



Module 5

PHP Database Handling

Dr. L.M. Jenila Livingston

VIT Chennai

Database Handling

- **PHP can connect to**
 - MySQL/MySQLi (i-improved)
 - MSSQL
 - Access and other databases like Oracle, Postgre, SQLite etc
- **There are separate methods available for connecting to the databases**

MySQL

- **MySQL is the most popular open-source database system.**
 - MySQL is a database server
 - MySQL is ideal for both small and large applications
 - MySQL supports standard SQL

MySQL

- Databases are useful for storing information category wise.
- Data in MySQL is stored in database objects called **tables**.
- A table is a collection of related data entries and it consists of **columns and rows**.
- University db may have: “Department”, “Faculty_info”, “Student_info” “Placement_details”etc.

University Database

- Student Table

RegNo	SName	Mobile
19BCE1001	Rahul	9000333333
19BCE1003	Liana	9000044444
19BAI1001	Harleen	9000044455

Part A: Steps to handle MySQL DB

1. Start MySQL Server
2. Go to localhost/phpmyadmin
3. Create User Account
4. Create Database
5. Create Tables
6. PHP Coding

1. Start MySQL server

XAMPP Control Panel v3.2.2 [Compiled: Nov 12th 2015]

XAMPP Control Panel v3.2.2

Service	Module	PID(s)	Port(s)	Actions
<input type="checkbox"/>	Apache	7828 9984	80, 443	Stop Admin Config Logs
<input type="checkbox"/>	MySQL	8296	3306	Stop Admin Config Logs
<input type="checkbox"/>	FileZilla			Start Admin Config Logs
<input type="checkbox"/>	Mercury			Start Admin Config Logs
<input type="checkbox"/>	Tomcat			Start Admin Config Logs

Modules

Config

Netstat

Shell

Explorer

Services

Help

Quit

2. localhost/phpmyadmin

8

localhost / 127.0.0.1 | phpMyAdmin

localhost/phpmyadmin/server_databases.php

Server: 127.0.0.1

Databases SQL Status User accounts Export Import

Databases

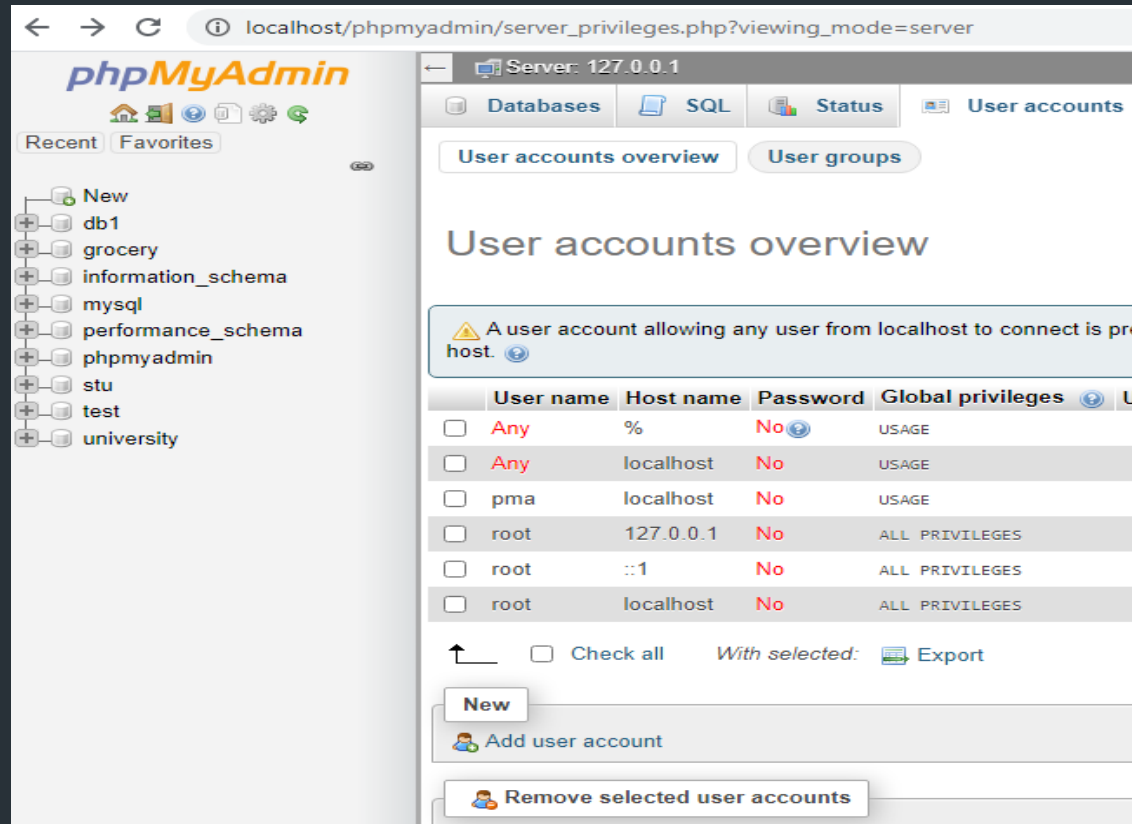
Create database

Database name latin1_swedish_ci Create

	Database	Collation	Action
<input type="checkbox"/>	db1	latin1_swedish_ci	Check privileges
<input type="checkbox"/>	grocery	latin1_swedish_ci	Check privileges
<input type="checkbox"/>	information_schema	utf8_general_ci	Check privileges

3. Create User Account

9

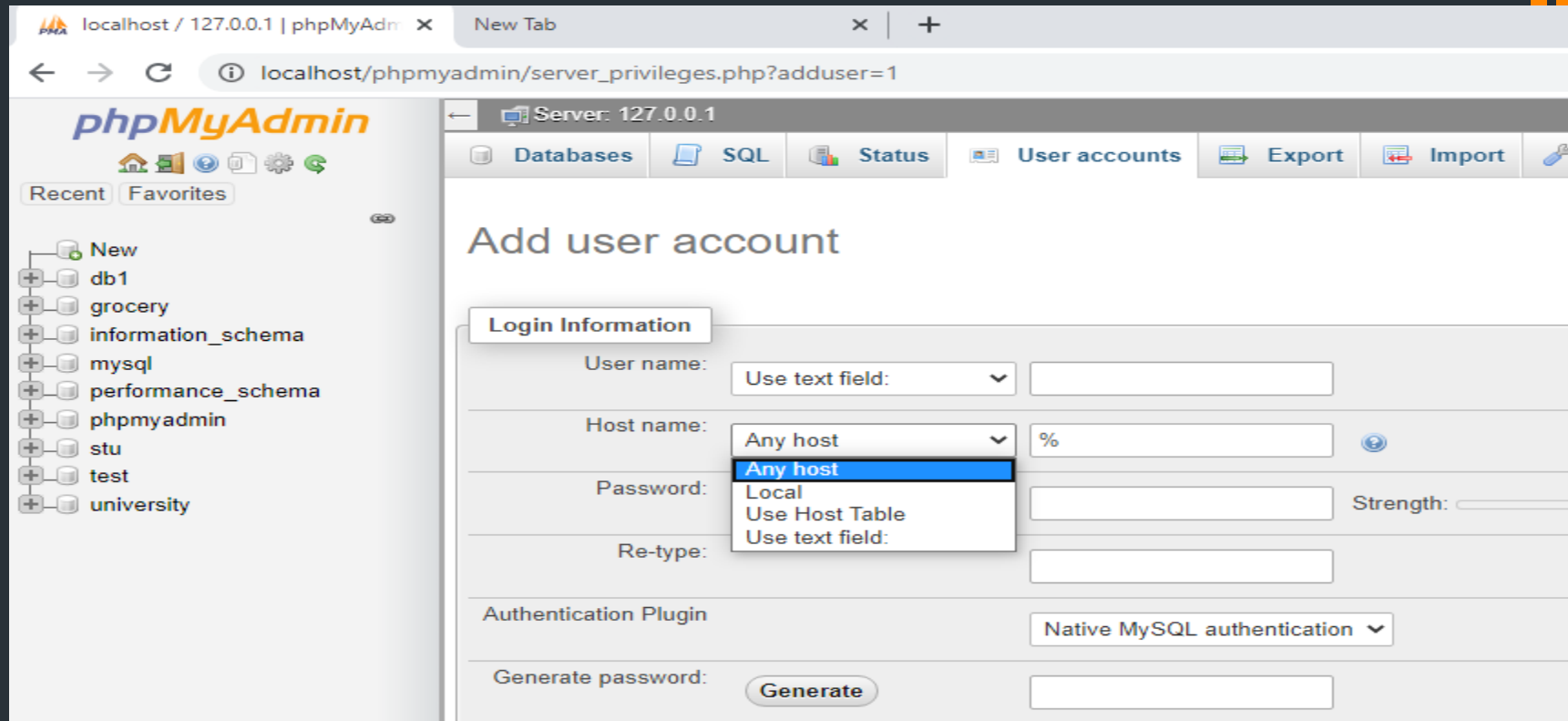


The screenshot shows the phpMyAdmin interface for managing user accounts. The left sidebar displays a tree view of databases and schemas, including 'New', 'db1', 'grocery', 'information_schema', 'mysql', 'performance_schema', 'phpmyadmin', 'stu', 'test', and 'university'. The main content area is titled 'User accounts overview' and features a table of existing user accounts. A warning message at the top states: 'A user account allowing any user from localhost to connect is present on this host.' The table lists users with columns for checkboxes, User name, Host name, Password, and Global privileges. Below the table, there are controls to 'Check all', 'Export', 'Add user account', and 'Remove selected user accounts'.

	User name	Host name	Password	Global privileges
<input type="checkbox"/>	Any	%	No	USAGE
<input type="checkbox"/>	Any	localhost	No	USAGE
<input type="checkbox"/>	pma	localhost	No	USAGE
<input type="checkbox"/>	root	127.0.0.1	No	ALL PRIVILEGES
<input type="checkbox"/>	root	::1	No	ALL PRIVILEGES
<input type="checkbox"/>	root	localhost	No	ALL PRIVILEGES

User Account

10



The screenshot shows the phpMyAdmin interface in a web browser. The browser's address bar displays the URL `localhost/phpmyadmin/server_privileges.php?adduser=1`. The phpMyAdmin header includes the logo and navigation tabs: Databases, SQL, Status, User accounts, Export, and Import. The left sidebar shows a tree view of databases, with 'New' at the top and others like 'db1', 'grocery', 'information_schema', 'mysql', 'performance_schema', 'phpmyadmin', 'stu', 'test', and 'university' below. The main content area is titled 'Add user account' and contains a 'Login Information' section. This section has four rows of input fields: 'User name:', 'Host name:', 'Password:', and 'Re-type:'. The 'Host name:' dropdown menu is open, showing options: 'Any host' (highlighted), 'Local', 'Use Host Table', and 'Use text field:'. To the right of the 'Password:' field is a 'Strength' indicator. Below the 'Re-type:' field is the 'Authentication Plugin' dropdown, currently set to 'Native MySQL authentication'. At the bottom, there is a 'Generate password:' section with a 'Generate' button and an empty text field.

localhost / 127.0.0.1 | phpMyAdmin x New Tab

localhost/phpmyadmin/server_privileges.php?adduser=1

phpMyAdmin

Recent Favorites

- New
- db1
- grocery
- information_schema
- mysql
- performance_schema
- phpmyadmin
- stu
- test
- university


Server: 127.0.0.1

Databases SQL Status User accounts Export Import

Add user account

Login Information

User name: Use text field:

Host name: Any host % 

Any host
Local
Use Host Table
Use text field:

Password: Strength:

Re-type:

Authentication Plugin Native MySQL authentication

Generate password:

User Account

11

localhost / 127.0.0.1 | phpMyAdmin x New Tab

localhost/phpmyadmin/server_privileges.php?adduser=1

phpMyAdmin

Recent Favorites

- New
- db1
- grocery
- information_schema
- mysql
- performance_schema
- phpmyadmin
- stu
- test
- university

Server: 127.0.0.1

Databases SQL Status User accounts Export Import Settings

Add user account

Login Information

User name:

Use text field:

Host name:

Any host %

Password:

No Password Strength: Strong

Re-type:

No Password
Use text field:

Authentication Plugin

Native MySQL authentication

Generate password:

G6TGjp9moEfAcfLg

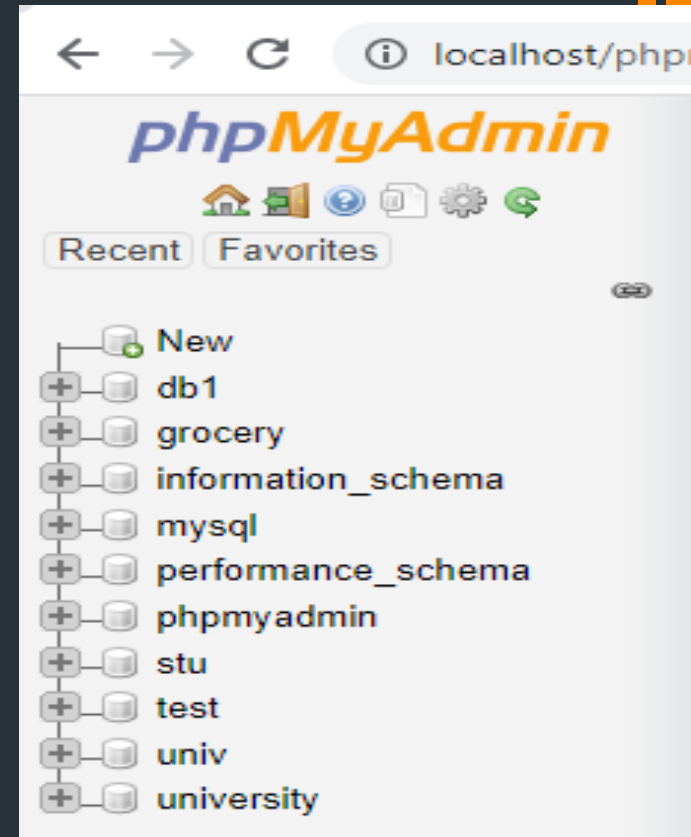
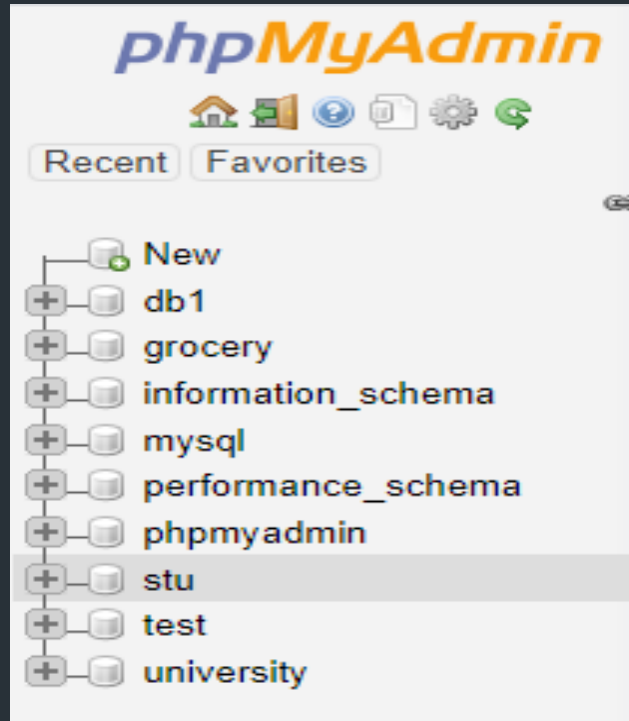
4. Create Database



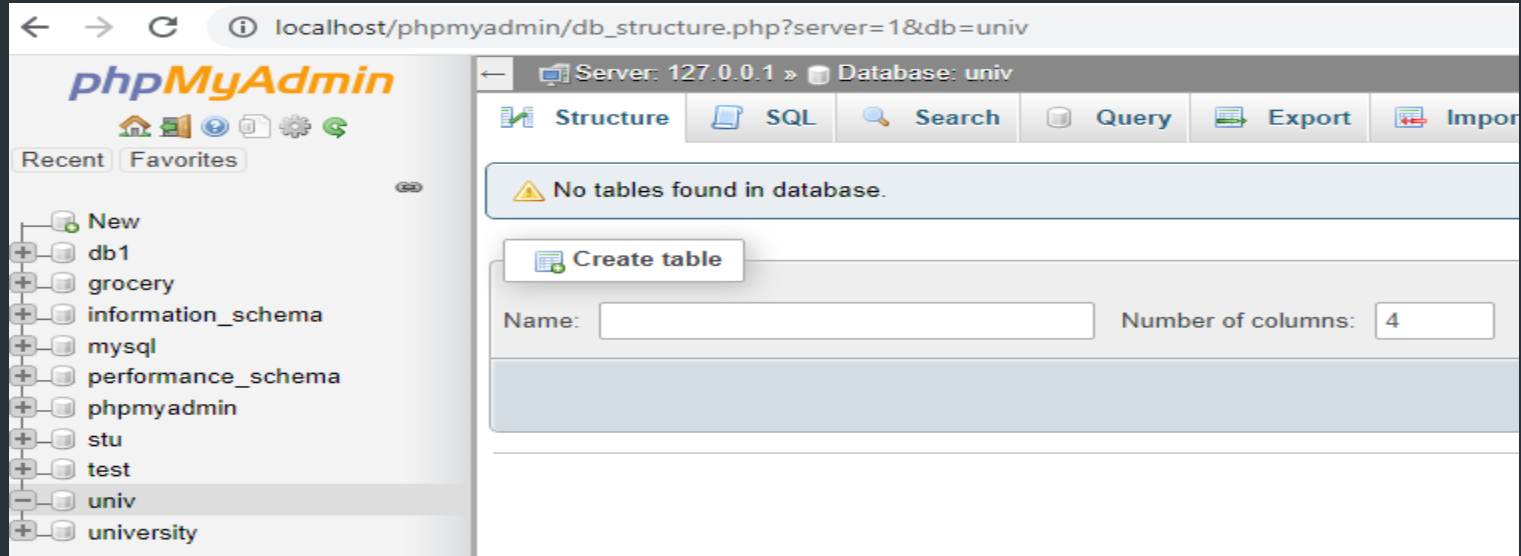
Univ database

13

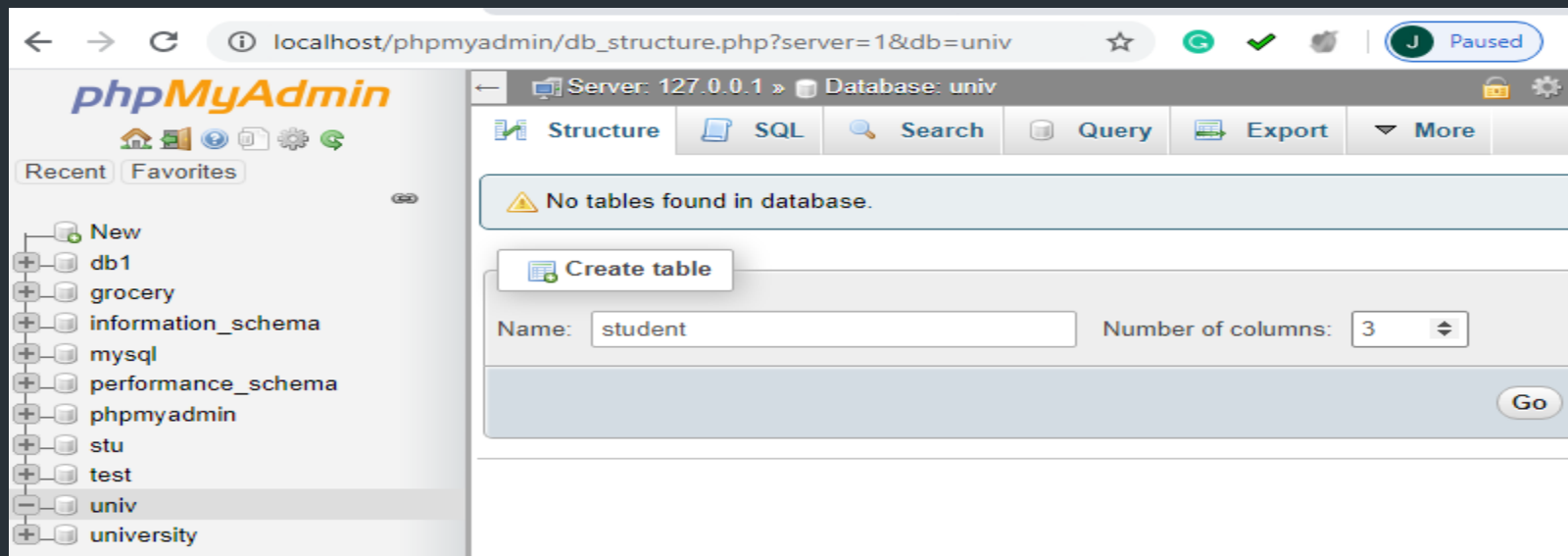
- before and after creation



5. Create Table



student table



Adding attributes

Server: 127.0.0.1 » Database: univ » Table: student

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Tracking](#)
[Triggers](#)

Table name: Add column(s)

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index
<input type="text"/> <small>Pick from Central Columns</small>	INT	<input type="text"/>	None	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---
<input type="text"/> <small>Pick from Central Columns</small>	INT	<input type="text"/>	None	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---
<input type="text"/> <small>Pick from Central Columns</small>	INT	<input type="text"/>	None	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>	---

Table comments:

Collation:

Storage Engine:

PARTITION definition:

Partition by: ()

Partitions:

Adding attributes

Server: 127.0.0.1 » Database: univ » Table: student

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Tracking](#)
[Triggers](#)

Table name: Add column(s)

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index
<input type="text" value="regno"/> <small>Pick from Central Columns</small>	CHAR	10	None			<input type="checkbox"/>	---
<input type="text" value="sname"/> <small>Pick from Central Columns</small>	VARCHAR	20	None			<input type="checkbox"/>	--- PRIMARY UNIQUE INDEX FULLTEXT SPATIAL ---
<input type="text" value="mobile"/> <small>Pick from Central Columns</small>	INT	10	None			<input type="checkbox"/>	---

student table schema

	#	Name	Type
<input type="checkbox"/>	1	regno	char(10)
<input type="checkbox"/>	2	sname	varchar(20)
<input type="checkbox"/>	3	mobile	int(11)

Insert values

The screenshot shows the phpMyAdmin web interface. On the left is a sidebar with a tree view of databases and tables. The 'univ' database is selected, and the 'student' table is highlighted. The main panel on the right shows the 'Structure' tab for the 'student' table. The table has 1 column and a 'Sum' row. The 'Insert' button is highlighted in the table's action menu. Below the table, there are options to 'Check all' and 'With selected:'. At the bottom, there is a 'Create table' button and fields for 'Name:' and 'Number of columns: 4'.

Server: 127.0.0.1 » Database: univ

Structure SQL Search Query Export Import

Filters

Containing the word:

Table	Action
<input type="checkbox"/> student	Browse Structure Search Insert Empty Drop
1 table	Sum

☐ Check all With selected:

[Print](#) [Data dictionary](#)

[Create table](#)

Name: Number of columns:

Insert values

20

← Server: 127.0.0.1 » Database: univ » Table: student

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#)

Column	Type	Function	Null	Value
regno	char(10)	<input type="text"/>		<input type="text"/>
sname	varchar(20)	<input type="text"/>		<input type="text"/>
mobile	int(10)	<input type="text"/>		<input type="text"/>

☒ Ignore

Column	Type	Function	Null	Value
regno	char(10)	<input type="text"/>		<input type="text"/>
sname	varchar(20)	<input type="text"/>		<input type="text"/>
mobile	int(10)	<input type="text"/>		<input type="text"/>

Student table after data Insertion

21

```
SELECT * FROM `student`
```

☐ Show all | Number of rows

+ Options

regno	sname	mobile
121	Jenila	9999
122	Hello	9999

6. Coding – db.php

22

```
<?php
$con = mysqli_connect("localhost","root","");
mysqli_select_db($con,"univ");
$sel='select * from student';
$sq=mysqli_query($con,$sel);

while ($row=mysqli_fetch_array($sq))
{
    echo $row['regno']." ".
        $row['sname']." ".
        $row['mobile'].'<br>';
}
?>
```

Part B: MySQLi Functions

1. `mysqli_connect()`
2. `mysqli_close()`
3. `mysqli_select_db()`
4. `mysqli_query()`
5. `mysqli_fetch_array()`
6. `mysqli_num_rows()`

1. MySQL dbserver Connection (mysqli_connect)

24

This method is used for connecting to MySQLi using a userid and password

Syntax:

`mysqli_connect(dbserver,userid,password)`

Example:

```
$con = mysqli_connect("localhost","root","root");
```

- This gets a connection to the local mysql server using the credentials root and root

```
$con = mysqli_connect("localhost","root","");
```


Parameters

Parameter	Description
servername	Optional. Specifies the server to connect to. Default value is "localhost"
username	Optional. Specifies the username to log in with. Default value is the name of the user that owns the server process
password	Optional. Specifies the password to log in with. Default is ""

Example

26



We store the connection in a variable (\$con) for later use in the script. The "die" part will be executed if the connection fails.

```
<?php
$con = mysqli_connect("localhost","jenila","leni");
if (!$con)
{
    die('Could not connect: ' . mysqli_error());
}
?>
```

Mysql_error()

- If server cannot be connected, it will throw an error stating the problem

- 1049: Unknown database

1046: No database selected

2. MySQLi Close

- To close a db connection we have close method
 - `mysqli_close(connection);`
- Example:
 - `mysqli_close($con);`

3. Connect Database (mysqli_select_db)

- Syntax:

`mysqli_select_db(connection,dbname)`

`mysqli_select_db($con,"univ");`

This will select the univ db under the server localhost

4. Working with Queries

30

To get PHP to execute the SQL statements we must use the `mysql_query()` function. This function is used to send a query or command to a MySQL connection.

Syntax:

- `mysql_query($con,"sqlquery");`
or
- `$sql="Any SQL Query";`
`mysql_query($con,$sql);`

Product Table

Productcode	ProductName	Cost	Manufacturer
P001	Cheese	150	Creamcenter
P002	Salt	15	Tata
P003	Butter	40	XYZ

4.1 Create database

- The CREATE DATABASE statement is used to create a database in MySQL.

Syntax:

```
$sql=CREATE DATABASE database_name;
```

```
$result=mysqli_query($con,$sql)
```


Create a Database

33

Example :The following example creates a database called "grocery"

```
$sql="CREATE database grocery";  
$result=mysqli_query($con,$sql) or  
die(mysqli_error());  
if ($result)  
echo "database created successfully";
```

4.2 Create a Table

- The **CREATE TABLE** statement is used to create a table in MySQL.

Syntax:

```
CREATE TABLE table_name
(
    column_name1 data_type, column_name2
    data_type, column_name3    data_type,
    ....
)
```

- We must add the **CREATE TABLE** to the statement **mysql_query()** function to execute the command.

Create a Table

Example:

```
$sql="create table product(productcode char(7),  
productname varchar(20),cost int, Manufacturer  
varchar2(15))";
```

```
mysqli_query($con,$sql);
```

4.3 Insert Data into a Table

36

- The **INSERT INTO** statement is used to add new records to a database table.

Syntax:

- **INSERT INTO** statement is in two forms.
- The first form doesn't specify the column names where the data will be inserted, only their values:

INSERT INTO table_name VALUES (value1, value2, ...)

- The second form specifies both the column names and the values to be inserted:

**INSERT INTO table_name (column1, column2, ...)
VALUES (value1, value2, ...)**

Insert Data into a Table

Example:

```
sql= "insert into product  
      (productcode,productname,cost)  
values ('P001','Cheese',150)";
```

```
mysqli_query($con,$sql)
```

Insert Data from a Form

Create an HTML form that can be used to add new records to the “product” table.

```
<html> <body>
<form action="fromform.php" method="post">
Product Code: <input type="text" name="pcode">
Product Name: <input type="text" name="pname">
Cost: <input type="text" name="cost">
<input type="submit">
</form> </body>
</html>
```

Insert values from a form

```
$sql="insert into product  
    (productcode,productname,cost)  
    values('$_POST[pcode]',  
          '$_POST[pname]',  
          '$_POST[cost]')";  
  
mysqli_query ($con,$sql);  
echo "No. of records added: 1";
```

4.4 Update data in a Table

- The UPDATE statement is used to update existing records in a table.
- Syntax:

UPDATE table_name

SET column1=value, column2=value2,...

WHERE some_column=some_value

- Example:

```
mysqli_query($con,"update product set  
cost=300 where productcode='P001'");
```


4.5 Delete data from a table

- The DELETE FROM statement is used to delete records from a database table.
- Syntax:

DELETE FROM table_name WHERE column_name = value

- Example:

```
mysqli_query($con,"delete from product  
where productcode='P002'");
```

4.6 Select Data from a Table

- The SELECT statement is used to select data from a table.

- Syntax:

SELECT column_name(s) FROM table_name;

SELECT column_name(s) FROM table_name where condition;

5. Select Data from a Table (`mysqli_fetch_array`)

43

- `$rs=mysqli_query($con,select stmt);`
- The rows of select statement will be stored in `$rs`
- `$row = mysqli_fetch_array($rs);`
 - This will fetch a row and store in `$row`
 - Values can be accessed like - `$row["ID"]` – returns value of column ID in the fetched row.

Using Associative array

44

```
$result=mysqli_query($con,"select * from product where  
    productcode='P001'");  
while ($row=mysqli_fetch_array($result))  
{  
    echo $row['productcode']." ".  
        $row['productname']." ".  
        $row['cost'].'<br>';  
}
```

Using Numerical array

```
$result=mysqli_query($con,"select * from products where  
prodcode='CHE'");
```

```
while ($row=mysqli_fetch_array($result))  
{  
    echo $row[0]. " ". $row[1]. " ". $row[2]. '<br>';  
}
```

Using orderby

46

```
$result=mysqli_query("select * from product
    order by cost desc");
while ($row=mysqli_fetch_array($result))
{
    echo $row['productcode']."".
        $row['prodname']. " ".
        $row['cost'].'<br>';
}
```

DB program

47

```
<?php
$con = mysqli_connect("localhost","root","");
mysqli_select_db($con,"univ");
$sel='select * from student';
$sq=mysqli_query($con,$sel);
while ($row=mysqli_fetch_array($sq))
{ echo $row['regno']." ".
    $row['sname']." ".
    $row['mobile'].'<br>';
}
?>
```

6. mysqli_num_rows

- Return the number of rows in a result set:

- Syntax:

```
$rowcount=mysqli_num_rows($result);
```


Functions Covered

- `mysqli_connect()`
- `mysqli_select_db()`
- `mysqli_close()`
- `mysqli_query()`
- `mysqli_fetch_array()`
- `mysqli_num_rows()`

Part C:

Write the table data into a file

- Write the table data into a file
- Write the file data into a table

To write the table data into a file – dbdisplayoutput.php

- `$result=mysqli_query($con,"select * from products
into outfile 'D:/products.txt'");`

To write the data file data into table - ⁵²

dbdisplaywrite.php

```
$result=mysqli_query($con,"load data local infile  
'D:\products.txt' into table abc");
```

```
$result=mysqli_query($con,"load data local infile  
'D:\products.txt' into table abc(productname,productcode,cost)");
```

