BRAC UNIVERSITY

Inspiring Excellence

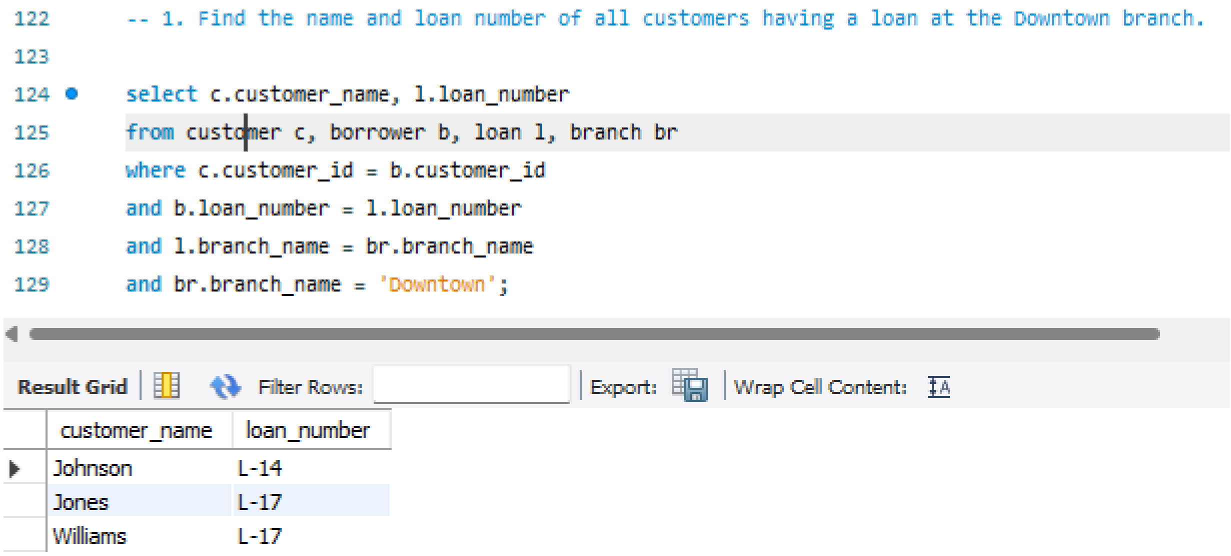
Course Title: Database Management
Course Code: CSE 370
Lab Assignment no: 03

Name: PowerxPochita_Raven
Student ID: RAVEN
Section: xx

Using the bank database, write MySQL queries for the following tasks:
1. Find the name and loan number of all customers having a loan at the Downtown branch. [2]

Command:

```
select c.customer_name, l.loan_number
from customer c, borrower b, loan l, branch br
where c.customer_id = b.customer_id
and b.loan_number = l.loan_number
and l.branch_name = br.branch_name
and br.branch_name = 'Downtown';
```



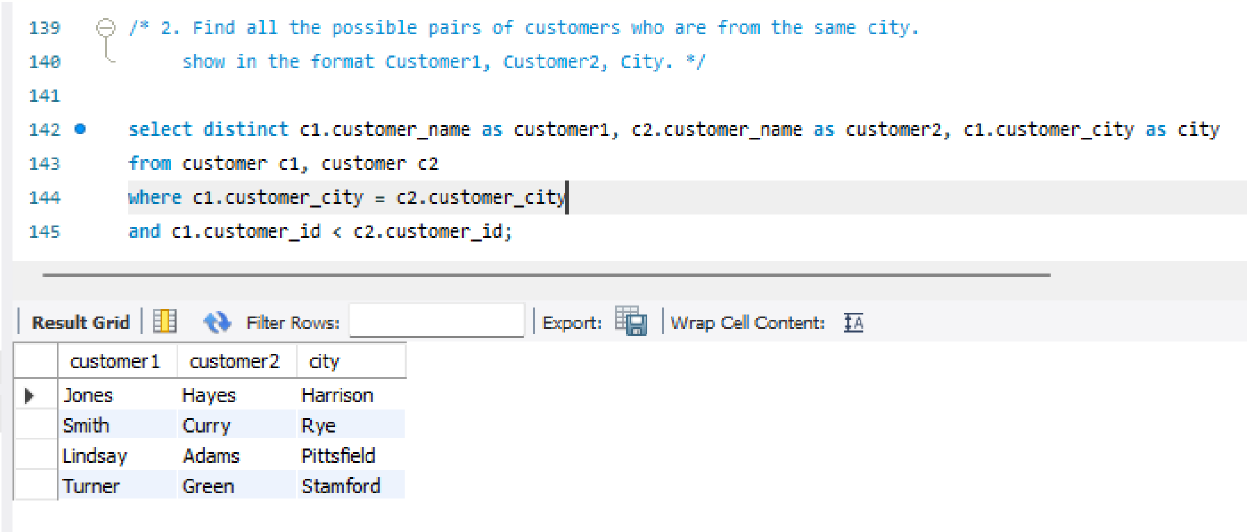
```
-- 1. Find the name and loan number of all customers having a loan at the Downtown branch.

select c.customer_name, l.loan_number
from customer c, borrower b, loan l, branch br
where c.customer_id = b.customer_id
and b.loan_number = l.loan_number
and l.branch_name = br.branch_name
and br.branch_name = 'Downtown';
```

customer_name	loan_number
Johnson	L-14
Jones	L-17
Williams	L-17

2. Find all the possible pairs of customers who are from the same city. show in the format Customer1, Customer2, City. [2]

Command:

```
select distinct c1.customer_name as customer1, c2.customer_name as customer2,
c1.customer_city as city
from customer c1, customer c2
where c1.customer_city = c2.customer_city
and c1.customer_id < c2.customer_id;
```



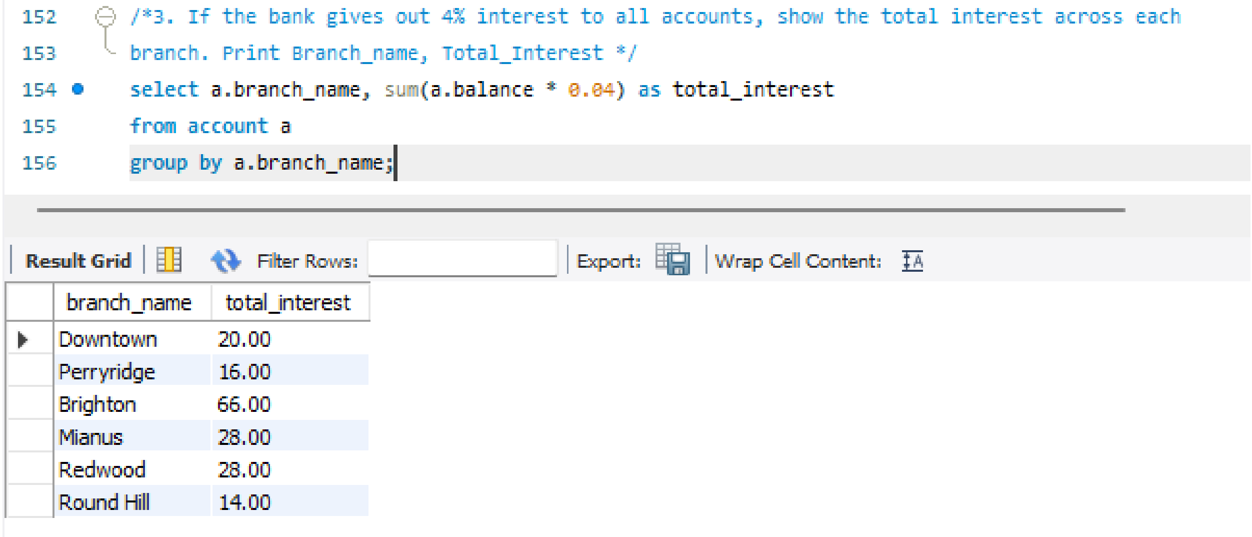
```
/* 2. Find all the possible pairs of customers who are from the same city.
    show in the format Customer1, Customer2, City. */

select distinct c1.customer_name as customer1, c2.customer_name as customer2, c1.customer_city as city
from customer c1, customer c2
where c1.customer_city = c2.customer_city
and c1.customer_id < c2.customer_id;
```

customer1	customer2	city
Jones	Hayes	Harrison
Smith	Curry	Rye
Lindsay	Adams	Pittsfield
Turner	Green	Stamford

3. If the bank gives out 4% interest to all accounts, show the total interest across each branch. Print Branch_name, Total_Interest [1]

Command:

```
select a.branch_name, sum(a.balance * 0.04) as total_interest
from account a
group by a.branch_name;
```



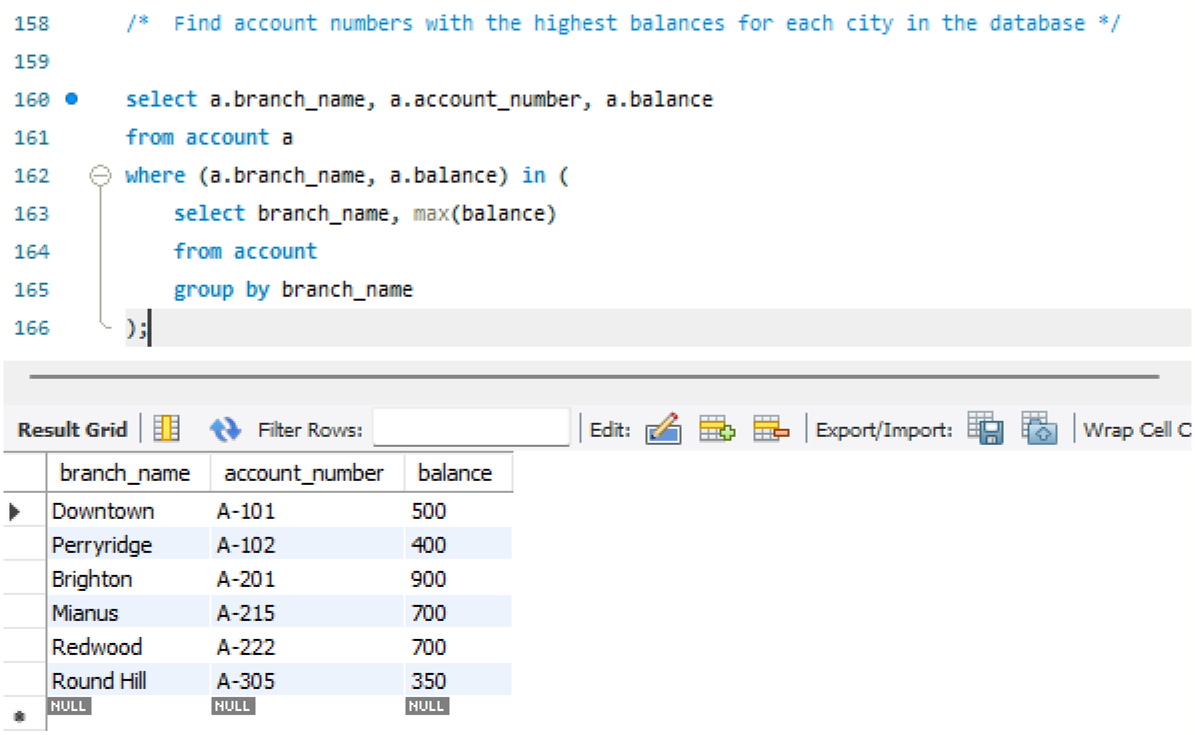
```
/*3. If the bank gives out 4% interest to all accounts, show the total interest across each
branch. Print Branch_name, Total_Interest */
select a.branch_name, sum(a.balance * 0.04) as total_interest
from account a
group by a.branch_name;
```

branch_name	total_interest
Downtown	20.00
Perryridge	16.00
Brighton	66.00
Mianus	28.00
Redwood	28.00
Round Hill	14.00

4. Find account numbers with the highest balances for each city in the database [1]

Command:

```
select a.branch_name, a.account_number, a.balance
from account a
where (a.branch_name, a.balance) in (
    select branch_name, max(balance)
    from account
    group by branch_name
);
```



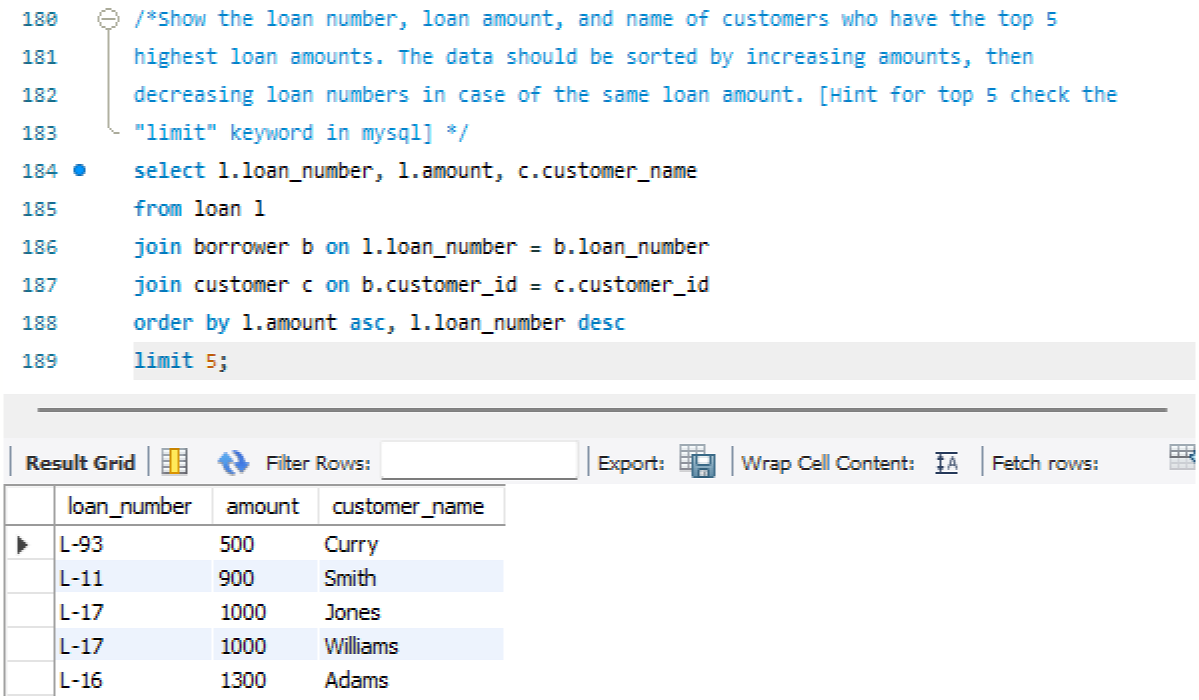
```
/*  Find account numbers with the highest balances for each city in the database */

select a.branch_name, a.account_number, a.balance
from account a
where (a.branch_name, a.balance) in (
    select branch_name, max(balance)
    from account
    group by branch_name
);
```

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

5. Show the loan number, loan amount, and name of customers who have the top 5 highest loan amounts. The data should be sorted by increasing amounts, then decreasing loan numbers in case of the same loan amount. [Hint for top 5 check the "limit" keyword in mysql] [2]

Command:

```
select l.loan_number, l.amount, c.customer_name
from loan l
join borrower b on l.loan_number = b.loan_number
join customer c on b.customer_id = c.customer_id
order by l.amount asc, l.loan_number desc
limit 5;
```



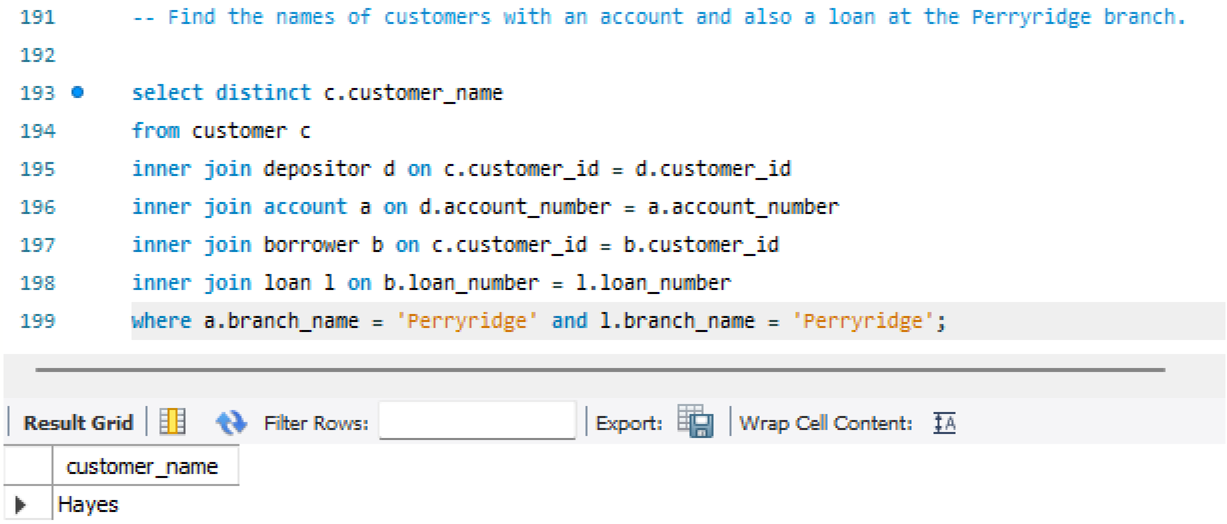
```
/*Show the loan number, loan amount, and name of customers who have the top 5
highest loan amounts. The data should be sorted by increasing amounts, then
decreasing loan numbers in case of the same loan amount. [Hint for top 5 check the
"limit" keyword in mysql] */
select l.loan_number, l.amount, c.customer_name
from loan l
join borrower b on l.loan_number = b.loan_number
join customer c on b.customer_id = c.customer_id
order by l.amount asc, l.loan_number desc
limit 5;
```

loan_number	amount	customer_name
L-93	500	Curry
L-11	900	Smith
L-17	1000	Jones
L-17	1000	Williams
L-16	1300	Adams

6. Find the names of customers with an account and also a loan at the Perryridge branch. [2]

Command:

```
select distinct c.customer_name
from customer c
inner join depositor d on c.customer_id = d.customer_id
inner join account a on d.account_number = a.account_number
inner join borrower b on c.customer_id = b.customer_id
inner join loan l on b.loan_number = l.loan_number
where a.branch_name = 'Perryridge' and l.branch_name = 'Perryridge';
```



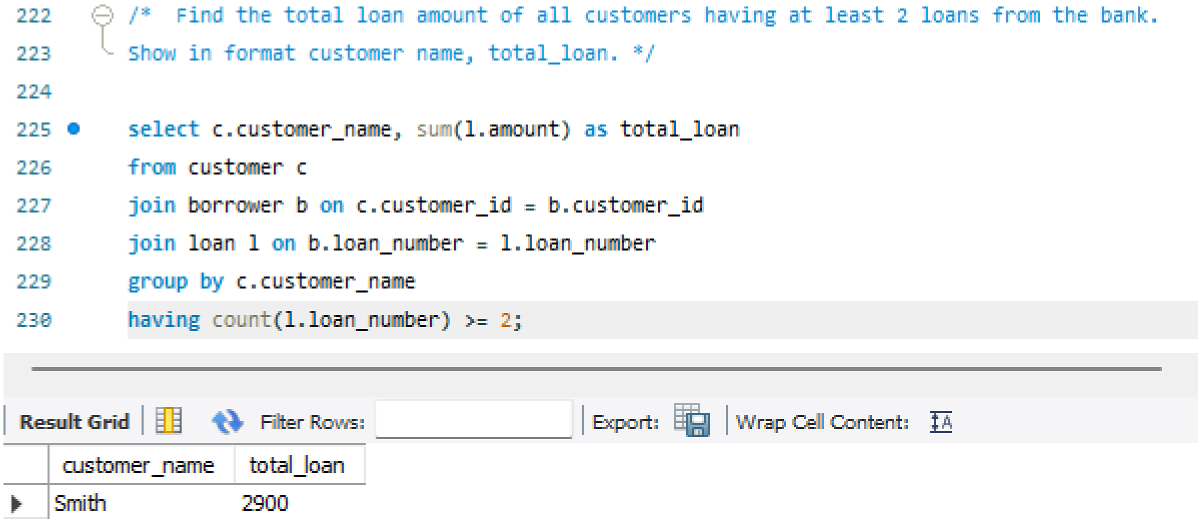
```
-- Find the names of customers with an account and also a loan at the Perryridge branch.

select distinct c.customer_name
from customer c
inner join depositor d on c.customer_id = d.customer_id
inner join account a on d.account_number = a.account_number
inner join borrower b on c.customer_id = b.customer_id
inner join loan l on b.loan_number = l.loan_number
where a.branch_name = 'Perryridge' and l.branch_name = 'Perryridge';
```

customer_name
Hayes

7. Find the total loan amount of all customers having at least 2 loans from the bank. Show in format customer name, total_loan. [2]

Command:

select c.customer_name, sum(l.amount) as total_loan
from customer c
join borrower b on c.customer_id = b.customer_id
join loan l on b.loan_number = l.loan_number
group by c.customer_name
having count(l.loan_number) >= 2;



```
/*  Find the total loan amount of all customers having at least 2 loans from the bank.
Show in format customer name, total_loan. */

select c.customer_name, sum(l.amount) as total_loan
from customer c
join borrower b on c.customer_id = b.customer_id
join loan l on b.loan_number = l.loan_number
group by c.customer_name
having count(l.loan_number) >= 2;
```

customer_name	total_loan
Smith	2900