

“Heaven’s Light is Our Guide”

Rajshahi University of Engineering & Technology
Rajshahi, Bangladesh



Department of Electrical & Computer Engineering

Course Code: ECE 2216

Course Title: Database System Sessional

Experiment No: 01

Date of Experiment: 10 September 2024

Date of Submission: 23 September 2024

Submitted To

Oishi Jyoti
Assistant Professor
Electrical & Computer Engineering
RUET

Submitted By

Nur-E-Sharin Trisha
Roll:2110034
Reg.:1088/2021-2022

Lab No: 01

Lab Task: Create a database containing following info for students.

1. Roll
2. Name
3. Semester
4. Major / Far subject
5. Obtained Marks

Problems:

1. Create database and table
2. Change a specific column name and data type
3. Add a new column named as log , set the values applicable and not applicable for the condition (<30)
4. Delete the students whose marks are below 30.

Objective:

The objective of these problems is to demonstrate the use of essential SQL operations, both Data Definition Language (DDL) and Data Manipulation Language (DML), to manage a student database. [1]Starting with the creation of a database and a table to store student information, the task highlights how to modify the database schema by renaming and changing the data type of a specific column. This involves altering the table structure and adding a new column named "Log," which stores the values 'Applicable' or 'Not Applicable' based on a condition (marks less than 30). Furthermore, the problem emphasizes the use of DML commands to update and manage data, including deleting records of students whose obtained marks fall below 30. By addressing these tasks, the exercise provides a comprehensive understanding of how to create, modify, and maintain a database effectively using SQL.

Problem Statement 1: Create Database and table.

Query:

Create Table:

```
Run SQL query/queries on server "127.0.0.1": ⓘ

1 CREATE TABLE student_data (
2   st_name VARCHAR,
3   st_roll INT,
4   st_semester VARCHAR,
5   major_subject VARCHAR,
6   obtained_marks INT );
```

Insert Information:

```
Run SQL query/queries on table student_info_table.student_data: ⓘ

1 INSERT INTO student_data (st_name, st_roll, st_semester,major_subject, obtained_marks)
2 VALUES ('Md.Kamal', 11, '1st', 'Mathematics', 28),
3         ('Jolly', 42, '2nd', 'Science', 75),
4         ('S.M.Michael ', 23, '1st', 'Literature', 45),
5         ('A.Koly ', 14, '2nd', 'History', 15),
6         ('M. Theroja', 52, '2nd', 'Science', 85);
7
```

Output :

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows:

Extra options

st_name	st_roll	st_semester	major_subject	obtained_marks
Md.Kamal	11	1st	Mathematics	28
Jolly	42	2nd	Science	75
S.M.Michael	23	1st	Literature	45
A.Koly	14	2nd	History	15
M. Theroja	52	2nd	Science	85

Problem Statement 2: Change a specific column name and datatype.

1. Change Column Name and Data Type Together

Query:

```
Run SQL query/queries on table student_info_table.student_data: ⓘ  
  
1 ALTER TABLE student_data  
2 CHANGE COLUMN obtained_marks cgpa INT(255);
```

Output:

Extra options				
st_name	st_roll	st_semester	major_subject	cgpa
Md.Kamal	11	1st	Mathematics	28
Jolly	42	2nd	Science	75
S.M.Michael	23	1st	Literature	45
A.Koly	14	2nd	History	15
M. Theroja	52	2nd	Science	85

2. Change Only Data Type (Without Renaming the Column)

Query:

```
Run SQL query/queries on table student_info_table.student_data: ⓘ  
  
1 ALTER TABLE student_data  
2 MODIFY COLUMN cgpa FLOAT;  
3
```

Problem Statement 3: Add a new column named as log . Set the value applicable and not applicable for the condition (<30).

Query:

```
Run SQL query/queries on table student_info_table.student_data: ⓘ
1
2 ALTER TABLE student_data
3 ADD Log VARCHAR(50);
4 UPDATE student_data
5 SET Log = CASE
6     WHEN obtained_marks >= 30 THEN 'Applicable'
7     ELSE 'Not Applicable'
8 END;
9
```

Output:

☐ Show all

Number of rows: 25

Filter rows:

Extra options

st_name	st_roll	st_semester	major_subject	obtained_marks	Log
Md Kamal	11	1st	Mathematics	28	Not Applicable
Jolly	42	2nd	Science	75	Applicable
S.M.Michael	23	1st	Literature	45	Applicable
A.Koly	14	2nd	History	15	Not Applicable
M. Theroja	52	2nd	Science	85	Applicable

☐ Show all

Number of rows: 25

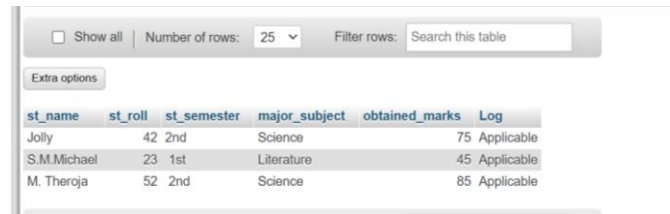
Filter rows:

Problem Statement 4: Delete the Students info. Whose marks are below 30.

Query:

```
Run SQL query/queries on table student_info_table.student_data: ⓘ
1 DELETE FROM student_data
2 WHERE obtained_marks < 30;
3
```

Output:

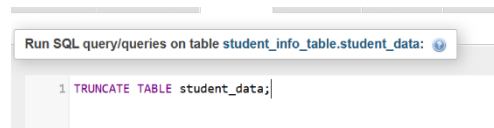


A screenshot of a database table view. At the top, there are controls: a checkbox for 'Show all', a 'Number of rows' dropdown set to '25', and a 'Filter rows' search box. Below these is an 'Extra options' button. The table itself has columns: st_name, st_roll, st_semester, major_subject, obtained_marks, and Log. It contains three rows of data.

st_name	st_roll	st_semester	major_subject	obtained_marks	Log
Jolly	42	2nd	Science	75	Applicable
S.M.Michael	23	1st	Literature	45	Applicable
M. Theroja	52	2nd	Science	85	Applicable

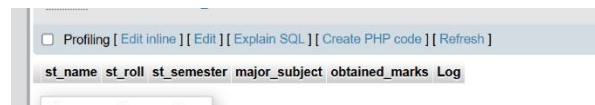
Truncate :

Query:



A screenshot of a SQL query editor. A toolbar at the top shows 'Run SQL query/queries on table student_info_table.student_data:'. Below the toolbar, a single line of SQL code is entered: `1 TRUNCATE TABLE student_data;`

Output:

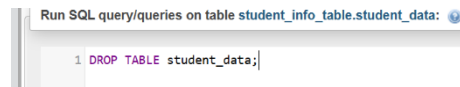


A screenshot of a database table view. At the top, there are controls: a checkbox for 'Profiling', and buttons for 'Edit inline', 'Edit', 'Explain SQL', 'Create PHP code', and 'Refresh'. Below these is a table header with columns: st_name, st_roll, st_semester, major_subject, obtained_marks, and Log. The table body is empty.

st_name	st_roll	st_semester	major_subject	obtained_marks	Log
---------	---------	-------------	---------------	----------------	-----

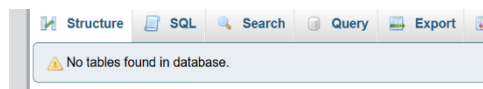
Drop:

Query:



A screenshot of a SQL query editor. A toolbar at the top shows 'Run SQL query/queries on table student_info_table.student_data:'. Below the toolbar, a single line of SQL code is entered: `1 DROP TABLE student_data;`

Output:



A screenshot of a database structure view. A toolbar at the top shows buttons for 'Structure', 'SQL', 'Search', 'Query', 'Export', and a red 'X' icon. Below the toolbar, a message box with a yellow warning triangle icon says: 'No tables found in database.'

Reference:

- [1] “What is DBMS (Database management system) ? | OVHcloud Worldwide.” Accessed: Sep. 22, 2024. [Online]. Available: <https://www.ovhcloud.com/en/learn/what-is-dbms/>