**Lab # 09:**

**Lab Title:**

**“DML Commands and Different Functions in MySQL”**

**Lab Objectives:**

* Learn about the data manipulation commands
* Learn about the different functions and how to use them in MySQL

**Introduction:**  
Data Manipulation Language (DML) commands in a database management system like MySQL are used to interact with and manipulate data stored within tables. These commands primarily focus on performing operations related to data retrieval, insertion, modification, and deletion. Here are the key DML commands:

1. **SELECT:** Retrieves data from one or more tables.
2. **INSERT:** Adds new data into a table.
3. **UPDATE:** Modifies existing data in a table.
4. **DELETE:** Removes specific data from a table.

In MySQL, various functions serve diverse purposes, aiding in data manipulation, computation, and transformation. Some of the common types of functions are Aggregate functions , String functions , Date Functions , Numeric Functions.

**Creating Tables:**

|  |
| --- |
| create database lab\_9\_db;  use lab\_9\_db;  -- Table product details  CREATE TABLE product\_details(  pro\_code int not null primary key,  pro\_name char(20) not null,  pro\_price decimal,  transaction\_no int,  pro\_date date);  INSERT INTO product\_details (pro\_code, pro\_name, pro\_price, transaction\_no, pro\_date)  VALUES  ('001', 'T-Shirt', 599, 33445555, '2014-11-07'),  ('002', 'Jeans', 1999, 222222, '2014-11-07'),  ('003', 'Shirt', 999, 5555603, '2014-12-07'),  ('004', 'Trouser', 1799, 17777303, '2014-12-07'),  ('005', 'Sweatshirt', 799, 22228801, '2014-07-13');  -- Table student info  CREATE TABLE student\_info (  St\_id INT PRIMARY KEY,  St\_Name VARCHAR(50) NOT NULL,  DOB DATE,  Addr VARCHAR(50),  Cont\_no CHAR(15)  );  INSERT INTO student\_info (St\_id, St\_Name, DOB, Addr, Cont\_no)  VALUES  ('11199', 'Rahul', '1994-09-22', '22/9/1994 DOB Addr 11c siwan', '8005555245'),  ('11112', 'Jignesh', '1995-08-02', 'Sector 26 chd', '8905555665'),  ('11122', 'Sushil', '1994-01-07', 'Jd colony amb', '9005555555'),  ('11155', 'Vikas', '1992-05-11', 'Yamuna nagar', '8090000245'),  ('11116', 'Rohan', '1992-07-18', 'New city chd', '7775585245');  -- Table Enroll information  CREATE TABLE enroll\_info (  enroll\_no INT PRIMARY KEY,  name VARCHAR(50) NOT NULL,  age INT NOT NULL,  contact VARCHAR(20) NOT NULL,  city VARCHAR(20) NOT NULL  );  -- Insert data into the table  INSERT INTO enroll\_info (enroll\_no, name, age, contact, city)  VALUES (12, 'Vivek', 21, '800444245', 'Chd'),  (13, 'Bhusan', 20, '894444665', 'Amb'),  (23, 'Tarun', 30, '900675555', 'Chd'),  (45, 'Prasoon', 25, '809987245', 'Chd'),  (48, 'Somya', 25, '799876245', 'Kota');  -- Salary mar  -- Create table  CREATE TABLE salary\_mar (  emp\_id INT PRIMARY KEY,  emp\_name VARCHAR(50) NOT NULL,  salary DECIMAL(10,2) NOT NULL,  sal\_da DECIMAL(10,2) NOT NULL,  sal\_bonus DECIMAL(10,2) NOT NULL  );  -- Insert data into the table  INSERT INTO salary\_mar (emp\_id, emp\_name, salary, sal\_da, sal\_bonus)  VALUES (1223, 'Anuj', 25000.00, 800.00, 2000.00),  (6332, 'Richa', 55000.00, 1500.00, 2500.00),  (6442, 'Rohit', 65000.00, 1800.00, 2100.00),  (3557, 'Sunny', 40000.00, 1200.00, 5000.00),  (8668, 'Viren', 35000.00, 1000.00, 1000.00);  -- Customer info  CREATE TABLE customer\_info (  C\_id INT PRIMARY KEY AUTO\_INCREMENT,  C\_name VARCHAR(50) NOT NULL,  C\_city VARCHAR(50) NOT NULL,  Sales\_id VARCHAR(10) NOT NULL,  Phone VARCHAR(20)  );  INSERT INTO customer\_info (C\_name, C\_city, Sales\_id, Phone)  VALUES ('Anil', 'Chd', '001', '8884455128'),  ('A.Sharma', 'Ambala', '002', '9997775523'),  ('SunilVerma', 'Delhi', '003', '8884444666'),  ('Ravi', 'Delhi', '004', '9026345234'),  ('John', 'Chd', '005', '9022266145'); |

**In-Lab Tasks:**

**Task # 01:**

|  |
| --- |
| DELETE FROM product\_details WHERE pro\_code > 0;  DELETE FROM student\_info WHERE St\_id = '11112'; |

**Task #02:**

|  |
| --- |
| SELECT \* FROM product\_details;  SELECT \* FROM student\_info; |

**Task#03:**

|  |
| --- |
| SELECT St\_Name, DOB FROM student\_info; |

**Task#04:**

|  |
| --- |
| SELECT \* FROM student\_info WHERE DOB = '1994-09-22'; |

**Task#05:**

|  |
| --- |
| SELECT COUNT(\*) AS total\_students FROM student\_info; |

**Task#06:**

|  |
| --- |
| INSERT INTO product\_details (pro\_code, pro\_name, pro\_price, transaction\_no, pro\_date)  VALUES  ('001', 'T-Shirt', 599, 33445555, '2014-11-07'),  ('002', 'Jeans', 1999, 222222, '2014-11-07'),  ('003', 'Shirt', 999, 5555603, '2014-12-07'),  ('004', 'Trouser', 1799, 17777303, '2014-12-07'),  ('005', 'Sweatshirt', 799, 22228801, '2014-07-13');  SELECT SUM(pro\_price) AS total\_product\_price FROM product\_details; |

**Task#07:**

|  |
| --- |
| SELECT AVG(age) AS average\_age FROM enroll\_info; |

**Task#08:**

|  |
| --- |
| SELECT MAX(pro\_price) AS max\_pro\_price FROM product\_details; |

**Task#09:**

|  |
| --- |
| SELECT MIN(pro\_price) AS min\_pro\_price FROM product\_details; |

**Post-Lab Tasks:**

**Task#01:**

DELETE is more flexible and can selectively remove rows based on conditions. It's logged in the transaction log and can take longer for large deletions.

TRUNCATE is faster as it removes all rows at once without logging each deletion. It deallocates data pages and resets the table, but it doesn't allow any conditions and is a non-logged operation.

**Task#02:**

|  |
| --- |
| CREATE TABLE Employee (  Emp\_no INT PRIMARY KEY,  E\_name VARCHAR(255) NOT NULL,  E\_address VARCHAR(255) NOT NULL,  E\_ph\_no VARCHAR(20) NOT NULL,  Dept\_no INT NOT NULL,  Dept\_name VARCHAR(255) NOT NULL,  Job\_id INT NOT NULL,  Designation VARCHAR(255) NOT NULL,  Salary DECIMAL(10,2) NOT NULL  );  CREATE TABLE Employee (  Emp\_no INT PRIMARY KEY,  E\_name VARCHAR(255) NOT NULL,  E\_address VARCHAR(255) NOT NULL,  E\_ph\_no VARCHAR(20) NOT NULL,  Dept\_no INT NOT NULL,  Dept\_name VARCHAR(255) NOT NULL,  Job\_id INT NOT NULL,  Designation VARCHAR(255) NOT NULL,  Salary DECIMAL(10,2) NOT NULL  );  INSERT INTO Employee (Emp\_no, E\_name, E\_address, E\_ph\_no, Dept\_no, Dept\_name, Job\_id, Designation, Salary)  VALUES  (1, 'John', 'ABC town', '234567', 567, 'Accounting', 1, 'CEO', 1000000),  (2, 'Marry', 'DEF town', '234568', 568, 'Finance', 2, 'General Manager', 50000),  (3, 'Bella', 'GHI town', '234569', 569, 'Managing', 3, 'Manager', 1000000),  (4, 'Sara', 'JKL town', '234570', 570, 'Managing', 4, 'Manager', 5000000),  (5, 'Meeam', 'MNO town', '234571', 571, 'Accounting', 5, 'IT professor', 20000),  (6, 'Pigeon', 'PQR town', '234572', 572, 'Managing', 6, 'Clerk', 10000),  (7, 'Sparrow', 'STU town', '234573', 573, 'Managing', 7, 'Clerk', 10000);  -- Queries  -- query1  SELECT Emp\_no, E\_name, Salary FROM Employee  WHERE Designation = 'Manager';  -- query2  SELECT \* FROM Employee  WHERE Salary > (SELECT MAX(Salary) FROM Employee WHERE Designation = 'IT professor');  -- query3  SELECT \* FROM Employee  WHERE Designation = 'Clerk'; |