Database Operations

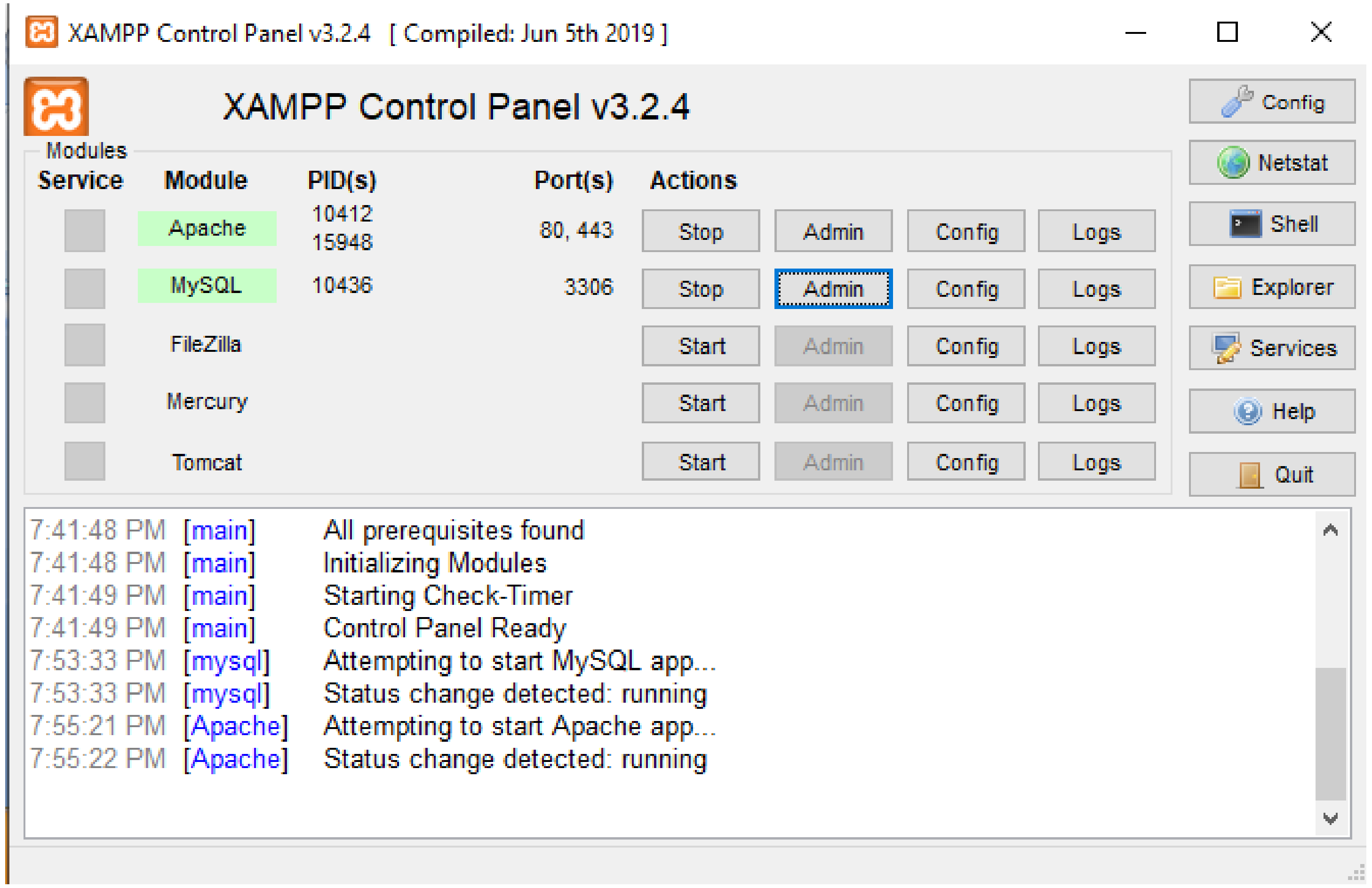
Unit - V

5.1 Introduction to MySQL

- MySQL is an open-source relational database management system.
- Its name is a combination of “My”, the name of co-founder Michael Widenius’s daughter, and “SQL”, the abbreviation for Structured Query Language.
- It is written in C and C++.

5.1 Introduction to MySQL

- Create a Database: Open XAMPP Control panel and start MySQL service, then click on Admin button to open phpMyAdmin user interface.

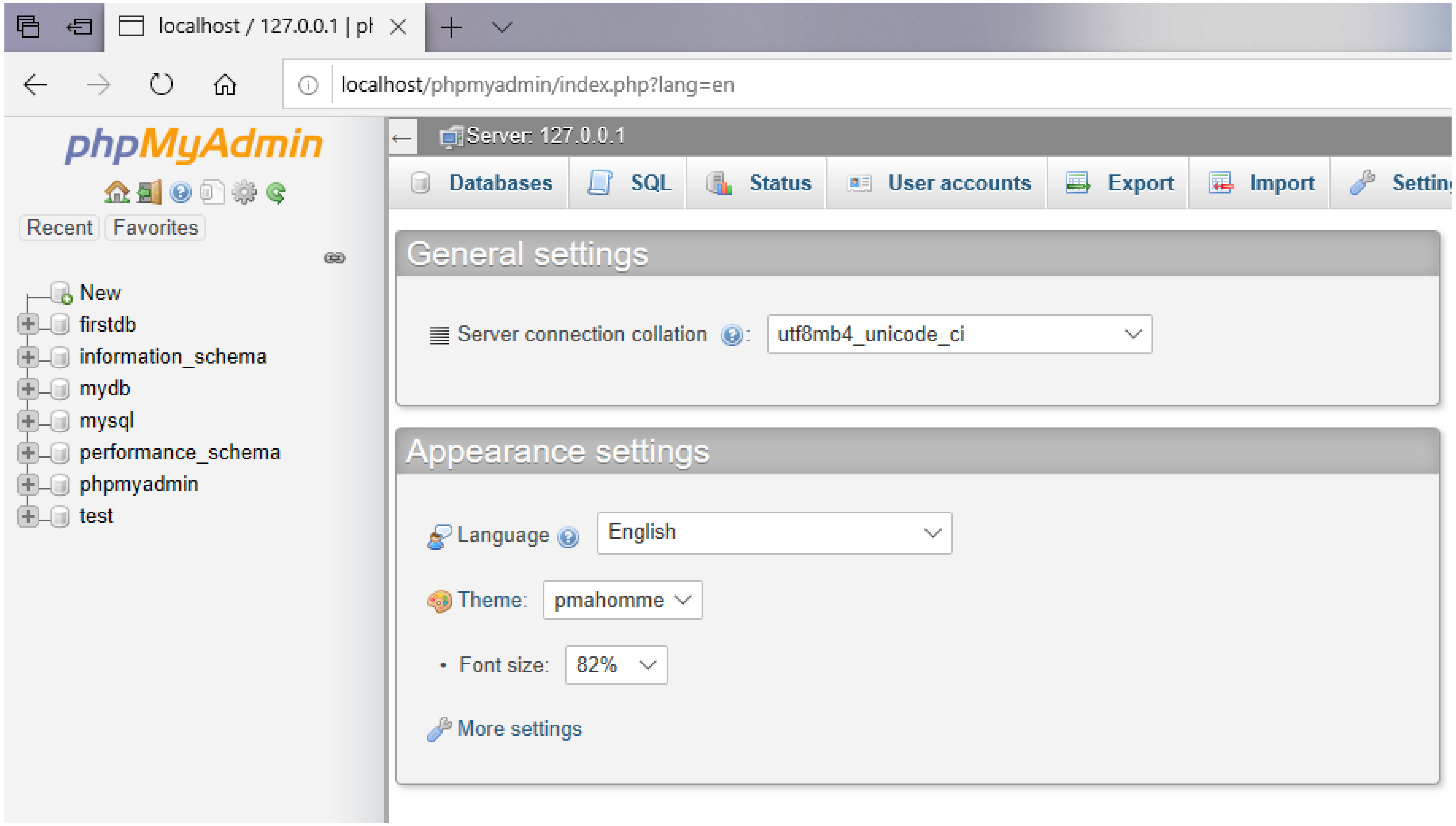

XAMPP Control Panel v3.2.4 [Compiled: Jun 5th 2019]
XAMPP Control Panel v3.2.4
Config
Netstat
Shell
Explorer
Services
Help
Quit
Modules

Service	Module	PID(s)	Port(s)	Actions			
	Apache	10412 15948	80, 443	Stop	Admin	Config	Logs
	MySQL	10436	3306	Stop	Admin	Config	Logs
	FileZilla			Start	Admin	Config	Logs
	Mercury			Start	Admin	Config	Logs
	Tomcat			Start	Admin	Config	Logs

```
7:41:48 PM  [main]    All prerequisites found
7:41:48 PM  [main]    Initializing Modules
7:41:49 PM  [main]    Starting Check-Timer
7:41:49 PM  [main]    Control Panel Ready
7:53:33 PM  [mysql]   Attempting to start MySQL app...
7:53:33 PM  [mysql]   Status change detected: running
7:55:21 PM  [Apache]  Attempting to start Apache app...
7:55:22 PM  [Apache]  Status change detected: running
```

5.1 Introduction to MySQL

- Create a Database: In order to work with database, click on the Database tab.



General settings

Server connection collation: utf8mb4_unicode_ci

Appearance settings

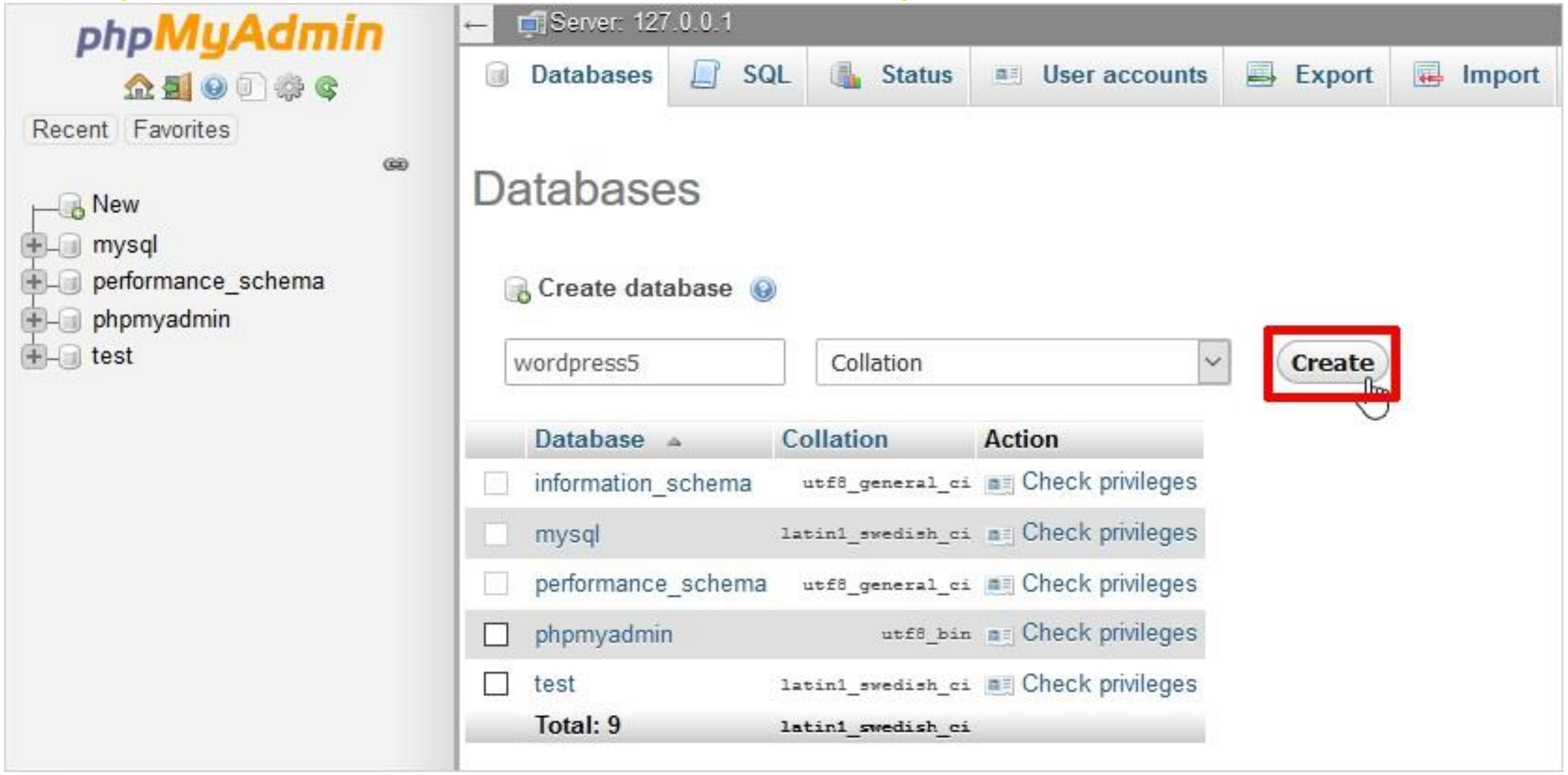
Language: English

Theme: pmahomme

Font size: 82%

5.1 Introduction to MySQL

- Create a Database: now you should see the option to create a Database, Write the Database name and click the 'Create' button. By default the host name is 'localhost', MySQL user is 'root' and have no password.



Databases

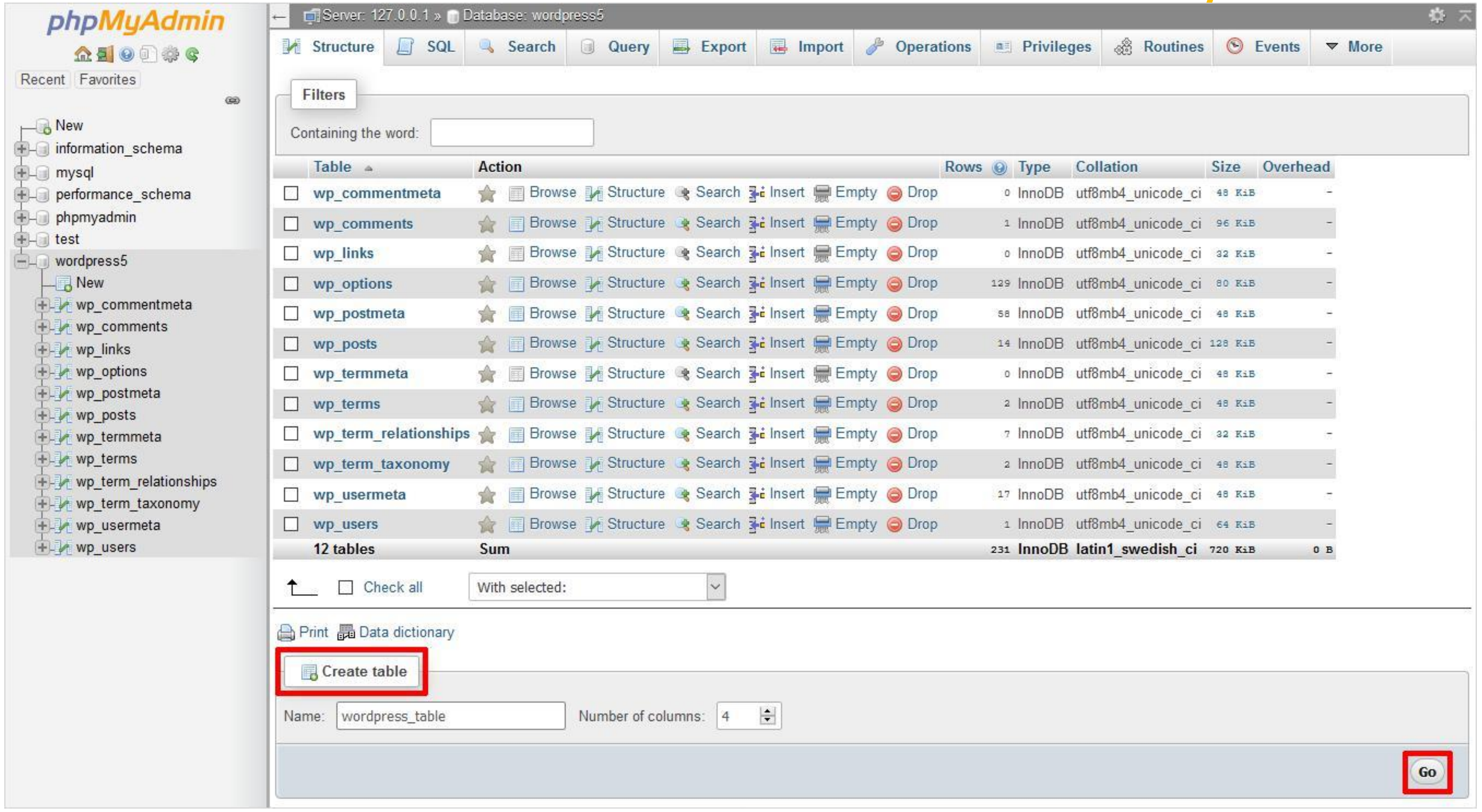
Create database

wordpress5 | Collation | Create

	Database	Collation	Action
☐	information_schema	utf8_general_ci	Check privileges
☐	mysql	latin1_swedish_ci	Check privileges
☐	performance_schema	utf8_general_ci	Check privileges
☐	phpmyadmin	utf8_bin	Check privileges
☐	test	latin1_swedish_ci	Check privileges
	Total: 9	latin1_swedish_ci	

5.1 Introduction to MySQL

- **Create a Table:** To create a table, first select the database and then click on the 'Structure' tab. Below the list at the bottom of the page you will see create table wizard, add the table name and total number of columns you need and click the 'Go' button.



phpMyAdmin
Recent Favorites
Server: 127.0.0.1 » Database: wordpress5
Structure SQL Search Query Export Import Operations Privileges Routines Events More
Filters

New
information_schema
mysql
performance_schema
phpmyadmin
test
wordpress5
New
wp_commentmeta
wp_comments
wp_links
wp_options
wp_postmeta
wp_posts
wp_termmeta
wp_terms
wp_term_relationships
wp_term_taxonomy
wp_usermeta
wp_users

Containing the word:

Table	Action	Rows	Type	Collation	Size	Overhead
wp_commentmeta	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48 KiB	-
wp_comments	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_ci	96 KiB	-
wp_links	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32 KiB	-
wp_options	Browse Structure Search Insert Empty Drop	129	InnoDB	utf8mb4_unicode_ci	80 KiB	-
wp_postmeta	Browse Structure Search Insert Empty Drop	58	InnoDB	utf8mb4_unicode_ci	48 KiB	-
wp_posts	Browse Structure Search Insert Empty Drop	14	InnoDB	utf8mb4_unicode_ci	128 KiB	-
wp_termmeta	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48 KiB	-
wp_terms	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_unicode_ci	48 KiB	-
wp_term_relationships	Browse Structure Search Insert Empty Drop	7	InnoDB	utf8mb4_unicode_ci	32 KiB	-
wp_term_taxonomy	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_unicode_ci	48 KiB	-
wp_usermeta	Browse Structure Search Insert Empty Drop	17	InnoDB	utf8mb4_unicode_ci	48 KiB	-
wp_users	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_ci	64 KiB	-
12 tables	Sum	231	InnoDB	latin1_swedish_ci	720 KiB	0 B

Check all With selected:

Print Data dictionary

Create table

Name: wordpress_table Number of columns: 4

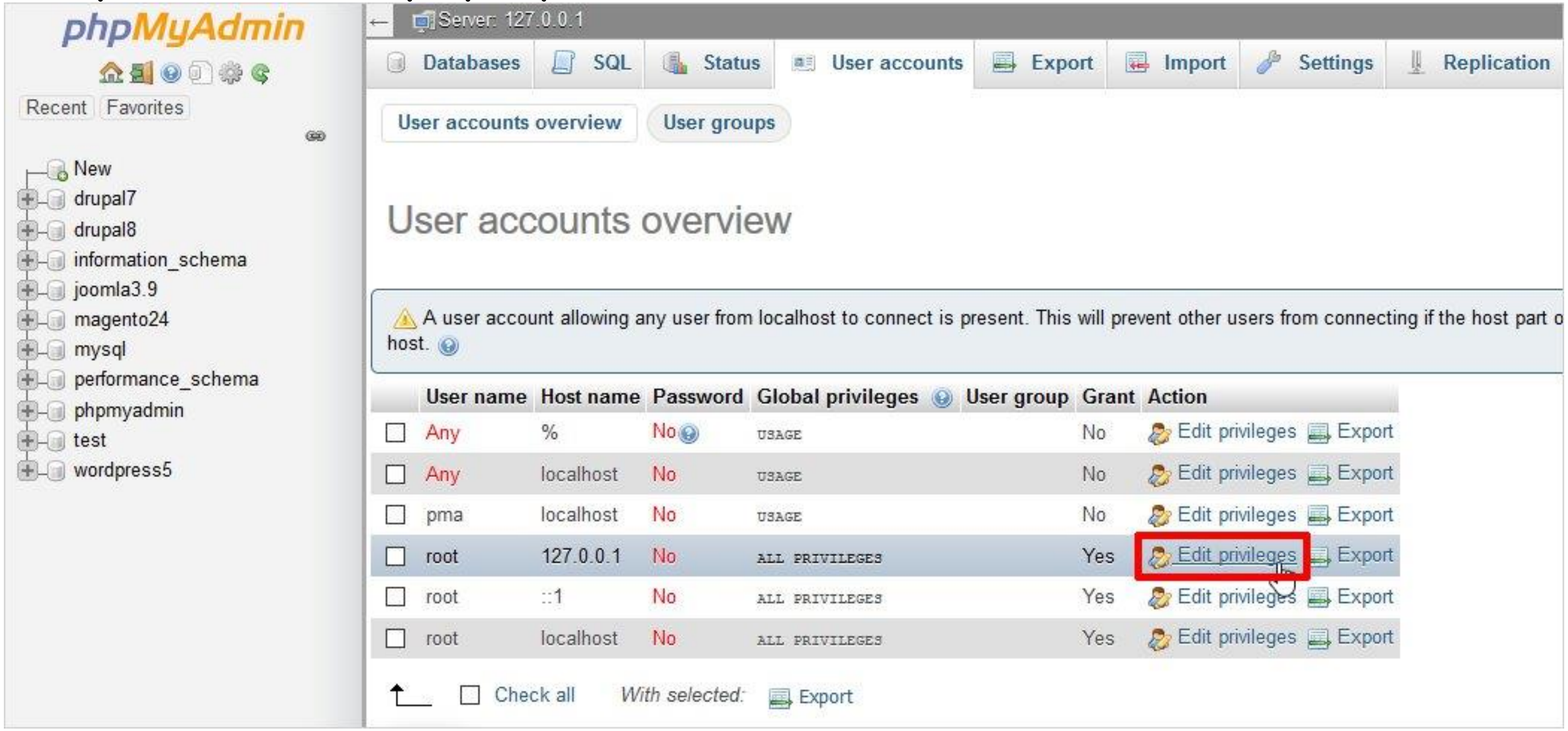
Go

5.1 Introduction to MySQL

- Set password to phpMyAdmin:
  - For phpMyAdmin, by default the username is 'root' and the password remains empty.
  - To change the password, go to phpMyAdmin home page.
  - Click on 'User Accounts' option at the top of the page.
  - Now click the 'Edit Privilidges' under 'Actions' option for the username 'root' and Hostname 'localhost'.
  - Now choose the third tab 'Change password' and type your password in the provided field, retype the password to confirm it and then finally click on the 'Go' key to conclude the process.

5.1 Introduction to MySQL

- Set password to phpMyAdmin:



User accounts overview

A user account allowing any user from localhost to connect is present. This will prevent other users from connecting if the host part o
host.

	User name	Host name	Password	Global privileges	User group	Grant	Action
☐	Any	%	No	USAGE		No	Edit privileges Export
☐	Any	localhost	No	USAGE		No	Edit privileges Export
☐	pma	localhost	No	USAGE		No	Edit privileges Export
☐	root	127.0.0.1	No	ALL PRIVILEGES		Yes	Edit privileges Export
☐	root	::1	No	ALL PRIVILEGES		Yes	Edit privileges Export
☐	root	localhost	No	ALL PRIVILEGES		Yes	Edit privileges Export

☐ Check all *With selected:* Export

5.2 Connecting to a MySQL Database

- MySQL : MySQLi : PDO
  - These are the APIs of PHP to access MySQL databases and tables.
  - Developers can choose either of them for their project.
  - MySQL was the main extension that was designed to help PHP applications send and receive data from MySQL database.
  - However MySQL has been deprecated and removed as of PHP7 and its newever versions.

5.2 Connecting to a MySQL Database

- MySQL : MySQLi : PDO
  - In MySQLi, i stands for Improved. It is an improved version of MySQL with procedural and object oriented approach.
  - PDO (PHP Data Objects): the main advantage of using PDO is that it supports, and provides a uniform method of access to 11 different databases.
  - PDO supported databases are:

    CUBRID, MS SQL Server, Firebird/Interbase, IBM, Informix, MySQL, Oracle, ODBC and DB2, PostgreSQL, SQLite, 4D
  - *However PDO doesn't allow the usage of all features available in present version of the MySQL server.*

5.2 Connecting to a MySQL Database

- MySQL : MySQLi : PDO

Parameters	MySQL	MySQLi	PDO

Connection	$connection_link = mysql_connect("host", "username", "password");
	
	mysql_select_db("database_name", $connection_link);
	
	mysql_set_charset('UTF-8', $connection_link);
		$mysqli_db = new mysqli('host', 'username', 'password', 'database_name');	$pdo = new PDO('mysql:host=host; dbname=database_name; charset=utf8',
			'username', 'password');

Error Handling	Error handling in MySQL is not considered to be a good approach.	Error handling in MySQLi is a bit easier.	PDO has the best error handling methods. It also provides error modes for error handling
Data Fetching	General programming loops such as for, or while can be used in MySQL.	Same as MySQL, code however will be a bit different.	PDO provides many built-in statements: fetchAll(), fetchColumn() etc.
API support	MySQL provides a Procedural way.	MySQLi provides both Procedural as well as Object Oriented way	PDO provides an Object Oriented approach

Functions to fetch data from Database

- mysqli_fetch_row()
- mysqli_fetch_assoc()
- mysqli_fetch_array()
- mysqli_fetch_object()
- mysqli_fetch_lengths()
- mysqli_fetch_field()

Functions to fetch data from Database

- mysqli_fetch_row():
  - This function will fetch data about the single row with which the row pointer currently exists.
  - After fetching the entire row details, it will be returned as an array with number indices corresponding to the MySQL field offset.
  - If no results found for the query, then mysqli_fetch_row() will return NULL.

Functions to fetch data from Database

- mysqli_fetch_row():

```
$conn = mysqli_connect("localhost", "root", "test", "blog_samples") or
die("Connection Error: " . mysqli_error($conn));

$query = "SELECT * from Users"; $result = mysqli_query($conn, $query) or
die(mysqli_error($conn));

$row = mysqli_fetch_row($result);

print "<pre>";

print_r($row);

print "<pre>";
```

Output:Array([0] => 1[1] => admin[2] => admin123[3] => student

Functions to fetch data from Database

- mysqli_fetch_assoc():
  - This function is similar to the mysqli_fetch_row(), except that, it will return an array of row information containing column values are indexed with the column name.
  - So the result type is an associative array where each column name and values of a single row are associated together as name, value pairs.

```
Output: Array
(
[user_id] => 1
[user_name] => admin
[password] => admin123
[user_type] => student
)
```

Functions to fetch data from Database

- mysqli_fetch_array():
  - This MySQL fetch method returns resultant array with both indices.
  - That is, field offset and field name. So, it would be used most probably by having both option of indexing.
  - mysqli_fetch_array() accepts an optional argument for specifying resultant array index type and its possible values are,

Functions to fetch data from Database

- mysqli_fetch_array():
  - *MYSQLI_BOTH* – It is the default value that would be taken if no second argument is provided for this function. It will provide resultant array with both indices.
  - *MYSQLI_NUM* – With this option, mysqli_fetch_array() will return array with offset indices as same as mysqli_fetch_row().
  - *MYSQLI_ASSOC* – With this option, mysqli_fetch_array() will return array with name indices as same as mysqli_fetch_assoc().

Functions to fetch data from Database

- mysqli_fetch_array():

```
Array
(
[0] => 1
[user_id] => 1
[1] => admin
[user_name] => admin
[2] => admin123
[password] => admin123
[3] => student
[user_type] => student
)
```

Functions to fetch data from Database

- mysqli_fetch_object():
  - mysqli_fetch_object() function will return MySQL data with same structure as returned by mysqli_fetch_assoc(), but its type is different.
  - mysqli_fetch_object() returns object where as mysqli_fetch_assoc() returns array.
  - So, the way of accessing these data will also be differed.

echo $row->user_name;

Functions to fetch data from Database

- mysqli_fetch_lengths():

  - It is used to returns the length of the fields in the result.

  - It returns an array of integer that represents the size of each column or FALSE if fails.

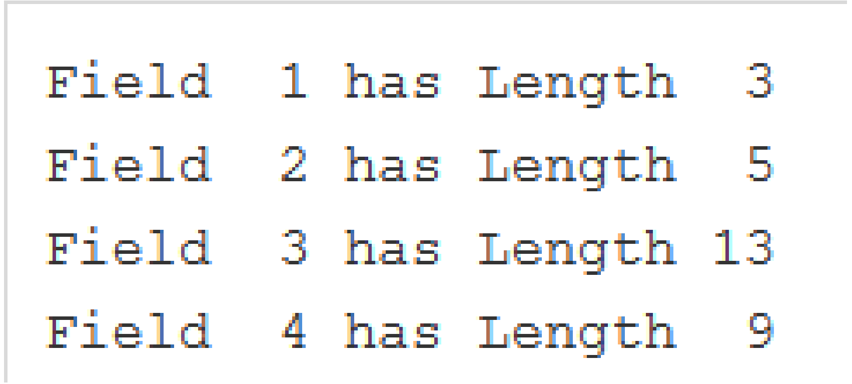
  - Parameter:

    Result: It specifies the result set identifier returned by mysqli_query(), mysqli_store_result() or mysqli_use_result()

Functions to fetch data from Database

- mysqli_fetch_lengths():

```
<?php
$link = mysqli_connect("localhost","my_user", "my_password", "world");
$query = "SELECT * from Country ORDER BY Code LIMIT 1";
if ($result = mysqli_query($link, $query)) {
    $row = mysqli_fetch_row($result);
    foreach (mysqli_fetch_lengths($result) as $i => $val) {
        printf("Field %2d has Length %2d\n", $i+1, $val);
    }
```



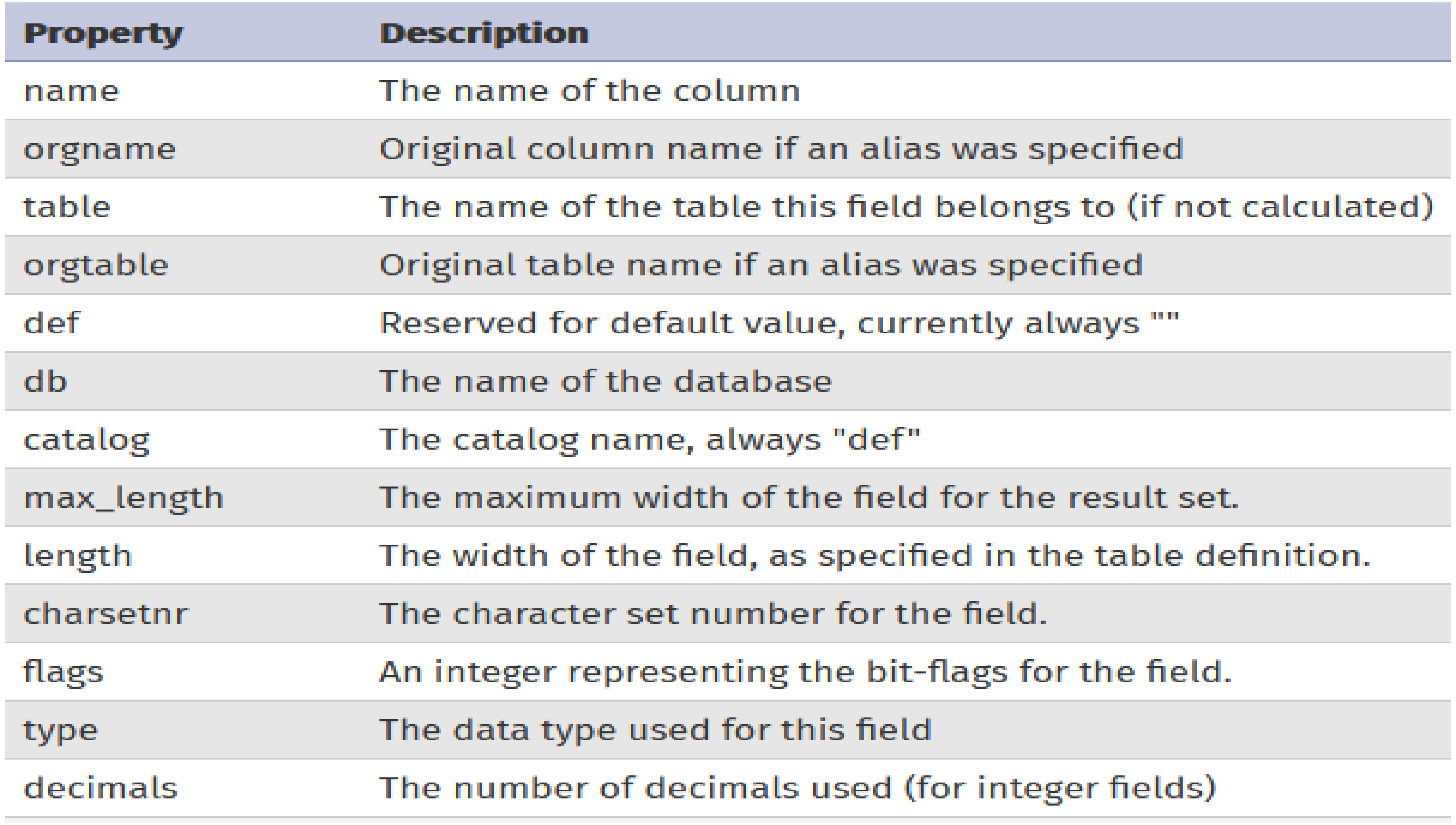
```
Field  1 has Length  3
Field  2 has Length  5
Field  3 has Length 13
Field  4 has Length  9
```

Functions to fetch data from Database

- mysqli_fetch_field():
  - It is used to retrieve the next field in the result set.
  - Returns the definition of one column of a result set as an object. Call this function repeatedly to retrieve information about all columns in the result set.
  - Returns an object which contains field definition information or **false** if no field information is available.

Functions to fetch data from Database

- mysqli_fetch_field():

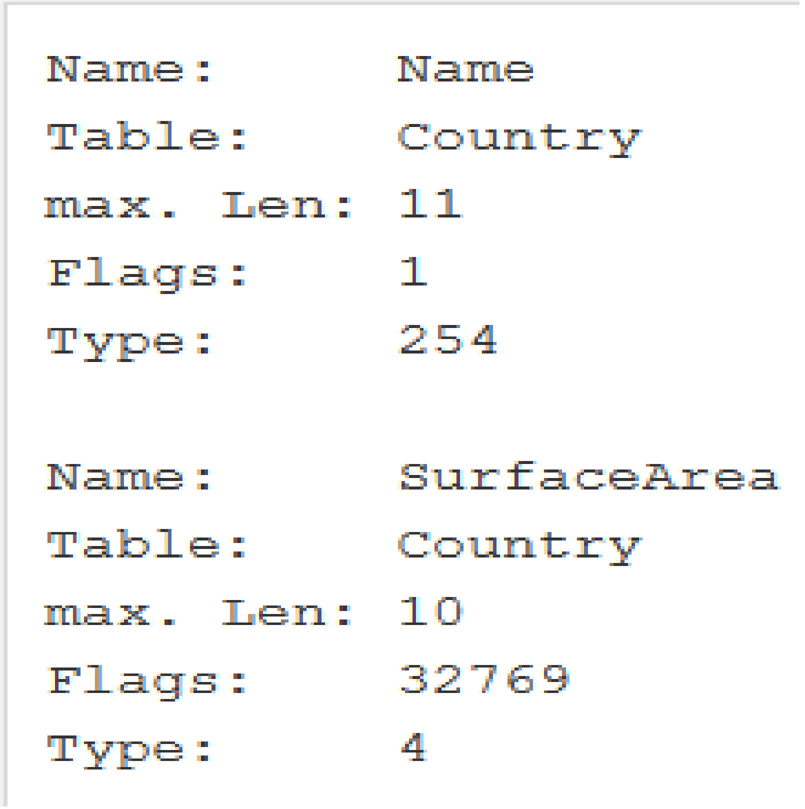


Property	Description
name	The name of the column
orgname	Original column name if an alias was specified
table	The name of the table this field belongs to (if not calculated)
orgtable	Original table name if an alias was specified
def	Reserved for default value, currently always ""
db	The name of the database
catalog	The catalog name, always "def"
max_length	The maximum width of the field for the result set.
length	The width of the field, as specified in the table definition.
charsetnr	The character set number for the field.
flags	An integer representing the bit-flags for the field.
type	The data type used for this field
decimals	The number of decimals used (for integer fields)

Functions to fetch data from Database

- mysqli_fetch_field():

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");
$query = "SELECT Name, SurfaceArea from Country ORDER BY Code LIMIT 5";
if ($result = mysqli_query($link, $query)) {
   while ($finfo = mysqli_fetch_field($result)) {
      printf("Name:     %s\n", $finfo->name);
      printf("Table:    %s\n", $finfo->table);
      printf("max. Len: %d\n", $finfo->max_length);
      printf("Flags:    %d\n", $finfo->flags);
      printf("Type:     %d\n\n", $finfo->type);
   }
```



```
Name:      Name
Table:     Country
max. Len:  11
Flags:     1
Type:      254

Name:      SurfaceArea
Table:     Country
max. Len:  10
Flags:     32769
Type:      4
```

5.2 Connecting to a MySQL Database

Login.php

```
<html>
<head>
<title>Login Form</title>
</head>
<body>
<form name="login_form" action="login_process.php" method="post">
<p><label>Username: </label>
<input type="text" name="username" id="username"/></p>
<p><label>Password: </label>
<input type="password" name="pass" id="pass"/></p>
<p><input type="submit" id='btn' value="Login"></p>
</form></body>
</html>
```

5.2 Connecting to a MySQL Database

Login_process.php

```
<?php
 //Accept values from Login page
 if($_SERVER['REQUEST_METHOD']=='POST')
 {
 if(!empty($_POST['username']))
 {
       $username=$_POST['username'];
 }
 if(isset($_POST['pass']))
 {
       $password=$_POST['pass'];
 }
```

5.2 Connecting to a MySQL Database

```
//Connecting to Server and Database
$con=mysqli_connect('localhost','root','','practical') or
die("connection not established");

//Creating query and executing it

$com=mysqli_query($con,"select * from login where
username='$username' and password='$password'") or die("Failed to
query Table".mysqli_error($con));
$row=mysqli_fetch_array($com);
```

Login_process.php

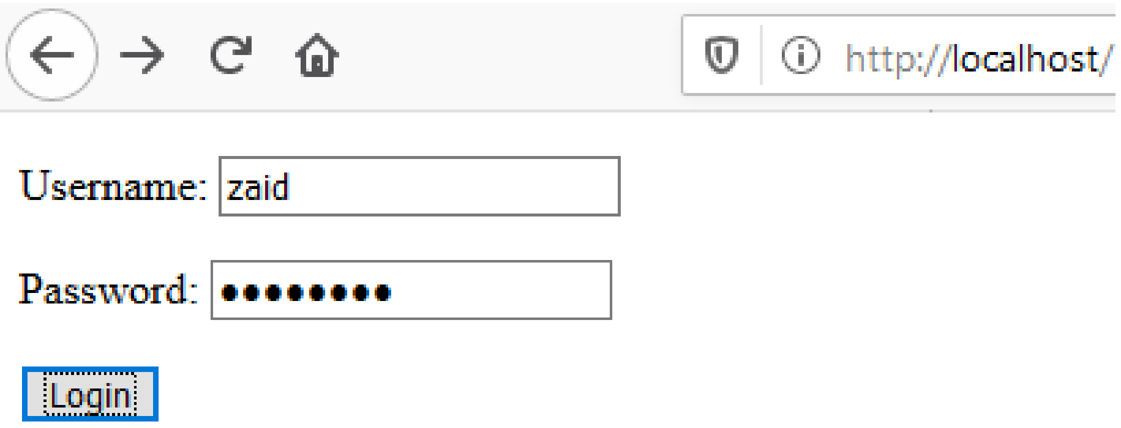
5.2 Connecting to a MySQL Database

```
//Checking if executed query has returned anything?
if($row)
  {
        echo"Login Successful... Welcome ".$row['username'];
  }
  else
  {
        echo"Incorrect username or password";
  }
  mysqli_close($con);
  }
?>
```

Login_process.php

5.2 Connecting to a MySQL Database

Login.php



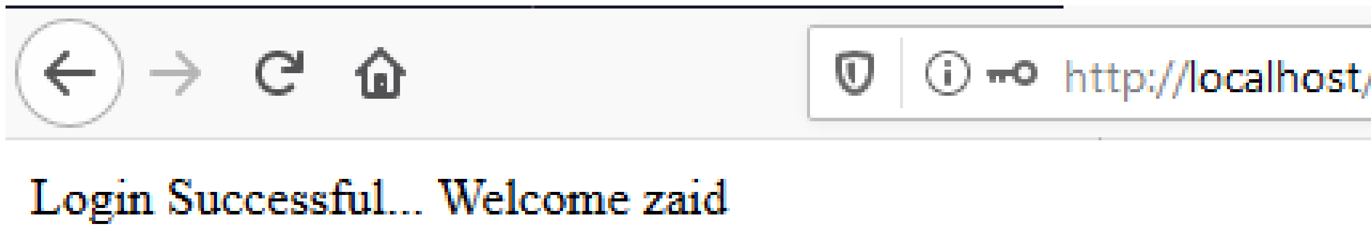
http://localhost/

Username: zaid

Password: ●●●●●●●●●

Login

Login_process.php



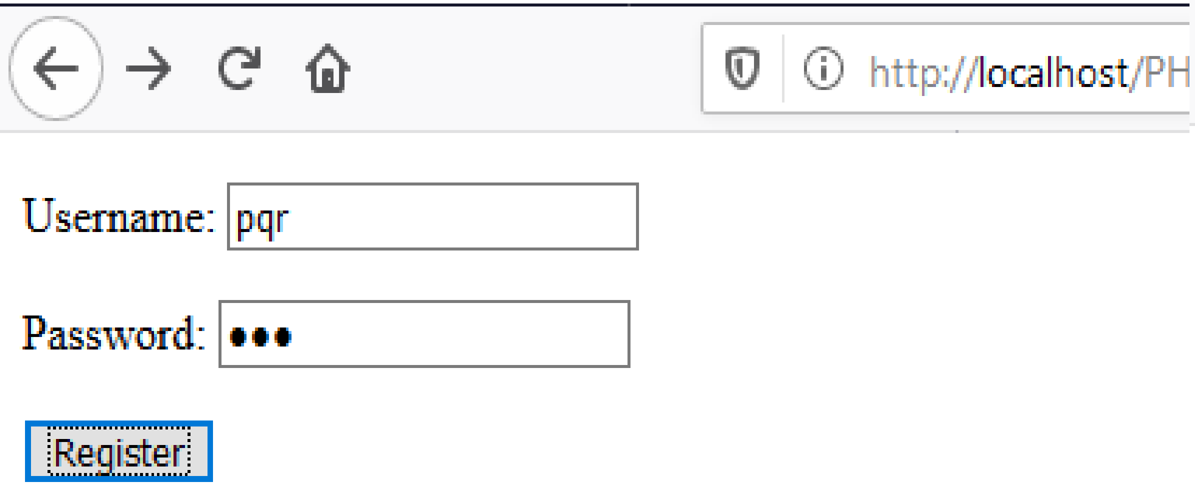
http://localhost/

Login Successful... Welcome zaid

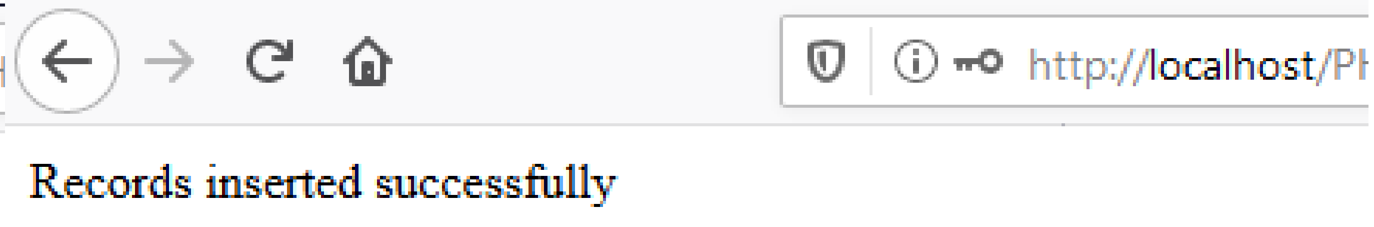
5.3 Database Operations: Insert Data

```
<?php
  if($_SERVER['REQUEST_METHOD']=='POST')
  {        if(!empty($_POST['username']))
           { $username=$_POST['username']; }
           if(!empty($_POST['pass']))
           { $password=$_POST['pass']; }
  }
  $con=mysqli_connect('localhost','root','','practical');
  mysqli_query($con,"insert into login (username, password) values ('$username','$password')");
  if($con)
           echo"Records inserted successfully";
  else
           echo"Request cannot be completed: ".mysqli_error($con);
  mysqli_close($con);
?>
```

5.3 Database Operations: Insert Data



Username:	pqr
Password:	•••

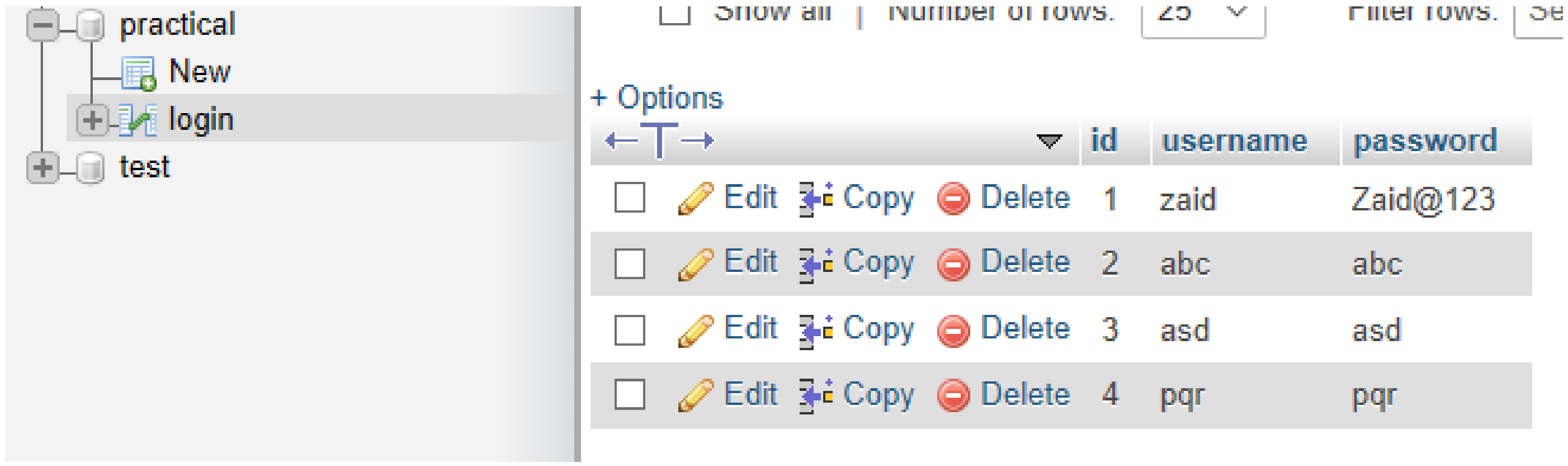


Records inserted successfully

Insert.html

Register_process.php

Login table

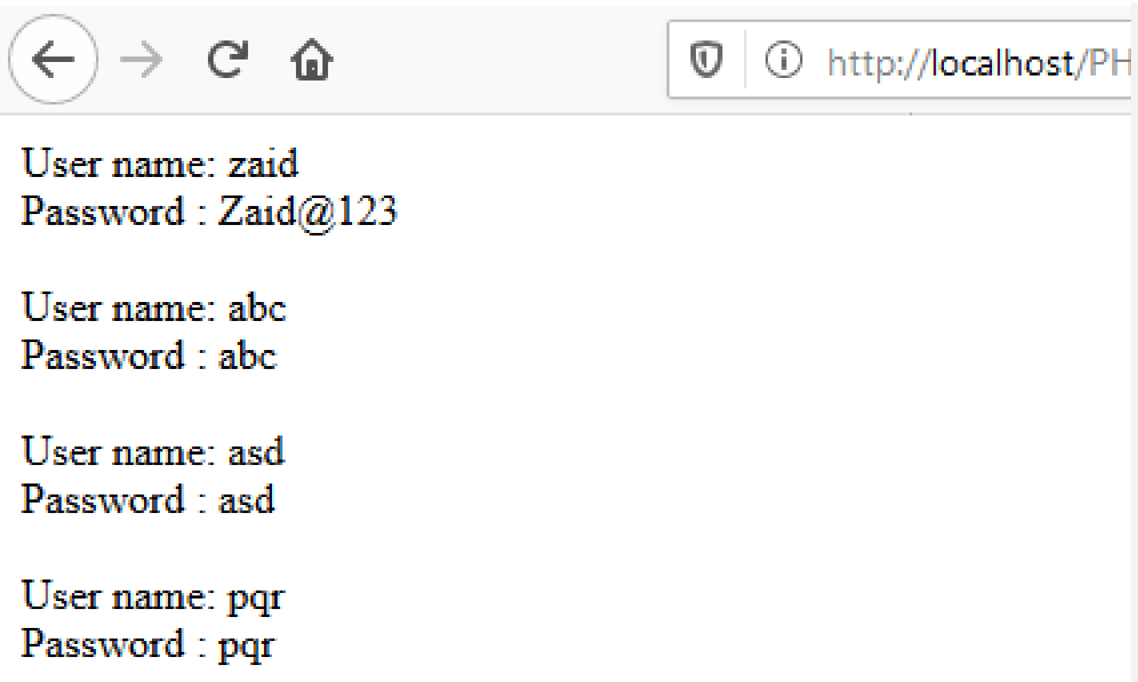


				id	username	password
	Edit	Copy	Delete	1	zaid	Zaid@123
	Edit	Copy	Delete	2	abc	abc
	Edit	Copy	Delete	3	asd	asd
	Edit	Copy	Delete	4	pqr	pqr

5.3 Database Operations: Retrieve Data

```
<?php
  $con=mysqli_connect('localhost','root','','practical') or die("Connection Failed");
  $result=mysqli_query($con,"select username, password from login");
  while($row=mysqli_fetch_assoc($result))
  {
        echo"User name: ".$row['username'];
        echo"<br>";
        echo"Password : ".$row['password'];
        echo"<br><br>";
  }
  mysqli_close($con);
?>
```

5.3 Database Operations: Retrieve Data



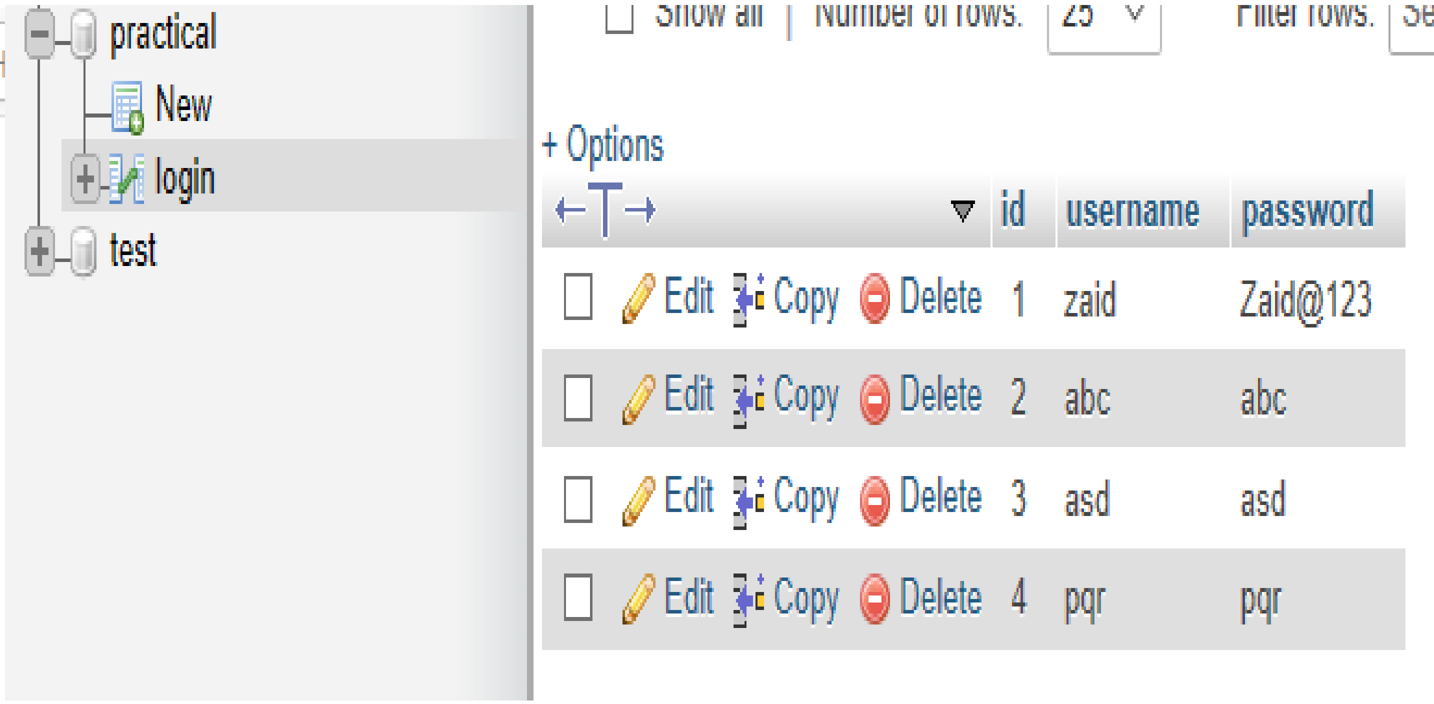
http://localhost/PH

User name: zaid
Password : Zaid@123

User name: abc
Password : abc

User name: asd
Password : asd

User name: pqr
Password : pqr



	id	username	password
Edit Copy Delete	1	zaid	Zaid@123
Edit Copy Delete	2	abc	abc
Edit Copy Delete	3	asd	asd
Edit Copy Delete	4	pqr	pqr

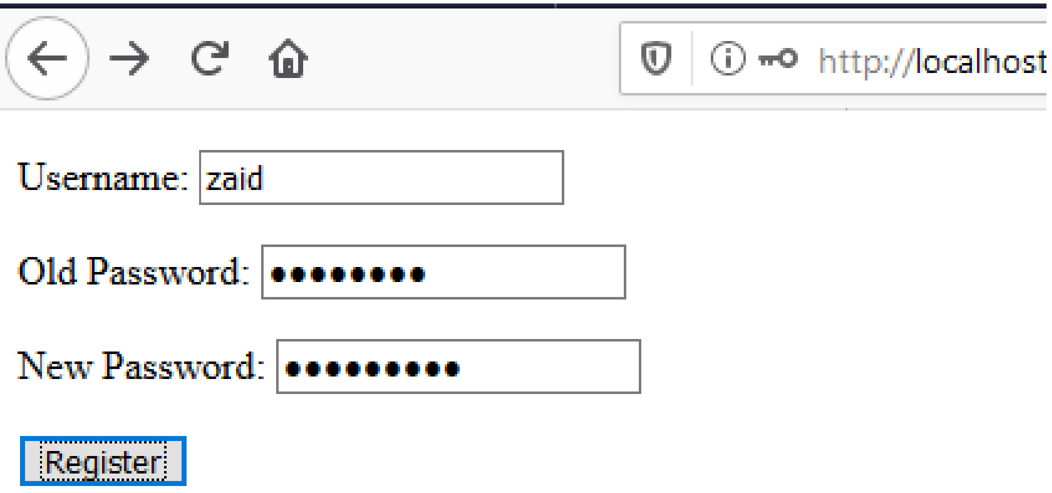
5.4 Database Operations: Update Data

```
<?php
  if($_SERVER['REQUEST_METHOD']=='POST')
  {       if(!empty($_POST['username']))
          {       $username=$_POST['username'];
          }
          if(!empty($_POST['oldpass']))
          {       $oldpassword=$_POST['oldpass'];
          }
          if(!empty($_POST['newpass']))
          {       $newpassword=$_POST['newpass'];
          }
  }
```

5.4 Database Operations: Update Data

```
$con=mysqli_connect('localhost','root','','practical') or die("Connection Failed..");
$result=mysqli_query($con,"update login set password='$newpassword' where
username='$username' and password='$oldpassword'");
if(!$result)
{       echo"Could not update";    }
else
{       echo"Password changed successfully..<br><br>";        }
$result=mysqli_query($con,"select password from login where
username='$username'");
$row=mysqli_fetch_row($result);
echo"you new password is : ".$row[0];
?>
```

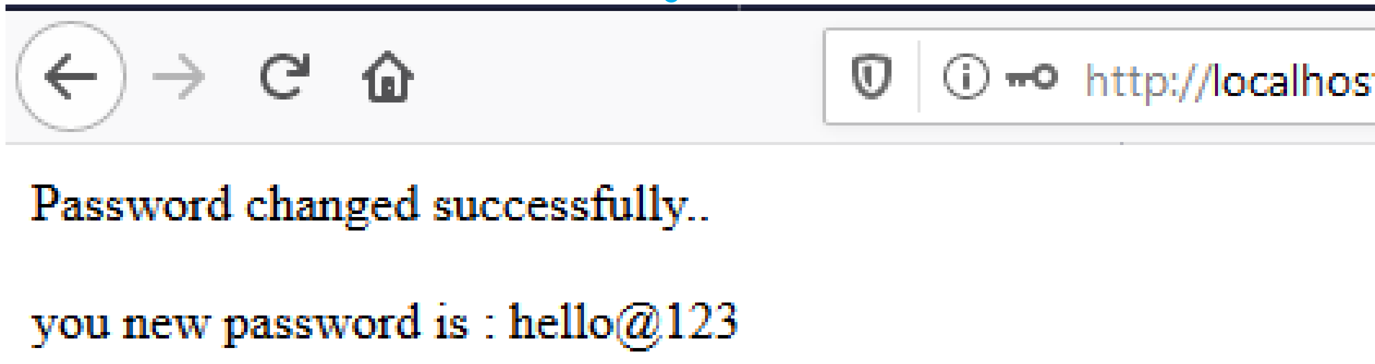
5.4 Database Operations: Update Data



Username: zaid

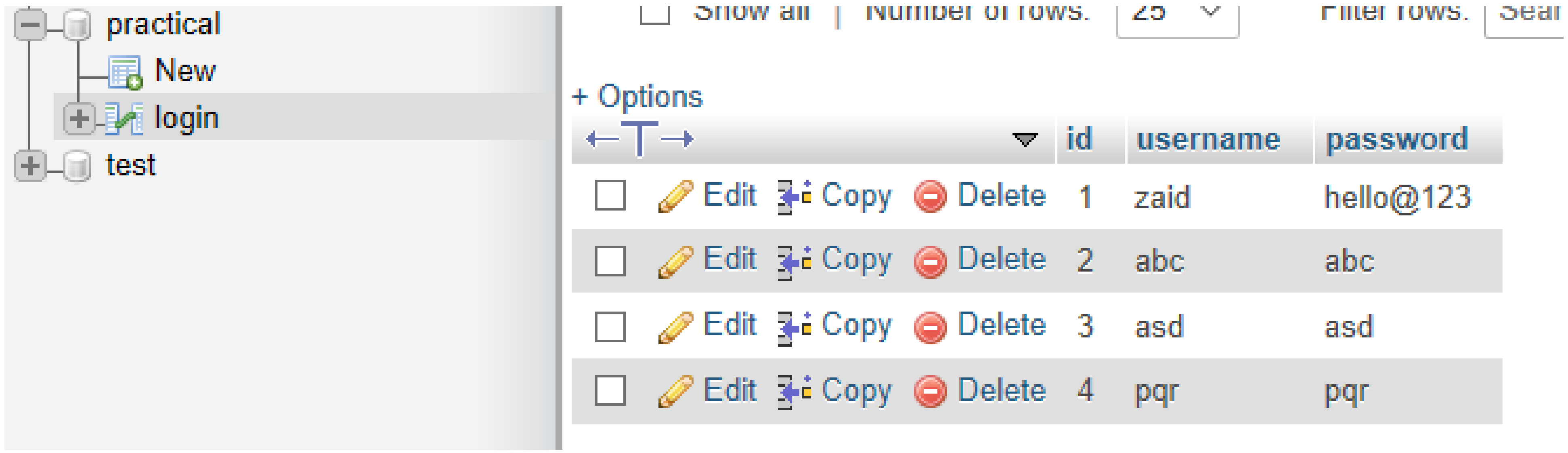
Old Password: ●●●●●●●●

New Password: ●●●●●●●●●



Password changed successfully..

you new password is : hello@123

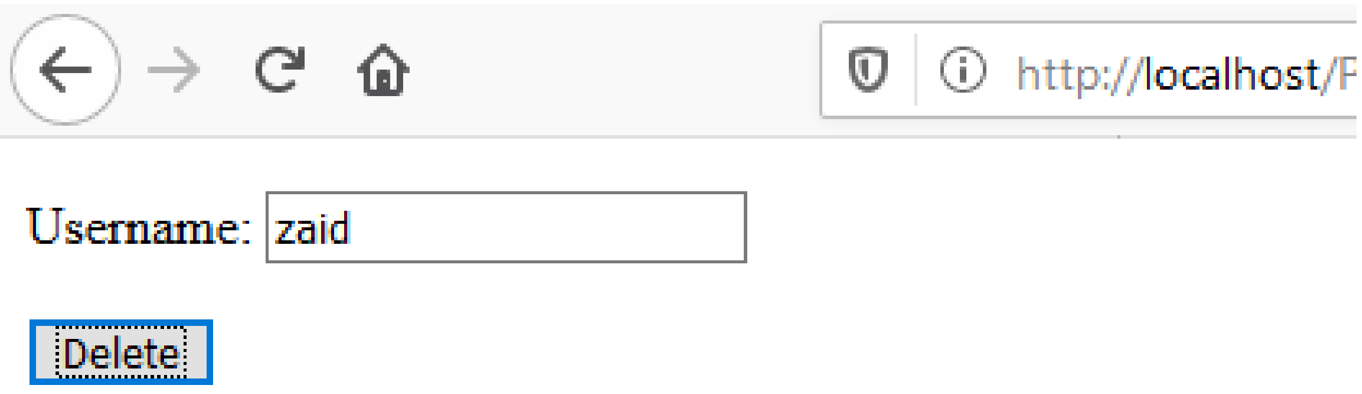


				id	username	password
☐	Edit	Copy	Delete	1	zaid	hello@123
☐	Edit	Copy	Delete	2	abc	abc
☐	Edit	Copy	Delete	3	asd	asd
☐	Edit	Copy	Delete	4	pqr	pqr

5.4 Database Operations: Delete Data

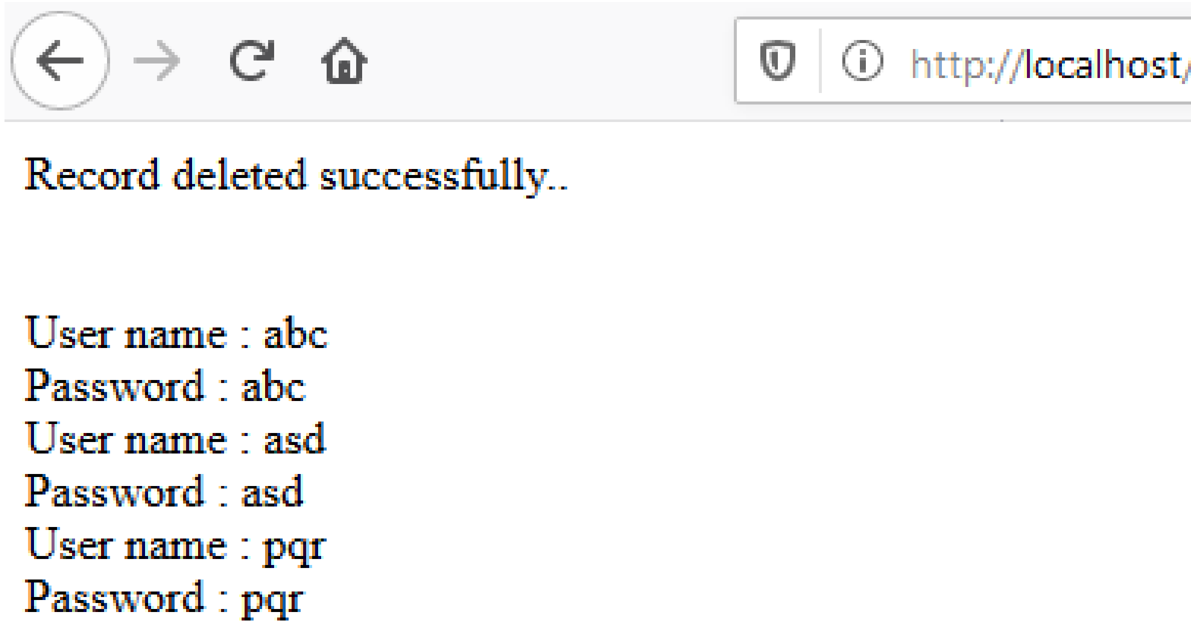
```
<?php
        if($_SERVER['REQUEST_METHOD']=='POST')
        {           if(!empty($_POST['username']))
                    {   $username=$_POST['username'];           }
        }
        $con=mysqli_connect('localhost','root','','practical') or die("Connection Failed..");
        $result=mysqli_query($con,"delete from login where username='$username'");
        if(!$result)
        {           echo"Could not delete“;     }
        else
        {           echo"Record deleted successfully..<br><br>“;            }
        $result=mysqli_query($con,"select username, password from login");
        while($row=mysqli_fetch_assoc($result))
        {
                    echo"<br>User name : ".$row['username'];
                    echo"<br>Password  : ".$row[‘password'];
        }
        mysqli_close($con);
?>
```

5.4 Database Operations: Delete Data



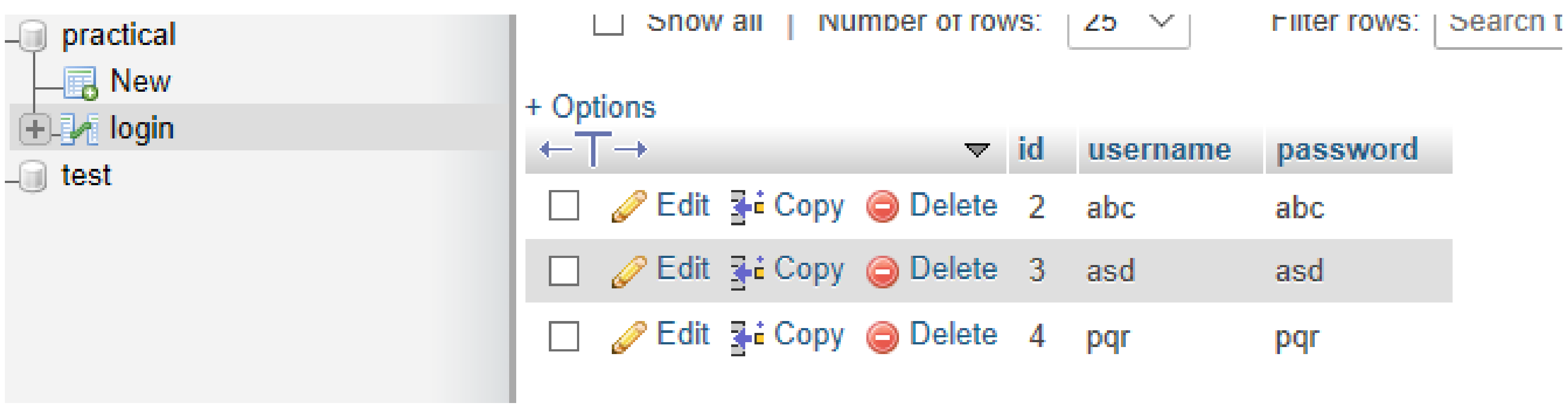
Username: zaid

Delete



```
Record deleted successfully..


User name : abc
Password : abc
User name : asd
Password : asd
User name : pqr
Password : pqr
```



				id	username	password
☐	Edit	Copy	Delete	2	abc	abc
☐	Edit	Copy	Delete	3	asd	asd
☐	Edit	Copy	Delete	4	pqr	pqr