

Heaven's Light is Our Guide

Rajshahi University of Engineering & Technology



Course code: ECE 2216

Course Name: Database Management System Sessional

Submitted by
Md. Fahim hossen
Roll no. 2110055
Session :2021-2022
Reg no. 1109 / 2021-2022
Department of ECE, RUET

Submitted To
Oishi Jyoti
Assistant Professor ,
Department of ECE, RUET

Experiment Number: 01

Experiment Name: DDL and DML commands with MySQL

Theory :

MySQL is an open-source relational database management system (RDBMS) that uses Structured Query Language (SQL) to interact with and manage databases. It is widely used for applications like web databases, data warehousing, and e-commerce. It allows users to create, read, update, and delete (CRUD) data in a database. MySQL is known for being fast, reliable, and scalable. It is a relational database. A relational database is a type of database that organizes data into structured tables (also called relations). Each table consists of rows (records) and columns (attributes), where each column represents a data field, and each row represents a data entry. The tables can be linked together through relationships (using foreign keys). Data in relational databases is typically accessed and managed using SQL. [1]

DDL (Data Definition Language) Commands: DDL commands are used to define and modify the structure of database objects like tables, indexes, and views. They are concerned with the schema (structure) of the database rather than the data itself. [2]

- **CREATE:** Creates new databases or tables.
- **ALTER:** Modifies the structure of an existing table, such as adding or changing columns.
- **RENAME:** Renames a table or column.
- **DROP:** Removes an entire table or database.
- **TRUNCATE:** Clears all data from a table while preserving its structure.

DML (Data Manipulation Language) : DML (Data Manipulation Language) is a subset of SQL (Structured Query Language) that focuses on manipulating data within database objects like tables. DML is used for tasks such as retrieving, inserting, updating, and deleting data in a database. The most common DML commands include: [2]

- **INSERT:** Inserts new data into a table.
- **UPDATE:** Updates existing records in a table.
- **DELETE:** Deletes specific records from a table based on a condition.
- **WHERE:** Filters records for use in SELECT, UPDATE, or DELETE operations.
- **MODIFY:** Changes the data type of a column.
- **SET:** Used in UPDATE statements to modify values under specific conditions.

Problem Statements:

1. Create a database and a table for storing information of 10 students.
2. Change a specific column name and data type.
3. Add a new column named "Log". Set the value to "Applicable" or "Not Applicable" based on the condition (marks)
4. Delete the student information for those whose marks are below 30

Software Used:

1. XAMPP

Problem 1: Create a database and a table for storing information of 10 student

SQL Commands To solve Following Query :

```
1 CREATE DATABASE student_details;
2 USE student_details;
3 CREATE TABLE student_info (
4 Roll INT,
5 Name VARCHAR(50),
6 Semester INT,
7 Major VARCHAR(50),
8 Obtained_Marks FLOAT
9 );
10 INSERT INTO student_info (Roll, Name, Semester, Major, Obtained_Marks) VALUES
11 (1, 'fahim', 4, 'Aeronautical Engineering', 23),
12 (2, 'Hossen', 2, 'Physics', 35),
13 (3, 'Anas', 2, 'Mathmatics', 54),
14 (4, 'Rupa', 7, 'Mathematics', 92),
15 (5, 'Rakib', 4, 'Civil Engineering', 42),
16 (6, 'Rana', 3, 'Mechanical Engineering', 25),
17 (7, 'Roni', 5, 'Mechanical Engineering', 63),
18 (8, 'Messy', 2, 'Aeronautical Engineering', 30),
19 (9, 'Rina', 4, 'Physics', 12),
20 (10, 'Rocky', 3, 'Mathematics', 29);
```

Output After executing the SQL Query:

Roll	Name	Semester	Major	Obtained_Marks
1	fahim	4	Aeronautical Engineering	23
2	Hossen	2	Physics	35
3	Anas	2	Mathmatics	54
4	Rupa	7	Mathematics	92
5	Rakib	4	Civil Engineering	42
6	Rana	3	Mechanical Engineering	25
7	Roni	5	Mechanical Engineering	63
8	Messy	2	Aeronautical Engineering	30
9	Rina	4	Physics	12
10	Rocky	3	Mathematics	29

Figure 1: Database and table creation output

Discussion:

This SQL script begins by creating a new database called student_details to store student-related data. It then selects this database for use in subsequent operations with the USE command. Afterward, a table named Student_info is created within the student_details database. This table includes five columns: Roll (an integer for student identification), Name (a variable-length string for student names), Semester (an integer representing the semester the student is in), Major (a string for the student's field of study), and Obtained_Marks (a floating-point number to store the student's marks). Finally, several records of students are inserted into the Student_info table, each including values for the corresponding columns.

Problem 2: Change a specific column name and data type

Ans : SQL Commands To solve Following problem :

```
1 ALTER TABLE student_info CHANGE COLUMN Major Fav_Subject VARCHAR(50);
2 ALTER TABLE student_info MODIFY Obtained_Marks DOUBLE|
```

Output After executing the SQL Query :

Roll	Name	Semester	Fav_Subject	Obtained_Marks
1	fahim	4	Aeronautical Engineering	23
2	Hossen	2	Physics	35
3	Anas	2	Mathmatics	54
4	Rupa	7	Mathematics	92
5	Rakib	4	Civil Engineering	42
6	Rana	3	Mechanical Engineering	25
7	Roni	5	Mechanical Engineering	63
8	Messy	2	Aeronautical Engineering	30
9	Rina	4	Physics	12
10	Rocky	3	Mathematics	29

Figure 2: Output After Column change

	#	Name	Type
<input type="checkbox"/>	1	Roll	int(11)
<input type="checkbox"/>	2	Name	varchar(50)
<input type="checkbox"/>	3	Semester	int(11)
<input type="checkbox"/>	4	Fav_Subject	varchar(50)
<input type="checkbox"/>	5	Obtained_Marks	double

Figure 3: output after changing Datatype

Discussion:

This SQL script makes two modifications to the student_info table. First, it renames the column Major to Fav_Subject using the CHANGE COLUMN command, while keeping the data type as VARCHAR (50) for storing variable-length strings up to 50 characters. The second modification is to the Obtained_Marks column, where the data type is changed from its original type to DOUBLE using the MODIFY command, allowing for more precision in storing decimal values. Both alterations adjust the structure of the student_info table to better fit the updated data requirements.

Problem 3: Add a new column named "Log". Set the value to "Applicable" or "Not Applicable" based on the condition (marks < 30)

Ans : SQL Commands To solve Following Problem :

```

1 ALTER TABLE student_info ADD Log VARCHAR(20);
2 UPDATE student_info SET Log = 'Applicable' WHERE Obtained_Marks >= 30;
3 UPDATE student_info SET Log = 'Not Applicable' WHERE Obtained_Marks < 30;

```

Output After executing the SQL Query :

Roll	Name	Semester	Fav_Subject	Obtained_Marks	Log
1	fahim	4	Aeronautical Engineering	23	Not Applicable
2	Hossen	2	Physics	35	Applicable
3	Anas	2	Mathmatics	54	Applicable
4	Rupa	7	Mathematics	92	Applicable
5	Rakib	4	Civil Engineering	42	Applicable
6	Rana	3	Mechanical Engineering	25	Not Applicable
7	Roni	5	Mechanical Engineering	63	Applicable
8	Messy	2	Aeronautical Engineering	30	Applicable
9	Rina	4	Physics	12	Not Applicable
10	Rocky	3	Mathematics	29	Not Applicable

Figure 4: Output after adding and updating the 'Log' column

Discussion :

This SQL script introduces a new column and updates its values based on conditions. First, the ALTER TABLE command adds a new column called Log to the student_info table with a data type of VARCHAR(20), allowing for strings up to 20 characters. Then, two UPDATE statements are used to set values for the Log column. The first update sets the value of Log to 'Applicable' for students whose Obtained_Marks are less than 30, indicating some condition or status. The second update sets the Log column to 'Not Applicable' for students with Obtained_Marks greater than or equal to 30. This categorizes the students based on their marks.

Problem 4: Delete the student information for those whose marks are below 30

Ans : SQL Commands To solve Following Problem :

```
1 DELETE FROM student_info WHERE Obtained_Marks < 30;
```

Output After executing the SQL Query :

Roll	Name	Semester	Fav_Subject	Obtained_Marks	Log
2	Hossen	2	Physics	35	Applicable
3	Anas	2	Mathmatics	54	Applicable
4	Rupa	7	Mathematics	92	Applicable
5	Rakib	4	Civil Engineering	42	Applicable
7	Roni	5	Mechanical Engineering	63	Applicable
8	Messy	2	Aeronautical Engineering	30	Applicable

Figure 5: Output after deleting records with marks below 30

Discussion :

This SQL command removes records from the Students table based on a specific condition. The DELETE FROM statement targets the Students table and specifies that any student with Obtained_Marks less than 30 should be deleted. This operation helps to eliminate entries that do not meet a certain performance threshold, effectively cleaning up the dataset by removing underperforming students. After executing this command, only the records of students who have obtained marks of 30 or higher will remain in the table.

Reference:

[1] [Online]. Available: <https://www.tutorialspoint.com/sql/sql-discussion.htm>.

[2] [Online]. Available: https://www.w3schools.com/sql/sql_syntax.asp.