What is Database?

The database is an organized collection of data so that it can be easily managed and accessible.

A database is an organized collection of structured information or data, typically stored electronically in a computer system.

The database is a collection of inter-related data which is used to retrieve, insert, and delete the data efficiently. It is also used to organize the data in the form of a table, schema, views, and report etc.



For example: The college database organizes the data about the admin, students, libraries, faculty..etc Using the database, you can easily retrieve, insert, and delete the information.

College database:

Students

Students

Alumnus

New admission

Admission Inquiry

Dropout

Faculty

Former faculties

Faculties

Hostel

Hostel Block

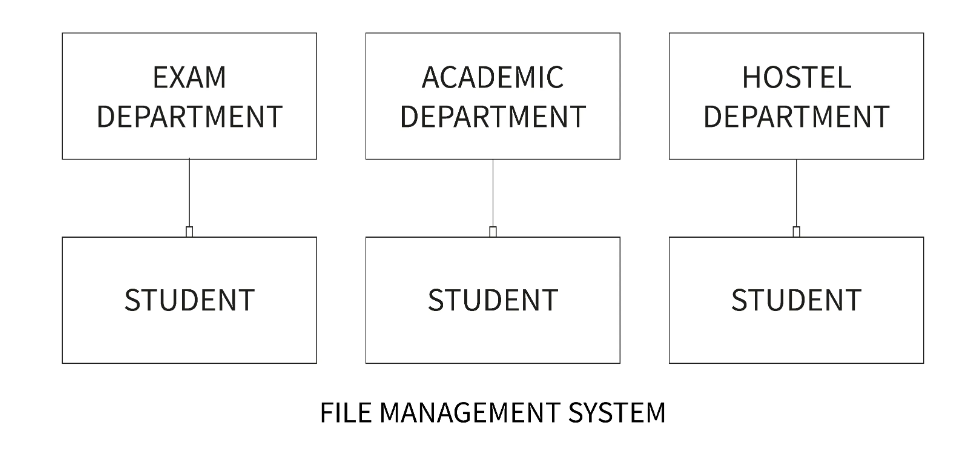
Hostel Staff

Traditional File based approach vs Database approach

* File base approach/File Management system used to manage data needed for a specific use case or application. Each use stores separate data for the application even if the same data stored by another user.
* For example: Like College, there are multiple department such as admin, hostel, library, exam each departments are maintaining student details separately

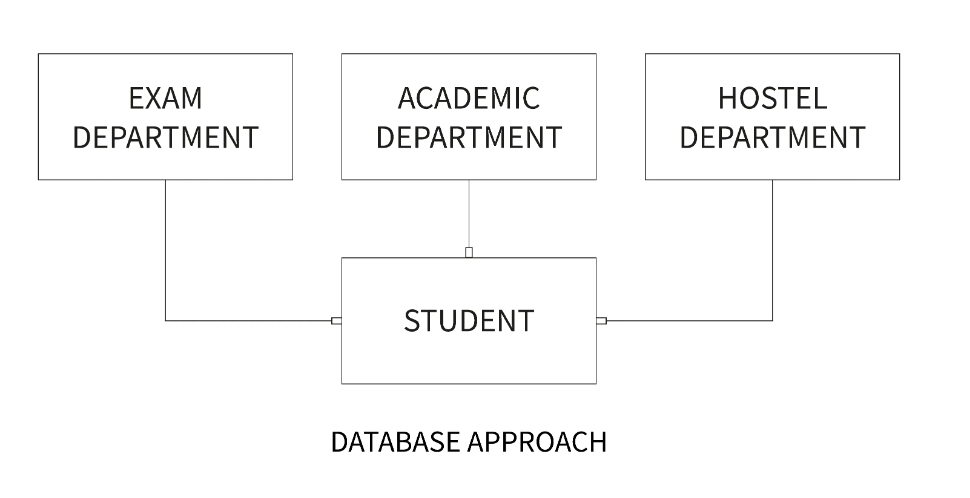
If there are 100 students and 5 departments then will be storing 5x100 = 500 records

* Major drawback of file-based approach is repetition of data and wastage of resources.



Database approach:

* A database is used for storing and maintaining the data where data defined once and stored in single location, it available for multiple users or departments.



Characteristics of Database approach:

1. Self-Describing Nature of a Database System ie. database store data in a specified format so it’s easy to understand and maintaining data properly. ie. **USN**, **Student Name**
2. Insulation between **Programs and Data**, and **Data Abstraction** ie. Database system is works separately modifying columns/records doesn’t have issue with the application running with database, it will get latest data once application get refreshed.
3. Sharing of Knowledge and Multi-user Transaction Processing
4. Support of Multiple Views of the Data
5. Security
6. Supports Query Language to perform database operation effectively.

**What Is Database Management?**

A database management system (DBMS) is a software tool that helps organize, store and retrieve data from a database. It involves a number of functions that collectively work together to ensure that the data is accurate, available and accessible.

A database management system consists of three main elements:

1. A physical database that contains the data
2. A database engine that helps to access the data and modify its contents.
3. A database scheme which provides the logical structure of the data stored in the database.

Types of Database Management System

1. **Relational** database management system
2. **Distributed** database management system
3. **Network** database management system
4. **Object-oriented** database management system
5. **Hierarchical** database management system

Examples of DBMS:



**MySQL**

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company.

RDBMS

Relational Database Management System

A relational database management system or RDBMS is a database system that stores and fetches data in the form of tables.

How Does a Relational Database Work?

* Relational Databases use tables to store data about related objects. Each column contains data attributes, whereas each row holds a record of unique data known as Key.
* Relational Databases or RDBMS are managed using SQL short form for Structured Query Language. Therefore SQL codes are used to retrieve information from relational databases by doing various interactive operations like JOIN, TRUNCATE, etc.

**How is a Relational Database Model Structured?**

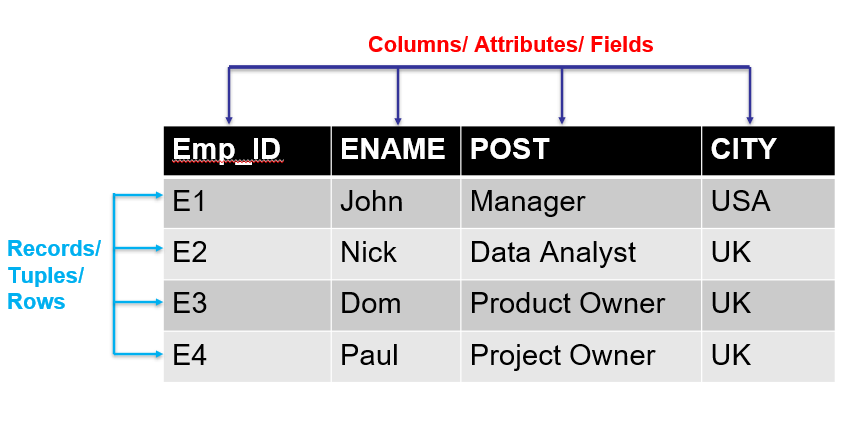
A relational database model represents how data is stored in tables, which helps form relations between data values.

Consider a relational table named EMPLOYEE with attributes EMP\_ID, NAME, ADDRESS, AGE.

**EMPLOYEE TABLE:**

| **EMP\_ID** | **NAME** | **ADDRESS** | **AGE** |
| --- | --- | --- | --- |
| 1 | Rahul | Bengaluru | 23 |
| 2 | Suman | Kolkata | 25 |
| 3 | Raj | Delhi | 30 |

Some important terminologies that you should keep in mind while forming a relational database.



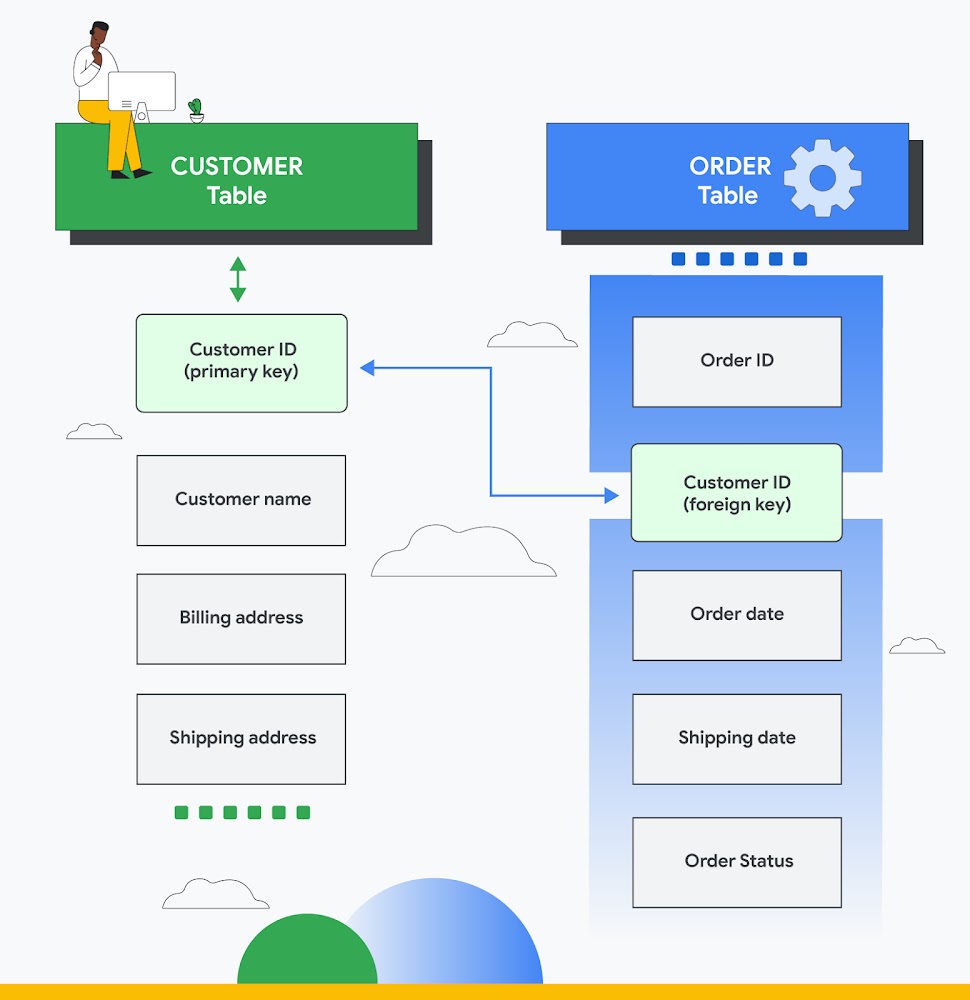
**Attribute**: Attributes are properties that define the relational database. Eg: EMP\_ID, NAME etc

**Relation Schema**: A relational schema defines its relation with other attributes altogether. E.g., EMPLOYEE (EMP\_ID, NAME, ADDRESS, AGE)

**Degree**: Degree is defined by several attributes we have in a relational table. E.g., The degree of the EMPLOYEE table is 4

**Cardinality**: Cardinality is defined by the number of tuples in a relation. E.g., The cardinality of the EMPLOYEE table is 3.

**NULL Values**: The values or data which are unknown are kept as NULL.



**SQL**

SQL is a standard programming language used to operate Relational Databases and carry out various operations such as inserting, manipulating, updating, and retrieving data from relational databases.

SQL is not a database system, but it is a query language.

SQL is a short-form of the structured query language, and it is pronounced as S-Q-L or sometimes as See-Quell.

This database language is mainly designed for maintaining the data in relational database management systems.

It is a special tool used by data professionals for handling structured data (data which is stored in the form of tables)

**Why SQL**

* RDBMS only understand SQL command and instruction to perform any kind of operation.

**Features of SQL**

* SQL is used to access data within the relational database.
* SQL is very fast in extracting large amounts of data very efficiently.
* SQL is flexible as it works with multiple database systems from Oracle, IBM, Microsoft, etc.
* SQL helps you manage databases without knowing a lot of coding.

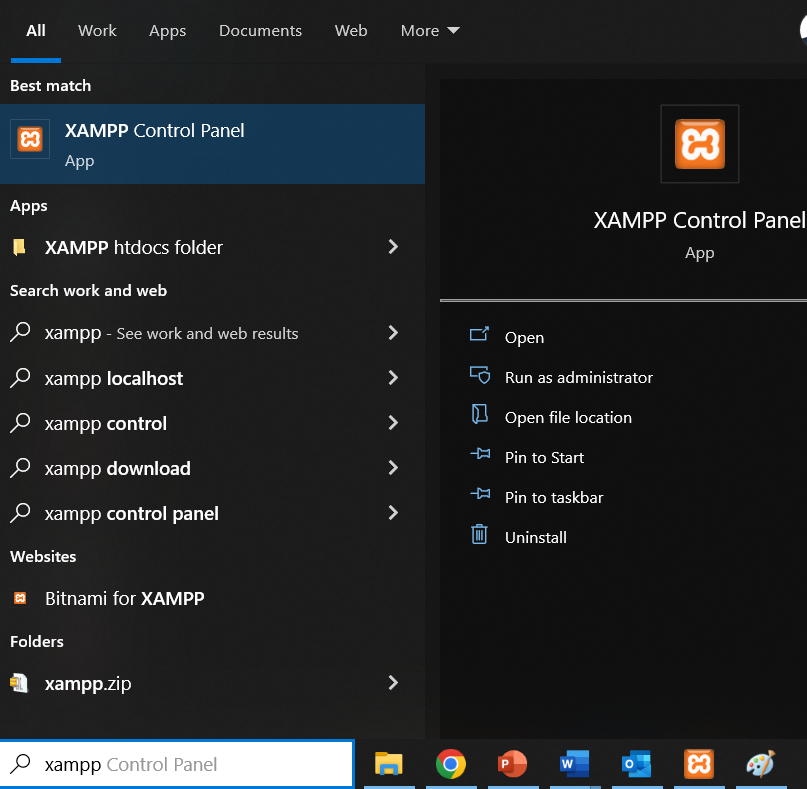
**Some SQL Commands**

The SQL commands help in creating and managing the database. The most common SQL commands which are highly used are mentioned below:

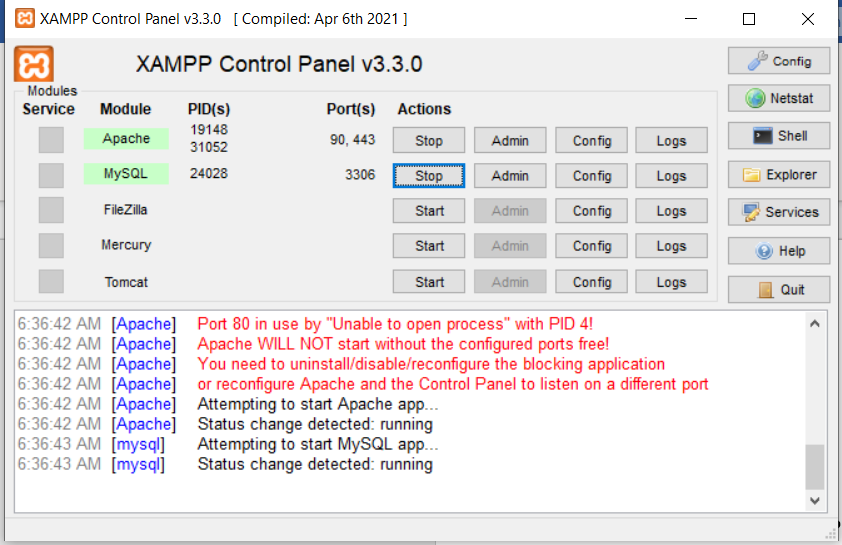
1. CREATE command
2. UPDATE command
3. DELETE command
4. SELECT command
5. DROP command
6. INSERT command

**Working with MYSQL (XAMPP phpMyAdmin)**

Start XAMPP application from all programs in Windows laptop



Start **Apache** & **MySQL** services on XAMPP control panel

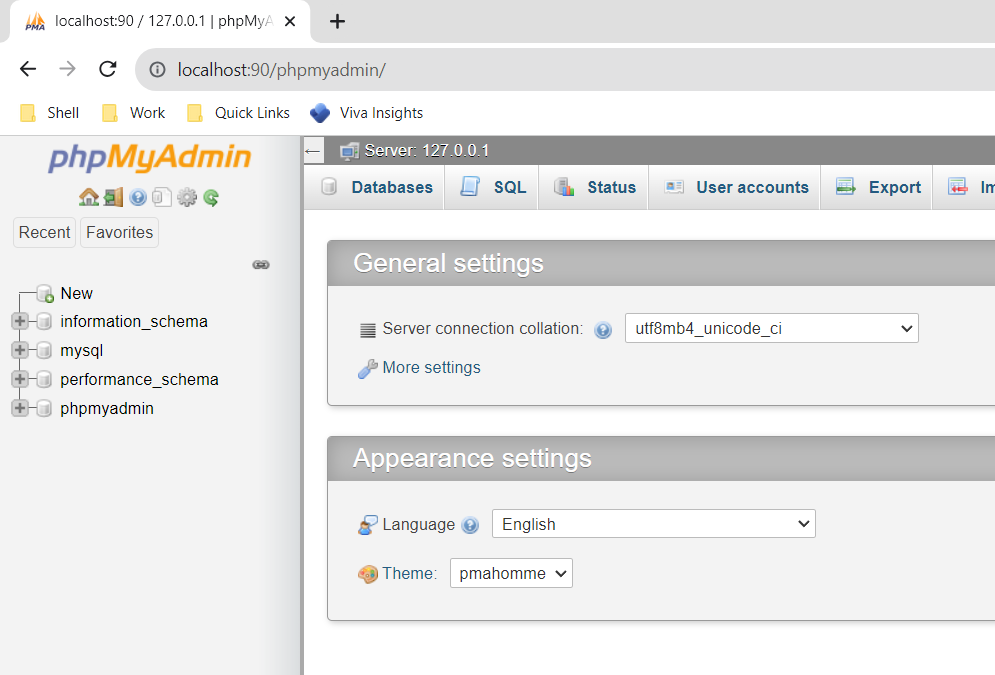


Once both services are running click on **Admin** button of MySQL services. It will take you on web browser with URL like

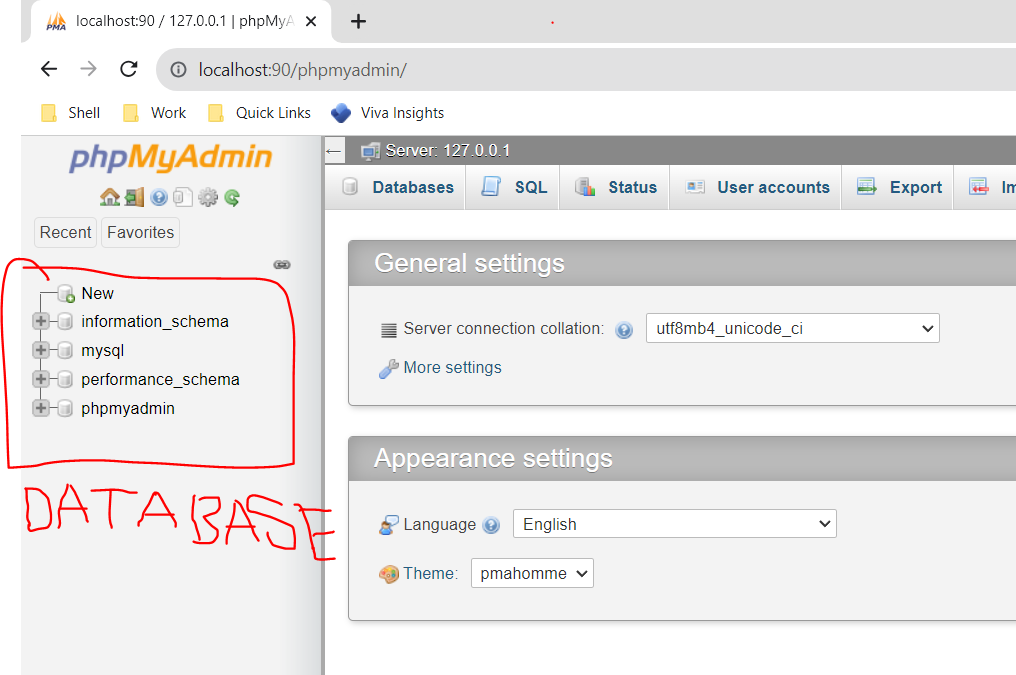
<http://localhost/phpmyadmin>

<http://localhost:8080/phpmyadmin>

<http://localhost:90/phpmyadmin>

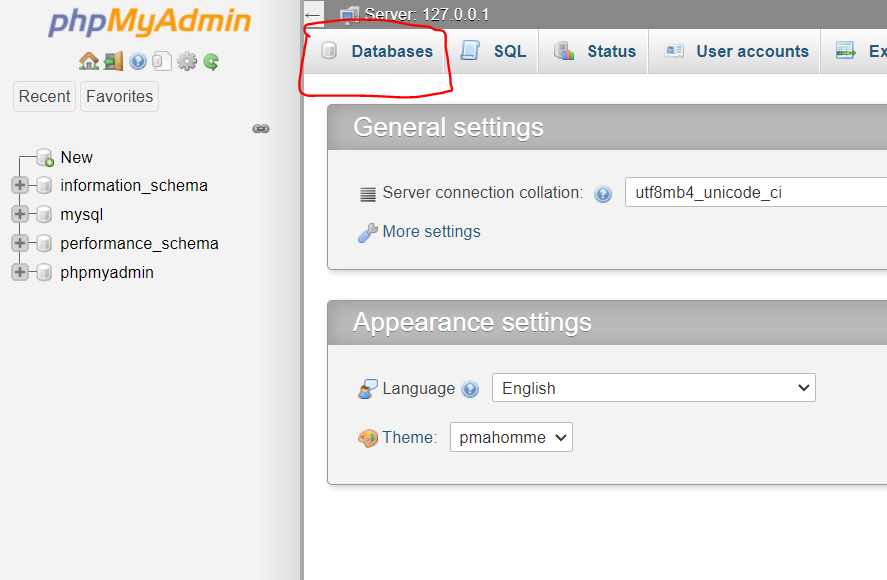


On left panel it contains all the database available on MySQL server.

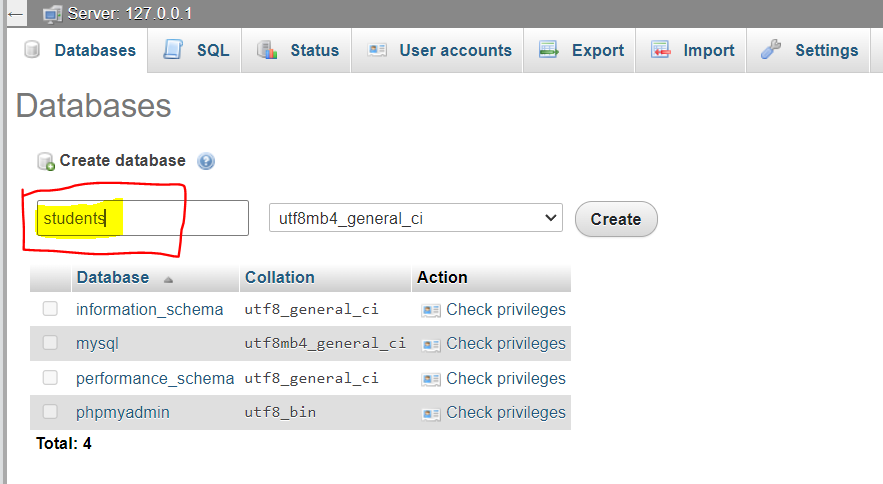


Creation of Database:

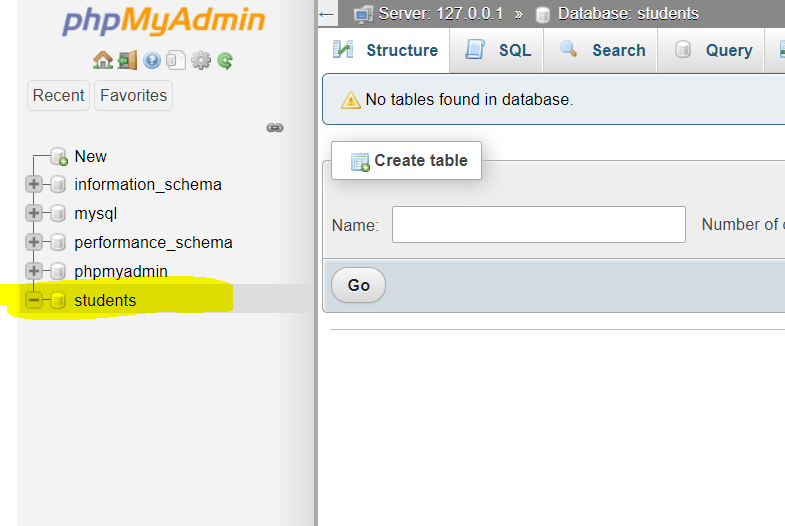
Click on database on top row of phpMyAdmin



Type database name and click on **Create** button

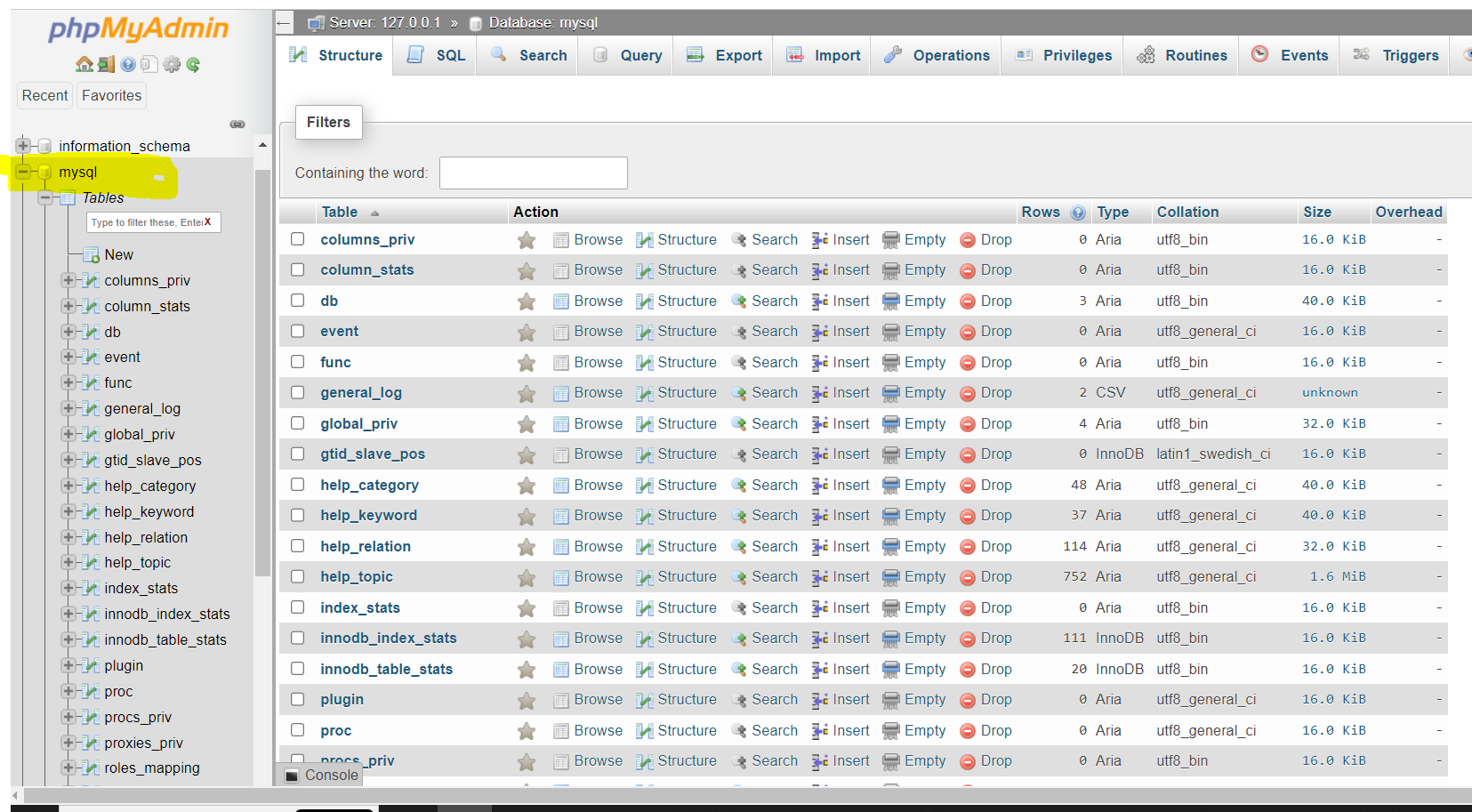


Once database created it will show in left panel windows.



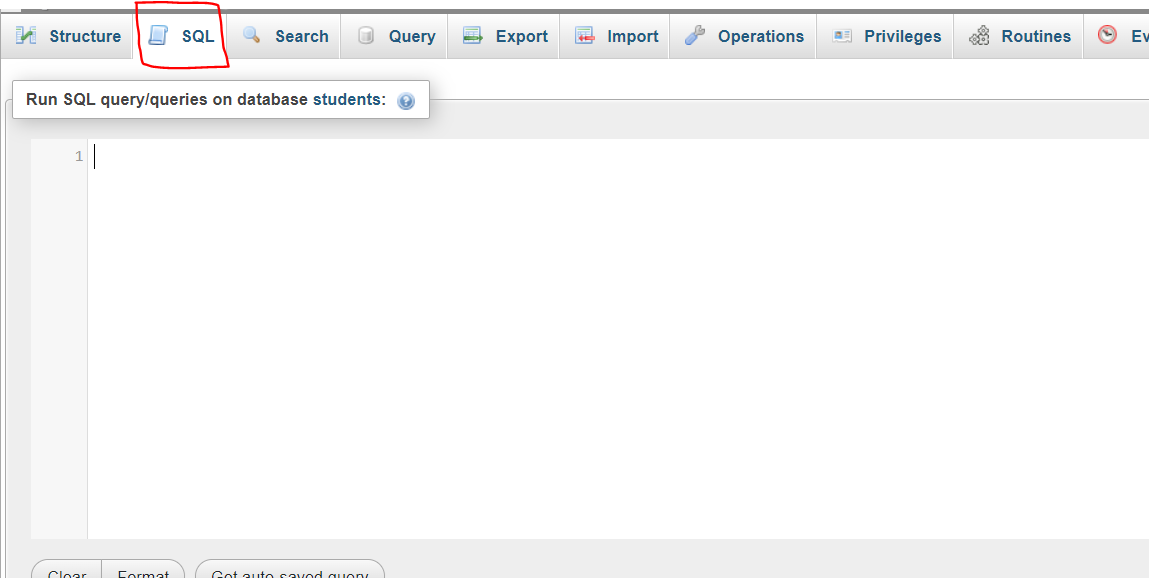
Selecting Databases:

Clicking on database name it will get selected for data manipulation, and display lists of tables available in selected database.

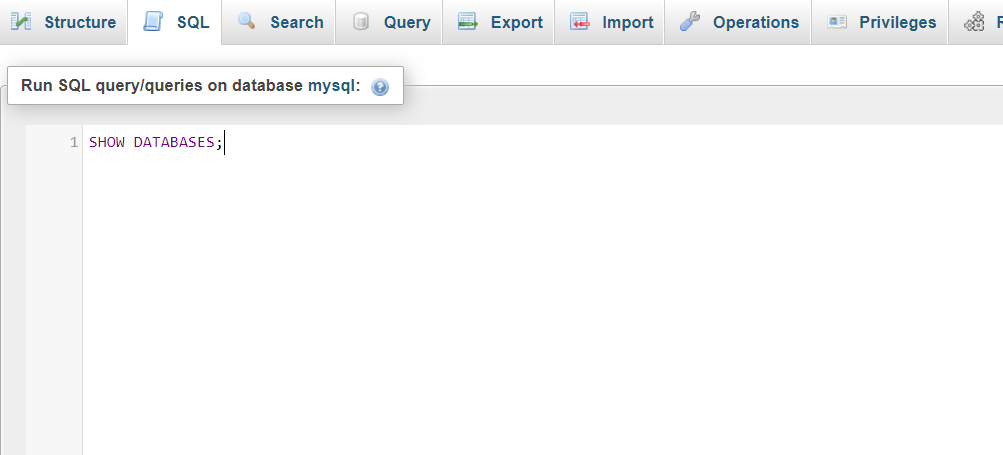


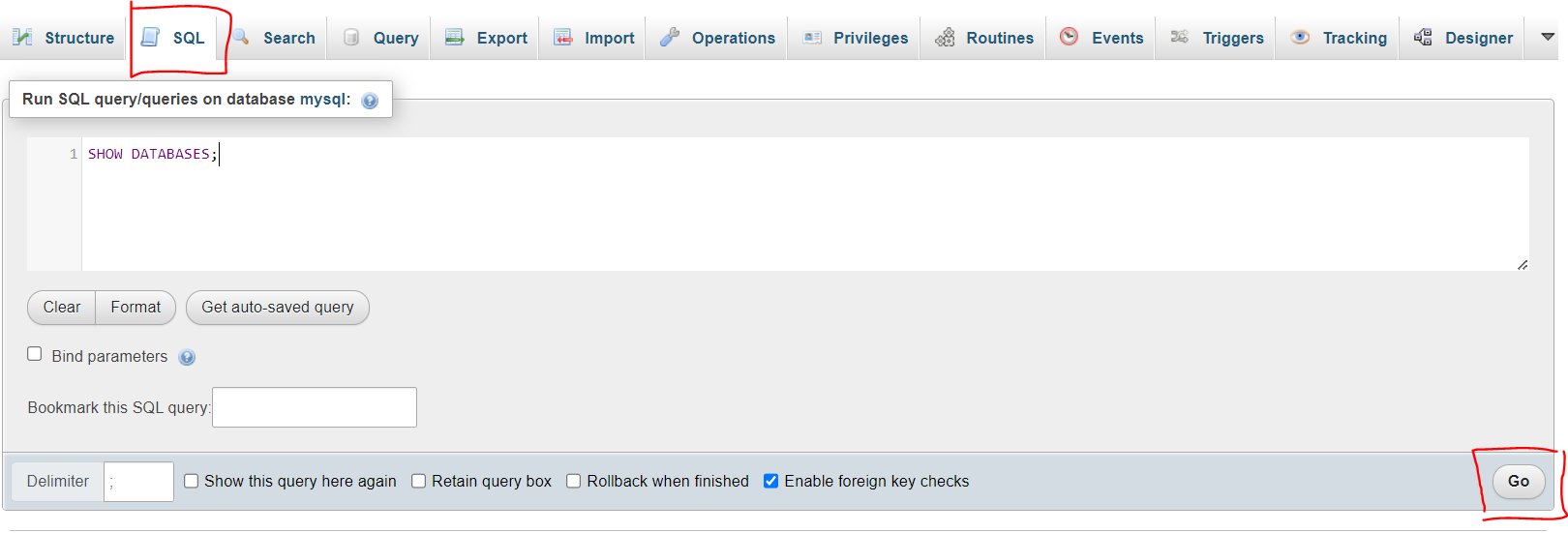
SQL Windows :

On top row of phpMyAdmin SQL button is available, click on SQL button gives SQL windows to write SQL commands / statements/ instructions.

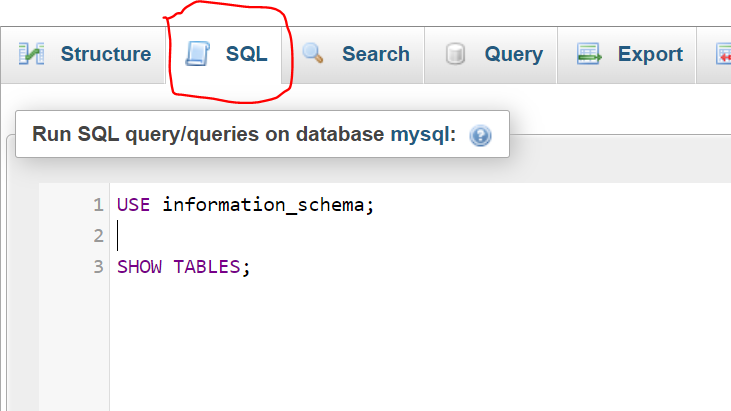


Statement execution:

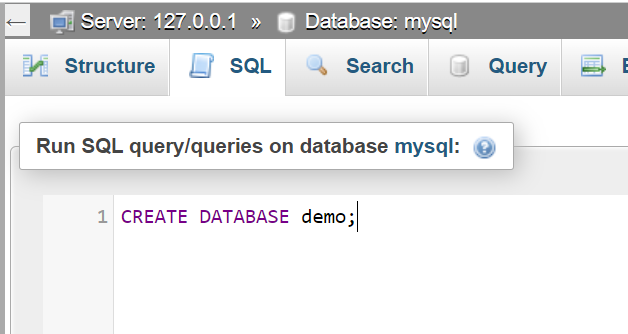




Selecting database using USE statement

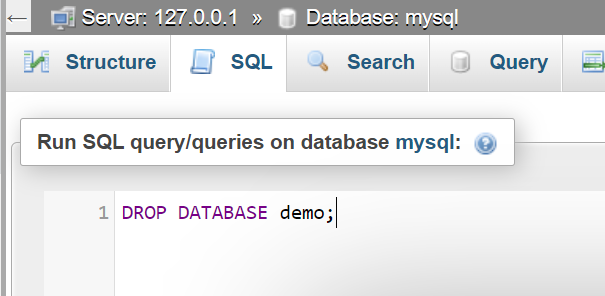


Creating new database using SQL statement

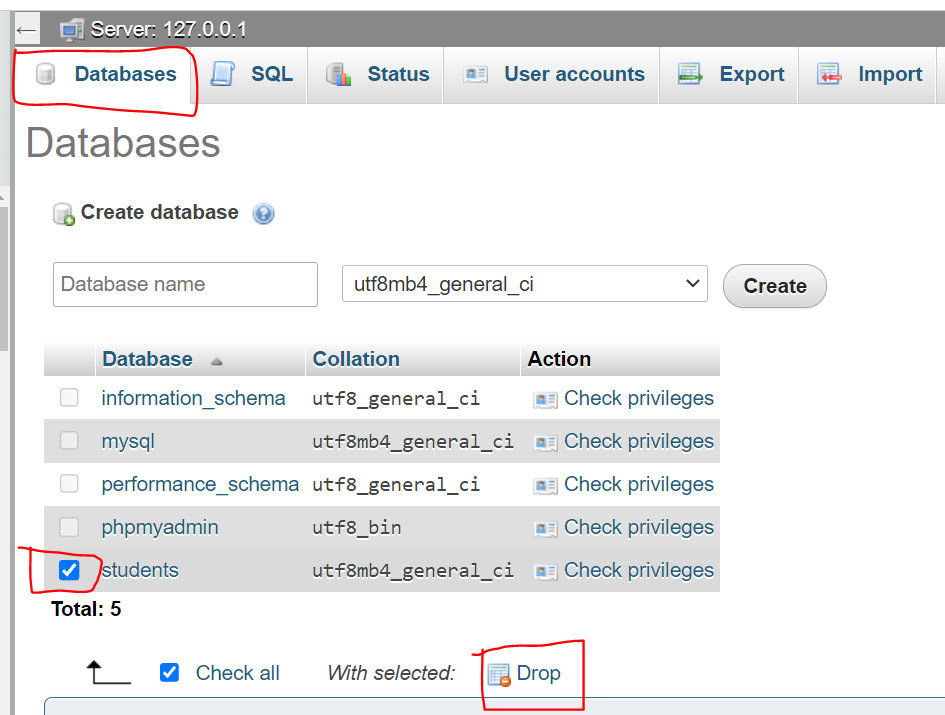
Create new database **demo** in mysql 

Deleting database using SQL statement

Delete **demo** database from mysql



Delete student database from Navigation bar



Recap:

* Selecting database using SQL command and Navigation bar
* Creating database using SQL command and Navigation bar
* Deleting database using SQL command and Navigation bar

Exercises:

Perform database exercises with below database

1. Create EMPLOYEE database using SQL Command, write all the commands in notebook and then execute in SQL windows.
2. LIST ALL the Database available in MySQL
3. CREATE EMPLOYEE statement
4. LIST ALL Database available in MySQL
5. USE EMPLOYEE statement
6. LIST ALL TABLE present in EMPLOYEE database
7. Create MOVIE database using Navigation bar.
8. After refreshing page check all the database in left window
9. CREATE MOVIE database
10. After refreshing page check all the database in left window
11. SELECT MOVIE database
12. Check are there any tables with MOVIE database
13. SELECT phpmyadmin database from left panel
14. Check are there any tables with phpmyadmin