_name)
WHERE EXISTS (
    SELECT 1
    FROM customer
    LEFT JOIN depositor USING (customer_name)
    LEFT JOIN borrower USING (customer_name)
    WHERE customer.customer_city = branch.branch_city
    AND depositor.customer_name IS NULL
    AND borrower.customer_name IS NULL
);
```

I am finding the person who doesnot have any loan or bank account. For this I have joined loan, account with branch and checking that if there is a customer in the customer table left joined by depositor and borrower who’s city and branch city is same but his name in depositor and borrower table is null. Simply he has no entry in those table.



branch_name	branch_city	assets
Downtown	Brooklyn	900000