ORMEDIAN RESEARCH INSTITUTE

TOPIC:

DATABASE

SUBTOPIC: SQL FUNDAMENTALS (PART 5)

By:

ADEKOLA OLUWASEUN O.

DECEMBER 29, 2022

CONTENT COVERED

- *Referenced Tables*
- *Creating a New Database*
- *Getting List of Existing Database*
- *Creating a New Database Table*
- *Dropping a Database*
- *How to Back Up Any Database*
- *Creating Tables in SQL from Scratch and Copying of Data From One Table to Another.*
- *Dropping a Database Table*
- *Renaming a table in SQL*
- *Truncating a Database Table*
- *The Use of ALTER TABLE Statement*
- *Dropping/Deletion of Column From A Table*
- *Adding Column to Table*
- *Renaming/Changing of Column Name*
- *Column Modification using MODIFY statement*

IN THIS PRESNTATION, THE FOLLOWING TABLES WILL BE USED AS OUR REFERENCES AT EVERY LEVEL OF WRITING QUERIES

TABLENAME: ECE501

STUDENTID	SCORE	DEPARTMENT	LEVEL
160211012	65	ECE	500
160221014	89	MECH	500
160231044	89	CPE	500
160211059	89	ECE	500
160221088	53	MECH	500
160211014	88	ECE	500

TABLENAME: ECE507

STUDENTID	SCORE	CURRENT_CGPA	LEVEL	DEPARTMENT
160221015	89	4.67	500	ECE
160211012	89	4.67	500	ECE
160211014	83	4.6	500	MECH
160231044	81	4.6	500	CPE
160211059	89	4.63	500	ECE
160211088	89	4.6	500	ECE

Link to this implementation is available @ https://github.com/SafersTechnologies/SQL-Tutorials/blob/master/ece501.sql

Link to this implementation is available @ https://github.com/SafersTechnologies/SQL-Tutorials/blob/master/ece507.sql

HOW TO CREATE A NEW DATABASE

To create a new SQL database, the syntax is given as:

CREATE DATABASE databasename;

Let's take the database name of choice to be ***Ormedian.*** *Then our database creation query will look like:*

CREATE DATABASE Ormedian;

TO SEE THE LIST OF ALL DATABASES CREATED

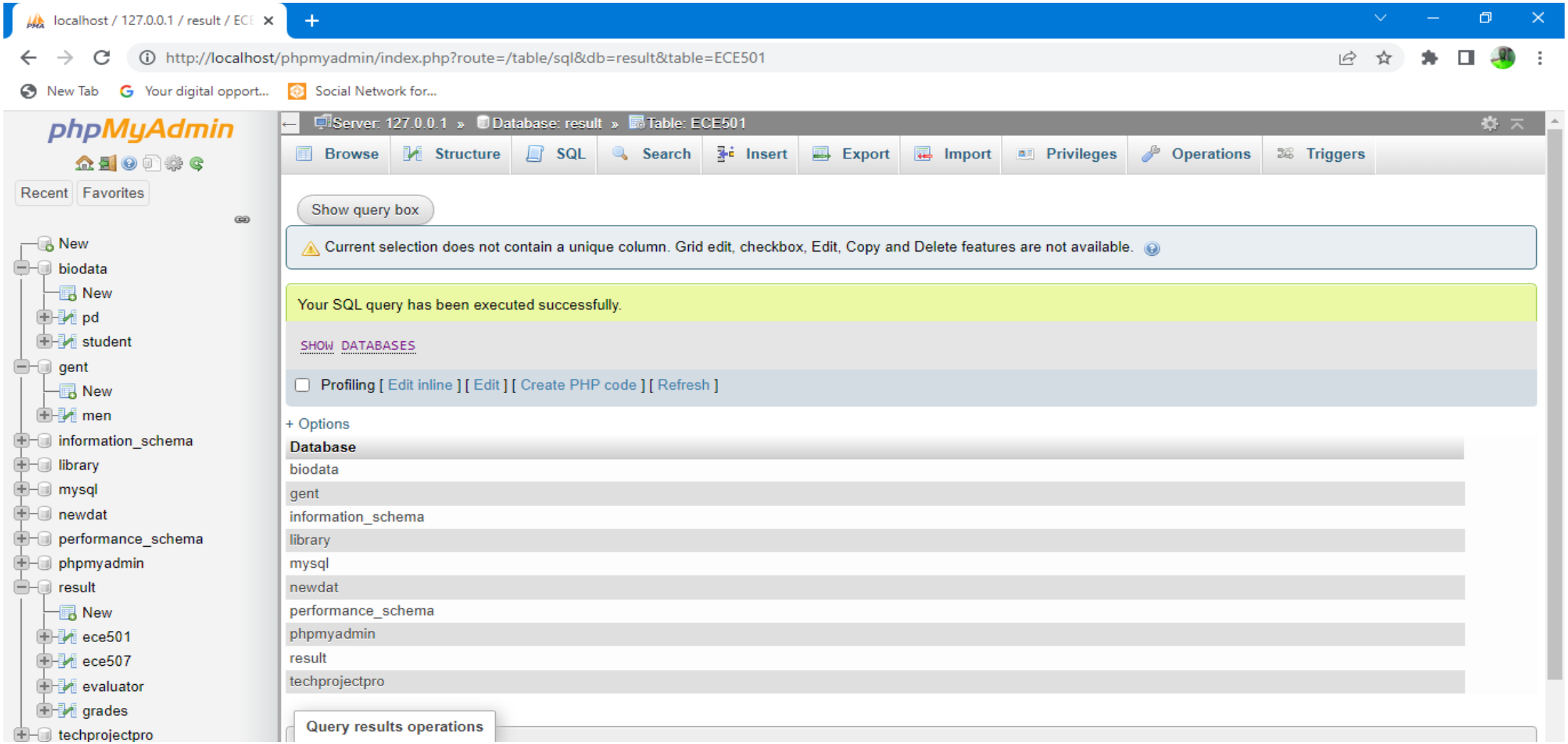
Whenever there is need for us to create a new database name, we might need to check for existing databases we already created to avoid conflict or duplication of database. Since duplication of database is not allowed, then we need to make sure data redundancy is completely eliminated.

SYNTAX TO CHECK FOR EXISTING DATABASES

SHOW DATABASES;

For example, databases existing in the server used in this presentation are shown in the figure display in the next slide

LIST OF EXISTING DATABASES SHOWN FROM THE QUERY



DROPPING A DATABASE

This is simply a way of deleting any database from an instance of SQL server and this at the same time deletes physical disk space occupied by that such database.

SYNTAX TO ACHIEVE THIS:

DROP DATABASE databasename;

Let us assume that we want to drop the Ormedian database that we previously created. Our Syntax will then become;

DROP DATABASE Ormedian;

HOW TO BACKUP DATABASE

To create back full back up of an existing file in SQL database, the BACKUP DATABASE statement is used. This could also be regarded as a complete copy that stores all database objects such as views, indexes, tables, functions procedures etc. Backup is an integral part of the database which allows database content to be recovered.

Backup in database could be of two different forms:

I. *Full backup of an existing file*

II. *SQL BACKUP with differential statement: this backs up only the part of the database that has changed since the full database backup.*

SYNTAX USED TO BACK UP DATABASE

BACKUP DATABASE databasename to DISK = "filepath";

Assuming that I need to save a backup for the content of the Ormedian database I created earlier on my laptop desktop area. Then the backup syntax can further be modified to:

BACKUP DATABASE Ormedian TO DISK="C:\Users\SHOPINVERSE\Desktop";

If I am only interested in the changes made since the last full database backup, then the syntax will take this form:

BACKUP DATABASE Ormedian TO DISK="C:\Users\SHOPINVERSE\Desktop" WITH DIFFERENTIAL;

CREATION OF TABLES IN SQL

When creating a table in a database system, it is worth noting that we will need certain data to be recorded into this table. Each data must be filled into rows they are will be separated into columns. Therefore the name of each column must be specified and data types for the values they are to accommodate, and also we must specify the length of character and limit that it can accommodate.

SYNTAX FOR TABLE CREATION IN SQL DATABASE

CREATE TABLE tablename (Column1 datatype, Column2 datatype, Column3 datatype, Column4 datatype, …….);

Assuming we need to create a simple table called student_details to hold the Studentid, FirstName, Level, and Age of several students in an institution.

CREATE TABLE student_details (Studentid int(5), FirstName varchar(25), Level int(4), Age int(5));

TABLE CREATION

……Continuation

There are several ways in which a table can be created. We can create a new table without making any reference to other tables. But there are some cases whereby we will need exact data from another table to form a new table. Instead of creating new table and insert this values into columns one after the other manually, this will really take a lot of time. For this reason, we need to look for a way to copy the content needed from an existing table to the new table we intend to create.

SYNTAX TO ACHIEVE THIS:

CREATE TABLE new_table_name AS SELECT column1, column2, …. FROM old_table;

OR

CREATE TABLE new_table_name SELECT column1, column2, …. FROM old_table;

Implementation of the technique to copy from one table to another

SCENARIO

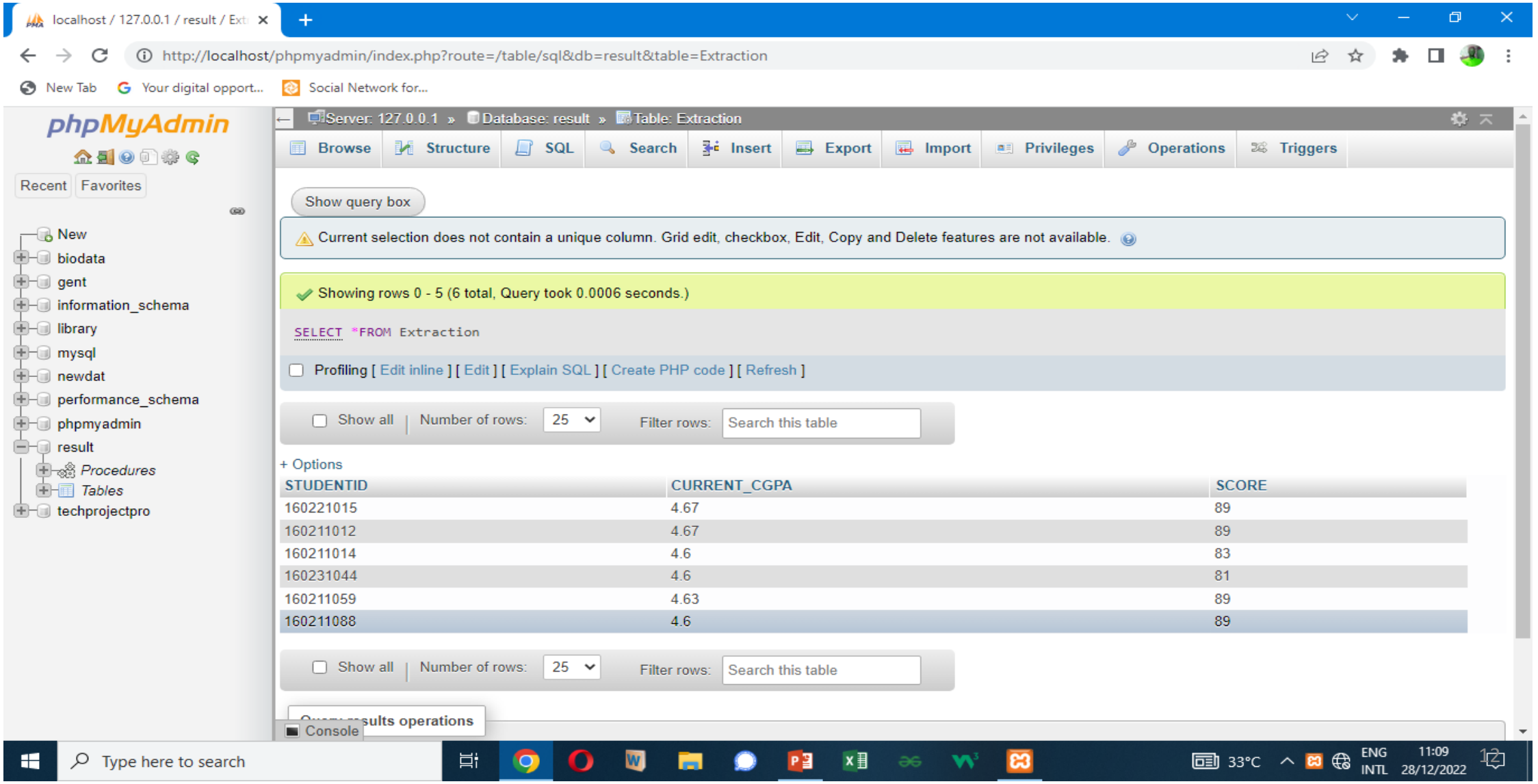
Let us assume that we need to copy STUDENTID, SCORE, and CURRENT_CGPA from ECE507 table to another table called Extraction.

CODE EXAMPLE

```
CREATE TABLE Extraction AS SELECT STUDENTID, CURRENT_CGPA, SCORE FROM ECE507;
```

Link to this implementation is available @ https://github.com/SafersTechnologies/SQL-Tutorials/blob/master/Extraction.sql

COPYING DATA FROM ONE TABLE TO NEWLY CREATED TABLE



DROPPING A DATABASE TABLE

DROP statement is used as a keyword for initiating the deletion of a database table in SQL. To drop a database table is the same as saying we want to delete a database table entirely.

SYNTAX

DROP TABLE Tablename;

SCENARIO

Let us say that we need to completely delete a table named Extraction from the database. Then the syntax becomes:

DROP TABLE Extraction;

TRUNCATING A DATABASE TABLE

TRUNCATE statement is basically used in SQL to ensure that data stored inside a table are deleted but the table itself is not deleted. Unlike in the case of DROP where the entire database table is deleted together with its entire content. Table truncation in SQL is just a way of getting all records stored inside a table deleted without the table itself being deleted.

SYNTAX FOR THIS:

TRUNCATE TABLE Tablename;

Let us say for example we have some records stored in a given table, the name of the table is GRADES. If we need to delete these records without the table itself being deleted.

TRUNCATE TABLE GRADES;

RENAMING A TABLE IN SQL

To rename a table in SQL, we use the RENAME statement with ALTER TABLE statement to achieve that.

SYNTAX

ALTER TABLE Table_name RENAME to New_Tablename;

SCENARIO

Let us say we need to rename the table called GRADES to FINAL_GRADES;

ALTER TABLE GRADES RENAME TO FINAL_GRADES;

ALTER TABLE STATEMENT

ALTER TABLE *statement enables us to add, modify or delete existing columns in a database table. It has other functionalities to add and drop constraints on existing database. It can also be used to rename column or a table.*

Things we can use alter statement to achieve in SQL

- *To delete columns from the database table*
- *To add columns to the database table*
- *To rename or change column name*
- *To modify datatype*

DELETION OF COLUMN FROM A TABLE

To delete a column from any table in a database, we need to specify which table exactly in our database we want to alter its content. This is very important because a single database can house as many tables as possible. Hence we have to specify which table we want to work on.

SYNTAX TO DROP A COLUMN FROM A PARTICULAR TABLE

ALTER TABLE Table_name DROP COLUMN Column_name;

SCENARIO

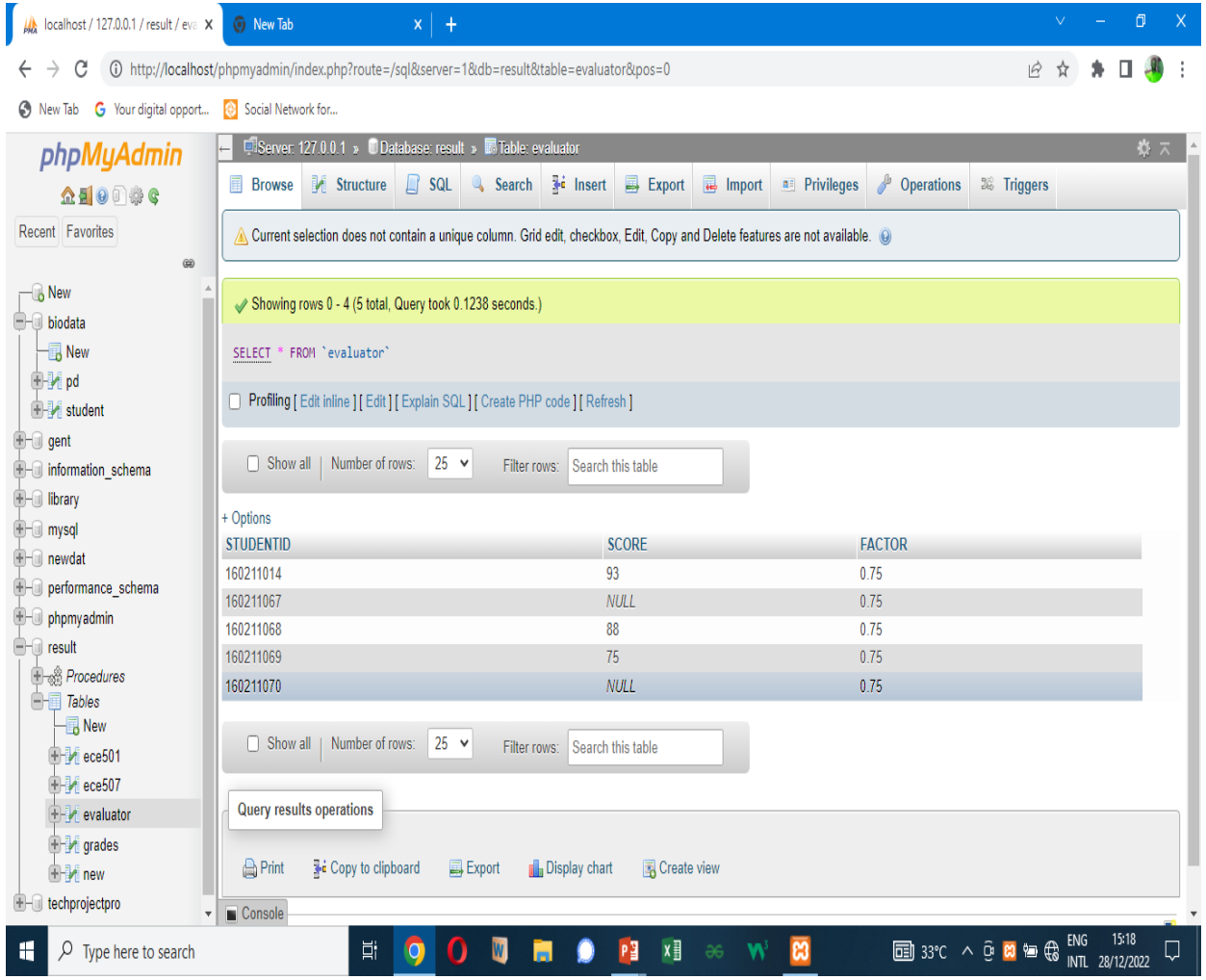
I want to write SQL query that will drop a column named FACTOR from a table called Evaluator.

CODE EXAMPLE

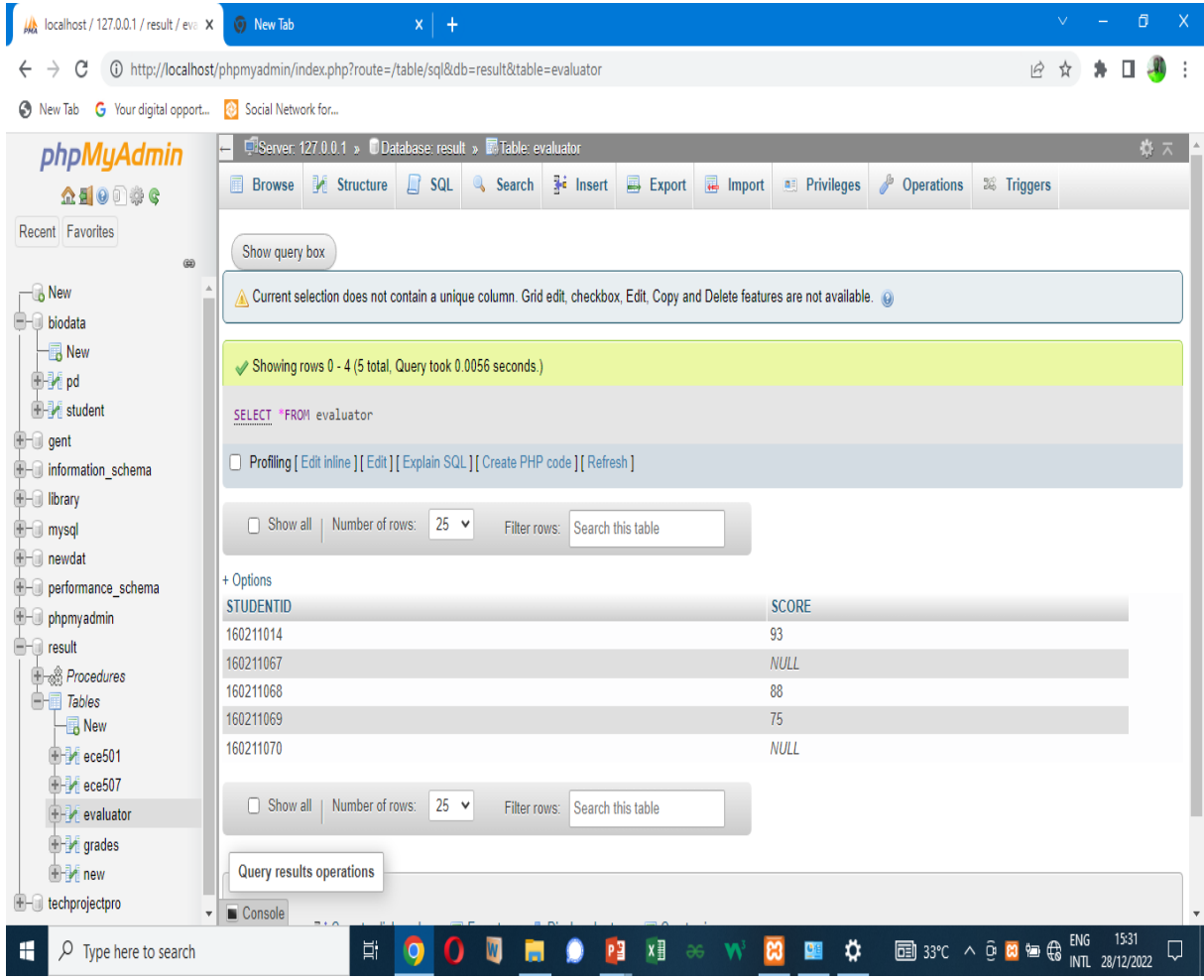
ALTER TABLE Evaluator DROP COLUMN FACTOR;

SAMPLE OF TABLE WHOSE COLUMN IS DROPPED

BEFORE



AFTER



HOW TO ADD COLUMNS TO A TABLE IN SQL

Adding new columns could become a necessity in some cases where initially created database table had not included other attributes of a record. It could be that a table called ***student_details*** *ought to have the following attributes (FirstName, LastName, StudentID, Age, Sex) but during the process of creating the table, the Age attribute(column) was omitted. While this has happened, the next thing to do is to look for a way to get such column included into the table. While doing this, character length limit and the datatype must be specified. To get this done, the syntax given below will help achieve that.*

SYNTAX

```
ALTER TABLE Table_name ADD COLUMN Column_name;
```

CODE EXAMPLE FOR THE SCENARIO

```
ALTER TABLE student_details ADD COLUMN Age Int(4);
```

RENAMING/CHANGING A COLUMN_NAME IN A DATABASE TABLE

*Column must be given a very descriptive name and must denote what it is meant to represent. For example, it makes sense to use the word **GENDER** than **TYPE_OF_HUMAN** to describe whether a person is a male or female. To also get a column renamed, it could be that there was a typographical error while naming a column during the table creation process, and we need to give such column its proper name.*

It is pretty easy to use the CHANGE statement together with datatype keyword to achieve that.

SYNTAX

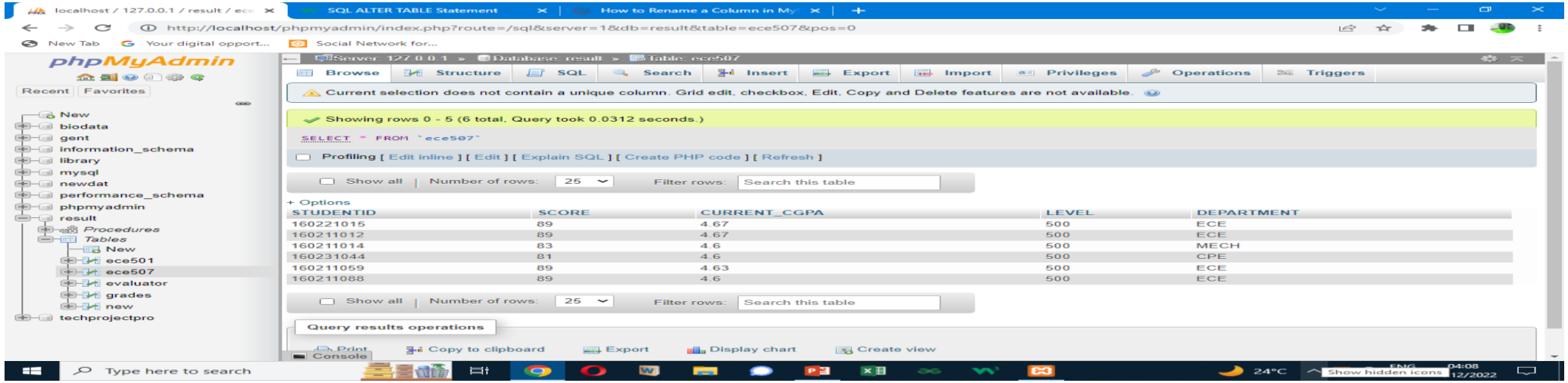
ALTER TABLE Tablename CHANGE old_column new_column datatype;

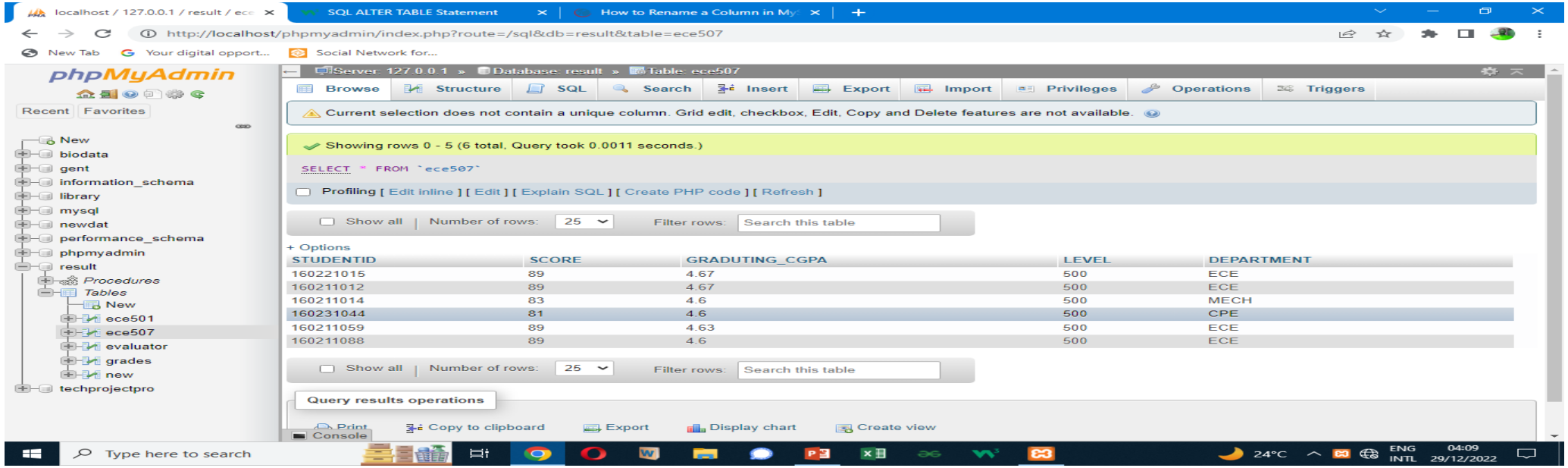
SCENARIO

Say we need to change a column name in ECE507 table. For example let's say the CURRENT_CGPA column is to be renamed to GRADUATING_CGPA.

ALTER TABLE ECE507 CHANGE CURRENT_CGPA GRADUATING_CGPA float (14);

IMPLEMENTATION RESULT OF COLUMN RENAMING (BEFORE AND AFTER)





COLUMN MODIFICATION

Modification of columns could be required at some points of editing table records and its fields. Unlike the change/rename functionality of SQL which allows us to change one field name to another. The MODIFY COLUMN or MODIFY enables us to change datatypes without the need to rename the column.

SYNTAX

ALTER TABLE Table_name MODIFY COLUMN column_name new_datatype;

SCENARIO

Let's say I need to change the datatype of the SCORE field of ECE507 from Integer to float.

CODE EXAMPLE

ALTER TABLE ECE507 MODIFY COLUMN SCORE FLOAT(4);

OR

ALTER TABLE ECE507 MODIFY SCORE FLOAT(4);

THANKS FOR VIEWING

NOTE: ***Subsequent topics under SQL will be discussed in Part 6***

Also, all Implementation as far as this presentation is concerned are in line with MySQL Syntax. All have been tested on XAMPP Server.

THE NEXT PART WILL REVOLVE AROUND DEEPER UNDERSTANDING OF HOW TO BUILD SQL DATABASES