

SQL MODULE

LAB - 3

Bandi Amrutha

AF0366376

Questions

Task 1: Update the Student table with the following information:

Change the email to 'jane_Smith@example.com'

Where FirstName is 'Jane' and LastName is 'Smith';

Code:

```
mysql> INSERT INTO Student Values (6, 'Jane', 'Smith', '1998-09-02', 'm', 'jane@gmail.com', '9867564387');
Query OK, 1 row affected (0.06 sec)

mysql> UPDATE Student
-> SET Email = 'jane_Smith@example.com'
-> WHERE FirstName = 'Jane' AND LastName = 'Smith';
Query OK, 1 row affected (0.04 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

Output:

```
mysql> select * from Student;
+-----+-----+-----+-----+-----+-----+-----+
| StudentId | FirstName | LastName | DateOfBirth | Gender | Email | Phone |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | Amrutha | Bandi | 2001-02-01 | f | amrutha.b@gmail.com | 9783687598 |
| 2 | Bhavani | Dhagudu | 2003-03-02 | f | bhavani.d@gmail.com | 3776726797 |
| 3 | Keerthi | Kothhinti | 2003-05-04 | f | keerthi.k@gmail.com | 6472838978 |
| 4 | Akshaya | Gundagala | 2002-06-05 | f | akshaya.g@gmail.com | 7652364776 |
| 5 | Anitha | Myla | 2003-06-06 | f | anitha.m@gmail.com | 8749857612 |
| 6 | Jane | Smith | 1998-09-02 | m | jane_Smith@example.com | 9867564387 |
+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.01 sec)
```

Update the Instructor with the following information:

Change the email to 'rogerwhite@example.com'

Where FirstName of the instructor is 'Roger' and LastName is 'White';

Code:

```
mysql> UPDATE Instructor
      -> SET Email = 'rogerwhite@example.com'
      -> WHERE FirstName = 'Dr. Roger' AND LastName = 'White';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

Output:

```
mysql> select * from Instructor;
```

| InstructorID | FirstName | LastName | Email |
|--------------|-------------|----------|----------------------------|
| 1 | Dr. Emily | Clark | emily.clark@example.com |
| 2 | Dr. Michael | Wang | michael.wang@example.com |
| 3 | Dr. Sarah | Lee | sarah.lee@example.com |
| 4 | Dr. Roger | Miller | robert.miller@example.com |
| 5 | Dr. Linda | Martinez | linda.martinez@example.com |
| 6 | Dr. Roger | White | rogerwhite@example.com |

```
6 rows in set (0.00 sec)
```

Task 2:

Delete record from the Student table on following condition:

Delete student/students records from the Student table where last name is Smith.

Code:

```
mysql> DELETE FROM Student
      -> WHERE LastName = 'Smith';
Query OK, 0 rows affected (0.08 sec)
```

Output:

```
mysql> select * from Student;
```

| StudentId | FirstName | LastName | DateOfBirth | Gender | Email | Phone |
|-----------|-----------|-----------|-------------|--------|---------------------|------------|
| 1 | Amrutha | Bandi | 2001-02-01 | f | amrutha.b@gmail.com | 9783687598 |
| 2 | Bhavani | Dhagudu | 2003-03-02 | f | bhavani.d@gmail.com | 3776726797 |
| 3 | Keerthi | Kothhinti | 2003-05-04 | f | keerthi.k@gmail.com | 6472838978 |
| 4 | Akshaya | Gundagala | 2002-06-05 | f | akshaya.g@gmail.com | 7652364776 |
| 5 | Joshitha | Myla | 2003-06-06 | f | Joshitha@gmail.com | 8749857612 |

```
5 rows in set (0.02 sec)
```

Task 3: List the student whose first name starts with J.

Code:

```
mysql> SELECT FirstName from Student where FirstName like 'J%';
```

Output:

```
+-----+  
| FirstName |  
+-----+  
| Joshitha |  
+-----+  
1 row in set (0.01 sec)
```

Lab 2. Database Schema:

Code:

```
mysql> CREATE DATABASE DB;  
Query OK, 1 row affected (0.18 sec)
```

Consider a simple database with one tables: Employee

Employee Table:

- Columns: emp_id (Primary Key), first_name, last_name, age, email

Task 1: Insert Data

Write an SQL INSERT statement to insert data into the Employee table.

Code:

```
mysql> INSERT INTO Employee (first_name, last_name, age, email) VALUES  
-> ('John', 'Doe', 28, 'john.doe@example.com'),  
-> ('Jane', 'Smith', 35, 'jane.smith@example.com'),  
-> ('Emily', 'Johnson', 40, 'emily.johnson@example.com'),  
-> ('Michael', 'Brown', 45, 'michael.brown@example.com'),  
-> ('Sarah', 'Davis', 25, 'sarah.davis@example.com');  
Query OK, 5 rows affected (0.07 sec)  
Records: 5 Duplicates: 0 Warnings: 0
```

Task 2: Retrieving Data

Write an SQL SELECT statement to retrieve the first_name and last_name of all employees from the Employee table.

Code:

```
mysql> SELECT first_name, last_name FROM Employee;
```

| first_name | last_name |
|------------|-----------|
| John | Doe |
| Jane | Smith |
| Emily | Johnson |
| Michael | Brown |
| Sarah | Davis |

```
5 rows in set (0.00 sec)
```

Task 3: Filtering Data

Write an SQL SELECT statement to retrieve the first_name, last_name, and age of employees who are older than 30 years.

Code:

```
mysql> SELECT first_name, last_name, age FROM Employee  
-> WHERE age > 30;
```

| first_name | last_name | age |
|------------|-----------|-----|
| Jane | Smith | 35 |
| Emily | Johnson | 40 |
| Michael | Brown | 45 |

```
3 rows in set (0.02 sec)
```

Task 4: Updating Data

Write an SQL UPDATE statement to increase the age of employees by 1 year for all employees older than 25.

Code:

```
mysql> UPDATE Employee
      -> SET age = age + 1
      -> WHERE age > 25;
Query OK, 4 rows affected (0.04 sec)
Rows matched: 4  Changed: 4  Warnings: 0
```

```
mysql> select * from Employee;
```

| emp_id | first_name | last_name | age | email |
|--------|------------|-----------|-----|---------------------------|
| 1 | John | Doe | 29 | john.doe@example.com |
| 2 | Jane | Smith | 36 | jane.smith@example.com |
| 3 | Emily | Johnson | 41 | emily.johnson@example.com |
| 4 | Michael | Brown | 46 | michael.brown@example.com |
| 5 | Sarah | Davis | 25 | sarah.davis@example.com |

5 rows in set (0.00 sec)

ChatGPT Exercise

Using ChatGPT generates SQL queries to update the Employee salary.

Scenario:

Due to a pricing adjustment, the company decided to increase the salary of all employees by 10%. Create an SQL update query to apply this change selectively to employees with a specific job title, say 'Manager'

Answer:

To update the salary of employees with a specific job title, such as 'Manager', by increasing it by 10%, you would first need to ensure your **Employee** table includes a **salary** column and a **job_title** column. Assuming these columns are already present in the table, you can use the following SQL UPDATE statement:

Adding Salary and Job Title Columns (if not already present)

If the **Employee** table does not yet have **salary** and **job_title** columns, you can add them with these statements:

```
mysql> ALTER TABLE Employee ADD COLUMN salary DECIMAL(10, 2) NOT NULL DEFAULT 0.00;
Query OK, 0 rows affected (0.31 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> ALTER TABLE Employee ADD COLUMN job_title VARCHAR(50) NOT NULL;
Query OK, 0 rows affected (0.03 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

Insert Sample Data with Job Titles and Salaries (if needed)

```
mysql> INSERT INTO Employee (first_name, last_name, age, email, salary, job_title) VALUES
-> ('John', 'Doe', 28, 'john.doe2@example.com', 50000, 'Developer'),
-> ('Jane', 'Smith', 35, 'jane.smith2@example.com', 60000, 'Manager'),
-> ('Emily', 'Johnson', 40, 'emily.johnson2@example.com', 70000, 'Manager'),
-> ('Michael', 'Brown', 45, 'michael.brown2@example.com', 55000, 'Developer'),
-> ('Sarah', 'Davis', 25, 'sarah.davis2@example.com', 48000, 'Designer');
Query OK, 5 rows affected (0.14 sec)
Records: 5 Duplicates: 0 Warnings: 0
```

Task: Update the Salary of Managers by 10%

To increase the salary of all employees with the job title 'Manager' by 10%, you can use the following SQL UPDATE statement:

Code:

```
mysql> UPDATE Employee
-> SET salary = salary * 1.10
-> WHERE job_title = 'Manager';
Query OK, 2 rows affected (0.09 sec)
Rows matched: 2 Changed: 2 Warnings: 0
```

Explanation:

- UPDATE Employee: Specifies the table to update.
- SET salary = salary * 1.10: Increases the current salary by 10%.
- WHERE job_title = 'Manager': Ensures only employees with the job title 'Manager' are affected by this update.

Running this query will adjust the salaries of all managers in the Employee table by increasing them by 10%.

