PHP and MYSQL

Dr. Prabu M, Ph.D.,(CSE)
Assistant Professor, Department of Computer Science,
CHRIST (Deemed to be University), Bangalore-560029.

What is MySQL?

MySQL is a database system used on the web
MySQL is a database system that runs on a server
MySQL is ideal for both small and large applications
MySQL is very fast, reliable, and easy to use
MySQL uses standard SQL
MySQL compiles on a number of platforms
MySQL is free to download and use
MySQL is developed, distributed, and supported by Oracle Corporation

Objectives :

Connecting to MySQL database
Create MySQL Database Using PHP
Delete MySQL Database Using PHP
Insert Data To MySQL Database
Retrieve Data From MySQL Database
Using Paging through PHP
Updating Data Into MySQL Database
Deleting Data From MySQL Database
Using PHP To Backup MySQL Database

Opening Database Connection

PHP provides mysql_connect function to open a database connection. This function takes five parameters and returns a MySQL link identifier on success, or FALSE on failure.

Syntax

```
connection mysql_connect
(server,user,passwd,new_link,client_flag);
```

Sr.No	Parameter & Description
1	**server** Optional − The host name running database server. If not specified then default value is **localhost:3306**.
2	**user** Optional − The username accessing the database. If not specified then default is the name of the user that owns the server process.
3	**passwd** Optional − The password of the user accessing the database. If not specified then default is an empty password.
4	**new_link** Optional − If a second call is made to mysql_connect() with the same arguments, no new connection will be established; instead, the identifier of the already opened connection will be returned.
5	**client_flags** Optional − A combination of the following constants − •**MYSQL_CLIENT_SSL** − Use SSL encryption •**MYSQL_CLIENT_COMPRESS** − Use compression protocol •**MYSQL_CLIENT_IGNORE_SPACE** − Allow space after function names •**MYSQL_CLIENT_INTERACTIVE** − Allow interactive timeout seconds of inactivity before closing the connection

Closing Database Connection

Its simplest function mysql_close PHP provides to close a database connection. This function takes connection resource returned by mysql_connect function. It returns TRUE on success or FALSE on failure.

Syntax

```
bool mysql_close ( resource $link_identifier
);
```

```php
<?php

  $dbhost = 'localhost:3036';
  $dbuser = 'guest';
  $dbpass = 'guest123';
  $conn = mysql_connect($dbhost, $dbuser, $dbpass);

  if(! $conn ) {
    die('Could not connect: ' . mysql_error());
  }

  echo 'Connected successfully';
  mysql_close($conn);
?>
```

Insert Data into MySQL Database

Data can be entered into MySQL tables by executing SQL INSERT statement through PHP function mysql_query. Below a simple example to insert a record into employee table.

In real application, all the values will be taken using HTML form and then those values will be captured using PHP script and finally they will be inserted into MySQL tables.

```
$conn = mysql_connect($dbhost, $dbuser, $dbpass);

  if(! $conn ) {
    die('Could not connect: ' . mysql_error());
  }

  $sql = 'INSERT INTO employee '.
    '(emp_name,emp_address, emp_salary, join_date) '.
    'VALUES ( "guest", "XYZ", 2000, NOW() )';

  mysql_select_db('test_db');
  $retval = mysql_query( $sql, $conn );

  if(! $retval ) {
    die('Could not enter data: ' . mysql_error());
  }

  echo "Entered data successfully\n";
```

Getting Data From MySQL Database

Data can be fetched from MySQL tables by executing SQL SELECT statement through PHP function mysql_query. You have several options to fetch data from MySQL.

The most frequently used option is to use function mysql_fetch_array(). This function returns row as an associative array, a numeric array, or both. This function returns FALSE if there are no more rows.

Below is a simple example to fetch records from employee table.

```
$conn = mysql_connect($dbhost, $dbuser, $dbpass);
   if(! $conn ) {
   die('Could not connect: ' . mysql_error());
  }
  $sql = 'SELECT emp_id, emp_name, emp_salary FROM employee';
  mysql_select_db('test_db');
  $retval = mysql_query( $sql, $conn );

  if(! $retval ) {
   die('Could not get data: ' . mysql_error());
  }
   while($row = mysql_fetch_array($retval, MYSQL_ASSOC)) {
   echo "EMP ID :{$row['emp_id']}  <br> ".
     "EMP NAME : {$row['emp_name']} <br> ".
     "EMP SALARY : {$row['emp_salary']} <br> ".
     "------------------------------<br>";
  }
   echo "Fetched data successfully\n";
   mysql_close($conn);
```

Updating Data into MySQL Database

Data can be updated into MySQL tables by executing SQL UPDATE statement through PHP function mysql_query.

Below is a simple example to update records into employee table. To update a record in any table it is required to locate that record by using a conditional clause. Below example uses primary key to match a record in employee table.

```
$conn = mysql_connect($dbhost, $dbuser, $dbpass);

        if(! $conn ) {
          die('Could not connect: ' . mysql_error());
        }

        $emp_id = $_POST['emp_id'];
        $emp_salary = $_POST['emp_salary'];

        $sql = "UPDATE employee ". "SET emp_salary = $emp_salary ".
          "WHERE emp_id = $emp_id" ;
        mysql_select_db('test_db');
        $retval = mysql_query( $sql, $conn );

        if(! $retval ) {
          die('Could not update data: ' . mysql_error());
        }
        echo "Updated data successfully\n";
```

Deleting Data from MySQL Database

Data can be deleted from MySQL tables by executing SQL DELETE statement through PHP function mysql_query.

Below is a simple example to delete records into employee table. To delete a record in any table it is required to locate that record by using a conditional clause. Below example uses primary key to match a record in employee table.

```
$conn = mysql_connect($dbhost, $dbuser, $dbpass);

        if(! $conn ) {
          die('Could not connect: ' . mysql_error());
        }

        $emp_id = $_POST['emp_id'];

        $sql = "DELETE FROM employee WHERE emp_id =
$emp_id" ;
        mysql_select_db('test_db');
        $retval = mysql_query( $sql, $conn );

        if(! $retval ) {
          die('Could not delete data: ' . mysql_error());
        }

        echo "Deleted data successfully\n";
```

REFERENCES :

- The Joy of PHP Programming: A Beginner's Guide – by Alan Forbes.
- PHP & MySQL Novice to Ninja – by Kevin Yank.
- Head First PHP & MySQL – by Lynn Beighley & Michael Morrison.
- Learning PHP, MySQL, JavaScript, and CSS: A Step-by-Step Guide to Creating Dynamic Websites – by Robin Nixon.
- PHP & MySQL Web Development – by Luke Welling & Laura Thompson.