Heaven's Light is Our Guide

Rajshahi University of Engineering and Technology

Course Code
ECE 2216

Course Title
Database Systems Sessional

Experiment Date: November 5, 2023,
Submission Date: November 12, 2023

Lab Report 4: Creating a database and doing operations on it using MySQL

Submitted to	**Submitted by**
Md. Robiul Islam	Md. Tajim An Noor
Assistant Professor	Roll: 2010025
Dept of ECE, Ruet	

Contents

Basic MySQL Operations

Md Tajim An Noor

1 Tools Used

- MySQL
- VS Code - as an IDE to use SQL
- MacTeX -LaTeXcompiler
- VS Code with LaTeX workshop extension as a text editor

2 Process

The task is to create two tables in a database and then do some operations on the database using sub-query, Group By operations. The first table, Order, has the columns ord_date, purch_amt, ord_date, customer_id, salesman_id. The second one, named Customer, has the columns customer_id, cust_name, city, grade, salesman_id.

2.1 SQL Codes:

Creating both tables and inserting data.

Code:

```
—
```

```
--creating the 1st table and adding info
CREATE TABLE ProductOrderDetailsLR3.Order(
    ord_no int Auto_Increment Primary Key,
    purch_amt decimal(6, 2),
    ord_date Date,
    customer_id int,
    salesman_id int -- )
    INSERT INTO ProductOrderDetailsLR3.Order (
        -- ord_no,
        purch_amt,
        ord_date,
        customer_id,
        salesman_id2
    )
VALUES
    (
        70001,
        '150.5',
        '2012-10-05',
        3005,
```

```
            5002
        );

(2480.4, '2012-10-10', 3009, 5003),
(110.5, '2012-08-17', 3009, 5003),
(2400.6, '2012-07-27', 3007, 5001);

(70007, 948.5, '2012-09-10', 3005, 5002);

(5760, '2012-09-10', 3002, 5001),
(270.65, '2012-09-10', 3001, 5005),
(1983.43, '2012-10-10', 3004, 5006),
(75.29, '2012-08-17', 3003, 5007),
(250.45, '2012-06-27', 3008, 5002),
(3045.6, '2012-04-25', 3002, 5001);

-- creating the 2nd table and adding info

CREATE TABLE ProductOrderDetailsLR3.Customer(
    customer_id int Auto_Increment Primary Key,
    cust_name varchar(40),
    city varchar(40),
    grade int,
    ord_date Date,
    salesman_id int
-- )
ALTER TABLE ProductOrderDetailsLR3.Customer drop column ord_date;
INSERT INTO ProductOrderDetailsLR3.Customer (cust_name, city, grade, salesman_id)
VALUES
(
    3001,
    'Brad Guzan',
    'London',
    null,
    5005
);
(
    'Nick Romando',
    'New York',
    100,
    5001
),
(
    'Jozy Altidor',
    'Moscow',
    200,
    5007
).
-- To save space, some entries are shown.
```

2.1.1 Calculate total purchase amount of all orders. Return total purchase amount.

Code:

```
SELECT
    SUM(purch_amt) as "Total Purchase Amount"
```

```
FROM
    ProductOrderDetailsLR3.Order
```

2.1.2 Count the number of unique salesperson. Return number of salesperson.

Code:

```
SELECT
    salesman_id as "Salesperson",
    Count(*)
from
    ProductOrderDetailsLR3.Order
Group by
    salesman_id
```

2.1.3 Find the highest grade of customers in each city. Return city, maximum grade.

Code:

```
SELECT
    city as City,
    grade as "Highest Grade"
FROM
    ProductOrderDetailsLR3.Customer Cust
WHERE
    grade = (
        SELECT
            max(grade)
        FROM
            ProductOrderDetailsLR3.Customer
        WHERE
            city = cust.city
        group by
            city
    );
```

2.1.4 Find the highest purchase amount ordered by each customer. Return customer ID, maximum purchase amount.

Code:

```
SELECT
    customer_id as "Customer ID",
    purch_amt as "Max Purchase Amount"
FROM
    ProductOrderDetailsLR3.Order Cust
WHERE
    purch_amt = (
        SELECT
            max(purch_amt)
        FROM
            ProductOrderDetailsLR3.Order
        WHERE
            customer_id = Cust.customer_id
    )
Order by
    customer_id
```

2.1.5 Find the highest purchase amount ordered by each customer on a particular date. Return order date and highest purchase amount.

Code: (Using sub-query)

```
SELECT
    ord_date as "Order Date",
    purch_amt as "Purchased Max Amount"
FROM
    ProductOrderDetailsLR3.Order Ord
WHERE
    purch_amt = (
        SELECT
            max(purch_amt)
        FROM
            ProductOrderDetailsLR3.Order
        WHERE
            ord_date = ord.ord_date
    )
Order by
    ord_date

    SELECT
```

```
    ord_date as "Order Date",
    max(purch_amt) as "Purchased Max Amount"
from
    ProductOrderDetailsLR3.Order
Group by
    ord_date
Order by
    ord_date
```

Code: (Using GROUP BY clause)

```
SELECT
    ord_date as "Order Date",
    max(purch_amt) as "Purchased Max Amount"
from
    ProductOrderDetailsLR3.Order
Group by
    ord_date
Order by
    ord_date
```

2.1.6 Determine the highest purchase amount made by each salesperson on 2012-08-17. Return salesperson ID and purchase amount.

Code:

```
SELECT
    salesman_id as "Salesperson ID",
    max(purch_amt) as "Purchased Max Amount",
    ord_date as "Ordered on"
from
    ProductOrderDetailsLR3.Order
WHERE
    ord_date = "2012-08-17"
Group by
    salesman_id
Order by
    salesman_id
```

2.1.7 Find the highest order *purchase* amount by each customer on a particular date. Filter the result by highest order purchase amount above 2000. Return customer ID, order date, highest purchase amount.

Code:

```
SELECT
    customer_id as Customer,
    ord_date as "Order Date",
    max(purch_amt) as "Max ordered on date"
from
    ProductOrderDetailsLR3.Order
WHERE
    purch_amt > 2000
group by
    customer_id,
    ord_date
```

3 Output

customer_id	cust_name	city	grade	salesman_id
3001	Brad Guzan	London	NULL	5005
3002	Nick Romando	New York	100	5001
3003	Jozy Altidor	Moscow	200	5007
3004	Fabian Johnson	Paris	300	5006
3005	Graham Zusi	California	200	5002
3006	Julian Green	London	300	5002
3007	Geoff Cameron	Berlin	100	5003

The complete Customer table.

ord_no	purch_amt	ord_date	customer_id	salesman_id
70001	150.50	2012-10-05	3005	5002
70002	65.26	2012-10-05	3002	5001
70003	2480.40	2012-10-10	3009	5003
70004	110.50	2012-08-17	3009	5003
70005	2400.60	2012-07-27	3007	5001
70007	948.50	2012-09-10	3005	5002
70008	5760.00	2012-09-10	3002	5001
70009	270.65	2012-09-10	3001	5005
70010	1983.43	2012-10-10	3004	5006
70011	75.29	2012-08-17	3003	5007
70012	250.45	2012-06-27	3008	5002
70013	3045.60	2012-04-25	3002	5001

The complete Order table.

Total Purchase Amount
17541.18

Total purchased amount.

Salesperson	Count(*)
5002	3
5001	4
5003	2
5005	1
5006	1
5007	1

Sales by each salesman.

City	Highest Grade
New York	100
Moscow	200
Paris	300
California	200
London	300
Berlin	100

Customers with highest grade in each city.

Customer ID	Max Purchase Amount
3001	270.65
3002	5760.00
3003	75.29
3004	1983.43
3005	948.50
3007	2400.60
3008	250.45
3009	2480.40

Max purchased amount by each customer.

Order Date	Purchased Max Amount
Filter...	Filter...
2012-04-25	3045.60
2012-06-27	250.45
2012-07-27	2400.60
2012-08-17	110.50
2012-09-10	5760.00
2012-10-05	150.50
2012-10-10	2480.40

Max purchased amount on each dates.

Salesperson ID	Purchased Max Amount	Ordered on
Filter...	Filter...	Filter...
5003	110.50	2012-08-17
5007	75.29	2012-08-17

Max purchased amount by salesman on 2012-08-17

Customer	Order Date	Max ordered on date
Filter...	Filter...	Filter...
3009	2012-10-10	2480.40
3007	2012-07-27	2400.60
3002	2012-09-10	5760.00
3002	2012-04-25	3045.60

Max purchased on dates that are above 2000.