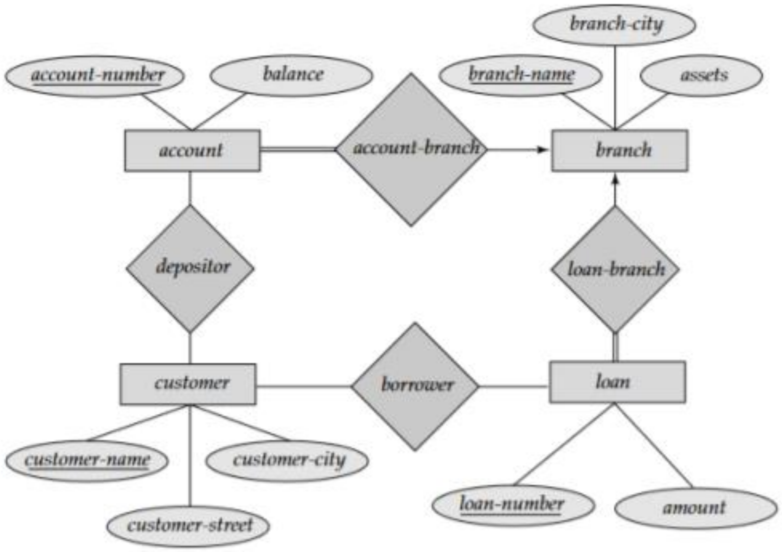
CSLR 51 : DBMS LAB-4

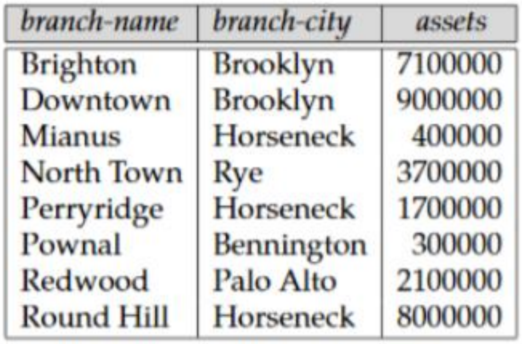
Roll no. : **106119100**

Name : **Rajneesh Pandey**

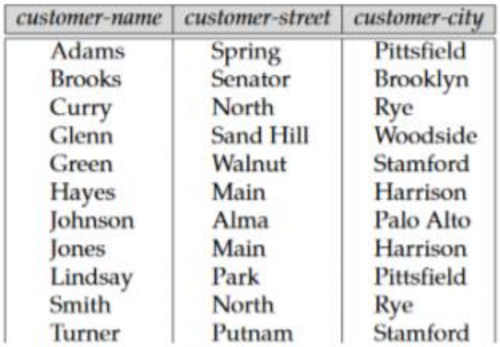
Section : **CSE-B**

PROBLEM 1

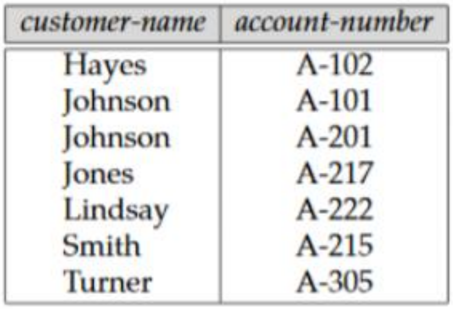




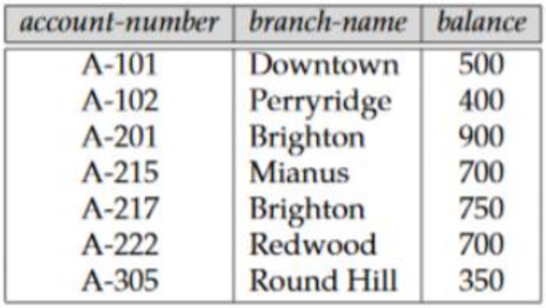
branch-name	branch-city	assets
Brighton	Brooklyn	7100000
Downtown	Brooklyn	9000000
Mianus	Horseneck	400000
North Town	Rye	3700000
Perryridge	Horseneck	1700000
Pownal	Bennington	300000
Redwood	Palo Alto	2100000
Round Hill	Horseneck	8000000



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



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



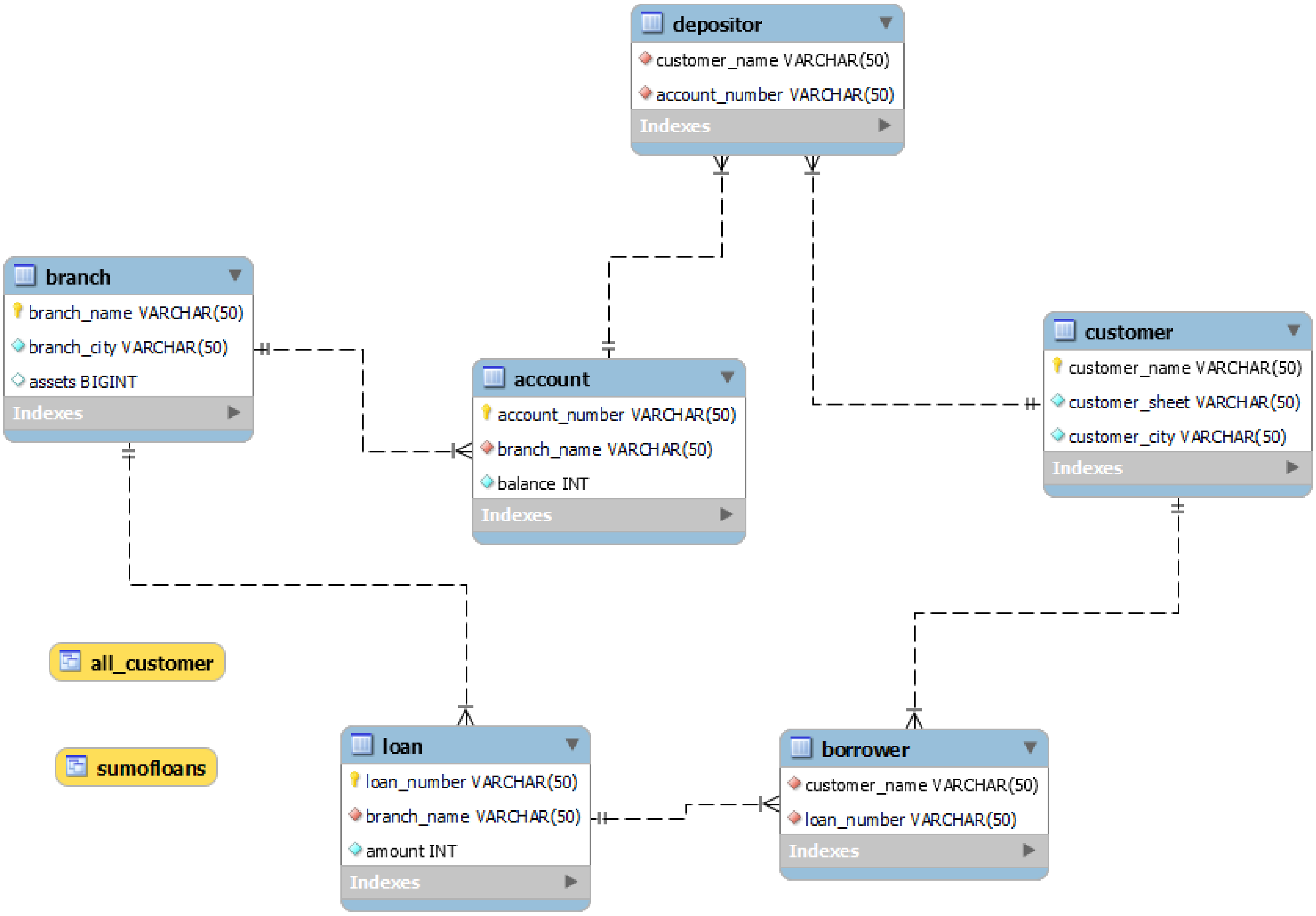
account-number	branch-name	balance
A-101	Downtown	500
A-102	Perryridge	400
A-201	Brighton	900
A-215	Mianus	700
A-217	Brighton	750
A-222	Redwood	700
A-305	Round Hill	350

A) **Given Above tables, perform the following queries**

1. Create a view consisting of branch names and the names of customers who have either an account or a loan at that branch. Assume that view to be called *all-customer*.
2. Create a view gives for each branch the sum of the amounts of all the loans at the branch.
3. Using the view all-customer, we can find *all customers* of the *Perryridge* branch.
4. Write a Query for below Relational algebraic notation :

$$\Pi_{customer\text{-}name}\ (borrower) \cup \Pi_{customer\text{-}name}\ (depositor)$$

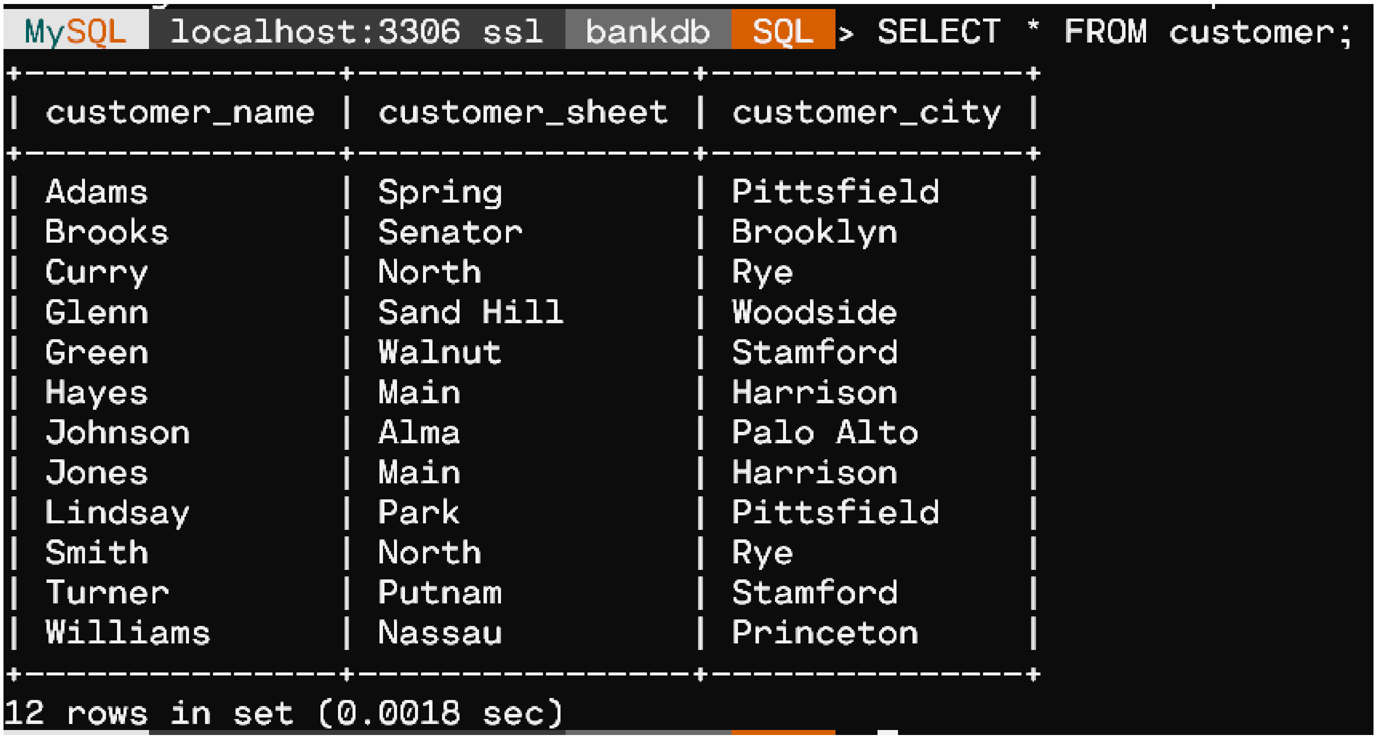
ER Diagram :



```
CREATE DATABASE bankDB;
USE bankDB;

CREATE TABLE branch
(
    branch_name varchar(50) PRIMARY KEY,
    branch_city varchar(50) NOT NULL,
    assets bigint
);
```

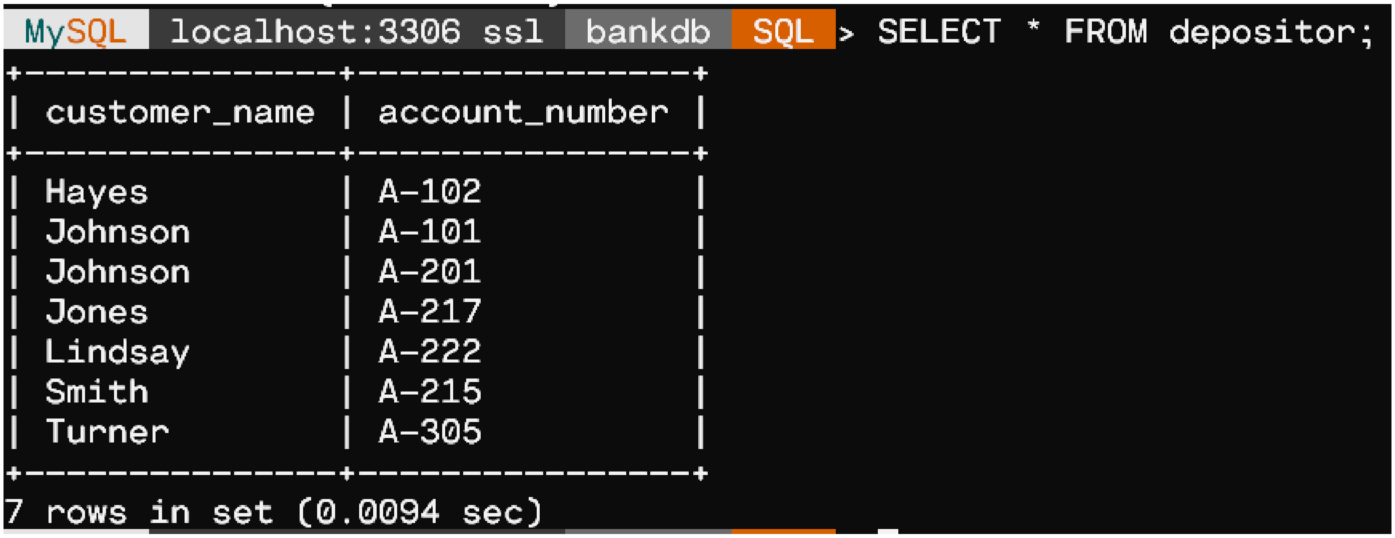
```
CREATE TABLE account
(
    account_number varchar(50) PRIMARY KEY,
    branch_name varchar(50) NOT NULL,
    balance INT NOT NULL,
    foreign key(branch_name) references branch(branch_name)
);

CREATE TABLE loan
(
    loan_number varchar(50) PRIMARY KEY,
    branch_name varchar(50) NOT NULL,
    amount INT NOT NULL,
    foreign key(branch_name) references branch(branch_name)
);

CREATE TABLE customer
(
    customer_name varchar(50) PRIMARY KEY,
    customer_sheet varchar(50) NOT NULL,
    customer_city varchar(50) NOT NULL
);

CREATE TABLE depositor
(
    customer_name varchar(50) NOT NULL,
    account_number varchar(50) NOT NULL,
    foreign key(customer_name) references
    customer(customer_name),
    foreign key(account_number) references account(account_number)
);

CREATE TABLE borrower
(
    customer_name varchar(50) NOT NULL,
    loan_number varchar(50) NOT NULL,
    foreign key(customer_name) references
    customer(customer_name),
    foreign key(loan_number) references loan(loan_number)
);

insert into branch
values
('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);

insert into account
values
('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);

insert into loan
values
('L-11','Round Hill',900),
('L-14','Downtown',1500),
('L-15','Perryridge',1500),
('L-16','Perryridge',1300),
('L-17','Downtown',1000),
('L-23','Redwood',2000),
('L-93','Mianus',500);

insert into customer
values
('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');

insert into depositor
values
('Hayes','A-102'),
('Johnson','A-101'),
('Johnson','A-201'),
('Jones','A-217'),
('Lindsay','A-222'),
('Smith','A-215'),
('Turner','A-305');

insert into borrower
values
('Adams','L-16'),
('Curry','L-93'),
('Hayes','L-15'),
('Johnson','L-14'),
('Jones','L-17'),
('Smith','L-11'),
('Smith','L-23'),
('Williams','L-17');

/*---------------------------- Querys----------------------------- */
```

```
SELECT * FROM customer;
```



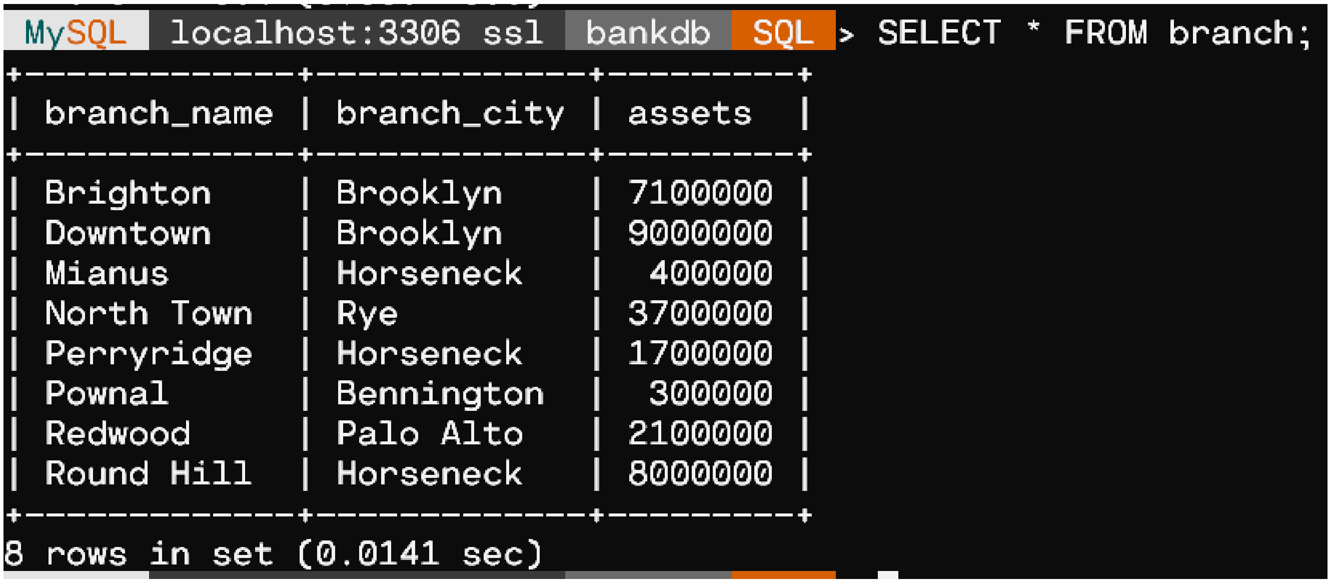
```
 MySQL  localhost:3306 ssl  bankdb  SQL > SELECT * FROM customer;
+---------------+----------------+---------------+
| customer_name | customer_sheet | customer_city |
+---------------+----------------+---------------+
| Adams         | Spring         | Pittsfield    |
| Brooks        | Senator        | 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     |
+---------------+----------------+---------------+
12 rows in set (0.0018 sec)
```

```
SELECT * FROM depositor;
```



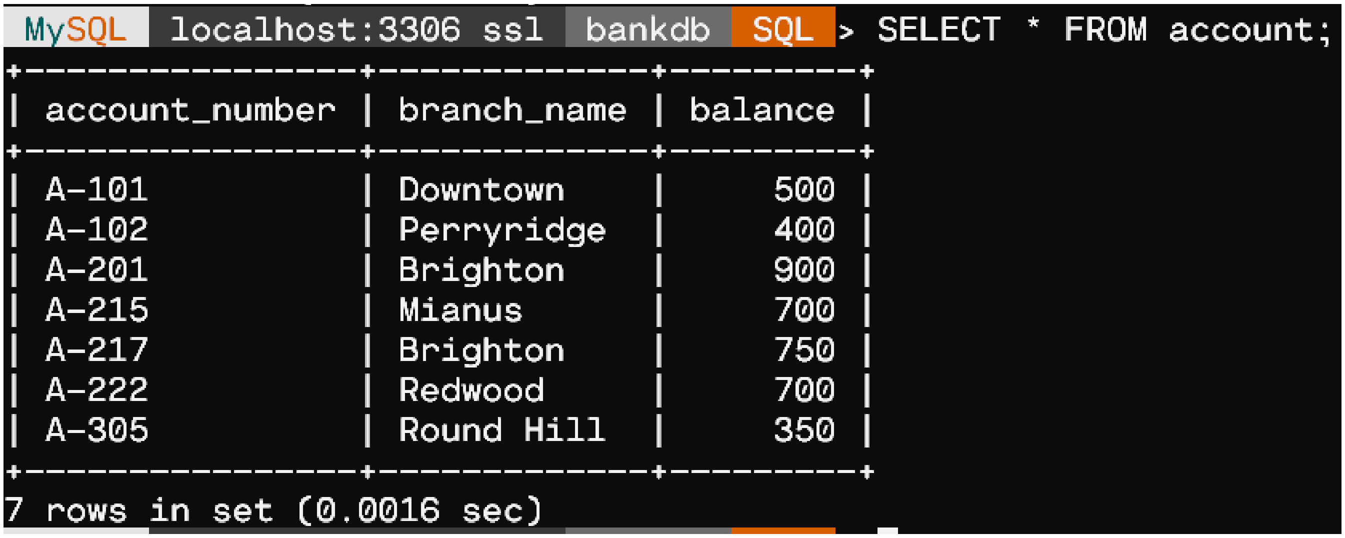
```
 MySQL  localhost:3306 ssl  bankdb  SQL > SELECT * FROM depositor;
+---------------+----------------+
| customer_name | account_number |
+---------------+----------------+
| Hayes         | A-102          |
| Johnson       | A-101          |
| Johnson       | A-201          |
| Jones         | A-217          |
| Lindsay       | A-222          |
| Smith         | A-215          |
| Turner        | A-305          |
+---------------+----------------+
7 rows in set (0.0094 sec)
```

```
SELECT * FROM branch;
```



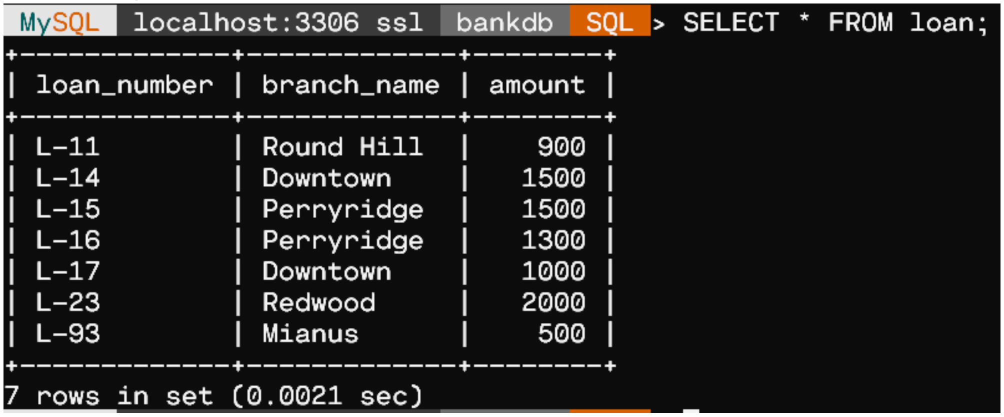
```
 MySQL  localhost:3306 ssl  bankdb  SQL > SELECT * FROM branch;
+-------------+-------------+---------+
| branch_name | branch_city | assets  |
+-------------+-------------+---------+
| Brighton    | Brooklyn    | 7100000 |
| Downtown    | Brooklyn    | 9000000 |
| Mianus      | Horseneck   |  400000 |
| North Town  | Rye         | 3700000 |
| Perryridge  | Horseneck   | 1700000 |
| Pownal      | Bennington  |  300000 |
| Redwood     | Palo Alto   | 2100000 |
| Round Hill  | Horseneck   | 8000000 |
+-------------+-------------+---------+
8 rows in set (0.0141 sec)
```

```
SELECT * FROM account;
```



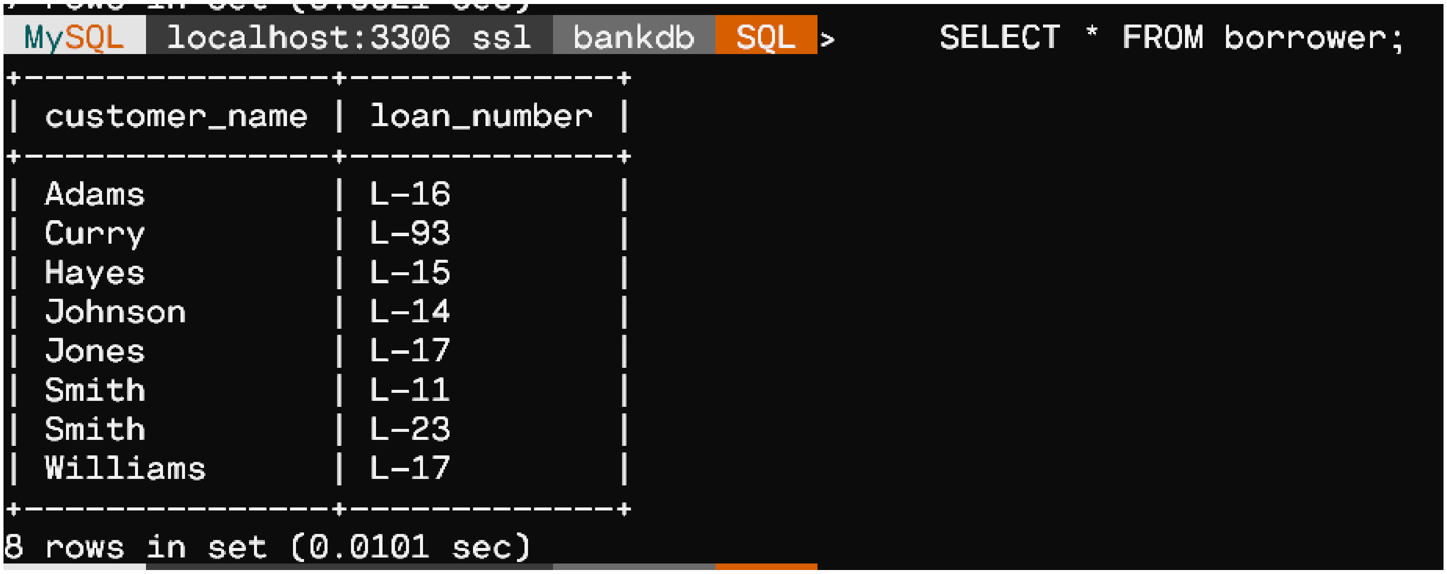
```
 MySQL  localhost:3306 ssl  bankdb  SQL > SELECT * FROM account;
+----------------+-------------+---------+
| account_number | branch_name | balance |
+----------------+-------------+---------+
| A-101          | Downtown    |     500 |
| A-102          | Perryridge  |     400 |
| A-201          | Brighton    |     900 |
| A-215          | Mianus      |     700 |
| A-217          | Brighton    |     750 |
| A-222          | Redwood     |     700 |
| A-305          | Round Hill  |     350 |
+----------------+-------------+---------+
7 rows in set (0.0016 sec)
```

```
SELECT * FROM loan;
```



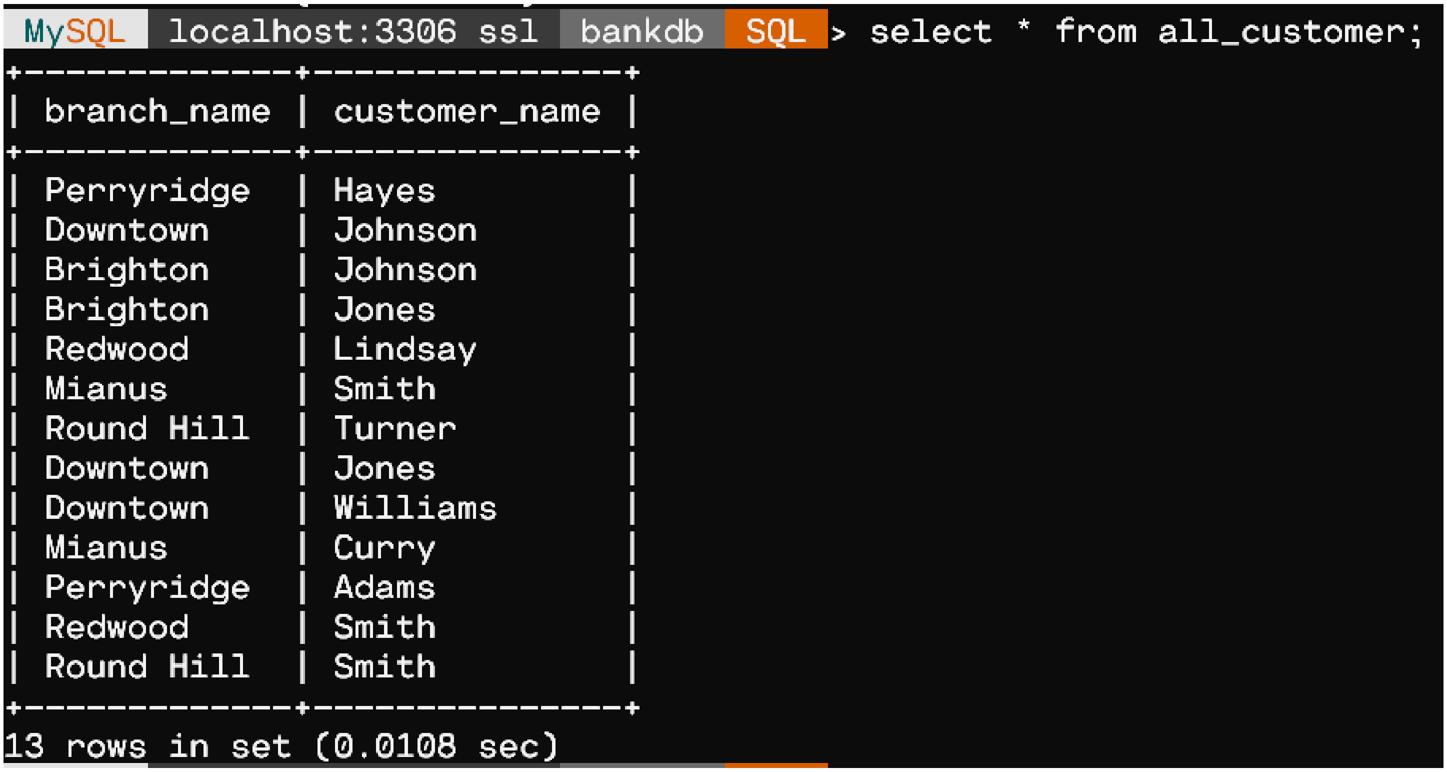
```
 MySQL  localhost:3306 ssl  bankdb  SQL > SELECT * FROM loan;
+-------------+-------------+--------+
| loan_number | branch_name | amount |
+-------------+-------------+--------+
| L-11        | Round Hill  |    900 |
| L-14        | Downtown    |   1500 |
| L-15        | Perryridge  |   1500 |
| L-16        | Perryridge  |   1300 |
| L-17        | Downtown    |   1000 |
| L-23        | Redwood     |   2000 |
| L-93        | Mianus      |    500 |
+-------------+-------------+--------+
7 rows in set (0.0021 sec)
```

```
SELECT * FROM borrower;
```



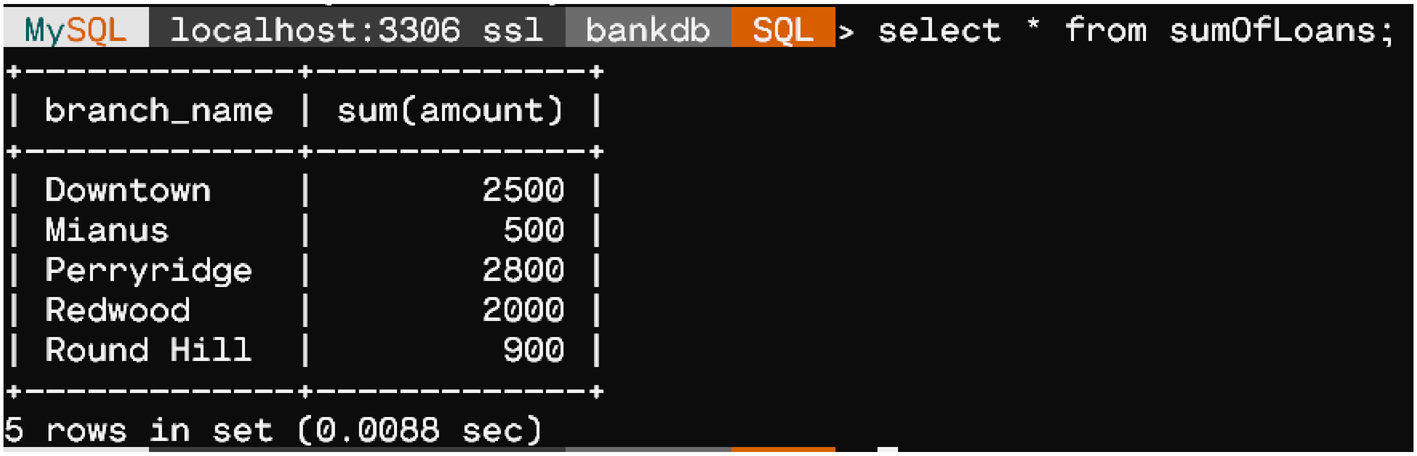
```
MySQL  localhost:3306 ssl  bankdb  SQL >     SELECT * FROM borrower;
+---------------+-------------+
| customer_name | loan_number |
+---------------+-------------+
| Adams         | L-16        |
| Curry         | L-93        |
| Hayes         | L-15        |
| Johnson       | L-14        |
| Jones         | L-17        |
| Smith         | L-11        |
| Smith         | L-23        |
| Williams      | L-17        |
+---------------+-------------+
8 rows in set (0.0101 sec)
```

```
/*1.Create a view consisting of branch names and the names of custome
rs who have either an account or a loan at that branch.
 Assume that view to be called all-customer.*/

    create view all_customer as
    select branch_name , customer_name
    from depositor , account
    where depositor.account_number = account.account_number
    union
    (select branch_name , customer_name
    from borrower , loan
    where borrower.loan_number = loan.loan_number);




 -- For Output
    select * from all_customer;
```



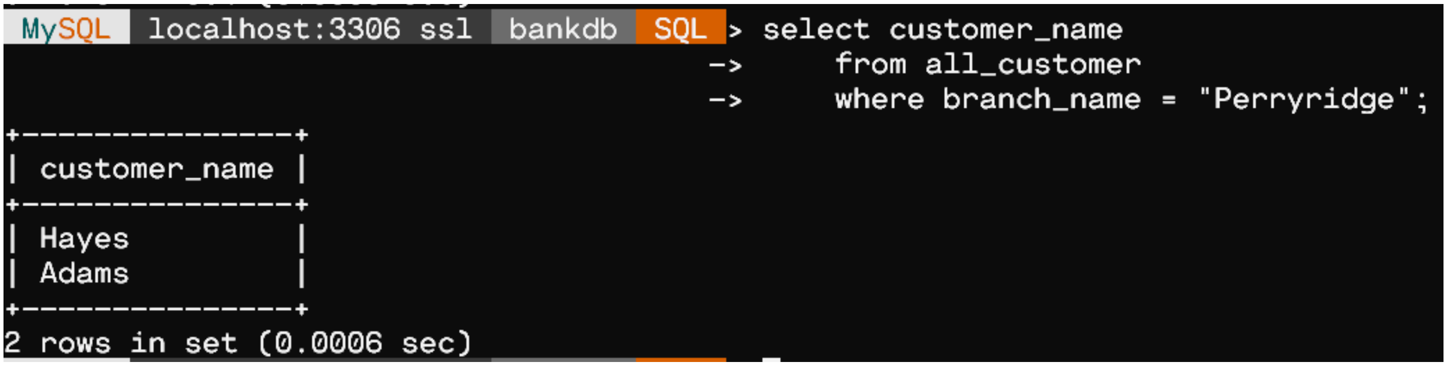
```
MySQL  localhost:3306 ssl  bankdb  SQL > select * from all_customer;
+-------------+---------------+
| branch_name | customer_name |
+-------------+---------------+
| Perryridge  | Hayes         |
| Downtown    | Johnson       |
| Brighton    | Johnson       |
| Brighton    | Jones         |
| Redwood     | Lindsay       |
| Mianus      | Smith         |
| Round Hill  | Turner        |
| Downtown    | Jones         |
| Downtown    | Williams      |
| Mianus      | Curry         |
| Perryridge  | Adams         |
| Redwood     | Smith         |
| Round Hill  | Smith         |
+-------------+---------------+
13 rows in set (0.0108 sec)
```

```
/*2.Create a view gives for each branch the sum of the amounts of all
 the loans at the branch.*/

    create view sumOfLoans as
    select branch_name , sum(amount)
    from loan
    group by branch_name;
    -- For Output
    select * from sumOfLoans;
```



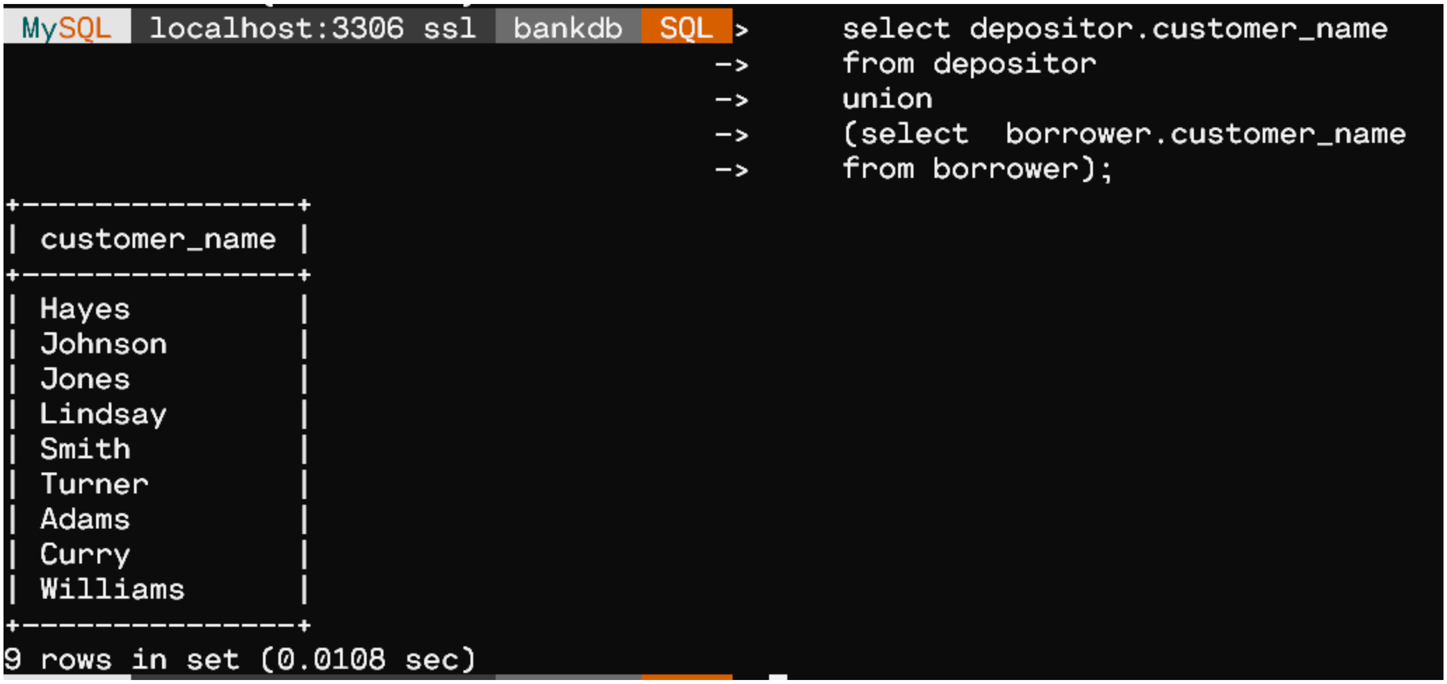
```
MySQL  localhost:3306 ssl  bankdb  SQL > select * from sumOfLoans;
+-------------+-------------+
| branch_name | sum(amount) |
+-------------+-------------+
| Downtown    |        2500 |
| Mianus      |         500 |
| Perryridge  |        2800 |
| Redwood     |        2000 |
| Round Hill  |         900 |
+-------------+-------------+
5 rows in set (0.0088 sec)
```

```
/*3.Using the view all-
customer, we can find all customers of the Perryridge branch.*/

    select customer_name
    from all_customer
    where branch_name = "Perryridge";
```



```
 MySQL  localhost:3306 ssl  bankdb  SQL > select customer_name
                                      ->     from all_customer
                                      ->     where branch_name = "Perryridge";
+---------------+
| customer_name |
+---------------+
| Hayes         |
| Adams         |
+---------------+
2 rows in set (0.0006 sec)
```

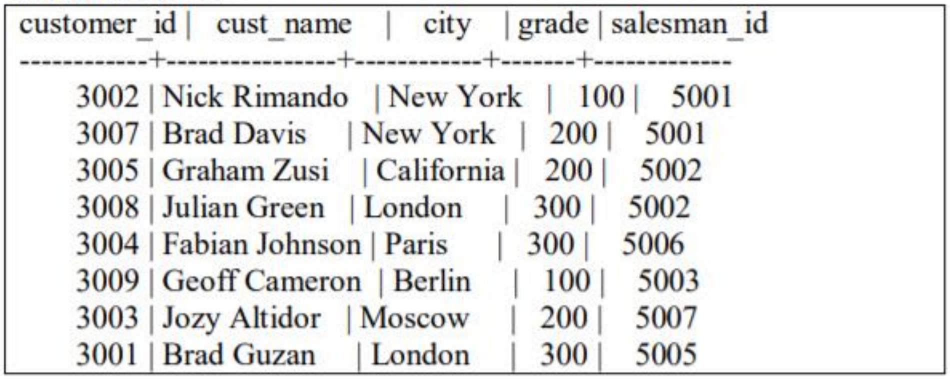
```
/*4.Write a Query for below Relational algebraic notation :*/

    select depositor.customer_name
    from depositor
    union
    (select  borrower.customer_name
    from borrower);
```



```
 MySQL  localhost:3306 ssl  bankdb  SQL >     select depositor.customer_name
                                      ->     from depositor
                                      ->     union
                                      ->     (select  borrower.customer_name
                                      ->     from borrower);
+---------------+
| customer_name |
+---------------+
| Hayes         |
| Johnson       |
| Jones         |
| Lindsay       |
| Smith         |
| Turner        |
| Adams         |
| Curry         |
| Williams      |
+---------------+
9 rows in set (0.0108 sec)
```

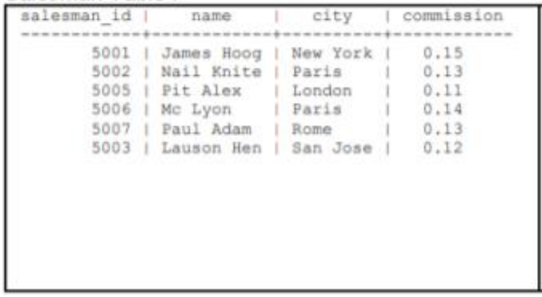
PROBLEM 2

Customer Table :



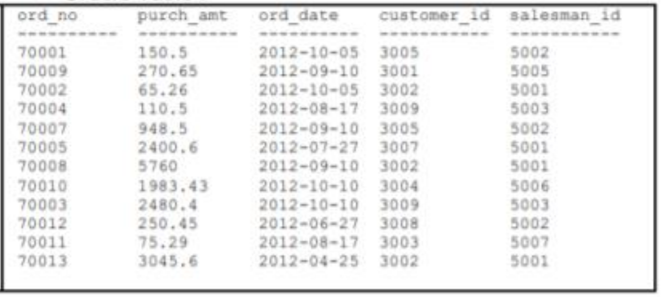
customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3007	Brad Davis	New York	200	5001
3005	Graham Zusi	California	200	5002
3008	Julian Green	London	300	5002
3004	Fabian Johnson	Paris	300	5006
3009	Geoff Cameron	Berlin	100	5003
3003	Jozy Altidor	Moscow	200	5007
3001	Brad Guzan	London	300	5005

Salesman Table :



salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5007	Paul Adam	Rome	0.13
5003	Lauson Hen	San Jose	0.12

Orders Table :



ord_no	purch_amt	ord_date	customer_id	salesman_id
70001	150.5	2012-10-05	3005	5002
70009	270.65	2012-09-10	3001	5005
70002	65.26	2012-10-05	3002	5001
70004	110.5	2012-08-17	3009	5003
70007	948.5	2012-09-10	3005	5002
70005	2400.6	2012-07-27	3007	5001
70008	5760	2012-09-10	3002	5001
70010	1983.43	2012-10-10	3004	5006
70003	2480.4	2012-10-10	3009	5003
70012	250.45	2012-06-27	3008	5002
70011	75.29	2012-08-17	3003	5007
70013	3045.6	2012-04-25	3002	5001

B) Given three tables, perform the following queries:

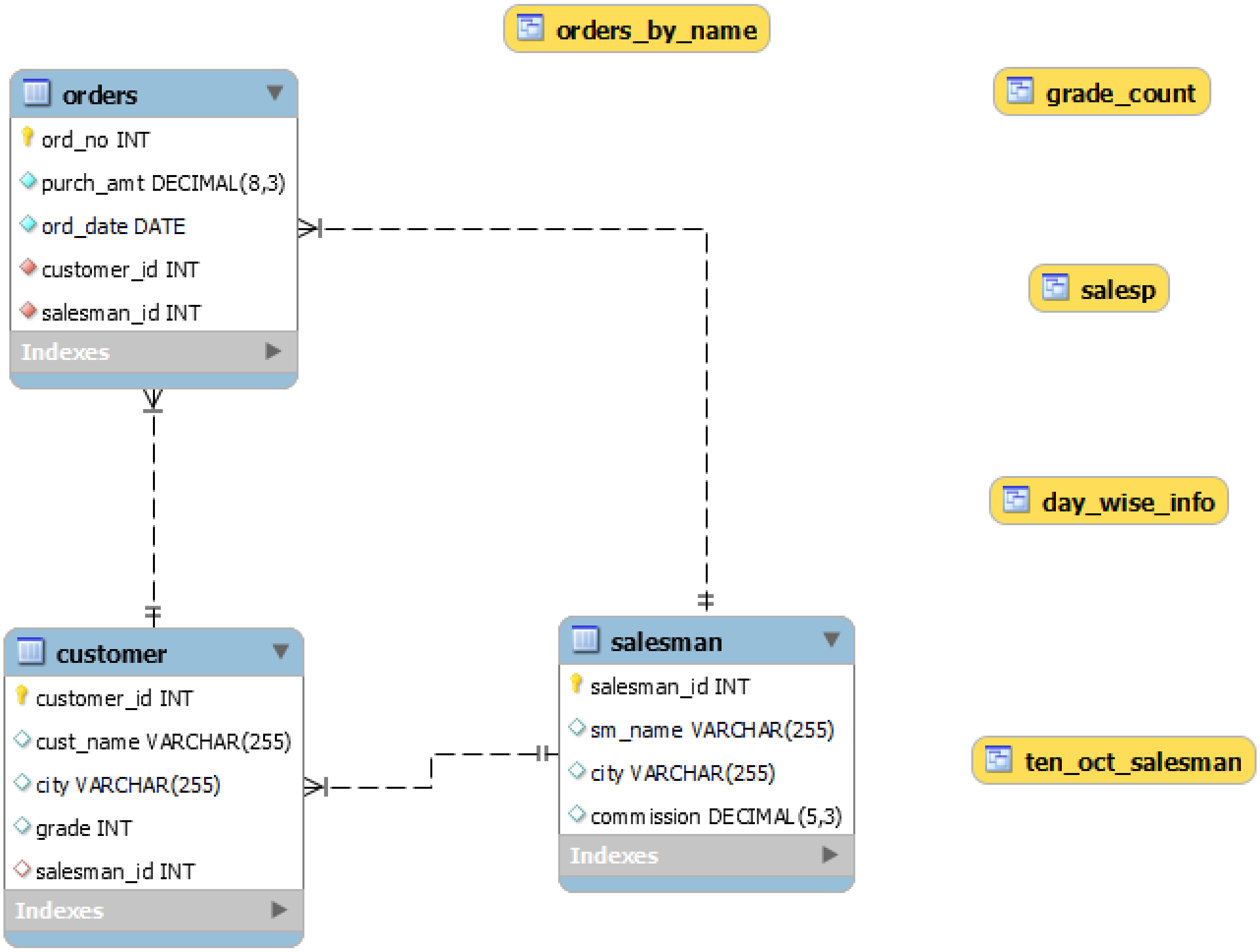
1 From the table, create a view to count the number of customers in each grade.

2 From the following table, create a view to count the number of unique customer, compute average and total purchase amount of customer orders by each date.

3 create a view to get the salesperson and customer by name. Return order name, purchase amount, salesperson ID, name, customer name.

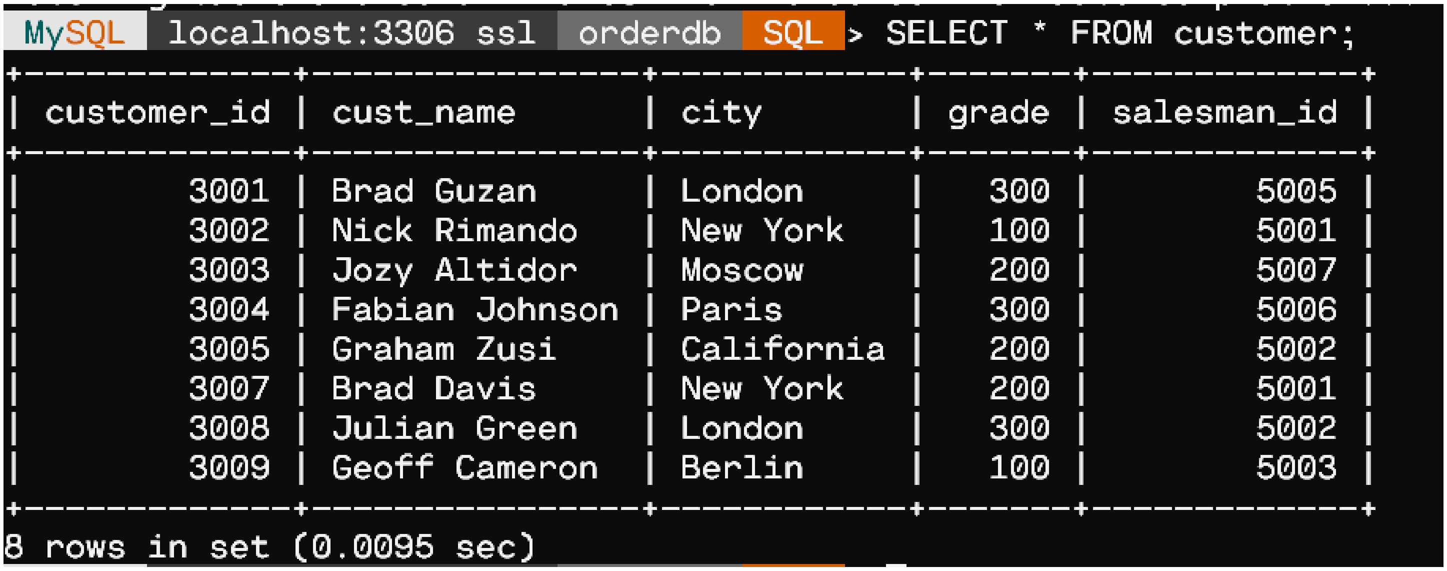
4 create a view to find the salespersons who issued orders on October 10th, 2012. Return all the fields of salesperson.

ER Diagram :



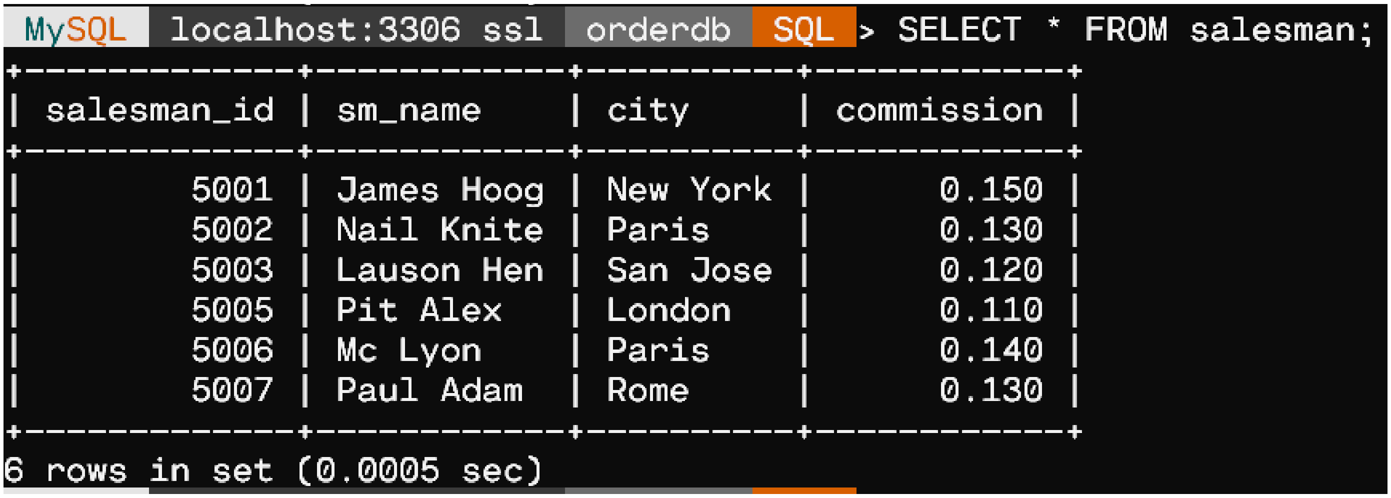
```
CREATE DATABASE orderDB;
USE orderDB;

CREATE TABLE salesman(
    salesman_id int NOT NULL,
    sm_name varchar(255),
    city varchar(255),
    commission DECIMAL(5,3),
    UNIQUE (salesman_id),
    PRIMARY KEY (salesman_id)
);
```

```
INSERT INTO salesman (salesman_id, sm_name, city, commission)
VALUES
        ('5001', 'James Hoog', 'New York', '0.15'),
        ('5002', 'Nail Knite', 'Paris', '0.13'),
        ('5005', 'Pit Alex', 'London', '0.11'),
        ('5006', 'Mc Lyon', 'Paris', '0.14'),
        ('5007', 'Paul Adam', 'Rome', '0.13'),
        ('5003', 'Lauson Hen', 'San Jose', '0.12');


CREATE TABLE customer(
    customer_id int NOT NULL,
    cust_name varchar(255),
    city varchar(255),
    grade int,
    salesman_id int,
    UNIQUE (customer_id),
    PRIMARY KEY (customer_id),
    FOREIGN KEY (salesman_id) REFERENCES salesman(salesman_id)
);

INSERT INTO customer(customer_id, cust_name, city,
                    grade, salesman_id)
 VALUES
      (3002 , 'Nick Rimando' , 'New York' , 100 , 5001),
      (3007 , 'Brad Davis' , 'New York' , 200 , 5001),
      (3005 , 'Graham Zusi' , 'California' , 200 , 5002),
      (3008 , 'Julian Green' , 'London' , 300 , 5002),
      (3004 , 'Fabian Johnson' , 'Paris' , 300 , 5006),
      (3009 , 'Geoff Cameron' , 'Berlin' , 100 , 5003),
      (3003 , 'Jozy Altidor' , 'Moscow' , 200 , 5007),
      (3001 , 'Brad Guzan' , 'London' , 300 , 5005);

CREATE TABLE orders(
    ord_no INT NOT NULL,
    purch_amt DECIMAL(8,3) NOT NULL,
    ord_date DATE NOT NULL,
    customer_id INT NOT NULL,
    salesman_id INT NOT NULL,
    PRIMARY KEY (ord_no),
    FOREIGN KEY (customer_id) REFERENCES customer(customer_id),
    FOREIGN KEY (salesman_id) REFERENCES salesman(salesman_id)
    );

INSERT INTO orders (ord_no, purch_amt, ord_date, customer_id, salesma
n_id)
VALUES
        ('70001', '150.5', '2012-10-05', '3005', '5002'),
        ('70009', '270.65', '2012-09-10', '3001', '5005'),
        ('70002', '65.26', '2012-10-05', '3002', '5001'),
        ('70004', '110.5', '2012-08-17', '3009', '5003'),
        ('70007', '948.5', '2012-09-10', '3005', '5002'),
        ('70005', '2400.6', '2012-07-27', '3007', '5001'),
        ('70008', '5760', '2012-09-10', '3002', '5001'),
        ('70010', '1983.43', '2012-10-10', '3004', '5006'),
        ('70003', '2480.4', '2012-10-10', '3009', '5003'),
        ('70012', '250.45', '2012-06-27', '3008', '5002'),
        ('70011', '75.29', '2012-08-17', '3003', '5007'),
        ('70013', '3045.6', '2012-04-25', '3002', '5001');


      /*------------------------Queries--------------------------*/

SELECT * FROM customer;
```



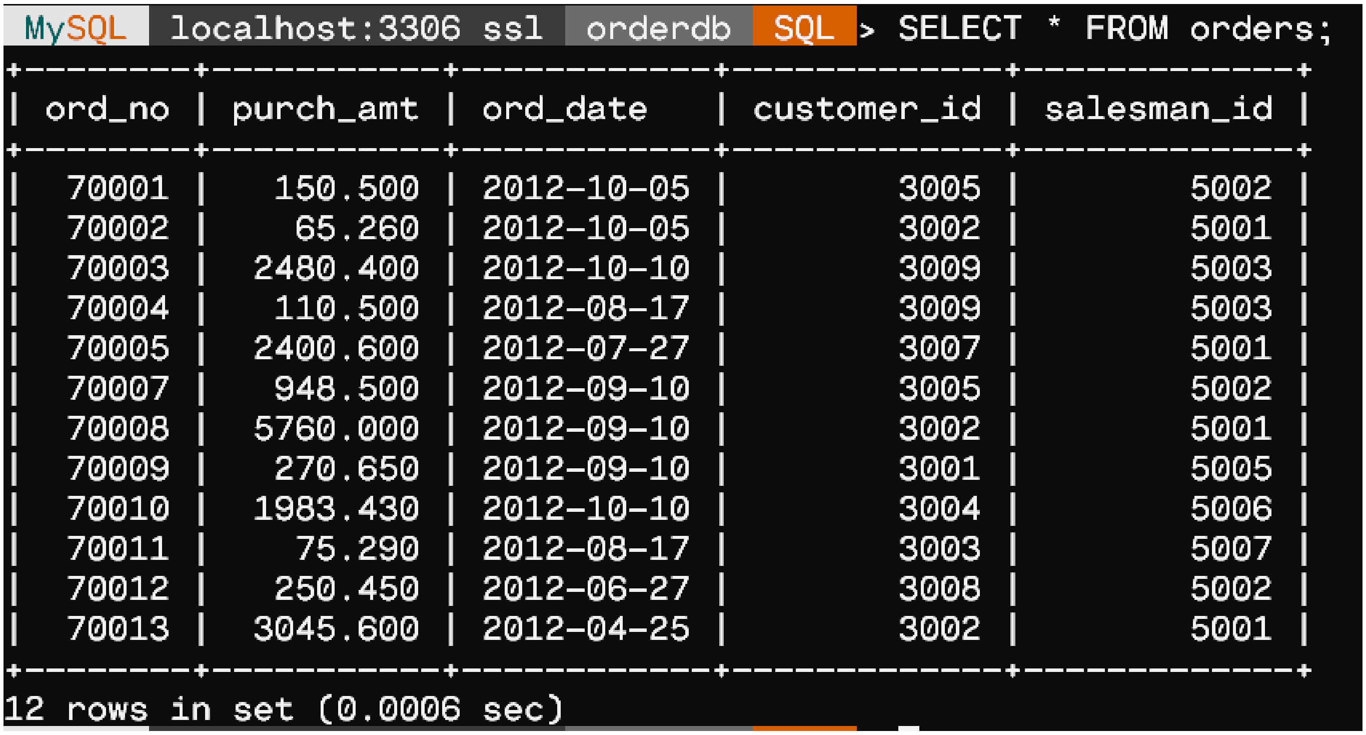
```
 MySQL  localhost:3306 ssl  orderdb  SQL > SELECT * FROM customer;
+-------------+----------------+------------+-------+-------------+
| customer_id | cust_name      | city       | grade | salesman_id |
+-------------+----------------+------------+-------+-------------+
|        3001 | Brad Guzan     | London     |   300 |        5005 |
|        3002 | Nick Rimando   | New York   |   100 |        5001 |
|        3003 | Jozy Altidor   | Moscow     |   200 |        5007 |
|        3004 | Fabian Johnson | Paris      |   300 |        5006 |
|        3005 | Graham Zusi    | California |   200 |        5002 |
|        3007 | Brad Davis     | New York   |   200 |        5001 |
|        3008 | Julian Green   | London     |   300 |        5002 |
|        3009 | Geoff Cameron  | Berlin     |   100 |        5003 |
+-------------+----------------+------------+-------+-------------+
8 rows in set (0.0095 sec)
```

```
SELECT * FROM salesman;
```



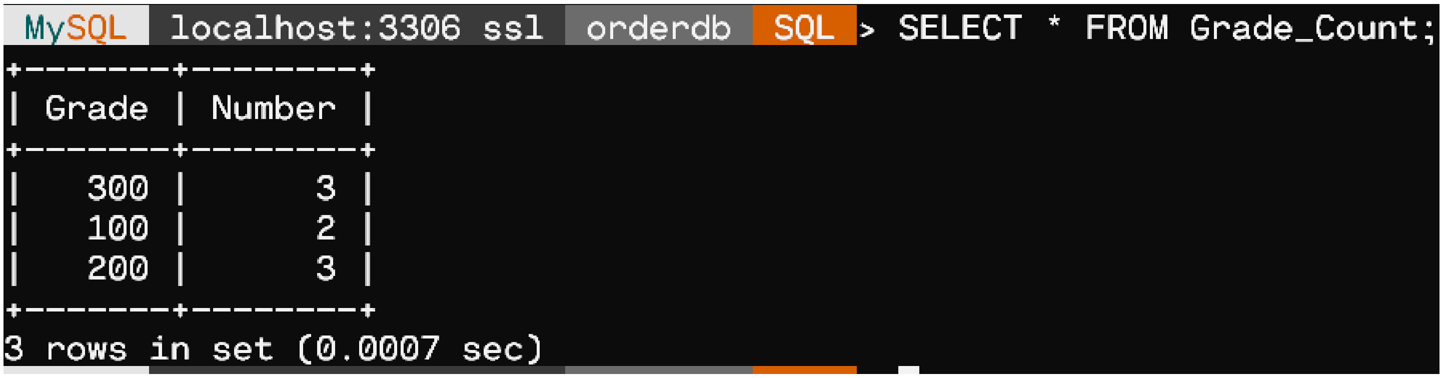
```
MySQL  localhost:3306 ssl  orderdb  SQL > SELECT * FROM salesman;
+-------------+------------+----------+------------+
| salesman_id | sm_name    | city     | commission |
+-------------+------------+----------+------------+
|        5001 | James Hoog | New York |      0.150 |
|        5002 | Nail Knite | Paris    |      0.130 |
|        5003 | Lauson Hen | San Jose |      0.120 |
|        5005 | Pit Alex   | London   |      0.110 |
|        5006 | Mc Lyon    | Paris    |      0.140 |
|        5007 | Paul Adam  | Rome     |      0.130 |
+-------------+------------+----------+------------+
6 rows in set (0.0005 sec)
```

```
SELECT * FROM orders;
```



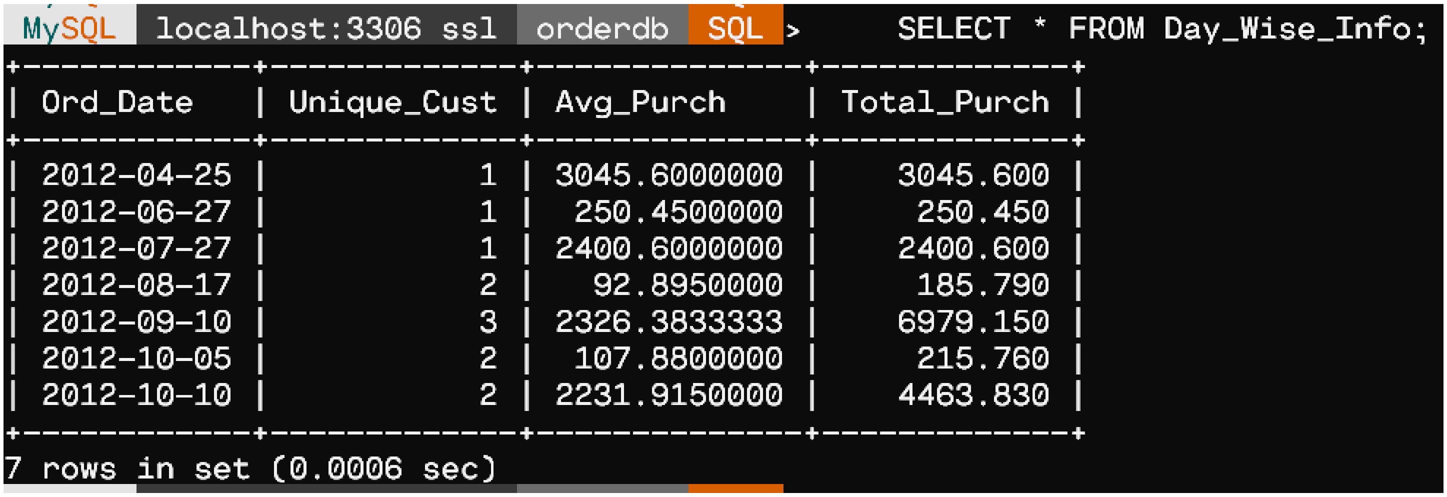
```
MySQL  localhost:3306 ssl  orderdb  SQL > SELECT * FROM orders;
+--------+-----------+------------+-------------+-------------+
| ord_no | purch_amt | ord_date   | customer_id | salesman_id |
+--------+-----------+------------+-------------+-------------+
|  70001 |   150.500 | 2012-10-05 |        3005 |        5002 |
|  70002 |    65.260 | 2012-10-05 |        3002 |        5001 |
|  70003 |  2480.400 | 2012-10-10 |        3009 |        5003 |
|  70004 |   110.500 | 2012-08-17 |        3009 |        5003 |
|  70005 |  2400.600 | 2012-07-27 |        3007 |        5001 |
|  70007 |   948.500 | 2012-09-10 |        3005 |        5002 |
|  70008 |  5760.000 | 2012-09-10 |        3002 |        5001 |
|  70009 |   270.650 | 2012-09-10 |        3001 |        5005 |
|  70010 |  1983.430 | 2012-10-10 |        3004 |        5006 |
|  70011 |    75.290 | 2012-08-17 |        3003 |        5007 |
|  70012 |   250.450 | 2012-06-27 |        3008 |        5002 |
|  70013 |  3045.600 | 2012-04-25 |        3002 |        5001 |
+--------+-----------+------------+-------------+-------------+
12 rows in set (0.0006 sec)
```

```
-
- 1 From the table, create a view to count the number of customers in
 each grade.

    CREATE VIEW Grade_Count (Grade, Number)
    AS SELECT grade, COUNT(*)
    FROM customer
    GROUP BY grade;

    SELECT * FROM Grade_Count;
```



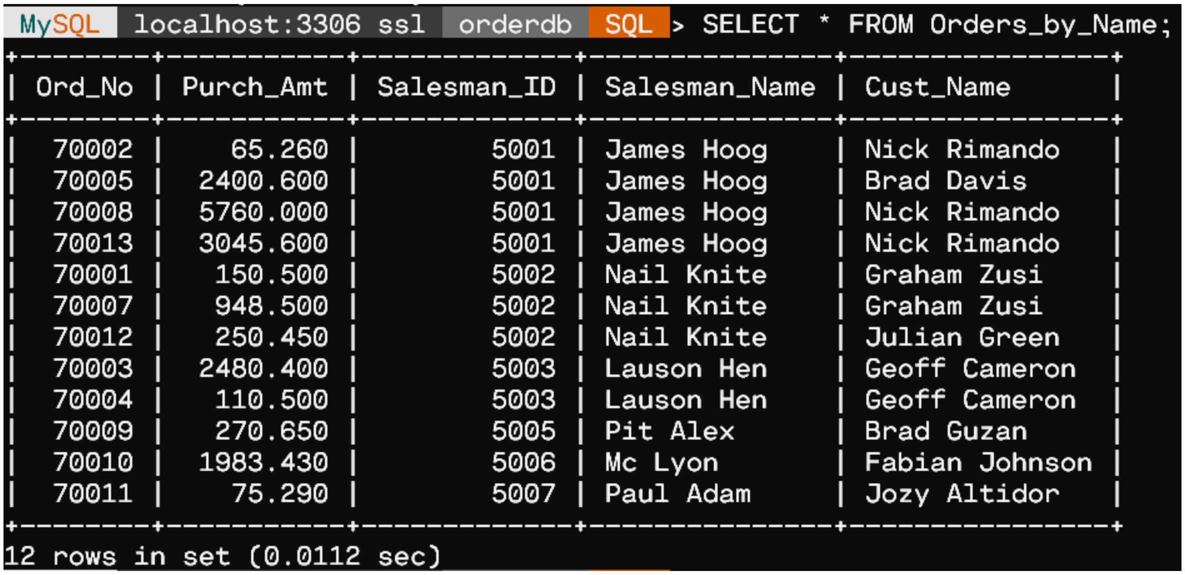
```
MySQL  localhost:3306 ssl  orderdb  SQL > SELECT * FROM Grade_Count;
+-------+--------+
| Grade | Number |
+-------+--------+
|   300 |      3 |
|   100 |      2 |
|   200 |      3 |
+-------+--------+
3 rows in set (0.0007 sec)
```

```
/* 2 From the following table, create a view to count the number of unique
customer, compute average and total purchase amount of customer orders by
each date. */

    CREATE VIEW Day_Wise_Info (Ord_Date, Unique_Cust, Avg_Purch, Total_Purch)
    AS SELECT ord_date, COUNT(DISTINCT customer_id),
    AVG(purch_amt), SUM(purch_amt)
    FROM orders
    GROUP BY ord_date;
    SELECT * FROM Day_Wise_Info;
```



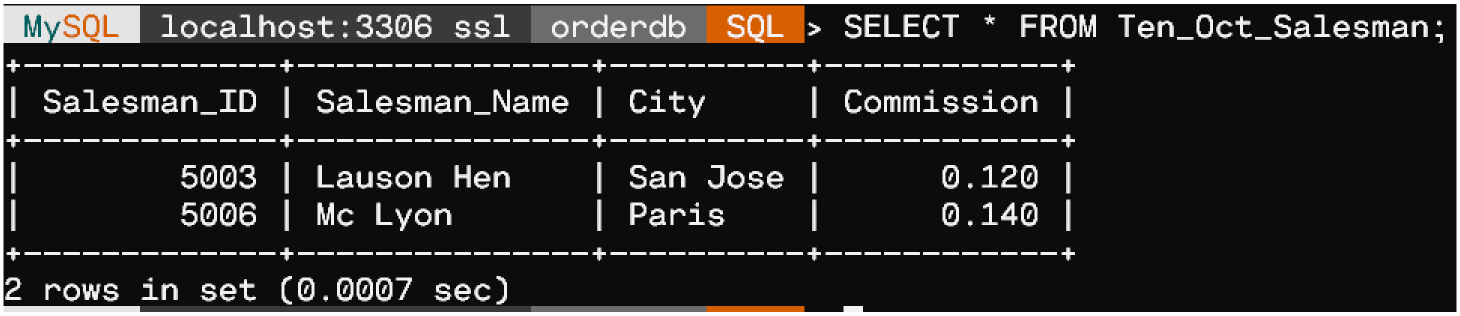
```
 MySQL  localhost:3306 ssl  orderdb  SQL >     SELECT * FROM Day_Wise_Info;
+------------+-------------+---------------+-------------+
| Ord_Date   | Unique_Cust | Avg_Purch     | Total_Purch |
+------------+-------------+---------------+-------------+
| 2012-04-25 |           1 | 3045.6000000  |    3045.600 |
| 2012-06-27 |           1 |  250.4500000  |     250.450 |
| 2012-07-27 |           1 | 2400.6000000  |    2400.600 |
| 2012-08-17 |           2 |   92.8950000  |     185.790 |
| 2012-09-10 |           3 | 2326.3833333  |    6979.150 |
| 2012-10-05 |           2 |  107.8800000  |     215.760 |
| 2012-10-10 |           2 | 2231.9150000  |    4463.830 |
+------------+-------------+---------------+-------------+
7 rows in set (0.0006 sec)
```

```
/* 3 create a view to get the salesperson and customer by name. Return order
name, purchase amount, salesperson ID, name, customer name. */
    CREATE VIEW Orders_by_Name (Ord_No,Purch_Amt,Salesman_ID, Salesman_Name, Cust_Name)
    AS SELECT ord_no, purch_amt, a.salesman_id, sm_name, cust_name
    FROM orders a, customer b, salesman c
    WHERE a.customer_id = b.customer_id
    AND a.salesman_id = c.salesman_id;

    SELECT * FROM Orders_by_Name;
```



```
 MySQL  localhost:3306 ssl  orderdb  SQL > SELECT * FROM Orders_by_Name;
+--------+----------+-------------+---------------+----------------+
| Ord_No | Purch_Amt | Salesman_ID | Salesman_Name | Cust_Name      |
+--------+----------+-------------+---------------+----------------+
|  70002 |   65.260 |        5001 | James Hoog    | Nick Rimando   |
|  70005 | 2400.600 |        5001 | James Hoog    | Brad Davis     |
|  70008 | 5760.000 |        5001 | James Hoog    | Nick Rimando   |
|  70013 | 3045.600 |        5001 | James Hoog    | Nick Rimando   |
|  70001 |  150.500 |        5002 | Nail Knite    | Graham Zusi    |
|  70007 |  948.500 |        5002 | Nail Knite    | Graham Zusi    |
|  70012 |  250.450 |        5002 | Nail Knite    | Julian Green   |
|  70003 | 2480.400 |        5003 | Lauson Hen    | Geoff Cameron  |
|  70004 |  110.500 |        5003 | Lauson Hen    | Geoff Cameron  |
|  70009 |  270.650 |        5005 | Pit Alex      | Brad Guzan     |
|  70010 | 1983.430 |        5006 | Mc Lyon       | Fabian Johnson |
|  70011 |   75.290 |        5007 | Paul Adam     | Jozy Altidor   |
+--------+----------+-------------+---------------+----------------+
12 rows in set (0.0112 sec)
```

```
/* 4 create a view to find the salespersons who issued orders on Octo
ber 10th,
2012. Return all the fields of salesperson. */

    CREATE VIEW Ten_Oct_Salesman (Salesman_ID, Salesman_Name, City, C
ommission)
    AS SELECT *
    FROM salesman
    WHERE salesman_id IN
        (SELECT salesman_id
             FROM orders
             WHERE ord_date = '2012-10-10');

    SELECT * FROM Ten_Oct_Salesman;
```



```
MySQL  localhost:3306 ssl  orderdb  SQL > SELECT * FROM Ten_Oct_Salesman;
+-------------+---------------+----------+------------+
| Salesman_ID | Salesman_Name | City     | Commission |
+-------------+---------------+----------+------------+
|        5003 | Lauson Hen    | San Jose |      0.120 |
|        5006 | Mc Lyon       | Paris    |      0.140 |
+-------------+---------------+----------+------------+
2 rows in set (0.0007 sec)
```