

PHP and AJAX

What is AJAX ?

- AJAX stands for **A**synchronous **J**avaScript and **X**ML. AJAX is a new technique for creating better, faster, and more interactive web applications with the help of XML, HTML, CSS and Java Script.
- Conventional web application transmit information to and from the sever using synchronous requests. This means you fill out a form, hit submit, and get directed to a new page with new information from the server.
- With AJAX when submit is pressed, JavaScript will make a request to the server, interpret the results and update the current screen. In the purest sense, the user would never know that anything was even transmitted to the server.

For complete learning on AJAX, please refer to [AJAX Tutorial](#).

PHP and AJAX Example

To clearly illustrate how easy it is to access information from a database using Ajax and PHP, we are going to build MySQL queries on the fly and display the results on "ajax.html". But before we proceed, lets do ground work. Create a table using the following command.

NOTE – We are assuming you have sufficient privilege to perform following MySQL operations.

```
CREATE TABLE `ajax_example` (  
  `name` varchar(50) NOT NULL,  
  `age` int(11) NOT NULL,  
  `sex` varchar(1) NOT NULL,  
  `wpm` int(11) NOT NULL,  
  PRIMARY KEY (`name`)  
)
```

Now dump the following data into this table using the following SQL statements.

```
INSERT INTO `ajax_example` VALUES ('Jerry', 120, 'm', 20);  
INSERT INTO `ajax_example` VALUES ('Regis', 75, 'm', 44);  
INSERT INTO `ajax_example` VALUES ('Frank', 45, 'm', 87);  
INSERT INTO `ajax_example` VALUES ('Jill', 22, 'f', 72);  
INSERT INTO `ajax_example` VALUES ('Tracy', 27, 'f', 0);  
INSERT INTO `ajax_example` VALUES ('Julie', 35, 'f', 90);
```

Client Side HTML file

Now lets have our client side HTML file which is ajax.html and it will have following code

```

<html>
  <body>

    <script language = "javascript" type = "text/javascript">
      <!--
        //Browser Support Code
        function ajaxFunction(){
          var ajaxRequest; // The variable that makes Ajax
possible!

          try {
            // Opera 8.0+, Firefox, Safari
            ajaxRequest = new XMLHttpRequest();
          } catch (e) {
            // Internet Explorer Browsers
            try {
              ajaxRequest = new
ActiveXObject("Msxml2.XMLHTTP");
            } catch (e) {
              try{
                ajaxRequest = new
ActiveXObject("Microsoft.XMLHTTP");
              } catch (e){
                // Something went wrong
                alert("Your browser broke!");
                return false;
              }
            }
          }

          // Create a function that will receive data
          // sent from the server and will update
          // div section in the same page.

          ajaxRequest.onreadystatechange = function(){
            if(ajaxRequest.readyState == 4){
              var ajaxDisplay =
document.getElementById('ajaxDiv');
              ajaxDisplay.innerHTML =
ajaxRequest.responseText;
            }
          }

          // Now get the value from user and pass it to
          // server script.

          var age = document.getElementById('age').value;
          var wpm = document.getElementById('wpm').value;
          var sex = document.getElementById('sex').value;

```

```

        var queryString = "?age=" + age ;

        queryString += "&wpm=" + wpm + "&sex=" + sex;
        ajaxRequest.open("GET", "ajax-example.php" +
queryString, true);
        ajaxRequest.send(null);
    }
    //-->
</script>

<form name = 'myForm'>
    Max Age: <input type = 'text' id = 'age' /> <br />
    Max WPM: <input type = 'text' id = 'wpm' />
    <br />

    Sex: <select id = 'sex'>
        <option value = "m">m</option>
        <option value = "f">f</option>
    </select>

    <input type = 'button' onclick = 'ajaxFunction()' value =
'Query MySQL' />

</form>

<div id = 'ajaxDiv'>Your result will display here</div>
</body>
</html>

```

NOTE – The way of passing variables in the Query is according to HTTP standard and the have formA.

URL?variable1=value1;&variable2=value2;

Now the above code will give you a screen as given below

NOTE – This is dummy screen and would not work.

Max Age:

Max WPM:

Sex:

Your result will display here

Server Side PHP file

So now your client side script is ready. Now we have to write our server side script which will fetch age, wpm and sex from the database and will send it back to the client. Put the following code into "ajax-example.php" file.

```
<?php

$dbhost = "localhost";
$dbuser = "dbusername";
$dbpass = "dbpassword";
$dbname = "dbname";

//Connect to MySQL Server
mysql_connect($dbhost, $dbuser, $dbpass);

//Select Database
mysql_select_db($dbname) or die(mysql_error());

// Retrieve data from Query String
$age = $_GET['age'];
$sex = $_GET['sex'];
$wpm = $_GET['wpm'];

// Escape User Input to help prevent SQL Injection
$age = mysql_real_escape_string($age);
$sex = mysql_real_escape_string($sex);
$wpm = mysql_real_escape_string($wpm);

//build query
$query = "SELECT * FROM ajax_example WHERE sex = '$sex'";

if(is_numeric($age))
    $query .= " AND age <= $age";

if(is_numeric($wpm))
    $query .= " AND wpm <= $wpm";

//Execute query
$query_result = mysql_query($query) or die(mysql_error());

//Build Result String
$display_string = "<table>";
$display_string .= "<tr>";
$display_string .= "<th>Name</th>";
$display_string .= "<th>Age</th>";
$display_string .= "<th>Sex</th>";
$display_string .= "<th>WPM</th>";
$display_string .= "</tr>";

// Insert a new row in the table for each person returned
```

```
while($row = mysql_fetch_array($qry_result)) {  
    $display_string .= "<tr>";  
    $display_string .= "<td>$row[name]</td>";  
    $display_string .= "<td>$row[age]</td>";  
    $display_string .= "<td>$row[sex]</td>";  
    $display_string .= "<td>$row[wpm]</td>";  
    $display_string .= "</tr>";  
}  
echo "Query: " . $query . "<br />";  
  
$display_string .= "</table>";  
echo $display_string;  
?>
```

Now try by entering a valid value in "Max Age" or any other box and then click Query MySQL button.

Max Age:

Max WPM:

Sex:

Your result will display here

If you have successfully completed this lesson then you know how to use MySQL, PHP, HTML, and Javascript in tandem to write Ajax applications.