

Beginners Guide



LEARN MySQL

for Absolute Beginners

First Edition

PHPBOOTCAMP.COM

EASY TO PRACTISE CODE SAMPLES

About this Book

MYSQL is a Database Query language that allows web developers to access MySQL database using PHP Libraries. MySQL is database used to store data in table format. This book will help you understand the basics of SQL Language and how to put it in practice to build Websites.

Audience

This tutorial has been designed to meet the requirements of all those readers who are keen to learn the basics of MySQL.

Prerequisites

This book assumes you have no prior knowledge on Programming knowledge and assume you are at a beginner level.

How to use this Book

This book contains SQL Language Basics, Exercises and Examples which are part of the PHP Bootcamp Program. This bootcamp has helped many students to become PHP Full Stack Web Developer in Just 30 days.

[>>Check out more about this program here...](#)

[Become PHP Full Stack Web Developer in Just 30 Days](#)

Copyright & Disclaimer

© Copyright 2019 by PHPBootcamp.com.

All the content and graphics published in this e-book are the property of PHPBootcamp.com. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. PHPBootcamp.com provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@phpbootcamp.com.

Found Typos & Broken Link

If you found any typos and broken links in the course or book, then please let me know using this email address.

contact@phpbootcamp.com

Support

You can reach me for technical discussions and other support related queries from here.

support@phpbootcamp.com

Thanks for your support!

[Become PHP Full Stack Web Developer in Just 30 Days](#)

Free Courses

	<u>Learn HTML in 1 Hour</u>
	<u>Learn CSS in 1 Hour</u>
	<u>Learn JAVASCRIPT in 1 Hour</u>

[Become PHP Full Stack Web Developer in Just 30 Days](#)

Table of Contents

About this Book	1
Audience	1
Prerequisites	1
How to use this Book	1
Copyright & Disclaimer	2
Found Typos & Broken Link	2
Support	2
1 MySQL Basics	7
1.1 Introduction to Databases and SQL	7
1.2 What is MySQL?.....	10
1.3 What is phpMyAdmin?	12
1.4 Datatypes MySQL	15
1.5 Keywords MySQL.....	16
2 phpMyAdmin	18
2.1 Login and Logout phpMyAdmin	18
2.2 Create Users.....	20
2.3 Create Database and Table.....	23
2.4 Run Simple SQL Statements	29
3 MySQL Statements	35
3.1 Create a Table	35
3.2 Drop a Table.....	35
3.3 INSERT Statements.....	38

3.4 SELECT Statements.....	38
3.5 Clause WHERE, LIMIT	40
3.6 Operators IS NULL, LIKE, ORDER BY.....	41
3.7 UPDATE Statements.....	43
3.8 DELETE Statements	44
4 mySQL & PHP with PDO	47
4.1 Database Connectivity	47
4.2 Simple Query from PHP to mySQL.....	48
4.3 Form to Add, Edit, Update and Delete	52
5 mySQL & PHP with mysqli.....	55
5.1 Database Connectivity	55
5.2 Simple Query from PHP to mySQL.....	56
5.3 Form to Add, Edit, Update and Delete	60

1. MySQL BASICS

1 MySQL Basics

1.1 Introduction to Databases and SQL

What is a Database?

A Database is a structured ways of storing the data on your computer so that it can be easily searched, managed and updated.

Data stored in a file are not easy to search because it is not properly organized. This is solved by using Database Software.

Database software helps to store the data in such a way that it can be retrieved faster. Even Database software has the capacity to hold large amounts of data.

How Data are stored in Database?

Data in the database are stored in one or more tables. Each table will have data organized in row and column format.

Sample Marks Table will look like this.

ID	SUBJECT	MARKS
1	MATHS	98
2	SCIENCE	56
3	ENGLISH	45
4	SOCIAL	22
5	COMPUTER	19

Table Name: Marks

ID, SUBJECT and MARKS are called as Columns.
Value on each line are called as Rows.

What is Relational Database?

You can create relation between tables and avoid duplicating the data by using references between the tables.

Student ID Reference is used in Marks Table to make a relation between the tables.

RELATION		
ID	REFKEY	MARKS
1	101	87
2	201	79
3	322	55
4	411	67
5	533	78

ID	NAME	CLASS
101	JOHN	A
201	ELVIN	A
322	AMIT	B
411	JULIE	B
533	AMAR	A

Marks

STUDENT

Relationship of tables also helps to break a bigger table structure into small tables and link them.

What is SQL?

SQL Stands for “Structured Query Language” it is a language used to access the data in the database.

SQL is pronounced as ‘S-Q-L’ or ‘sequel’.

SQL Language is predefined with keywords that you can use to do the following things:

[Become PHP Full Stack Web Developer in Just 30 Days](#)

Create a Table

Delete a Table

Search Table with Conditions

Insert Rows

Update Rows

Delete Rows

SQL language can be used to perform such actions on the database.

Example:

To delete a table you can say:

DROP TABLE <tablename>

Usage of Database:

Instead of storing the data in files and access them. All the data are stored in the Database.

In a Website you can store the following things:

User Information

Product Information

Product Pricing

Orders

Invoice

Enquiry

Contacts

and many more

Every website uses Database to store its information because it is easy and faster to store and access it.

[Become PHP Full Stack Web Developer in Just 30 Days](#)

1.2 What is MySQL?

MySQL is a open source relational database management system which is free to use.

Open source means you can even download its source code and change it for your needs.

Relational Database means you can create relation between the tables when you use MySQL.

Download the MySQL from: <https://www.mysql.com/>

With MySQL Database Software, you can

- Insert, Update or Delete the data in Tables.
- Query the Database for a specific row based on unique data condition.
- Use SQL language to access the data in the database.
- handle large amount of data.

You have to download the MySQL database to your local machine to start using it.

MySQL is under maintenance of Oracle. You can even purchase a license from Oracle for business purpose.

It also runs on various platforms like Linux, Unix and Windows and it works pretty well with PHP because PHP has many libraries to access the MySQL Database.

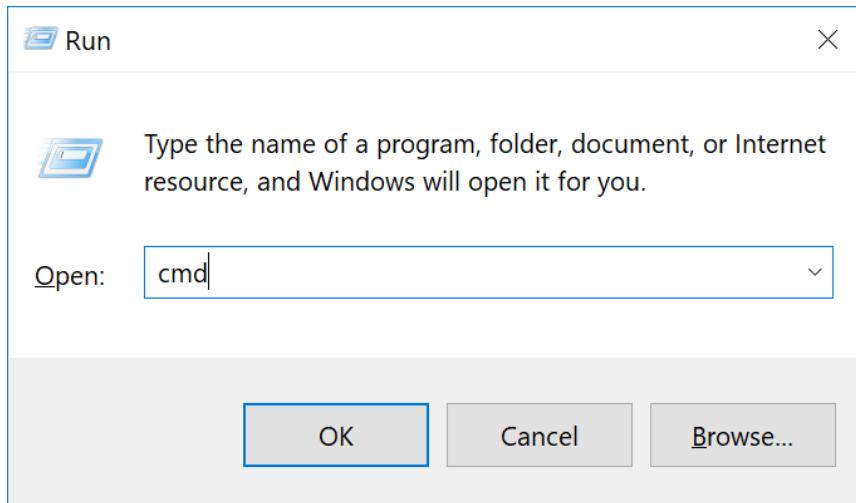
How to Access the MySQL from Command Line

Step 1: Make sure you have installed [WAMP Server from this guide.](#)

Step 2: Open Command Line with CMD + R and Type cmd.

[Become PHP Full Stack Web Developer in Just 30 Days](#)

1 MySQL Basics



Step 3: Go to folder

"cd C:\wamp64\bin\mysql\mysql5.7.21\bin"

A screenshot of a Windows Command Prompt window. The title bar says "Command Prompt". The main area shows the command prompt line: "C:\wamp64\bin\mysql\mysql5.7.21\bin>".

Step 4: Type >mysql -u root -p

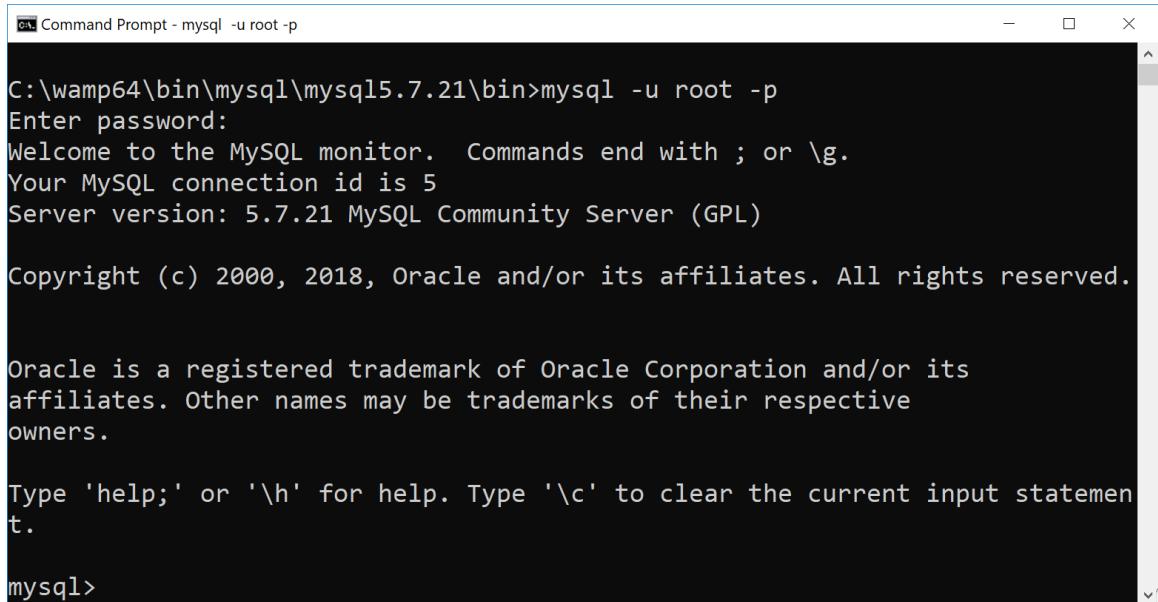
Press enter when it ask for password.

There is no password.

Step 5: This is your MySQL Database

[Become PHP Full Stack Web Developer in Just 30 Days](#)

1 MySQL Basics



```
C:\wamp64\bin\mysql\mysql5.7.21\bin>mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.7.21 MySQL Community Server (GPL)

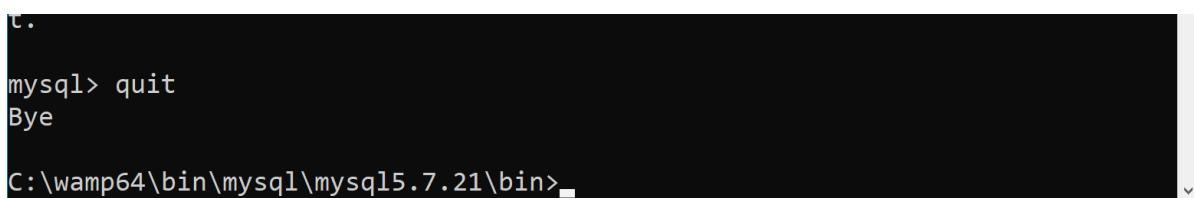
Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Step 6: Type 'quit' to exit.



```
t.

mysql> quit
Bye

C:\wamp64\bin\mysql\mysql5.7.21\bin>
```

1.3 What is phpMyAdmin?

How to access the MySQL Database?

Once you install the MySQL database there are three ways to access the database:

- Command Line Client
- Graphical Dashboard called as phpMyAdmin.
- PHP Program

[Become PHP Full Stack Web Developer in Just 30 Days](#)

What is phpMyAdmin?

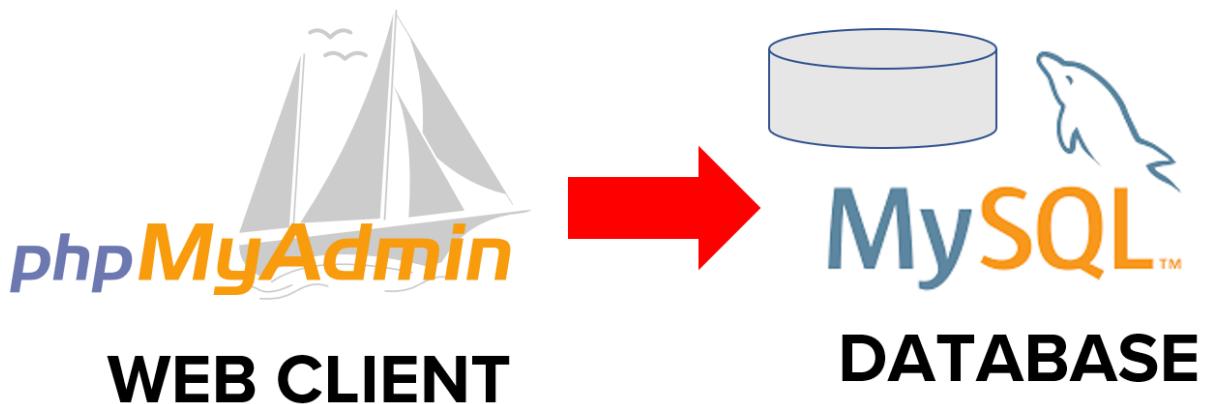
phpMyAdmin is a web based client using which we can access the MySQL Database and Tables.

We can perform the database operations like:

- Creating Users
- Creating Database, Tables
- Inserting, Updating and Deleting the Data

This is a Web based client using which we can perform the database operation on the MySQL.

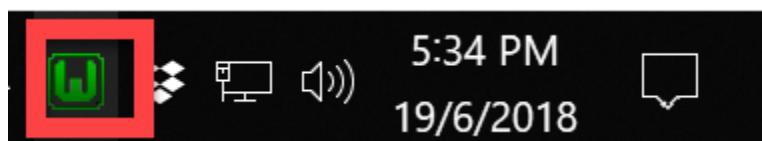
MySQL is the Database and phpMyAdmin is the web Client to access the database.



How to Access the MySQL from phpMyAdmin

Step 1: Make sure you have installed [WAMP Server from this guide.](#)

Step 2: Make sure the WAMP Server is running.

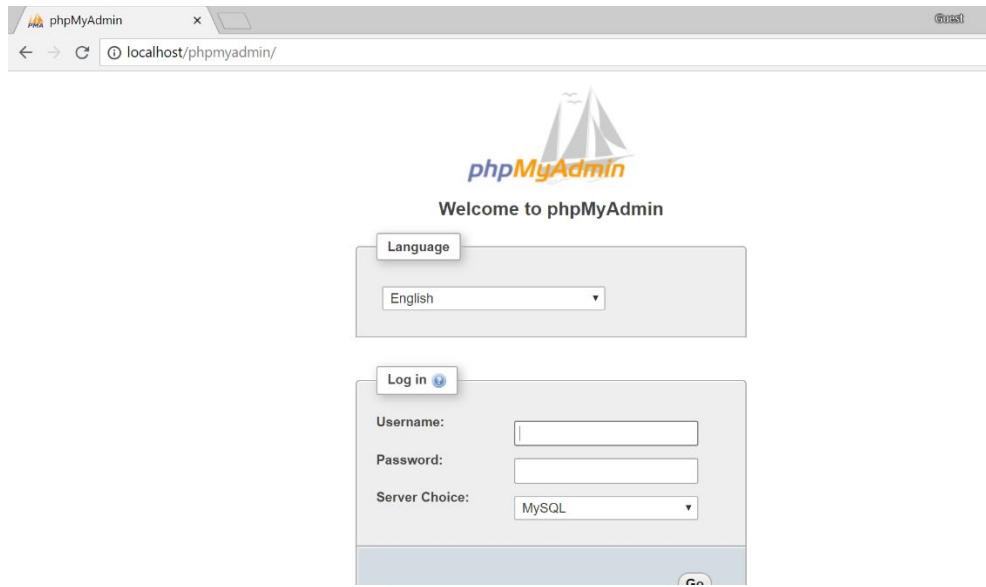


[Become PHP Full Stack Web Developer in Just 30 Days](#)

1 MySQL Basics

Step 3: Open Browser and type this url

http://localhost/phpmyadmin

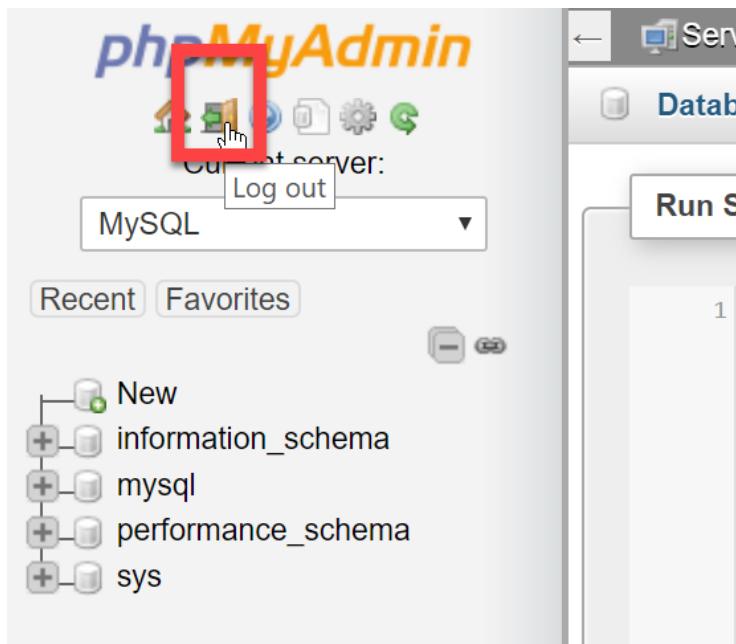


Step 4: Username is root and password is blank. Press enter.

A screenshot of the phpMyAdmin configuration page. The top navigation bar shows "Server: MySQL 3306" and various tabs: Databases, SQL, Status, User accounts, Export, Import, Settings, and More. On the left, there's a sidebar with "Recent" and "Favorites" sections and a tree view of databases: New, information_schema, mysql, performance_schema, and sys. The main content area is divided into several panels: "General settings" (Change password, Server connection collation set to utf8mb4_unicode_ci), "Appearance settings" (Language set to English, Theme set to pmahomme, Font size set to 82%, More settings), "Database server" (Server: MySQL (127.0.0.1 via TCP/IP), Server type: MySQL, Server version: 5.7.21 - MySQL Community Server (GPL), Protocol version: 10, User: root@localhost, Server charset: UTF-8 Unicode (utf8)), and "Web server" (Apache/2.4.33 (Win64) PHP/7.2.4, Database client version: libmysql - mysqlnd 5.0.12-dev - 20150407 - \$Id: 38fea24f2847fa7519001be390c98ae0acafe38\$, PHP extension: mysqli, curl, mbstring, PHP version: 7.2.4). A "Console" button is at the bottom left.

Step 5: Press the logout button to exit from the application

[Become PHP Full Stack Web Developer in Just 30 Days](#)



1.4 Datatypes MySQL

Datatypes in MySQL are divided into this 3 categories:

- Numeric
- Strings
- Date and Time

Numbers in MySQL are:

- INT
- TINYINT
- SMALLINT
- MEDIUMINT
- BIGINT
- FLOAT
- DOUBLE
- DECIMAL

String in MySQL are:

[Become PHP Full Stack Web Developer in Just 30 Days](#)

1 MySQL Basics

- CHAR
- VARCHAR
- BLOB (TINYBLOB, MEDIUMBLOB, LONGBLOB)
- TEXT (TINYTEXT, MEDIUMTEXT, LONGTEXT)
- ENUM

Date and Time in MySQL are:

- DATE
- TIME
- DATETIME
- TIMESTAMP

Most Commonly Used Datatypes:

- INT OR FLOAT OR DECIMAL – To Store numbers and decimals
- CHAR OR VARCHAR – To Store String
- DATETIME – To Store Date and Time

1.5 Keywords MySQL

Commonly used Keywords in MySQL SQL Statements:

- SELECT
- INSERT
- UPDATE
- DELETE
- AS
- DROP
- DESC
- TABLE
- DATABASE
- WHERE
- ISNULL
- ORDER BY

Complete list is [found here from MySQL](#)

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2.phpMyAdmin

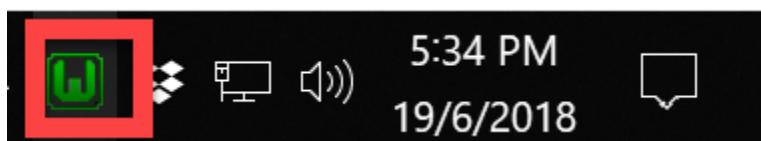
2 phpMyAdmin

2.1 Login and Logout phpMyAdmin

How to Login into phpMyAdmin

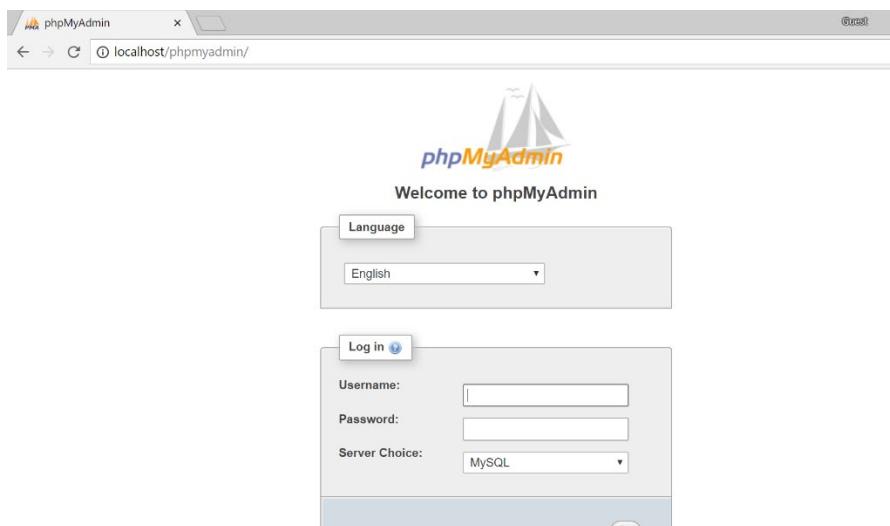
Step 1: Make sure you have installed [WAMP Server from this guide.](#)

Step 2: Make sure the WAMP Server is running.



Step 3: Open Browser and type this url

<http://localhost/phpmyadmin>



Step 4: Username is root and password is blank. Press enter.

2 phpMyAdmin

The screenshot shows the phpMyAdmin dashboard. On the left, there's a sidebar with icons for New, information_schema, mysql, performance_schema, and sys. The main area has tabs for Databases, SQL, Status, User accounts, Export, Import, Settings, and More. Under General settings, there's a 'Change password' link and a dropdown for 'Server connection collation' set to utf8mb4_unicode_ci. Under Appearance settings, there are options for Language (English), Theme (pmahomme), and Font size (82%). A 'More settings' link is also present. On the right, the Database server section lists MySQL details like version 5.7.21 and user root@localhost. The Web server section shows Apache 2.4.33, PHP 7.2.4, and various extensions.

How to Logout from phpMyAdmin

Step 1: Press the logout button to exit from the application

This screenshot shows the same phpMyAdmin interface as above, but with a red box highlighting the 'Log out' button in the top navigation bar. The rest of the interface, including the sidebar and main content areas, remains the same.

How to Change Password into phpMyAdmin

Step 1: Click on Home Icon



Step 2: Click on the Change Password.

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

The screenshot shows the phpMyAdmin interface with the 'General settings' tab selected. A red box highlights the 'Change password' link under 'General settings'. To the right, a sidebar titled 'Database server' provides server information. Below the main area, a modal window titled 'Change password' is open. It contains two input fields: 'Enter:' and 'Re-type:', both highlighted with a red box. A large red arrow points from the 'Change password' link to these fields. To the right of the fields, a progress bar indicates 'Strength: Good'. Below the fields, the 'Password Hashing' option is set to 'Native MySQL authentication'. A 'Generate' button is available for generating a new password. At the bottom right of the modal, there are 'Go' and 'Cancel' buttons, with 'Go' also highlighted with a red box. The word 'CHANGE PASSWORD' is overlaid in red text above the 'Change password' link, and 'ENTER NEW PASSOWRD' is overlaid in red text above the input fields.

Step 3: After you change password. Login again with the new password.

2.2 Create Users

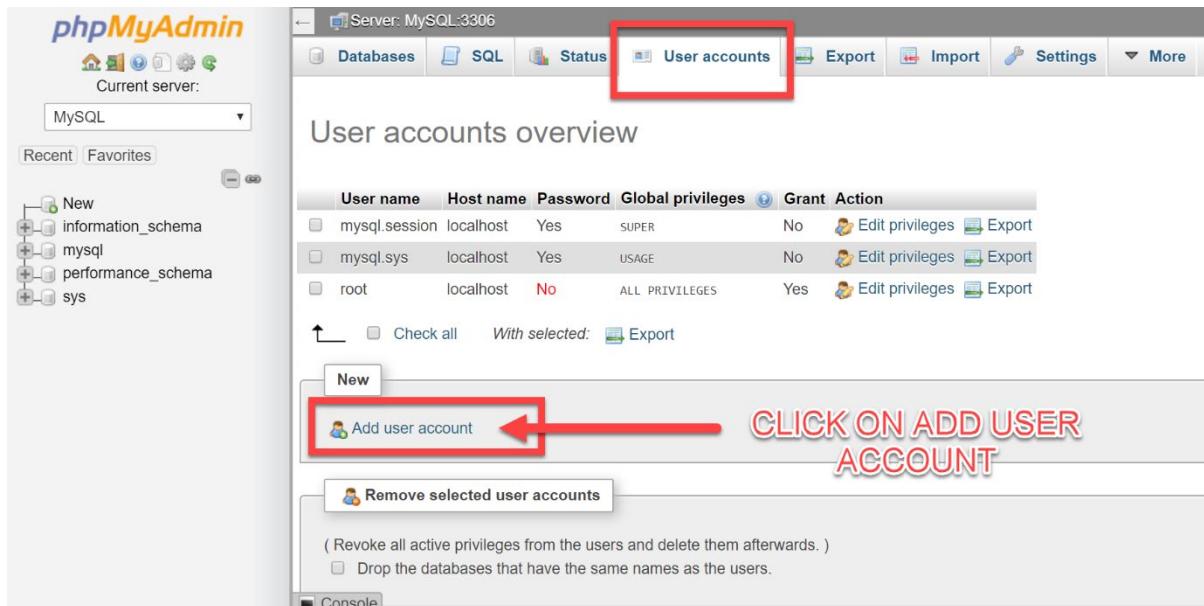
How to Create user with phpMyAdmin

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

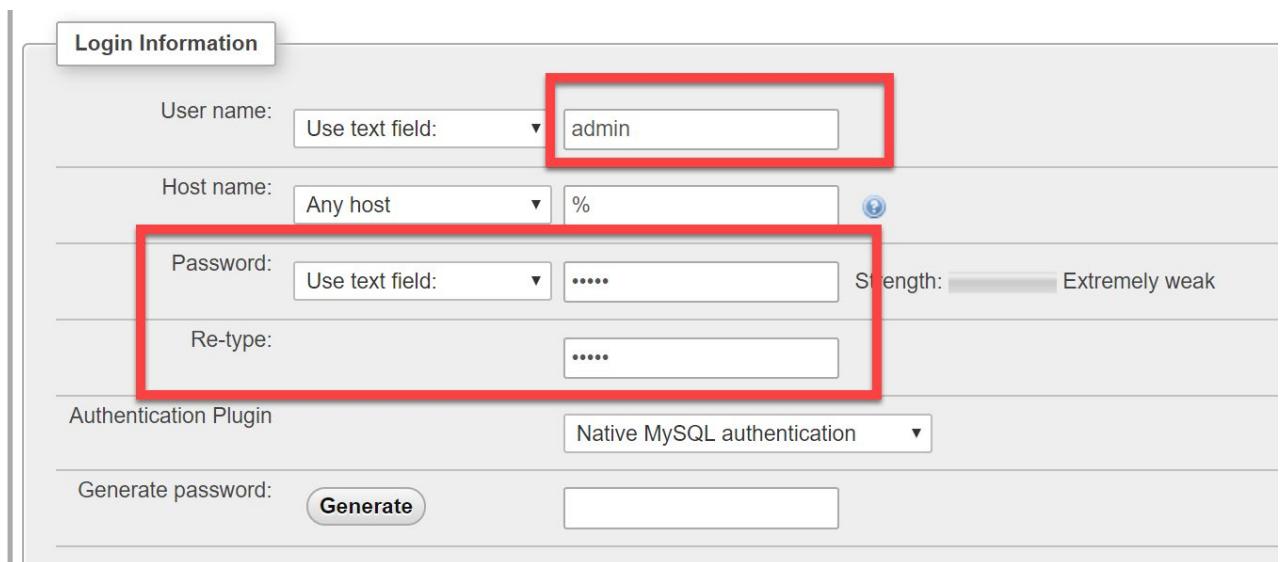
Step 1: Login into phpMyAdmin with the url: <http://localhost/phpmyadmin>

Step 2: Click on User Accounts -> Add New Account on the Home Page.



The screenshot shows the phpMyAdmin interface for MySQL. The left sidebar shows databases like information_schema, mysql, performance_schema, and sys. The top navigation bar has tabs for Databases, SQL, Status, User accounts (which is highlighted with a red box), Export, Import, Settings, and More. The main content area is titled 'User accounts overview' and lists three existing users: mysql.session, mysql.sys, and root. Below the list is a 'New' section with a 'Add user account' button, which is also highlighted with a red box. A large red arrow points from the text 'CLICK ON ADD USER ACCOUNT' to this button. There are also options to 'Remove selected user accounts' and checkboxes for revoking privileges or dropping databases.

Step 3: Add the Username and Password and click on Global Privileges.



This screenshot shows the 'Add user account' form. It includes fields for 'User name' (set to 'admin'), 'Host name' (set to 'Any host'), 'Password' (set to '.....'), 'Re-type' (also set to '.....'), and an 'Authentication Plugin' dropdown set to 'Native MySQL authentication'. A 'Generate password' button with a 'Generate' button next to it is also visible. The 'User name' and 'Host name' fields, as well as the 'Password' and 'Re-type' fields, are all highlighted with red boxes. The 'Generate password' section is also highlighted with a red box.

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

Note: MySQL privilege names are expressed in English.

Data	Structure	Administration
<input checked="" type="checkbox"/> SELECT	<input checked="" type="checkbox"/> CREATE	<input checked="" type="checkbox"/> GRANT
<input checked="" type="checkbox"/> INSERT	<input checked="" type="checkbox"/> ALTER	<input checked="" type="checkbox"/> SUPER
<input checked="" type="checkbox"/> UPDATE	<input checked="" type="checkbox"/> INDEX	<input checked="" type="checkbox"/> PROCESS
<input checked="" type="checkbox"/> DELETE	<input checked="" type="checkbox"/> DROP	<input checked="" type="checkbox"/> RELOAD
<input checked="" type="checkbox"/> FILE	<input checked="" type="checkbox"/> CREATE TEMPORARY TABLES	<input checked="" type="checkbox"/> SHUTDOWN
	<input checked="" type="checkbox"/> SHOW VIEW	<input checked="" type="checkbox"/> SHOW DATABASES
	<input checked="" type="checkbox"/> CREATE ROUTINE	<input checked="" type="checkbox"/> LOCK TABLES
	<input checked="" type="checkbox"/> ALTER ROUTINE	<input checked="" type="checkbox"/> REFERENCES

Click on GO Button at the bottom of the page.



Step 4: Check the User Created

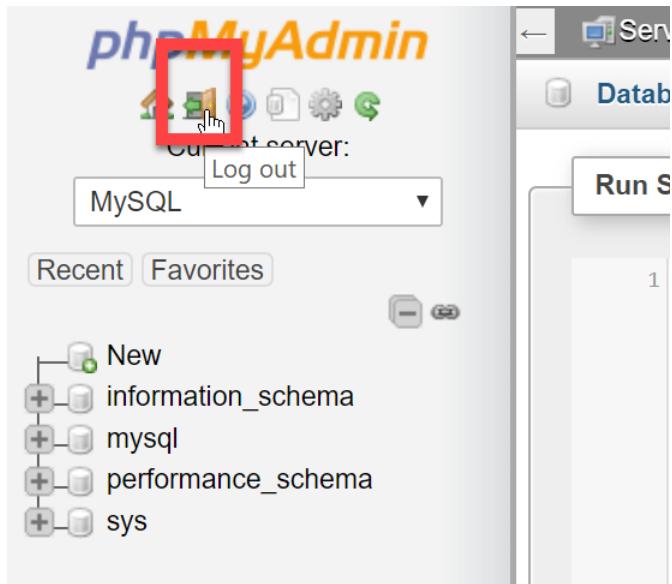
User name	Host name	Password	Global privileges	Grant	Action
admin	%	Yes	ALL PRIVILEGES	Yes	
mysql.session	localhost	Yes	SUPER	No	
mysql.sys	localhost	Yes	USAGE	No	
root	localhost	No	ALL PRIVILEGES	Yes	

CHECK THE USER

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

Step 5: Press the logout button to exit from the application



Step 6: Login again with new user credentials.

2.3 Create Database and Table

What is Database & Table?

Database is a collection of Tables. One Database can have multiple tables.

One Table can have defined column and all the data is stored in each row.

2 phpMyAdmin

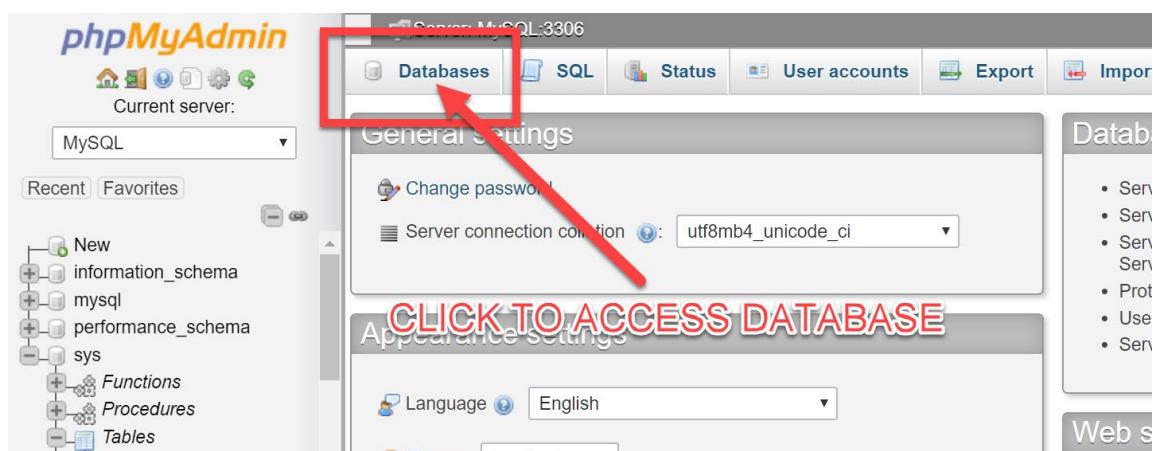
DATABASE																									
student	marks																								
<table border="1"><thead><tr><th>ID</th><th>NAME</th></tr></thead><tbody><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr></tbody></table>	ID	NAME											<table border="1"><thead><tr><th>ID</th><th>MARKS</th></tr></thead><tbody><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr></tbody></table>	ID	MARKS										
ID	NAME																								
ID	MARKS																								

Create Database in phpMyAdmin

Create Database 'schooldb'

Step 1: Open the phpMyAdmin Dashboard

Step 2: Click on Database



Step 3: Enter the new Database Name – **schooldb**

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

The screenshot shows the phpMyAdmin interface for MySQL. In the top navigation bar, 'Server: MySQL:3306' is selected. Below it, tabs for 'Databases', 'SQL', 'Status', 'User accounts', and 'Export' are visible. A red box highlights the 'Databases' tab. On the left, a tree view shows the database structure with 'schooldb' listed under 'Tables'. In the main area, a 'CREATE NEW DB' section has a 'Create database' form with 'schooldb' in the input field and a 'Create' button highlighted with a red box. Below this, a table lists existing databases: information_schema, mysql, performance_schema, and sys. The table shows 'Total: 4' databases.

Step 4: Check the Database

The screenshot shows the phpMyAdmin interface for MySQL. The 'Database: schooldb' tab is selected. A red arrow points from the 'schooldb' entry in the database list on the left to the 'Database: schooldb' tab at the top. The main area displays a message: 'No tables found in database.' Below it is a 'Create table' form with 'Name:' set to 'student' and 'Number of columns:' set to '4'. A red box highlights the 'Query' tab in the top menu. The text 'CHECK IF THE DB CREATED' is overlaid in red at the bottom of the interface.

Step 5: Create Two tables – student and marks

The screenshot shows the phpMyAdmin interface for MySQL. The 'Database: schooldb' tab is selected. A red box highlights the 'Structure' tab in the top menu. The main area displays a message: 'No tables found in database.' Below it is a 'Create table' form with 'Name:' set to 'student' and 'Number of columns:' set to '4'. A red box highlights the 'student' input field. A red box also highlights the 'Go' button at the bottom right of the form.

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

Step 6: Create Two tables – student

The screenshot shows the 'Create table' interface in phpMyAdmin. A red box highlights the 'Table name' field containing 'student'. Below it, a red box highlights the first two columns: 'ID' (INT) and 'NAME' (VARCHAR). Another red box highlights the 'Length/Values' field for the 'NAME' column, which contains '500'. The 'Default' field for this column is set to 'None'. The 'Add' button is visible at the top right, along with a '1' indicating one column is added.

Save Button is bottom right



Step 7: Verify the table

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

The screenshot shows the phpMyAdmin interface for the 'student' table in the 'schooldb' database. The 'Structure' tab is selected, highlighted by a red box. The table structure is displayed in a grid:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID	int(11)	latin1_swedish_ci		No	None			Change Drop More
2	NAME	varchar(500)	latin1_swedish_ci		No	None			Change Drop More

The sidebar on the left shows the database structure with 'schooldb' and 'student' tables highlighted by a red box. The 'Indexes' section below the table structure shows a warning: 'No index defined!'.

Step 8: Create the marks table

The screenshot shows the phpMyAdmin interface with the 'Create table' form highlighted by a red box. A red arrow points from the 'Name:' input field to the text 'enter table name'. Another red arrow points from the 'Go' button to the text 'here'.

The 'Create table' form has the following fields:

- Name: [empty input field]
- Number of columns: 4
- Go button (highlighted by a red box)

The sidebar on the left shows the database structure with 'schooldb' and 'student' tables highlighted by a red box. A red arrow points from the 'Home' icon in the top-left to the text 'click home to come here'.

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

The screenshot shows the 'Structure' tab of the phpMyAdmin 'Create' interface. A red box highlights the 'Name' and 'Type' columns for the first two columns: 'ID' (INT) and 'marks' (INT). Below this, the 'Table comments', 'Collation', and 'Storage Engine' (MyISAM) are set. Under 'PARTITION definition', there are fields for 'Partition by' and 'Partitions'. At the bottom right are 'Preview SQL' and 'Save' buttons, with 'Save' being highlighted by a red box.

Step 9: Verify the Table

The screenshot shows the 'Structure' tab for the 'marks' table. A red box highlights the 'Columns' section under the 'schooldb' schema. The 'student' table is also visible. In the center, a red box highlights the 'Indexes' section with the text 'No index defined'. At the bottom, there's a note 'Create an index on 1 columns Go'. The top navigation bar includes 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', and 'More'.

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

The screenshot shows the phpMyAdmin interface. At the top, the server is MySQL 3.306 and the database is selected as 'schooldb'. Below the top navigation bar, there are tabs for Structure, SQL, Search, Query, Export, Import, Operations, Privileges, and More. A 'Filters' button is present. A search bar contains the placeholder 'Containing the word:'. The main area displays a table of databases with columns: Table, Action, Rows, Type, Collation, and Size. Two tables are listed: 'marks' and 'student', both of which have 0 rows and are MyISAM type with latin1_swedish_ci collation. The total size for both is 2 KiB. Below the table, there are buttons for 'Check all' and 'With selected:'. At the bottom, there are links for Print and Data dictionary, and a 'Create table' button with fields for Name (marked) and Number of columns (4).

Table	Action	Rows	Type	Collation	Size
marks	Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 KiB
student	Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 KiB
2 tables Sum		0	MyISAM	latin1_swedish_ci	2 KiB

2.4 Run Simple SQL Statements

In this exercise, you will

- Create Table
- Insert Data in Table
- View the Table Data

Step 1: Login into phpMyAdmin

Type the url: <http://localhost/phpmyadmin> in the browser.

Step 2: Click on Database

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

phpMyAdmin

Server: MySQL:3306

Databases SQL Status User accounts Export Import Settings More

Current server: MySQL

New information_schema mysql performance_schema schooldb

New marks Columns New ID marks student Columns New ID NAME

General settings

Change password Server connection collation: utf8mb4_unicode_ci

CLICK ON DATABASE

Language: English

Theme: pmahomme

Font size: 82%

More settings

Database server

- Server: MySQL (127.0.0.1 via TCP/IP)
- Server type: MySQL
- Server version: 5.7.21 - MySQL Community Server (GPL)
- Protocol version: 10
- User: root@localhost
- Server charset: UTF-8 Unicode (utf8)

Web server

- Apache/2.4.33 (Win64) PHP/7.2.4
- Database client version: libmysql - mysqlnd 5.0.12-dev - 20150407 - \$Id: 38fea24f2847fa7519001be390c98ae0acafe \$
- PHP extension: mysqli curl mbstring
- PHP version: 7.2.4

Step 3: Open the Database

phpMyAdmin

Server: MySQL:3306

Databases SQL Status User accounts Export

Current server: MySQL

New information_schema mysql performance_schema schooldb

New marks Columns New ID marks student Columns New ID NAME

Databases

Create database

Database	Collation	Action
information_schema	utf8_general_ci	<input type="checkbox"/> Check privileges
mysql	latin1_swedish_ci	<input type="checkbox"/> Check privileges
performance_schema	utf8_general_ci	<input type="checkbox"/> Check privileges
schooldb	latin1_swedish_ci	<input type="checkbox"/> Check privileges
sys	utf8_general_ci	<input type="checkbox"/> Check privileges

Total: 5

Check all With selected: Drop

Note: Enabling the database statistics here might cause heavy traffic between the web

Step 4: Click the insert link

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

phpMyAdmin

Server: MySQL:3306 » Database: schooldb

Structure SQL Search Query Export Import Operations

Containing the word:

Table Action Rows Type Collation

marks Browse Structure Search Insert Empty Drop 0 MyISAM latin1_swed

student Browse Structure Search Insert Empty Drop 0 MyISAM latin1_swed

2 tables Sum

Check all With selected:

Print Data dictionary

Create table

Name: Number of columns: 4

Containing the word:

Table Action Rows

marks Browse Structure Search Insert Empty Drop 0

student Browse Structure Search Insert Empty Drop 0

2 tables Sum 0

CLICK THE STUDENT TABLE

click here

Step 5: Insert the data

Column	Type	Function	Null	Value
ID	int(11)			10
NAME	varchar(500)			JOHN

Go

Step 6: Verify the Rows in the Table

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

The screenshot shows the phpMyAdmin interface for the 'student' table in the 'schooldb' database. A red box highlights the message '1 row inserted.' and the SQL query: 'INSERT INTO `student`(`ID`, `NAME`) VALUES ('10', 'JOHN');'. Below this, the text 'ROW INSERTED' is overlaid in red. The table structure is visible on the left, and a list of tables is shown on the right.

Click on the “Browse” link beside the table.

The screenshot shows the phpMyAdmin interface for the 'schooldb' database. A red box highlights the 'Browse' link next to the 'student' table in the list. Below it, the text 'click browse to see the data' is overlaid in red.

[Become PHP Full Stack Web Developer in Just 30 Days](#)

2 phpMyAdmin

The screenshot shows the phpMyAdmin interface for the 'student' table in the 'schooldb' database. The left sidebar shows the database structure with tables 'marks' and 'student'. The 'marks' table has columns 'New', 'ID', and 'marks'. The 'student' table has columns 'New', 'ID', and 'NAME'. A red box highlights the first row of the 'student' table results, which contains 'ID' (10) and 'NAME' (JOHN). An arrow points from the text 'TABLE ROWS' to this highlighted row.

ID	NAME
10	JOHN

Step 7: Add the data for 'marks' table and browse it.

The screenshot shows the phpMyAdmin interface for the 'marks' table in the 'schooldb' database. The left sidebar shows the database structure with tables 'marks' and 'student'. The 'marks' table has columns 'New', 'ID', and 'marks'. The 'student' table has columns 'New', 'ID', and 'NAME'. A red box highlights the first row of the 'marks' table results, which contains 'ID' (10) and 'marks' (97). An arrow points from the text 'TABLE ROWS' to this highlighted row.

ID	marks
10	97

[Become PHP Full Stack Web Developer in Just 30 Days](#)

3. MySQL STATEMENTS



3 MySQL Statements

3.1 Create a Table

Follow this two Guide to create the table:

- [2.3 – Create Database and Table](#)
- [2.4 – Run Simple SQL Statements](#)

Exercise:

- Create a table name 'fees' with two fields
 - ID – INT
 - MARKS – INT
- Add some data to the fees table.

3.2 Drop a Table

Dropping a table means deleting a table.

DROP is a keyword to delete the table.

SQL SYNTAX:

DROP TABLE <TABLENAME>

SQL QUERY:

DROP TABLE 'fees';

Step 1: Create a new Table 'fees' with ID and Amount Fields

[Become PHP Full Stack Web Developer in Just 30 Days](#)

3 MySQL Statements

The screenshot shows the phpMyAdmin interface for the 'schooldb' database. On the left, the database structure is visible with tables 'marks' and 'student'. A red arrow points from the text 'CREATE TABLE' to the 'Create table' dialog box. Inside the dialog, the table name 'fees' is entered in the 'Name:' field, which is highlighted with a red box. The 'Number of columns:' field contains the value '4'. A red box also surrounds the 'Go' button at the bottom right of the dialog.

Step 2: Add some Data to the 'fees' Table

The screenshot shows the phpMyAdmin interface for the 'schooldb' database, specifically the 'fees' table. A red box highlights the table name 'fees' in the top navigation bar. The table structure is shown on the left, and the data grid is highlighted with a red box. The data grid contains two rows of information:

ID	AMOUNT
10	450
20	890

A red arrow points from the text 'TABLE DATA' to the data grid. Below the grid, there are buttons for 'Query results operations' including 'Console', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view'.

Step 3: Delete the 'fees' table

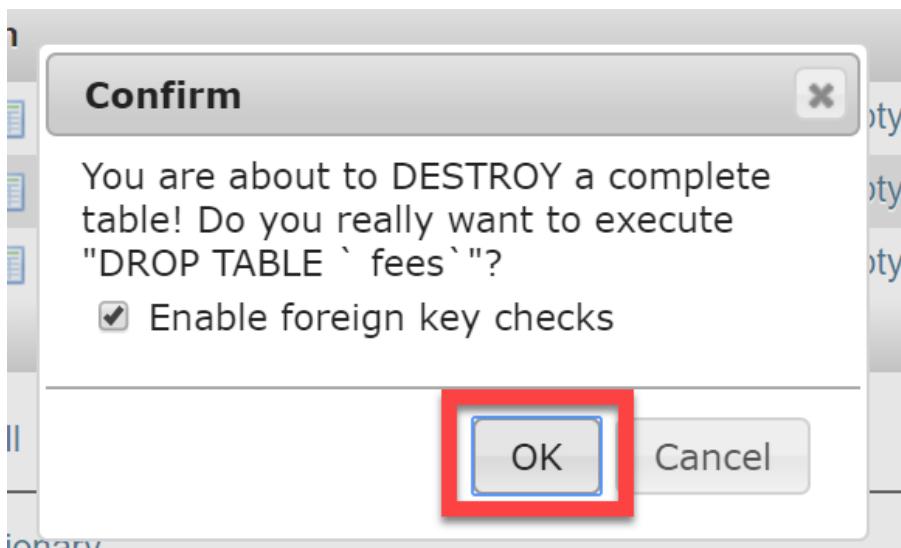
[Become PHP Full Stack Web Developer in Just 30 Days](#)

3 MySQL Statements

CLICK THIS TO DELETE THE TABLE

The screenshot shows the phpMyAdmin interface with the 'fees' table selected in the 'Tables' list. A red arrow points to the 'Drop' link next to the 'fees' table. Below the table list, there is a confirmation dialog box.

Table	Action	Rows	Type	Collation	Size
fees	Browse Structure Search Insert Empty Drop	2	MyISAM	latin1_swedish_ci	1 KiB
marks	Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	1 KiB
student	Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	1 KiB
3 tables Sum		4	MyISAM	latin1_swedish_ci	3 KiB



The screenshot shows the phpMyAdmin interface after the 'fees' table has been deleted. The 'Tables' list now shows only two tables: 'marks' and 'student'. A red arrow points to the 'Sum' entry in the table list. A message at the bottom right of the interface says "'fees' table deleted".

Table	Action	Rows	Type	Collation	Size
marks	Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	1 KiB
student	Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	1 KiB
2 table(s) Sum		2	MyISAM	latin1_swedish_ci	2 KiB

[Become PHP Full Stack Web Developer in Just 30 Days](#)

3.3 INSERT Statements

INSERT is a keyword to INSERT the data in the table.

SQL SYNTAX:

```
INSERT INTO table_name VALUES (value1, value2, value3);
```

SQL QUERY:

```
INSERT INTO fees VALUES(10, 86);
```

Exercise:

Add some data to the fees table.

3.4 SELECT Statements

SELECT is a keyword to select the data from the tables.

SQL SYNTAX:

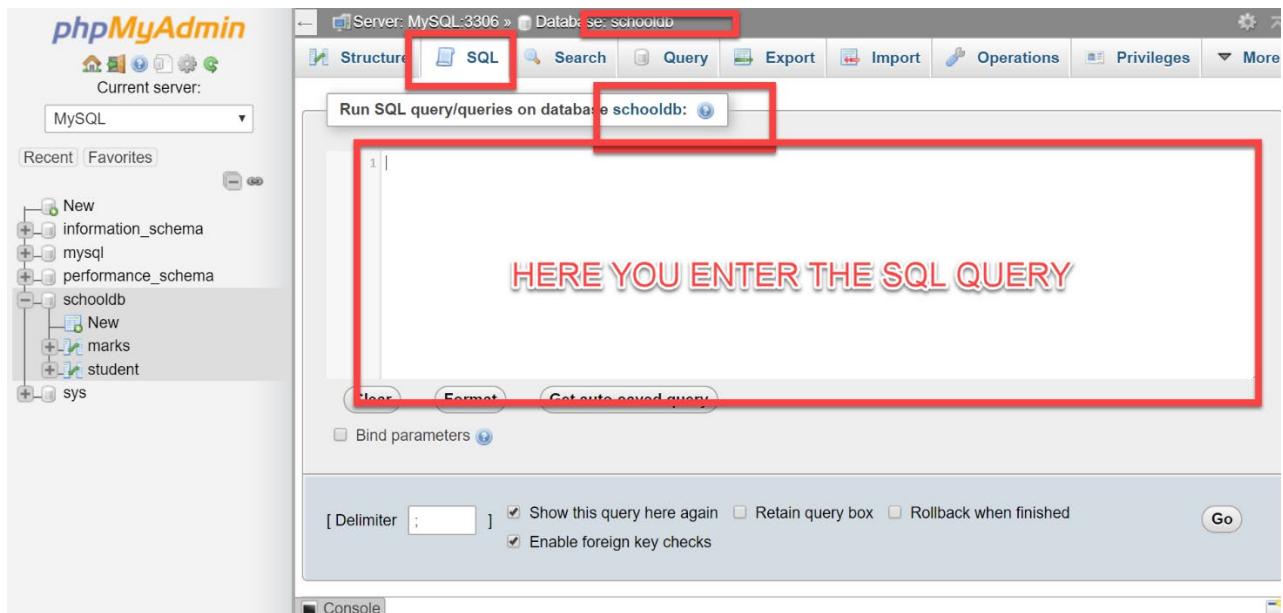
```
SELECT * FROM table_name;
```

SQL QUERY:

```
SELECT * FROM student;
```

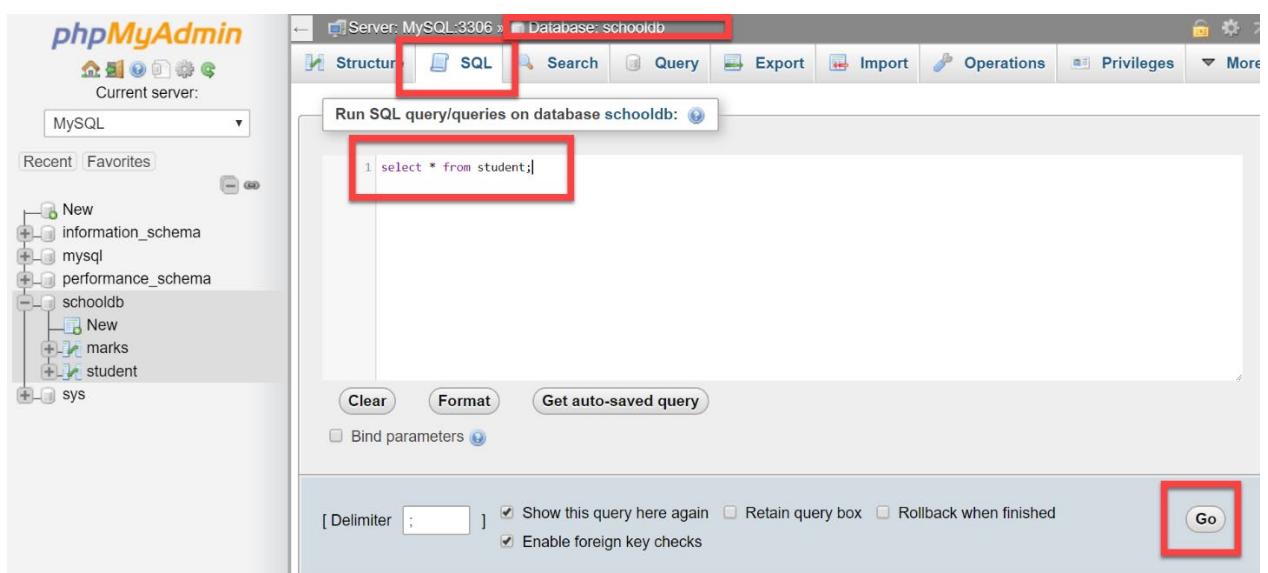
Step 1: Login into phpMyAdmin and open the 'studentdb'

3 MySQL Statements



Step 2: Enter the SELECT SQL Query

SELECT * FROM student;



Step 3: Verify the Data

[Become PHP Full Stack Web Developer in Just 30 Days](#)

3 MySQL Statements

The screenshot shows the phpMyAdmin interface for a MySQL database named 'schooldb'. The 'student' table is selected. The SQL query 'select * from student' is run, resulting in one row being displayed: ID 10 and NAME JOHN. A red box highlights the table data area, and a red arrow points to the word 'TABLE DATA'.

ID	NAME
10	JOHN

3.5 Clause WHERE, LIMIT

WHERE and LIMIT are called as Clause which are used along with SQL statement to apply the condition.

SQL SYNTAX:

```
SELECT * FROM table_name WHERE ID <= 100;
```

SQL QUERY:

```
SELECT * FROM student WHERE ID <= 100 LIMIT 2;
```

This query tell the MySQL to fetch student record where ID field value is less than 100 and fetch only two rows.

EXERCISE 1:

Fetch only 2 records from student table.

```
SELECT * FROM student LIMIT 2;
```

[Become PHP Full Stack Web Developer in Just 30 Days](#)

3 MySQL Statements

Showing rows 0 - 1 (2 total, Query took 0.0004 seconds.)

```
SELECT * FROM student LIMIT 2
```



+ Options

ID	NAME
----	------

10	JOHN
----	------

70	AMIT
----	------

EXERCISE 2:

Fetch records where ID > 50 and LIMIT to 1 record.

```
SELECT * FROM student WHERE student.ID <= 50 LIMIT 1;
```

Showing rows 0 - 0 (1 total, Query took 0.0002 seconds.)

```
SELECT * FROM student WHERE student.ID <= 50 LIMIT 1
```



+ Options

ID	NAME
----	------

10	JOHN
----	------

3.6 Operators IS NULL, LIKE, ORDER BY

IS NULL, LIKE and ORDER BY are called as operators that you can apply on the condition to check and sort the records.

[Become PHP Full Stack Web Developer in Just 30 Days](#)

3 MySQL Statements

IS NULL will check if the field is NULL or NOT.

ORDER BY will order / sort the records based on the field.

LIKE has two wild characters:

- % – The percent sign represents zero, one, or multiple characters
- _ – The underscore represents a single character

SQL SYNTAX:

`SELECT * FROM table_name WHERE columnN IS NOT NULL`

`SELECT * FROM table_name WHERE columnN LIKE pattern;`

`SELECT * FROM table_name WHERE columnN LIKE pattern ORDER BY columnN;`

SQL QUERY:

`SELECT * FROM student WHERE student.NAME IS NOT NULL;`

This query tell the to MySQL to fetch student record where NAME is not null.

`SELECT * FROM student WHERE student.NAME LIKE 'J%';`

This query tell the to MySQL to fetch student record where NAME matches with J.

`SELECT * FROM student WHERE student.NAME LIKE 'J%' ORDER BY student.NAME;`

This query tell the to MySQL to fetch student record where NAME matches with J and sort the records by NAME.

EXERCISE 1:

Execute the above 3 Queries.

[Become PHP Full Stack Web Developer in Just 30 Days](#)

3.7 UPDATE Statements

UPDATE statement is used to update an existing row in the table based on a condition specified with WHERE clause.

SQL SYNTAX:

```
UPDATE table_name SET column1 = value1, column2 = value2  
WHERE condition;
```

SQL QUERY:

```
UPDATE student  
SET student.name = 'Julie' WHERE ID = 100;
```

EXERCISE 1:

Update the record name when ID = 45

```
UPDATE student  
SET student.name = 'Julie' WHERE ID = 45;
```

The screenshot shows the phpMyAdmin interface. The left sidebar displays the database structure with databases like information_schema, mysql, performance_schema, and schooldb, and tables such as marks and student under schooldb. The top navigation bar shows 'Server: MySQL:3306 » Database: schooldb » Table: student'. The main area has tabs for Browse, Structure, SQL, Search, Insert, and Export. The SQL tab contains the query: `UPDATE student SET student.name = 'Julie' WHERE ID = 45`. A success message in a green box indicates '1 row affected. (Query took 0.0003 seconds.)'.

EXERCISE 2:

Update the record ID = 100 where NAME = 'WordPress'

[Become PHP Full Stack Web Developer in Just 30 Days](#)

3 MySQL Statements

UPDATE student SET student.id = 100 WHERE student.NAME = 'WordPress'

Show query box

✓ 0 rows affected. (Query took 0.0002 seconds.)

```
UPDATE student SET student.id = 100 WHERE student.NAME = 'WordPress'
```

3.8 DELETE Statements

DELETE statement is used to delete an existing row in the table based on a condition specified with WHERE clause.

SQL SYNTAX:

DELETE from table_name WHERE *condition*;

SQL QUERY:

DELETE FROM student WHERE student.name='John';

EXERCISE 1:

Delete a row where ID = 100 from student table.

[Become PHP Full Stack Web Developer in Just 30 Days](#)

3 MySQL Statements

DELETE FROM student WHERE student.ID = 100;

EXERCISE 2:

Delete a row where name is NULL

DELETE FROM student WHERE student.NAME IS NULL;

4. MySQL with PDO

4 mySQL & PHP with PDO

4.1 Database Connectivity

What is PDO?

PDO stands for PHP Data Objects it is a library that can be used to connect to MySQL from PHP code.

PDO gives object oriented database functions to perform the database operations on MySQL.

The biggest advantage of using PDO is that you can change database any time from MySQL to Oracle or Microsoft SQL and the underlying PDO code will not change.

How to Connect to MySQL DB with PDO?

To connect MySQL DB we need following things:

- Hostname / IP of the server on which MySQL is running.
- Database Name
- Userid
- Password

We have seen [how to work with MySQL with phpMyAdmin](#).

Define the parameters:

```
$dns = 'mysql:host=localhost;dbname=studentdb';  
$username = 'root';  
$password = 'root';  
$db = new PDO($dns, $username, $password);
```

[Become PHP Full Stack Web Developer in Just 30 Days](#)

\$dns will hold the parameters separated by semicolon (:).

mysql is the name of the database

localhost means mysql server is running on local machine.

dbname is the database name that [we created here](#).

\$username and \$password are the user credentials to login into the MySQL.

\$db will hold the connection to the MySQL and then we can execute the SQL query to work on the database tables.

Here are the high level steps to connect to DB:

- Define the DNS variable with hostname and database name
- Create a PDO class with \$dns, user name and password.
- This will create an instance using which we can access the database.

4.2 Simple Query from PHP to mySQL

Write a Simple SELECT query using PDO

Credentials:

Database Name: studentdb

Table Name: student

Username: root

password: root

4 mySQL & PHP with PDO

The screenshot shows the phpMyAdmin interface. At the top, it displays 'Server: MySQL-3306' and 'Database: schooldb > table: student'. Below this is a toolbar with 'Browse', 'Structure', 'SQL', 'Set', 'Insert', 'Export', 'Import', 'Privileges', and 'More'. A red box highlights the 'SQL' tab. A message bar at the top says 'Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.' A red arrow points from the text 'DATABASE AND TABLE NAME' to this message. Below the message is a SQL query: 'SELECT * FROM `student`'. Another red box highlights this query. A red arrow points from the text 'TABLE ENTRIES' to the table data. The table has columns 'ID' and 'NAME', with rows: ID 10, NAME JOHN; ID 70, NAME AMIT; ID 45, NAME Julie.

Sample Example

[Download the Example](#)

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="description" content="Page Description">
    <title>PDO - SELECT Query</title>
</head>

<body>
<h1>PDO - SELECT Query</h1>
<?php

$query = "SELECT * FROM student;";
$dns = 'mysql:host=localhost;dbname=schooldb';
$username = 'root';
$password = 'root';
try{

    $db = new PDO($dns, $username, $password);

    //Prepared Statement
    $statement = $db->prepare($query);

    //Execute the Query
    $statement->execute();

    //Loop all the records using fetch records
    while ($student = $statement->fetch()) {
        echo "ID: " . $student['ID']."<br />\n";
        echo "NAME: " . $student['NAME']."<br />\n";
    }

}
```

[Become PHP Full Stack Web Developer in Just 30 Days](#)

```
//close the connection to DB
$statement->closeCursor();

}catch(Exception $e)
{
    $error_message = $e->getMessage();
    echo "<p>Error message: $error_message </p>";
}

?>
</body>
</html>
```

PDO - SELECT Query

ID: 10
NAME: JOHN
ID: 70
NAME: AMIT
ID: 45
NAME: Julie

[Live Preview](#)

Exercise 1

[Download the Exercise 1](#)

Exercise 1: Write a Select query to fetch student where ID > 50.

SELECT * FROM student where ID > 50;

[Become PHP Full Stack Web Developer in Just 30 Days](#)

PDO - SELECT Query

ID: 70

NAME: AMIT

[Live Preview](#)

Exercise 2

[Download the Exercise 2](#)

Exercise 1: Write a Select query to fetch student and sort in ascending order by name field.

SELECT * FROM student order by name;

PDO - SELECT Query

ID: 70

NAME: AMIT

ID: 10

NAME: JOHN

ID: 45

NAME: Julie

[Live Preview](#)

[Become PHP Full Stack Web Developer in Just 30 Days](#)

4.3 Form to Add, Edit, Update and Delete

Write a Student form to Add, Edit and Delete the Entries from MySQL DB.

Credentials:

Database Name: studentdb

Table Name: student

Username: root

password: root

The screenshot shows the phpMyAdmin interface. At the top, it displays the server as 'MySQL:3306', the database as 'schooldb', and the table as 'student'. A red box highlights the 'Structure' tab in the navigation bar. Below the tabs, a message states: 'Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.' A red arrow points from this message to the 'Search' button in the toolbar. The main area shows a green banner with the text 'DATABASE AND TABLE NAME' and a SQL query: 'SELECT * FROM `student`'. Another red box highlights this SQL query. To the right, there is a 'TABLE ENTRIES' section containing a table with four rows of data: ID, NAME, 10, JOHN; ID, NAME, 70, AMIT; ID, NAME, 45, Julie. A red arrow points from the 'TABLE ENTRIES' label to this data table. On the left sidebar, the database structure is shown with 'schooldb' selected, and its tables 'New', 'marks', and 'student' are highlighted with red boxes.

ID	NAME
10	JOHN
70	AMIT
45	Julie

Sample Example

[Download the Example](#)

[Become PHP Full Stack Web Developer in Just 30 Days](#)

PDO - Table

DISPLAY - TABLE ENTRIES

ID	NAME
100	ELLY
70	JULIE
45	AMIT
40	TEST
12	HENRY

ADD - TABLE ENTRIES

Student Form

ID:	<input type="text"/>
Name:	<input type="text"/>

EDIT - TABLE ENTRIES

#	ID	NAME	OPERATION
100	ELLY	<input type="button" value="Edit"/>	
70	JULIE	<input type="button" value="Edit"/>	
45	AMIT	<input type="button" value="Edit"/>	
40	TEST	<input type="button" value="Edit"/>	
12	HENRY	<input type="button" value="Edit"/>	

DELETE - TABLE ENTRIES

#	ID	NAME	OPERATION
100	ELLY	<input type="button" value="Delete"/>	
70	JULIE	<input type="button" value="Delete"/>	

[Live Preview](#)

Exercise 1

Exercise 1: Create your own form and do the Add, Edit and Delete Operation.

[Become PHP Full Stack Web Developer in Just 30 Days](#)

5. MySQLi with PHP



5 mySQL & PHP with mysqli

5.1 Database Connectivity

What is mysqli?

mysqli is a library that can be used to connect to MySQL from PHP code.

mysqli is the just the extension of the mysql library (*i* stands for improved).

mysqli has more features and function to work with mysql and it very specific to mysql.

The disadvantage of using mysqli is that you cannot change database once your code written with this library.

How to Connect to MySQL DB with mysqli?

To connect MySQL DB we need following things:

- Hostname / IP of the server on which MySQL is running.
- Database Name
- Userid
- Password

We have seen [how to work with MySQL with phpMyAdmin](#).

Define the parameters:

```
$hostname = 'localhost';  
$db_name = 'schooldb';  
$username = 'root';
```

[Become PHP Full Stack Web Developer in Just 30 Days](#)

5 mySQL & PHP with mysqli

```
$password = 'root';
```

```
@ $db = mysqli_connect($hostname, $username, $password,  
$db_name);
```

@ is the error suppressor operator that is used to suppress any error throw by that statement.

localhost means mysql server is running on local machine.

\$db_name is the database name that [we created here](#).

\$username and \$password are the user credentials to login into the MySQL.

\$db will hold the connection to the MySQL and then we can execute the SQL query to work on the database tables.

Here are the high level steps to connect to DB:

- Define the DNS variable with hostname and database name
- Create a mysqli class with host, database, user name and password.
- This will create an instance using which we can access the database.

5.2 Simple Query from PHP to MySQL

Write a Simple SELECT query using PDO

Credentials:

Database Name: studentdb

Table Name: student

[Become PHP Full Stack Web Developer in Just 30 Days](#)

5 mySQL & PHP with mysqli

Username: root

password: root

The screenshot shows the phpMyAdmin interface. At the top, it says "Server: MySQL:3306 Database: schooldb > Table: student". Below this is a toolbar with "Browse", "Structure", "SQL", "Select", "Insert", "Export", "Import", "Privileges", and "More". A red box highlights the "Select" button. A yellow banner at the top states: "Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available." Below the banner, a red box highlights the SQL query: "SELECT * FROM `student`". To the right of the query, a red arrow points to the table results with the text "DATABASE AND TABLE NAME". The results table shows three rows of data:

ID	NAME
10	JOHN
70	AMIT
45	Julie

A red arrow points from the table to the text "TABLE ENTRIES".

Sample Example

[Download the Example](#)

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="description" content="Page Description">
    <title>mysqli - SELECT Query</title>
</head>

<body>
<h1>mysqli - SELECT Query</h1>
<?php

$query = "SELECT * FROM student";
$hostname = 'localhost';
$db_name = 'schooldb';
$username = 'root';
$password = 'root';

try{
    $db = new mysqli($hostname, $username, $password, $db_name);

    // Check connection
    if ($db->connect_error) {
        die("Connection failed: " . $db->connect_error);
}
```

[Become PHP Full Stack Web Developer in Just 30 Days](#)

5 mySQL & PHP with mysqli

```
}

//Select Query
$result = $db->query($query);

//Loop all the records using fetch records
while ($student = $result->fetch_assoc()) {
    echo "ID: " . $student['ID']."<br />\n";
    echo "NAME: " . $student['NAME']."<br />\n";
}

//close the connection to DB
$db->close();

} catch(Exception $e)
{
    $error_message = $e->getMessage();
    echo "<p>Error message: $error_message </p>";
}

?>
</body>
</html>
```

mysqli - SELECT Query

ID: 45
NAME: AMIT
ID: 100
NAME: ELLY
ID: 12
NAME: HENRY
ID: 70
NAME: JULIE
ID: 40
NAME: TEST

[Live Preview](#)

Exercise 1

[Download the Exercise 1](#)

[Become PHP Full Stack Web Developer in Just 30 Days](#)

Exercise 1: Write a Select query to fetch student where ID > 50.

```
SELECT * FROM student where ID > 50;
```

mysqli - SELECT Query

ID: 100

NAME: ELLY

ID: 70

NAME: JULIE

[Live Preview](#)

Exercise 2

[Download the Exercise 2](#)

Exercise 1: Write a Select query to fetch student and sort in ascending order by name field.

```
SELECT * FROM student order by name;
```

[Become PHP Full Stack Web Developer in Just 30 Days](#)

mysqli - SELECT Query

ID: 45
NAME: AMIT
ID: 100
NAME: ELLY
ID: 12
NAME: HENRY
ID: 70
NAME: JULIE
ID: 40
NAME: TEST

[Live Preview](#)

5.3 Form to Add, Edit, Update and Delete

Write a Student form to Add, Edit and Delete the Entries from MySQL DB.

Credentials:

Database Name: studentdb
Table Name: student
Username: root
password: root

[Become PHP Full Stack Web Developer in Just 30 Days](#)

5 mySQL & PHP with mysqli

The screenshot shows the phpMyAdmin interface. At the top, it displays "Server: MySQL:3306", "Database: schooldb", and "Table: student". Below this, there are tabs for "Browse", "Structure", "SQL", "Search", "Insert", "Export", "Import", "Privileges", and "More". A red box highlights the "Search" tab. A red arrow points from the "Search" tab to the "SELECT * FROM `student`" query in the SQL panel. Another red box highlights the "student" table in the database tree on the left. A red arrow points from this box to the table entries in the results panel. The results panel shows the following data:

+ Options	ID	NAME
	10	JOHN
	70	AMIT
	45	Julie

Sample Example

[Download the Example](#)

[Become PHP Full Stack Web Developer in Just 30 Days](#)

mysqli - Table

DISPLAY - TABLE ENTRIES

ID	NAME
100	ELLY
70	JULIE
40	TEST
12	HENRY

ADD - TABLE ENTRIES

Student Form

ID:	<input type="text"/>
Name:	<input type="text"/>

EDIT - TABLE ENTRIES

#	ID	NAME	OPERATION
100	ELLY	<input type="button" value="Edit"/>	
70	JULIE	<input type="button" value="Edit"/>	
40	TEST	<input type="button" value="Edit"/>	
12	HENRY	<input type="button" value="Edit"/>	

DELETE - TABLE ENTRIES

#	ID	NAME	OPERATION
100	ELLY	<input type="button" value="Delete"/>	
70	JULIE	<input type="button" value="Delete"/>	
40	TEST	<input type="button" value="Delete"/>	
12	HENRY	<input type="button" value="Delete"/>	

[Live Preview](#)

Exercise 1

Exercise 1: Create your own form and do the Add, Edit and Delete Operation.

[Become PHP Full Stack Web Developer in Just 30 Days](#)