Practicing Data Manipulation in SQL

Step 1: Prepare for the Application

You'll create a small application to manage a sample database using MySQL. The application will allow users to perform INSERT, UPDATE, and DELETE operations on a table called Users.

Instructions:

1. Open Visual Studio Code and connect it to a MySQL database using the terminal.
2. Create a new database called SampleDB using the command: CREATE DATABASE SampleDB;
3. Use the new database: USE SampleDB;
4. Create a table named Users with the following structure: CREATE TABLE Users (UserID INT AUTO_INCREMENT PRIMARY KEY, FirstName VARCHAR(50), LastName VARCHAR(50), Email VARCHAR(100), Age INT);
5. Populate the Users table with sample data: INSERT INTO Users (FirstName, LastName, Email, Age) VALUES ('Aisha', 'Khan', 'aisha.khan@example.com', 29), ('Carlos', 'Garcia', 'carlos.garcia@example.com', 35), ('Mei', 'Chen', 'mei.chen@example.com', 24);

Step 2: Implementing an INSERT Operation

You will write a query to add a new user to the Users table.

Instructions:

1. Add a new user with the following details:
    - FirstName: Arjun
    - LastName: Patel
    - Email: arjun.patel@example.com
    - Age: 41
2. Ensure that the user has been successfully added by querying the table.

Step 3: Implementing an UPDATE Operation

You will write a query to update the age of an existing user.

Instructions:

1. Update the age of the user whose FirstName is Mei to 26.
2. Ensure that only the targeted record has been updated by querying the table.

Step 4: Implementing a DELETE Operation

You will write a query to delete a user from the Users table.

Instructions:

1. Delete the user with the LastName Garcia.
2. Verify that the record has been deleted by querying the table.

Step 5: Practicing Safety Measures

You will practice safe SQL practices by using the WHERE clause to avoid unintentional changes.

Instructions:

1. Attempt to update the age of all users to 30 without using a WHERE clause. Observe the behavior.
2. Roll back the changes using the following command: ROLLBACK;
3. Retry the update for a specific user, this time using a WHERE clause to avoid unintended changes.

SampleDB.sql:

```sql
-- ============================================
-- Lab: Practicing Data Manipulation in SQL
-- Database: SampleDB
-- ============================================

-- Step 1. Prepare for the Application
-- --------------------------------------------
-- Create new database
CREATE DATABASE IF NOT EXISTS SampleDB;

-- Use the new database
USE SampleDB;

-- Drop table if exists (clean state)
DROP TABLE IF EXISTS Users;

-- Create Users table
CREATE TABLE Users (
    UserID INT AUTO_INCREMENT PRIMARY KEY,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    Email VARCHAR(100),
    Age INT
);

-- Insert sample data
INSERT INTO Users (FirstName, LastName, Email, Age) VALUES
('Aisha', 'Khan', 'aisha.khan@example.com', 29),
('Carlos', 'Garcia', 'carlos.garcia@example.com', 35),
('Mei', 'Chen', 'mei.chen@example.com', 24);

-- Check initial state
SELECT * FROM Users;

-- ============================================
-- Step 2. INSERT Operation
-- --------------------------------------------
-- Add a new user Arjun Patel
INSERT INTO Users (FirstName, LastName, Email, Age)
VALUES ('Arjun', 'Patel', 'arjun.patel@example.com', 41);

-- Verify insert
SELECT * FROM Users;

-- ============================================
-- Step 3. UPDATE Operation
-- --------------------------------------------
-- Update Mei's age to 26
UPDATE Users
SET Age = 26
WHERE FirstName = 'Mei';

-- Verify update
SELECT * FROM Users WHERE FirstName = 'Mei';

-- ============================================
-- Step 4. DELETE Operation
```

```sql
-- ------------------------------------------
-- Delete Carlos Garcia
DELETE FROM Users
WHERE LastName = 'Garcia';

-- Verify delete
SELECT * FROM Users;

-- ==========================================
-- Step 5. Practicing Safety Measures
-- ------------------------------------------
-- Danger: Update all users' age to 30 (without WHERE)
-- (In real work, this should be avoided!)
START TRANSACTION;
UPDATE Users
SET Age = 30;

-- Check effect
SELECT * FROM Users;

-- Rollback the change
ROLLBACK;

-- Verify rollback worked
SELECT * FROM Users;

-- Safe update: only for Arjun
UPDATE Users
SET Age = 42
WHERE FirstName = 'Arjun';

-- Verify safe update
SELECT * FROM Users WHERE FirstName = 'Arjun';
```