

PHP

How Web Applications interact with server side databases

CRUD

Connecting and using MySQL from PHP

PHP provides many MySQL specific functions

<code>mysql_connect</code>	Open a link/connection to a mysql database
<code>mysql_select_db</code>	Choose a specific database on a MySQL server
<code>mysql_query</code>	run an SQL statement on an opened database
<code>mysql_fetch_array</code>	process a result set
<code>mysql_close</code>	Close a MySQL connection

Opening a connection to a mysql server

Use `mysql_connect`

`mysql_connect` (PHP 4, PHP 5)

Open a connection to a MySQL Server

Description

resource `mysql_connect` ([string `$server` [, string `$username` [, string `$password` [, bool `$new_link` [, int `$client_flags`]]]])

Opens or reuses a connection to a MySQL server.

Opening a connection to a mysql server

Use `mysql_connect`

```
<?php
// we connect to example.com and port 3307
$link = mysql_connect('example.com:3307', 'user5', 'qwerty5');

echo 'Connected successfully';
mysql_close($link);
```

Opens a connection to the mysql server on **example.com:3307**, using **user5** with password **qwerty5**

Local version

To connect to the local mysql database - mamp

```
<?php
// Create connection
$link = mysql_connect('localhost:8889', 'root', 'root');

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

Demo

Kingston version

To connect to the local mysql database - mamp

```
<?php
// Create connection
$link = mysql_connect('studentnet.kingston.ac.uk:3306', 'kxxxx', 'password');

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

Local version

We can test `$link` to see if the connection works .. this example will fail ...

```
<?php
// Create connection
$link = mysql_connect('localhost:8889', 'root', 'missing');
if (!$link) {
    die('Could not connect: ' . mysql_error());
}
echo 'Connected successfully';
mysql_close($link);
?>
```

Demo

Local version

We can test `$link` to see if the connection works .. this example will fail ...

```
<?php
// Create connection
$link = mysql_connect('localhost:8889', 'root', 'missing');
if (!$link) {
    die('Could not connect: ' . mysql_error());
}
echo 'Connected successfully';
mysql_close($link);
?>
```

`$link` is only really used for testing the connection and for closing the connection - `$link` is of type **resource** - a built in PHP type for this kind of connection

Select a particular database on a mysql server

Use `mysql_select_db`

`mysql_select_db` (PHP 4, PHP 5)

Select a MySQL database

Description

`bool mysql_select_db (string $database_name [, resource $link_identifier])`

Sets the current active database on the server that's associated with the specified link identifier. Every subsequent call to `mysql_query()` will be made on the active database.

Opens a distinct connection to a particular named database on the previously opened mysql server

Select a particular database on a mysql server

Use `mysql_select_db`

```
<?php
// we connect to example.com and port 3307
$link = mysql_connect('example.com:3307', 'user5', 'qwerty5');
echo 'Connected';

$db_selected = mysql_select_db('foo', $link);

mysql_close($link);
```

Opens a connection to the database `foo`, using the `$link` resource

MAMP version

To open the `empdb` database

```
<?php
// we connect to localhost
$link = mysql_connect('localhost:8889', 'root', 'root');
echo 'Connected';

$db_selected = mysql_select_db('employee', $link);
print "Opened database";

mysql_close($link);
?>
```

Demo

Kingston version

To open the `empdb` database

```
<?php
// we connect to localhost
$link = mysql_connect('studentnet.kingston.ac.uk:3306', 'kxxxx', 'password');
echo 'Connected';

$db_selected = mysql_select_db('db_kxxxxxxx', $link);
print "Opened database";

mysql_close($link);
?>
```

Running a SQL statement against the database

Use `mysql_query`

`mysql_query` (PHP 4, PHP 5)

Send a MySQL query

Description

resource `mysql_query` (string `$query` [, resource `$link_identifier`])

`mysql_query()` sends an unique query (multiple queries are not supported) to the currently active database on the server that's associated with the specified `link_identifier`.

Runs an SQL statement against the opened DB

Running a SQL statement against the database

Use `mysql_query`

```
$result = mysql_query('insert into foo (field1, field2, field3) values (120, 2000, 22000)');
```

Runs the SQL query - if **DDL** or **DML** the result will indicate whether the query ran successfully or not (**bool**)

Local version

Use `mysql_query`

```
$result = mysql_query('insert into emp (empno, ename, sal) values (120, "WALKER", 22000)');
```

This inserts a single record (or fails)

To close a database connection

Use `mysql_close`

`mysql_close` (PHP 4, PHP 5)

Close MySQL connection

Description

`bool mysql_close ([resource $link_identifier])`

`mysql_close()` closes the non-persistent connection to the MySQL server that's associated with the specified link identifier. If `link_identifier` isn't specified, the last opened link is used.

Closes the connection and releases the resources

Local version

To close the `empdb` database

```
<?php
// we connect to localhost
$link = mysql_connect('localhost:8889', 'root', 'root');

....

mysql_close($link);
?>
```

So in summary

Complete sequence is

```
<?php
// Create connection to database server
$link = mysql_connect('localhost:8889', 'root', 'root');

// Select a database on the server
$db_selected = mysql_select_db('empdb', $link);

// Send a SQL query to the database server
$result = mysql_query('insert into emp (empno, ename, sal) values (120,
"WALKER", 22000)');

// Close connection
mysql_close($link);

?>
```

CRUD

Create record (s)

Retrieve record (s)

Update record (s)

Delete record (s)

CRUD

Create record (s) SQL insert

Retrieve record (s) SQL select

Update record (s) SQL update

Delete record (s) SQL delete

CRUD

Create record (s) SQL insert 1

Retrieve record (s) SQL select 4

Update record (s) SQL update 3

Delete record (s) SQL delete 2

Create record (s)

This example - collect data from a form

Enter an employee

This form will allow you to enter details for an employee (inserts into the emp table)

Provide the details and click Continue

Enter empno:

Enter ename:

Job

Enter mgrno:

Enter hiredate:

Enter salary:

Enter commission:

Enter deptno:

Press continue to insert the record

Create record (s)

This example - collect data from a form

```
<form name="feedback" action="empinsertresponse.php" method="POST">
<p>Enter empno: <input type="text" name="empno"></p>
<p>Enter ename: <input type="text" name="ename"></p>
<p>Job
  <select name="job">
    <option value="Admin">Admin</option>
    <option value="Analyst">Analyst</option>
    <option value="Manager">Manager</option>
    <option value="President">President</option>
    <option value="Sales">Sales</option>
  </select>
</p>
<p>Enter mgrno: <input type="text" name="mgrno"></p>
<p>Enter hiredate: <input type="text" name="hiredate"></p>
<p>Enter salary: <input type="text" name="sal"></p>
<p>Enter commission: <input type="text" name="comm"></p>
<p>Enter deptno: <input type="text" name="deptno"></p>
<p>Press continue to insert the record<input type="submit" name="continue" value="Continue" /></p>
</form>
```

Create record (s)

response activity

Collect data from form
connect to database server
choose a database
send off the SQL insert statement

Create record (s)

Collect data from form

```
//Get data from Form
$empno=$_POST["empno"];
$name=$_POST["ename"];
$job=$_POST["job"];
$mgrno=$_POST["mgrno"];
$hiredate=$_POST["hiredate"];
$sal=$_POST["sal"];
$comm=$_POST["comm"];
$deptno=$_POST["deptno"];
```

Create record (s)

connect to database server choose a database

```
// Connecting, selecting database
$link = mysql_connect('localhost:8888','root','root');

mysql_select_db('employee');
```

Create record (s)

send off the SQL insert statement

```
// Performing SQL query
// This example insert statement would insert a single record
// $query = "INSERT INTO emp VALUES(980, 'PATEL', 'MANAGER', 938, '13/06/2005', 36000, NULL, 2)";

$query = "INSERT INTO emp VALUES('$empno.', '$name.', '$job.', '$mgrno.', '$hiredate.', '$sal.', '$comm.', '$deptno.')";
$result = mysql_query($query);
```

The hard bit - getting all the quotes in
the SQL (in the right places)

Demo

Delete record (s)

This example - collect person to be deleted from a form

Delete an employee

This form will allow you to delete details for an employee (deletes a row from the emp table)

Provide the details and click Continue

Enter empno:

Press continue to delete the record

Delete record (s)

response activity

Collect data from form
connect to database server
choose a database
send off the SQL delete statement

Delete record (s)

```
<?php
//Get data from form
$empno=$_POST["empno"];

// Connecting, selecting database
$link = mysql_connect('localhost:8888','root','root');

mysql_select_db('employee');

// Performing SQL query
// Example SQL to delete a single record
// $query = "delete from emp where empno=980";

$query = "delete from emp where empno=".$empno;
$result = mysql_query($query);

// Closing connection
mysql_close($link);
?>
```

SQL statement easier here

Demo

Update record (s)

This example - collect person to be updated from a form

Update an employee

This form will allow you to update the details for an employees salary (increase it by 10%)

Provide the details and click Continue

Enter empno:

Press continue to increase the salary

Update record (s)

response activity

Collect data from form
connect to database server
choose a database
send off the SQL update statement

Update record (s)

```
<?php
//Get data from form
$empno=$_POST["empno"];

// Connecting, selecting database
$link = mysql_connect('localhost:8888','root','root');
mysql_select_db('employee');

// Performing SQL query
// Example SQL to update a single record
// UPDATE emp
// SET xxxxx=xxxx value from the form WHERE xxxxx=xxxx value from the form

$query = "update emp set sal=sal*1.1 where empno=".$empno;
$result = mysql_query($query);

// Closing connection
mysql_close($link);
?>
```

Demo