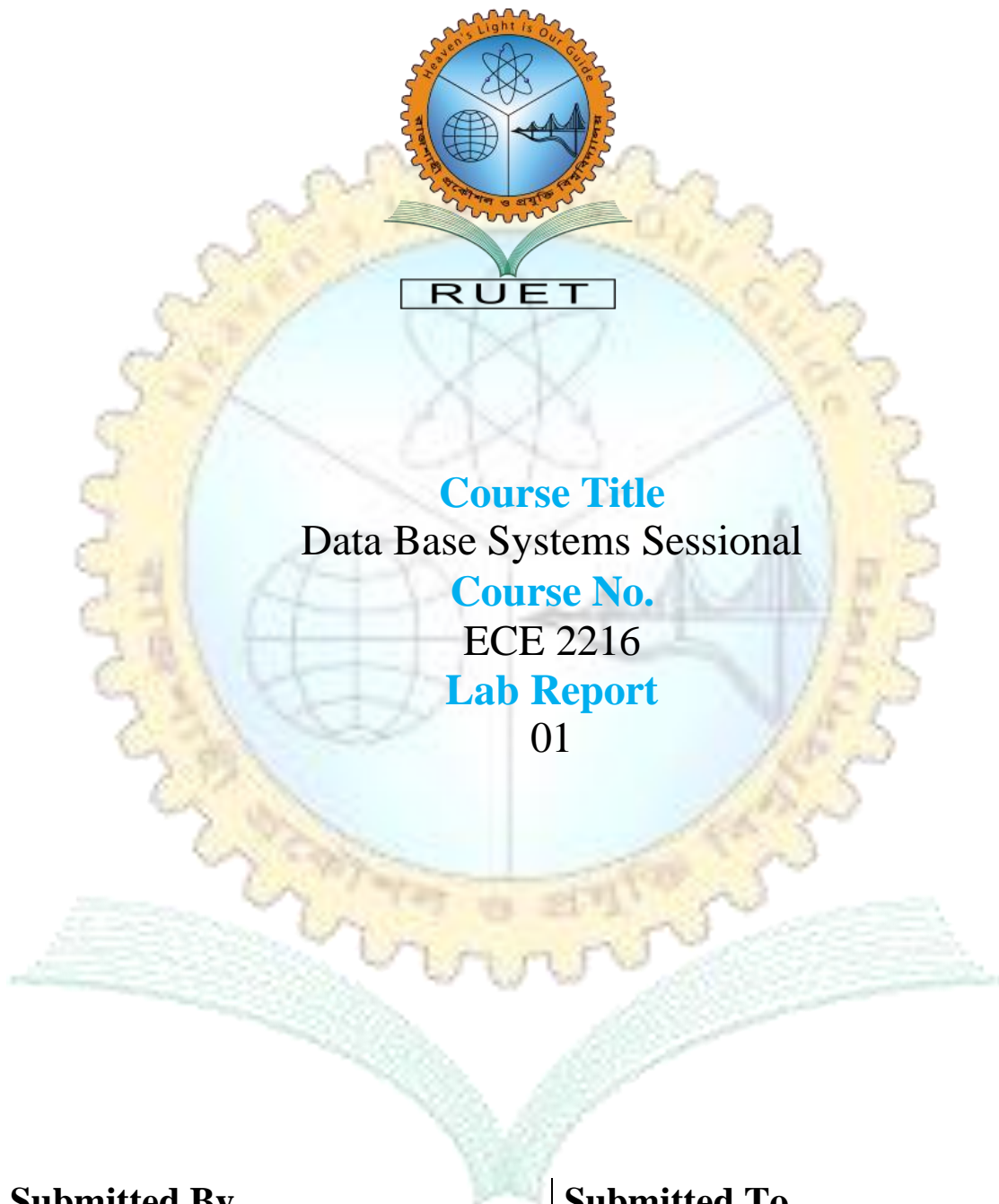


Heaven's Light is Our Guide
Rajshahi University of Engineering & Technology



Course Title

Data Base Systems Sessional

Course No.

ECE 2216

Lab Report

01

Submitted By

Ribbie Mohammad Omar
Roll no: 2110060
Reg no: 1114/2021-22
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 utilizes Structured Query Language (SQL) for managing and processing data within databases. It is widely used in web applications and large-scale systems due to its flexibility, scalability, and support for multi-user environments. MySQL enables users to perform various database operations, from creating databases to retrieving complex datasets.

SQL operations are categorized into two key types: Data Definition Language (DDL) and Data Manipulation Language (DML).

Data Definition Language (DDL)

DDL commands are used to define, modify, and remove database objects such as tables, indexes, and constraints. They deal with the structure of the database schema. The most commonly used DDL commands include:

- a. CREATE: To create new databases, tables, or other objects.
- b. ALTER: To modify the structure of existing database objects.
- c. RENAME: To change the name of an existing object.
- d. DROP: To remove database objects permanently.
- e. TRUNCATE: To delete all records from a table while retaining its structure for future use.

Data Manipulation Language (DML)

DML commands are used to manage the data within database objects. They focus on inserting, updating, or deleting data records from tables. Common DML commands include:

- a. INSERT: To add new data records to a table.
- b. UPDATE: To modify existing data records in a table.
- c. DELETE: To remove specific data records from a table.
- d. MODIFY: To alter existing data values or structures in certain SQL dialects.
- e. SET: To assign values to variables or update data in specified conditions.

Usage of DDL and DML in Database Operations

In practice, DDL and DML commands work together to manage databases. For instance, using DDL commands, a database and table can be created with specific column structures. If needed, column names and data types can be altered using the ALTER command. Data can be inserted into the table under specific conditions using INSERT, and unwanted data can be removed from the table using DELETE based on conditions like matching a certain value or ID. Similarly, existing records can be updated conditionally using UPDATE to reflect changes in the data stored in the database.

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 < 30).
4. Delete the student information for those whose marks are below 30.

Use the following DDLs:

CREATE

ALTER

RENAME

DROP

TRUNCATE

Use the following DMLs:

INSERT

UPDATE

DELETE

MODIFY

SET

Software Used:

1. XAMPP (used for running MySQL locally through the Apache server)

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

SQL Commands:

```
1 CREATE DATABASE Lab_1;
2 USE Lab_1;
3 CREATE TABLE st_info (Roll int(255), Name varchar(255), Semester varchar(255), Major_Subject varchar(255), Obtained_marks
  int(255), Batch int(255));
4
5 INSERT INTO st_info (Roll, Name, Semester, Major_Subject, Obtained_marks, Batch) VALUES
6 ('2110041', 'Ribbie Mohammad Omar', '2-1', 'Computer Engineering', '84', '21'),
7 ('2110042', 'Md. Abdullah Ibna Shad', '2-1', 'Mathematics', '29', '21'),
8 ('2210041', 'Tasnim Mehjabin Shaon', '1-2', 'Computer Engineering', '86', '22'),
9 ('2110043', 'Ripon Kumar Ghosh', '2-1', 'Electrical Engineering', '86', '21'),
10 ('2110044', 'Md. Hasin Sadaf', '2-1', 'Computer Engineering', '72', '21'),
11 ('2110045', 'Arifur Rahman', '2-1', 'Mathematics', '12', '21'),
12 ('2110048', 'EMon Mahmud', '2-1', 'Computer Engineering', '84', '21'),
13 ('2110050', 'Sazid Rahman', '2-1', 'Computer Engineering', '88', '21'),
14 ('2100076', 'Fariha Rahman Prova', '3rd Year', 'MBBS', '100', '21'),
15 ('2110052', 'Sheikh Golam Rabbani', '2-1', 'Computer Engineering', '84', '21');
```

Output:

Roll	Name	Semester	Major_Subject	Obtained_marks	Batch
2110041	Ribbie Mohammad Omar	2-1	Computer Engineering	84	21
2110042	Md. Abdullah Ibna Shad	2-1	Mathematics	29	21
2210041	Tasnim Mehjabin Shaon	1-2	Computer Engineering	86	22
2110043	Ripon Kumar Ghosh	2-1	Electrical Engineering	86	21
2110044	Md. Hasin Sadaf	2-1	Computer Engineering	72	21
2110045	Arifur Rahman	2-1	Mathematics	12	21
2110048	EMon Mahmud	2-1	Computer Engineering	84	21
2110050	Sazid Rahman	2-1	Computer Engineering	88	21
2100076	Fariha Rahman Prova	3rd Year	MBBS	100	21
2110052	Sheikh Golam Rabbani	2-1	Computer Engineering	84	21

Figure 1: Database and table creation output

Problem 2: Change a specific column name and data type

SQL Commands:

```
1 RENAME TABLE st_info to st_info_table;
2 ALTER TABLE st_info_table CHANGE COLUMN Batch HSC_Batch varchar(255);
3 ALTER TABLE st_info_table MODIFY Obtained_marks double;
```

Output:

Roll	Name	Semester	Major_Subject	Obtained_marks	HSC_Batch
2110041	Ribbie Mohammad Omar	2-1	Computer Engineering	84	21
2110042	Md. Abdullah Ibna Shad	2-1	Mathematics	29	21
2210041	Tasnim Mehjabin Shaon	1-2	Computer Engineering	86	22
2110043	Ripon Kumar Ghosh	2-1	Electrical Engineering	86	21
2110044	Md. Hasin Sadaf	2-1	Computer Engineering	72	21
2110045	Arifur Rahman	2-1	Mathematics	12	21
2110048	EMon Mahmud	2-1	Computer Engineering	84	21
2110050	Sazid Rahman	2-1	Computer Engineering	88	21
2100076	Fariha Rahman Prova	3rd Year	MBBS	100	21
2110052	Sheikh Golam Rabbani	2-1	Computer Engineering	84	21

Figure 2: Column name change output

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

SQL Commands:

```
1 ALTER TABLE st_info_table ADD LOG varchar(255);
2 UPDATE st_info_table SET LOG = 'APPLICABLE' WHERE Obtained_marks<30;
3 UPDATE st_info_table SET LOG = 'NOT APPLICABLE' WHERE Obtained_marks>30;
```

Output:

Roll	Name	Semester	Major_Subject	Obtained_marks	HSC_Batch	LOG
2110041	Ribbie Mohammad Omar	2-1	Computer Engineering	84	21	NOT APPLICABLE
2110042	Md. Abdullah Ibna Shad	2-1	Mathematics	29	21	APPLICABLE
2210041	Tasnim Mehjabin Shaon	1-2	Computer Engineering	86	22	NOT APPLICABLE
2110043	Ripon Kumar Ghosh	2-1	Electrical Engineering	86	21	NOT APPLICABLE
2110044	Md. Hasin Sadaf	2-1	Computer Engineering	72	21	NOT APPLICABLE
2110045	Arifur Rahman	2-1	Mathematics	12	21	APPLICABLE
2110048	EMon Mahmud	2-1	Computer Engineering	84	21	NOT APPLICABLE
2110050	Sazid Rahman	2-1	Computer Engineering	88	21	NOT APPLICABLE
2100076	Fariha Rahman Prova	3rd Year	MBBS	100	21	NOT APPLICABLE
2110052	Sheikh Golam Rabbani	2-1	Computer Engineering	84	21	NOT APPLICABLE

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

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

SQL Commands:

```
1 DELETE FROM st_info_table WHERE LOG = 'APPLICABLE';
```

Output:

Roll	Name	Semester	Major_Subject	Obtained_marks	HSC_Batch	LOG
2110041	Ribbie Mohammad Omar	2-1	Computer Engineering	84	21	NOT APPLICABLE
2210041	Tasnim Mehjabin Shaon	1-2	Computer Engineering	86	22	NOT APPLICABLE
2110043	Ripon Kumar Ghosh	2-1	Electrical Engineering	86	21	NOT APPLICABLE
2110044	Md. Hasin Sadaf	2-1	Computer Engineering	72	21	NOT APPLICABLE
2110048	EMon Mahmud	2-1	Computer Engineering	84	21	NOT APPLICABLE
2110050	Sazid Rahman	2-1	Computer Engineering	88	21	NOT APPLICABLE
2100076	Fariha Rahman Prova	3rd Year	MBBS	100	21	NOT APPLICABLE
2110052	Sheikh Golam Rabbani	2-1	Computer Engineering	84	21	NOT APPLICABLE

Figure 5: Output after deleting records with LOG “Applicable”

Truncation and Drop:

SQL Commands:

```
1 TRUNCATE st_info_table;
```

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0004 seconds.)

```
DROP DATABASE lab_1;
```

Output:

Roll	Name	Semester	Major_Subject	Obtained_marks	HSC_Batch	LOG
------	------	----------	---------------	----------------	-----------	-----

Query results operations

Figure 5: Output after deleting records with LOG “Applicable”

Discussions:

In this experiment, we successfully applied MySQL's DDL and DML commands to manage and manipulate a student database. We began by creating a table containing student details such as roll number, name, semester, major subject, obtained marks, and batch. Next, we used the ALTER command to rename the "batch" column to "HSC batch" for better clarity. Additionally, we added a new column, "LOGs", where we inserted "Applicable" for students with marks greater than 30 and "Not Applicable" for those with marks less than 30, demonstrating the use of conditional UPDATE. Subsequently, we deleted the records of students who were marked as "Applicable" using the DELETE command. Finally, we truncated the table, clearing all data while retaining its structure, and dropped the database to remove it entirely from the system. This experiment highlighted the practical applications of DDL and DML commands in database management and manipulation.

References:

- [1] Oracle Corporation, "MySQL 8.0 Reference Manual," [Online]. Available: <https://dev.mysql.com/doc/refman/8.0/en/>. Accessed: Sep. 24, 2024.
- [2] C. J. Date, *An Introduction to Database Systems*, 8th ed. Boston, MA: Addison-Wesley, 2003.
- [3] R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*, 7th ed. Boston, MA: Pearson, 2016.
- [4] P. DuBois, *MySQL Cookbook: Solutions for Database Developers and Administrators*, 4th ed. Sebastopol, CA: O'Reilly Media, 2014.
- [5] A. Silberschatz, H. F. Korth, and S. Sudarshan, *Database System Concepts*, 7th ed. New York, NY: McGraw-Hill, 2020.