SOS IN COMPUTER SCIENCE & APPLICATION
JIWAJI UNIVERSITY

Class : MBA (E-Commerce) IV Semester

Subject : PHP

Paper Code: (403)

Topic: Database Connectivity in PHP

PHP Connection to 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.

PHP Connection to MySQL Database

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);

If a resource is not specified then last opend database is closed.

Example

Try out following example to open and close a database connection −

```
<?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);
?>
```

Creating a Database-

To create and delete a database you should have admin privilege. Its very easy to create a new MySQL database. PHP uses **mysql_query** function to create a MySQL database. This function takes two parameters and returns TRUE on success or FALSE on failure.

Syntax

bool mysql_query(sql, connection);

Sr.No	Parameter & Description
1	**sql** Required - SQL query to create a database
2	**connection** Optional - if not specified then last opend connection by mysql_connect will be used.

Try out following example to create a database

```
<?php
$dbhost = 'localhost:3036';
$dbuser = 'root';
$dbpass = 'rootpassword';
   $conn = mysql_connect($dbhost, $dbuser, $dbpass);
  if(! $conn )
 {
die('Could not connect: ' . mysql_error());
}
  echo 'Connected successfully';
  $sql = 'CREATE Database test_db';
 $retval = mysql_query( $sql, $conn );
   if(! $retval )
{
die('Could not create database: ' . mysql_error());   }
     echo "Database test_db created successfully\n“;
 mysql_close($conn);
?>
_
```

Selecting a Database

Once you establish a connection with a database server then it is required to select a particular database where your all the tables are associated.

This is required because there may be multiple databases residing on a single server and you can do work with a single database at a time.

PHP provides function **mysql_select_db** to select a database.It returns TRUE on success or FALSE on failure.

Syntax

```
bool mysql_select_db( db_name, connection );
```

Sr.No	Parameter & Description
1	**db_name** Required - Database name to be selected
2	**connection** Optional - if not specified then last opend connection by mysql_connect will be used.

Creating Database Tables

To create tables in the new database you need to do the same thing as creating the database. First create the SQL query to create the tables then execute the query using mysql_query() function.

Example

Try out following example to create a table –

```
<?php
 $dbhost = 'localhost:3036';
 $dbuser = 'root';
 $dbpass = 'rootpassword';
   $conn = mysql_connect($dbhost, $dbuser, $dbpass);
if(! $conn )
 {
     die('Could not connect: ' . mysql_error());
   }
     echo 'Connected successfully';
    $sql = 'CREATE TABLE employee( '.      'emp_id INT NOT NULL AUTO_INCREMENT, '.      'emp_name
VARCHAR(20) NOT NULL, '.      'emp_address  VARCHAR(20) NOT NULL, '.      'emp_salary   INT NOT NULL,
'.      'join_date    timestamp(14) NOT NULL, '.      'primary key ( emp_id ))';
mysql_select_db('test_db');
   $retval = mysql_query( $sql, $conn );
if(! $retval )
{
      die('Could not create table: ' . mysql_error());
 }      echo "Table employee created successfully\n";
     mysql_close($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.

Example

Try out following example to insert record into employee table.

```
<?php
   $dbhost = 'localhost:3036';
  $dbuser = 'root';
  $dbpass = 'rootpassword';
  $conn = mysql_connect($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(! $retval )
 {
     die('Could not enter data: ' . mysql_error());
  }      echo "Entered data successfully\n";
   mysql_close($conn);
?>
```

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.

While doing data insert its best practice to use function **get_magic_quotes_gpc()** to check if current configuration for magic quote is set or not. If this function returns false then use function **addslashes()** to add slashes before quotes.

Insert Data into MySQL Database (cont..)
Example 2

Try out this example by putting this code into add_employee.php, this will take input using HTML Form and then it will create records into database.

```
<html>
    <head>
 <title>Add New Record in MySQL Database</title>
</head>
    <body>
  <?php
       if(isset($_POST['add']))
{
$dbhost = 'localhost:3036';
          $dbuser = 'root';
         $dbpass = 'rootpassword';
        $conn = mysql_connect($dbhost, $dbuser, $dbpass);           if(! $conn )
  die('Could not connect: ' . mysql_error());
       }
    if(! get_magic_quotes_gpc() )
{  $emp_name = addslashes ($_POST['emp_name']);  $emp_address = addslashes
($_POST['emp_address']);
  }
```

Insert Data into MySQL Database (cont..)
Example 2

```
else {
        $emp_name = $_POST['emp_name'];
  $emp_address = $_POST['emp_address'];
      }
$emp_salary = $_POST['emp_salary'];
       $sql = "INSERT INTO employee ". "(emp_name,emp_address, emp_salary,   join_date) ".
"VALUES('$emp_name','$emp_address',$emp_salary, NOW())"; mysql_select_db('test_db');
         $retval = mysql_query( $sql, $conn );
    if(! $retval )
{
          die('Could not enter data: ' . mysql_error());           }          echo
"Entered data successfully\n";
          mysql_close($conn);
        }
Else {     ?>
        <form method = "post" action = "<?php $_PHP_SELF ?>">     <table width = "400"
border = "0" cellspacing = "1"  cellpadding = "2"><tr> <td width = "100">Employee
Name</td>
    <td><input name = "emp_name" type = "text“ id = "emp_name"></td>        </tr>
 <tr> <td width = "100">Employee Address</td> <td><input name = "emp_address" type =
"text"  id = "emp_address"></td>  </tr><tr> <td width = "100">Employee Salary</td>
<td><input name = "emp_salary" type = "text" id = "emp_salary"></td>    </tr><tr>
<td width = "100"> </td>           <td> </td>        </tr><tr>           <td width =
"100"> </td>          <td> <input name = "add" type = "submit" id = "add"     value =
"Add Employee">           </td>        </tr>          </table>  </form>          <?php
}      ?>      </body></html>
```