PHP with MySQL Database

What is a database?

Database is a collection of inter-related data which helps in efficient retrieval, insertion and deletion of data from database and organizes the data in the form of tables, views, schemas, reports etc.

For Example, university database organizes the data about students, faculty, and admin staff etc. which helps in efficient retrieval, insertion, updation and deletion of data from it.

PHP has support for over 20 databases, including the most popular commercial and open source varieties of Relational and Non-Relational database systems such as Sqlite, MySQL, PostgreSQL, MariaDB, Oracle and MongoDB are the backbone of most modern dynamic web sites.

MySQL database server offers several advantages:

- MySQL is easy to use, yet extremely powerful, fast, secure, and scalable.
- MySQL runs on a wide range of operating systems, including UNIX or Linux, Microsoft Windows, Apple Mac OS X, and others.
- MySQL supports standard SQL (Structured Query Language).
- MySQL is ideal database solution for both small and large applications.
- MySQL is developed and distributed by Oracle Corporation.
- MySQL includes data security layers that protect sensitive data from intruders.

Structure:

MySQL database stores data into tables like other relational database. A table is a collection of related data and it is divided into rows and columns.

Each row in a table represents a data record that are inherently connected to each other such as information related to a particular person, whereas each column represents a specific field such as id, first_name, last_name, email, etc.

Plugin: mysql - deprecated, mysqli

There are three ways of working with MySQl and PHP:

1. MySQLi (object-oriented)
2. MySQLi (procedural)
3. PDO

<u>MySQLi Procedural:</u>

The basic steps to create MySQL database using PHP are:

- Establish a connection to MySQL server from your PHP script.
- If the connection is successful, write a SQL query to create a database and store it in a string variable.
- Execute the query.

PHP Database Connection:

PHP mysqli_connect() function is used to connect with MySQL database. It returns resource if connection is established or null.

Syntax:

resource mysqli_connect (server, username, password)

Example:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
```

```
// Creating connection
$conn = mysqli_connect($servername, $username, $password);

// Checking connection
if (!$conn) {
   die("Connection failed: " . mysqli_connect_error());
}
echo "Connected successfully";
?>
```

PHP Database Disconnection:

PHP mysqli_close() function is used to disconnect with MySQL database. It returns true if connection is closed or false.

Syntax

```
bool mysqli_close(resource $resource_link)
```

Example:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";

// Creating connection
$conn = mysqli_connect($servername, $username, $password);

// Checking connection
if (!$conn) {
   die("Connection failed: " . mysqli_connect_error());
}
echo "Connected successfully";
mysqli_close($conn);
```

```
?>
```

Execution of SQL Query:

mysqli_query() Function It accepts a SQL query as parameter (string) and executes/performs the given query on the database.

Syntax: mysqli_query($conn, query)

Basic CRUD Operations:

Create Table:

```
<?php
$dbname = 'db1';
$conn = mysqli_connect($servername, $username, $password, $dbname);

$sql = "create table employee (id INT AUTO_INCREMENT, name VARCHAR(20) NOT NULL,
 emp_salary INT NOT NULL, primary key (id))";
if(mysqli_query($conn, $sql)){
 echo "Employee table created successfully";
}else{
echo "Could not create table: ". mysqli_error($conn);
}
?>
```

Insert Record: mysqli_query() function is used to insert record in a table.

Example:

```
<?php
$sql = 'INSERT INTO employee(name,emp_salary) VALUES ("sonoo", 9000)';
if(mysqli_query($conn, $sql)){
 echo "Record inserted successfully";
}else{
```

```
echo "Could not insert record: ". mysqli_error($conn);
}
?>
```

Insert Multiple Records Using Array:

```
$a=[
   ['s1',3000],['s2',4000]];
echo count($a);
for($i=0;$i<count($a);$i++)
{
   $n=$a[$i][0];
   $s=$a[$i][1];
   $sql= "INSERT INTO employee (name,emp_salary) VALUES ('$n',$s)";
   if(mysqli_query($conn, $sql)){
      echo "Record inserted successfully";
     }else{
     echo "Could not insert record: ". mysqli_error($conn);
     }
}
```

Insert Records from HTML form:

```
<?php
if(isset($_POST['submit']))
{
$servername = "localhost";
$username = "root";
$password = "";
$dbname = 'db1';
// Creating connection
$conn = mysqli_connect($servername, $username,
$password,$dbname);
```

```
// Checking connection
if (!$conn) {
   die("Connection failed: " . mysqli_connect_error());
}
echo "Connected successfully";
$n=$_POST['name'];
$s=$_POST['salary'];
   $sql= "INSERT INTO employee (name,emp_salary) VALUES ('$n',$s)";
   if(mysqli_query($conn, $sql)){
      echo "Record inserted successfully";
      }else{
      echo "Could not insert record: ". mysqli_error($conn);
      }
}
?>
<form method="post">
   <input type="text" name="name" placeholder="Enter name"/>
   <input type="text" name="salary" placeholder="Enter salary"/>
   <input type="submit" value="submit" name="submit"/>
</form>
```

Database Selection:

mysqli_select_db() function is used to select a database.

Syntax:

```
<?php
mysqli_select_db($conn,$database_name);
?>
```

HERE,

“mysqli_select_db(…)” is the database selection function that returns either true or false

Identify Number of Records:

mysqli_num_rows() function is used to get the number of rows returned from a select query.

Syntax:

```
<?php
mysqli_num_rows($result);
?>
```

HERE,

“mysqli_num_rows(…)” is the row count function

“$result” is the mysqli_query result set

MySQL functions in php to fetch data from the database:

1. mysqli_fetch_array()
2. mysqli_fetch_assoc()
3. mysqli_fetch_row()
4. mysqli_fetch_object()
5. mysqli_fetch_field()
6. mysqli_fetch_lengths()

These mysql functions are used to extract the necessary information or data from the SQL database. Fetching data from the database helps the web developers to get necessary details and to manipulate it according to their need.

1. MYSQLI_FETCH_ARRAY(): This is used to fetch data from a row of a table. The data can be fetched as an associative array, numeric array or both depending on the type of result you want to generate.

Syntax : mysqli_fetch_array($result, $result_type);

Two parameters are used in this function whose description is shown below:

$result- It denotes the output (resource) of the php script. This is generated by using the mysqli_query() function.

$result_type- It denotes the type of result/output that is generated. You can put the following three values according to the type of result you desire.

(a) MYSQLI_ASSOC (used to get the output only as an associative array)

(b) MYSQLI_NUM (used to get output only as numeric array)

(c) MYSQLI_BOTH (used to get output as both associative and numeric array)

Result/Return Value: This function will return array values of a row if the row is present, and will return False if the row is not present.

2. MYSQLI_FETCH_ASSOC(): This function is used to fetch the data from the table only as an associative array. A single parameter $result is used in this function as shown below in the syntax. It denotes the output of the php script to be generated by using the mysqli_query() function.

Syntax : mysqli_fetch_assoc($result);

Result/Return Value: This function will return associative array corresponding to the values of a row if the row is present, and will return False if the row is not present.

3. MYSQLI_FETCH_ROW(): This function is used to extract data from a row of the database table only as an numeric array. Like mysqli_fetch_assoc() function, it also uses a single parameter $result which denotes the same as in the case above.

Syntax : mysqli_fetch_row($result);

Result/Return Value: The successful execution of this function will result a numeric array having values of the row to be fetched. And, it will return False, if it doesn't find the row.

4. MYSQLI_FETCH_OBJECT(): This function is used to extract the data from a row of the database table as an object.

Syntax: mysqli_fetch_object($result, $class_name, $additional parameter);

In this function, three parameters are used which are described below.
$result- the desired output (resource) being generated using mysqli_query() function.
$class_name- it is a string value that denotes the name of the class. If this parameter is not defined, the function will return stdClass object.
$additional parameter- this is an optional parameter which represents an array.
Result/Return Value: The successful execution of the function will result in an object value equivalent to the value of the row being fetched. And, it will return False, if the row is unavailable. It must be noted that the class name of the object must be declared.

5. MYSQLI_FETCH_FIELD(): This MySQL function is used to fetch the data from a column of the database table.

Syntax : mysqli_fetch_field($result, $field_offset);

Two parameters are used in this function that are discussed below.

$result- it represents the outcome (resource) of the php script using mysqli_query() function.

$field_offset- it is a numerical value starts at ‘0’ which if not defined will result in the execution of the next information.

Result/Return Value: It will return an object value equivalent to the value of the row being fetched.

6. MYSQLI_FETCH_LENGTHS(): This MySQL function is used in php to retrieve the length of the output data.

Syntax : mysqli_fetch_lengths($result);

Result/Return Value: On the successful execution of the function, you will get an array value of lengths equivalent to the value of each row or a False statement in case the execution is unsuccessful.

----Extra----

mysqli_fetch_array():

The mysqli_fetch_array function is used fetch row arrays from a query result set.

Syntax:

```
<?php
mysqli_fetch_array($result);
?>
```

HERE,

“mysqli_fetch_array(…)” is the function for fetching row arrays
“$result” is the result returned by the mysqli_query function.

```
<?php
        $conn = mysqli_connect('localhost', 'root', '');

        die("Unable to connect to MySQL: " . mysqli_error());
        //if connection failed output error message
   if (!mysqli_select_db($conn,’database_name’))
   {
        die("Unable to select database: " . mysqli_error());
   }
$sql_stmt = "SELECT * FROM my_contacts";
$result = mysqli_query($conn,$sql_stmt);

if (!$result)
        die("Database access failed: " . mysqli_error());
        //output error message if query execution failed
$rows = mysqli_num_rows($result);
// get number of rows returned

if ($rows) {
  while ($row = mysqli_fetch_array($result)) {
        echo 'ID: ' . $row['id'] . '<br>';
        echo 'Full Names: ' . $row['full_names'] . '<br>';
        echo 'Gender: ' . $row['gender'] . '<br>';
        echo 'Contact No: ' . $row['contact_no'] . '<br>';
        echo 'Email: ' . $row['email'] . '<br>';
        echo 'City: ' . $row['city'] . '<br>';
        echo 'Country: ' . $row['country'] . '<br><br>';
}
}
```

```
mysqli_close($conn); //close the database connection
?>
```

mysqli_fetch_all():

The fetch_all() / mysqli_fetch_all() function fetches all result rows and returns the result-set as an associative array, a numeric array, or both.

Syntax:

```
mysqli_fetch_all(result, resulttype)
```

mysqli_fetch_all(…) Returns an array of associative or numeric arrays holding the result rows

"$result" is the result returned by the mysqli_query function.

"resulttype" Specifies what type of array that should be produced. Can be one of the following values: MYSQLI_ASSOC, MYSQLI_NUM (this is default), MYSQLI_BOTH

Program:

```
<?php
$conn = mysqli_connect("localhost","root"," ","db_name");

$sql = "SELECT * FROM  my_contacts ";
$result = mysqli_query($conn, $sql);

// Fetch all
mysqli_fetch_all($result, MYSQLI_ASSOC);

// Free result set
mysqli_free_result($result);

mysqli_close($conn);
?>
```

mysqli_fetch_assoc():

The fetch_assoc() / mysqli_fetch_assoc() function fetches a result row as an associative array.

Note: Fieldnames returned from this function are case-sensitive.

Syntax:

mysqli_fetch_assoc(result)

mysqli_fetch_assoc() Returns an associative array of strings representing the fetched row. NULL if there are no more rows in result-set

“$result” Specifies a result set identifier returned by the mysqli_query function.

Program:

```
<?php
$conn = mysqli_connect("localhost","root"," ","db_name");
$sql = "SELECT * FROM my_contacts”;
$result = mysqli_query($conn, $sql);

// Associative array
$row = mysqli_fetch_assoc($result);
printf ("%s (%s)\n", $row["name"], $row["Age"]);

// Free result set
mysqli_free_result($result);

mysqli_close($conn);
?>
```