Architecting Web Applications using PHP

Session 15

PHP – AJAX

Session Overview

In this session, learners will be able to:

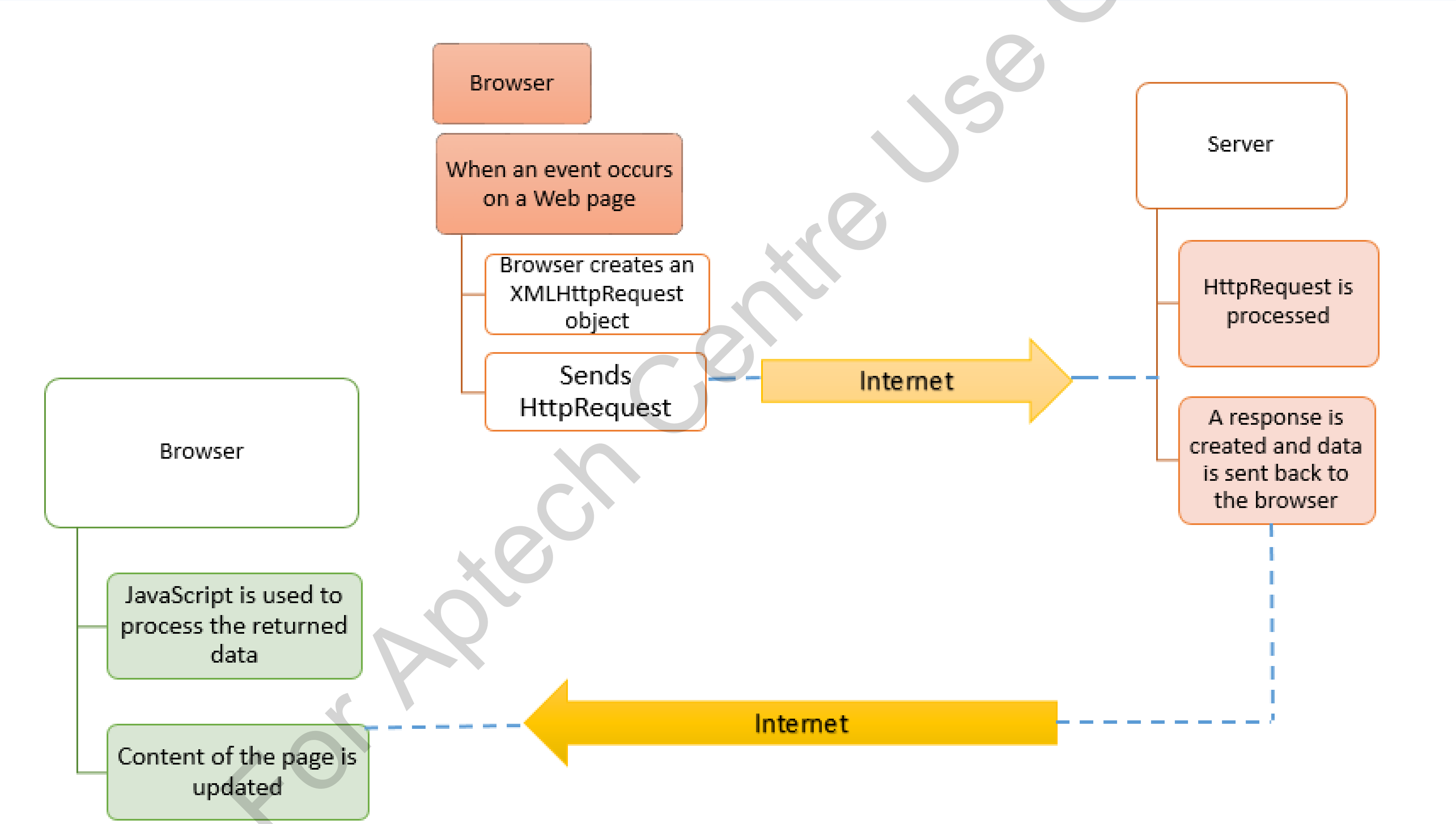
- Explain AJAX, work with AJAX, and Internet standards of AJAX
- Elaborate working with AJAX and PHP
- Illustrate working with AJAX and MySQL
- Explain how to use XML with AJAX
- Explain a program to perform AJAX live search technique

Introduction to AJAX

Asynchronous JavaScript and XML (AJAX) is not a new technology in itself, but a new approach to using JavaScript, HTML, CSS, and XML for creating faster and better Web applications.

Traditionally, Web applications transfer information to and fro the server in a Synchronous manner. This means that when the user fills a form and clicks submit, the user will be directed to a new page with updated information from the server.

Working of AJAX



Browser
When an event occurs on a Web page
Browser creates an XMLHttpRequest object
Sends HttpRequest
Internet
Server
HttpRequest is processed
A response is created and data is sent back to the browser
Internet
Browser
JavaScript is used to process the returned data
Content of the page is updated

Figure: Workflow of AJAX

Internet Standards of AJAX

XML HttpRequest object	• To send and receive data asynchronously with a server
JavaScript/DOM	• To display/interact with the information
CSS	• To make data more presentable
XML	• Normally, used as the format for transferring data

AJAX and PHP [1-2]

Example for AJAX PHP

```
CREATE TABLE customer (
name varchar(50) NOT NULL,
gender varchar(1) NOT NULL,
bill int(10) NOT NULL,
year int(10) NOT NULL,
PRIMARY KEY (name)
)
```

The user must also insert customer data into the table as shown here:

```
INSERT INTO 'customer' VALUES ('Paul', 'm', 10000, 1998);
INSERT INTO 'customer' VALUES ('Tyron', 'm', 5000, 1992);
INSERT INTO 'customer' VALUES ('Hans', 'm', 25000, 1994);
INSERT INTO 'customer' VALUES ('Erin', 'f', 12000, 1997);
INSERT INTO 'customer' VALUES ('Vinn', 'f', 2000, 2000);
```

AJAX and PHP [2-2]

Code Snippet:

```
<html>
<body>
<script language="javascript" type="text/javascript">
//Checking Browser Compatibility
function runAjax(){
var ajaxHttpRequest; // Key variable that is necessary for AJAX
try{
// Opera 8.0+, Firefox, Safari
ajaxHttpRequest = new XMLHttpRequest();
}
catch (e){
// Internet Explorer Browsers
try{
ajaxHttpRequest = new ActiveXObject("Msxml2.XMLHTTP");
}
catch (e) {
try{
ajaxHttpRequest = new ActiveXObject("Microsoft.XMLHTTP");
}
catch (e){
// Something went wrong
alert("Your browser is not working!");
return false;
}
}
}
//Setting up ajax to update the page on receiving the query results
ajaxHttpRequest.onreadystatechange = function(){
if(ajaxHttpRequest.readyState == 4){
var displayResponse = document.getElementById('result');
displayResponse.innerHTML = ajaxHttpRequest.responseText;
}
}
```

```
//Setting up ajax to update the page on receiving the query results
ajaxHttpRequest.onreadystatechange = function(){
if(ajaxHttpRequest.readyState == 4){
var displayResponse = document.getElementById('result');
displayResponse.innerHTML = ajaxHttpRequest.responseText;
}
}
//Sending the input data to server-side
var bill = document.getElementById('bill').value;
var year = document.getElementById('year').value;
var gender = document.getElementById('gender').value;
var queryString = "?bill=" + bill ;
queryString += "&year=" + year + "&gender=" + gender;
ajaxHttpRequest.open("GET", "customerdetail.php" +
queryString, true);
ajaxHttpRequest.send(null);
}
</script>
<form name='customerForm'>
Total bill of the customer: &nbsp;&nbsp;&nbsp;<input type='number'
id='bill' /> <br />
Year the customer joined: <input type='number' id='year' />
<br />
Gender: <select id='gender'>
    <option value="m">m</option>
    <option value="f">f</option>
    </select>
<input type='button' onclick='runAjax()'
value='Submit'/>
</form>
<div id='result'>Your result will be shown here</div>
</body>
</html>
```

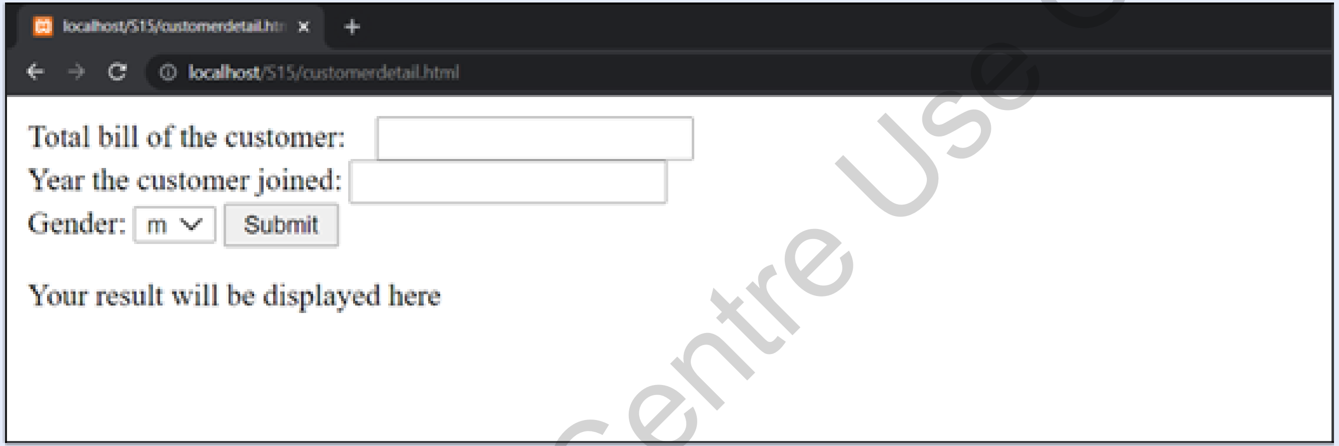
Server-Side PHP [1-2]

Code Snippet:

```
<?php
   //Passing the credentials to access the database
   $db_host = "localhost";
   $db_user = "root";
   $db_pass = "";
   $db_name = "test01";
   //Establishing the connection
   $con = mysqli_connect($db_host, $db_user, $db_pass, $db_name);
   $bill = $_GET['bill'];
   $year = $_GET['year'];
   $gender = $_GET['gender'];
   //Constructing the query with all conditions
   $query = "SELECT * FROM customer WHERE gender = '$gender'";
   if(is_numeric($bill))
   $query .= " AND bill >= $bill";
   if(is_numeric($year))
   $query .= " AND year <= $year";
```

```
//Running the query
$qry_result = mysqli_query($con,$query) or die(mysql_error());
//Table structure
$display_tb = "<table>";
$display_tb .= "<tr>";
$display_tb .= "<th>Name</th>";
$display_tb .= "<th>Total Bill</th>";
$display_tb .= "<th>Year Joined</th>";
$display_tb .= "<th>Gender</th>";
$display_tb .= "</tr>";
//Inserting data for each person
while($row = mysqli_fetch_array($qry_result)){
$display_tb .= "<tr>";
$display_tb .= "<td>$row[name]</td>";
$display_tb .= "<td>$row[bill]</td>";
$display_tb .= "<td>$row[year]</td>";
$display_tb .= "<td>$row[gender]</td>";
$display_tb .= "</tr>";
}
echo "Displaying the results for " . $query . "<br />";
$display_tb .= "</table>";
echo $display_tb;
?>
```

Server-Side PHP [2-2]



localhost/S15/customerdetail.html

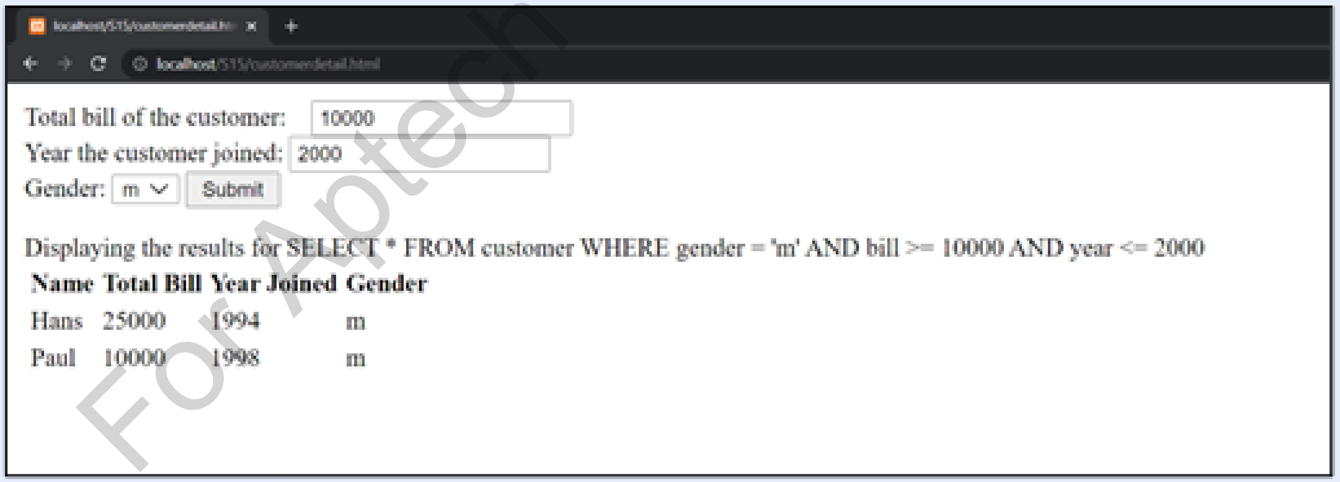
Total bill of the customer:

Year the customer joined:

Gender: m Submit

Your result will be displayed here

Figure: Web Page For User Inputs



localhost/S15/customerdetail.html

Total bill of the customer: 10000

Year the customer joined: 2000

Gender: m Submit

```
Displaying the results for SELECT * FROM customer WHERE gender = 'm' AND bill >= 10000 AND year <= 2000
 Name  Total Bill  Year Joined  Gender
 Hans  25000       1994         m
 Paul  10000       1998         m
```

Figure: Output of Server Side Script Using AJAX

AJAX and MySQL [1-4]

Train_Name	Train_No	Source	Destination
Inter	612158	Germany	France
Orient	829150	Paris	Budapest
Xpress	555432	France	Switzerland
BNF	125434	Vienna	Graz

AJAX and MySQL [2-4]

Code Snippet: (displayTrain.html)

```
<html>
<head>
<script>
//Function to use Ajax to transfer the data
function displayTrain(train_data) {
var ajaxHttpRequest = new XMLHttpRequest();
if (train_data == "") {
document.getElementById("result").innerHTML = "";
return;
}
ajaxHttpRequest.onreadystatechange = function(){
if(ajaxHttpRequest.readyState == 4){
var displayResponse = document.getElementById('result');
displayResponse.innerHTML = ajaxHttpRequest.responseText;
}
}
//The variable is passed to PHP to retrieve the result
ajaxHttpRequest.open("GET", "displayTrain.php?q=" +
train_data, true);
ajaxHttpRequest.send(null);
}
</script>
</head>
<body>
```

```
  <form>
    <select name="train_name" onchange =
"displayTrain(this.value)">
      <option value="">Select the train name:</option>
      <option value="Inter">Inter</option>
      <option value="Orient">Orient</option>
      <option value="Xpress">Xpress</option>
      <option value="BNF">BNF</option>
    </select>
    &nbsp;&nbsp;&nbsp;
    or
    &nbsp;&nbsp;&nbsp;
    <select name="train_number" onchange =
"displayTrain (this.value)">
      <option value="">Select the train number:</option>
      <option value="125434">125434</option>
      <option value="555432">555432</option>
      <option value="612158">612158</option>
      <option value="829150">829150</option>
    </select>
    </form>
    <br>
    <div id="result"><b>Train's information...</b></div>
    </body>

</html>
```

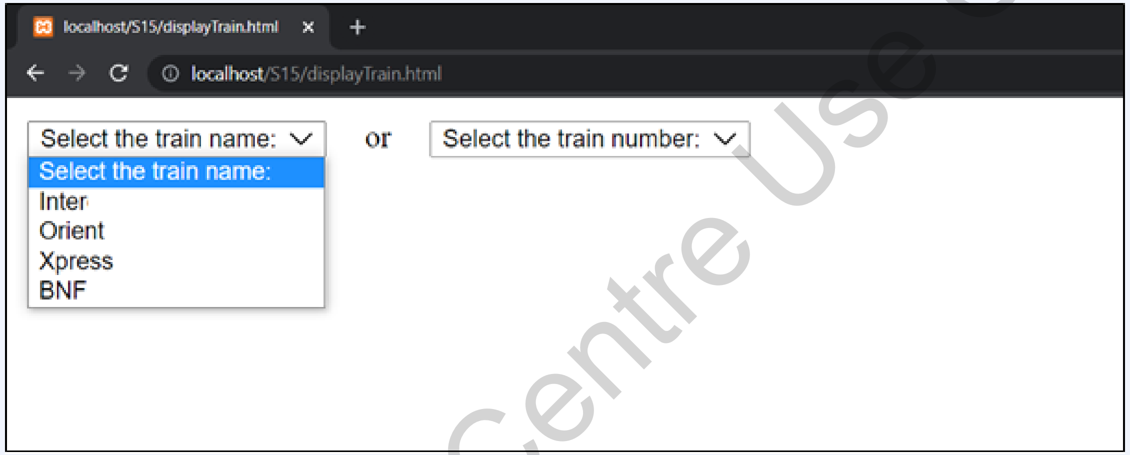
AJAX and MySQL [3-4]

Code Snippet: (`displayTrain.php`)

```
<html>
<head>
<style>
table {
  width: 100%;
  border-collapse: collapse;
}

table, td, th {
  border: 1px solid black;
  padding: 5px;
}

th {text-align: left;}
</style>
</head>
<body>

<?php
$q = $_GET['q'];
$con = mysqli_connect('localhost','root','','test01');
if((int)$q> 0){
$sql="SELECT * FROM train WHERE Train_No= '".$q."'";
}else {
$sql="SELECT * FROM train WHERE Train_Name= '".$q."'";
}
$result = mysqli_query($con,$sql);
```

```
echo "<table>
<tr>
<th>Train Number</th>
<th>Train Name</th>
<th>Source</th>
<th>Destination</th>
</tr>"
;
while($row = mysqli_fetch_array($result)) {
    echo "<tr>";
    echo "<td>" . $row['Train_No'] . "</td>";
    echo "<td>" . $row['Train_name'] . "</td>";
    echo "<td>" . $row['Source'] . "</td>";
    echo "<td>" . $row['Destination'] . "</td>";
    echo "</tr>";
  }
echo "</table>";
mysqli_close($con);
?>
</body>
</html>
```

AJAX and MySQL [4-4]

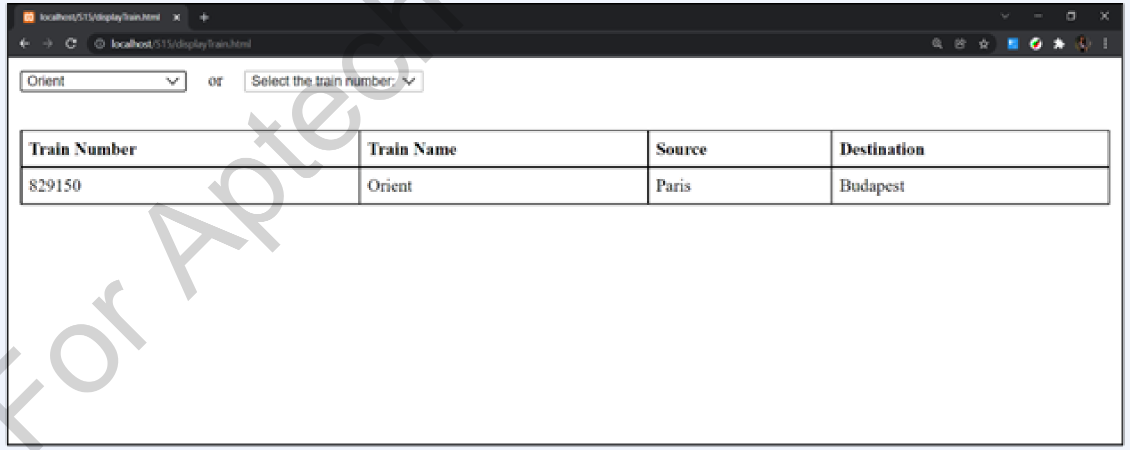


localhost/S15/displayTrain.html

Select the train name: or Select the train number:

Select the train name:
Inter
Orient
Xpress
BNF

Figure: Web Page for Selecting Trains



localhost/S15/displayTrain.html

Orient or Select the train number:

Train Number	Train Name	Source	Destination
829150	Orient	Paris	Budapest

Figure: Details of the Train Selected is Displayed

AJAX and XML [1-2]

There are two major differences between HTML and XML:

HTML defines a certain set of tags to be used compulsorily whereas XML does not.

XML is strict about document structure, how it is stored, and transferred, but HTML does not follow this strictly.

AJAX and XML [2-2]

Following example shows an invalid XML code:

Example:

```
<ul>
<li>It's raining
<li>It's sunny
   <li>It's cold
</ul>
```

Following example shows a valid XML code:

Example:

```
<ul>
   <li>Baked Pie</li>
   <li>Spiced Pies</li>
   <li>Ala Kiev with Minced Pie</li>
</ul>
```

AJAX Live Search [1-4]

A live search box is a search input box that displays the search results as and when the user starts typing words.

It uses a reliable database to retrieve information faster.

As an example, a live search box has been created that will search the employees table and show the results asynchronously.

Example:

```
CREATE TABLE emp (
    Emp_No INT(15) NOT NULL PRIMARY KEY,
    Name VARCHAR(50) NOT NULL
);
```

AJAX Live Search [2-4]

Code Snippet: `(employeeSearch.php)`

```
<html>
 <head>
<h1> PHP Live MySQL Database Search </h1>
<style>
/* Formatting search box */
.search-box{
width: 500px;
position: relative;
 display: inline-block;
 font-size: 14px;
}
.search-box input[type="text"]{
height: 32px;
padding: 5px 10px;
border: 1px solid #CCCCCC;
font-size: 14px;
}
</style>
<script>
function displayEmployee(){
    var ajaxHttpRequest = new XMLHttpRequest();
    var name = document.getElementById('name').value;
    if(name == ""){
        document.getElementById("result").innerHTML = "";
                return;
    }
```

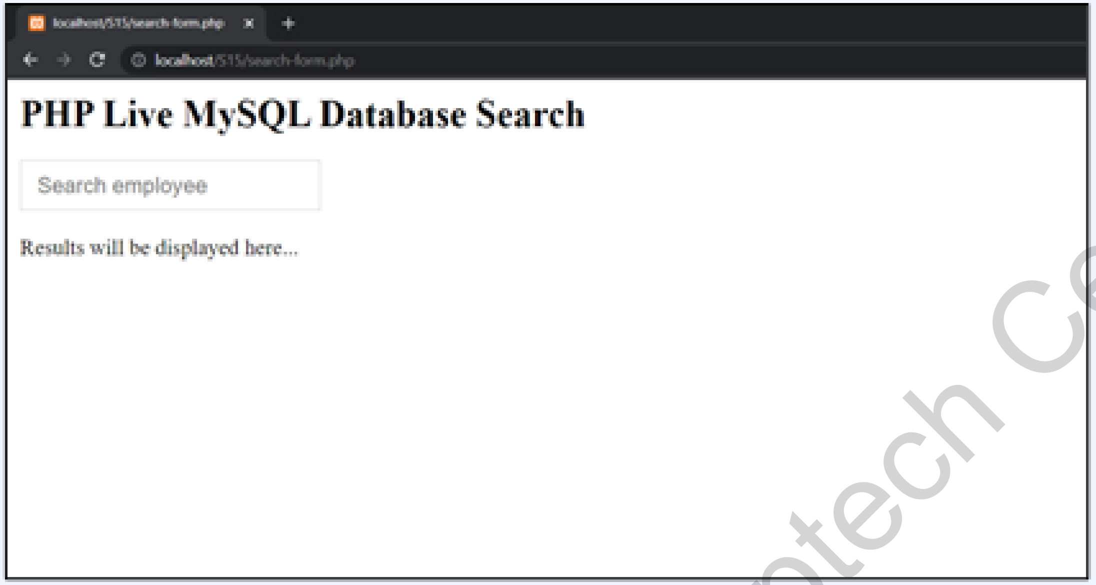
```
ajaxHttpRequest.onreadystatechange = function(){
if(ajaxHttpRequest.readyState == 4){
var displayResponse = document.getElementById('result');
displayResponse.innerHTML = ajaxHttpRequest.responseText;
}
}
var query = "?name=" + name ;
ajaxHttpRequest.open("GET", "getEmployee.php" +
query, true);
ajaxHttpRequest.send(null);
}

 </script>
</head>
<body> <div class="search-box">
<input id="name" type="text" autocomplete="off"
placeholder="Search employee" onkeyup="displayEmployee()"
/>
<br></br>
<div id="result">Results will be displayed here...</div>
</div>
</body>
 </html>
```

AJAX Live Search [3-4]

Code Snippet: (getEmployee.php)

```
<?php
 $db_host = "localhost";
 $db_user = "root";
 $db_pass = "";
 $db_name = "test01";

 //Establishing the connection
 $con = mysqli_connect($db_host, $db_user, $db_pass,$db_name) or die(mysql_error());;
 $name = $_GET['name'];
$age = mysqli_real_escape_string($con,$name);
 //Constructing the query with condition
$sql = "SELECT * FROM emp WHERE Name LIKE '$name%'";
//Retrieving the result from the database
    $result = mysqli_query($con,$sql);
    //Displaying the employee details
if(mysqli_num_rows($result) > 0){

                while($row = mysqli_fetch_array($result)){
                    echo "<p>" . $row["Emp_No"] . "&nbsp". "&nbsp". $row["Name"] . "</p>" ;
                }

            }else {
                echo "<p>No matches found</p>";
            }
     mysqli_close($con);
?>
```

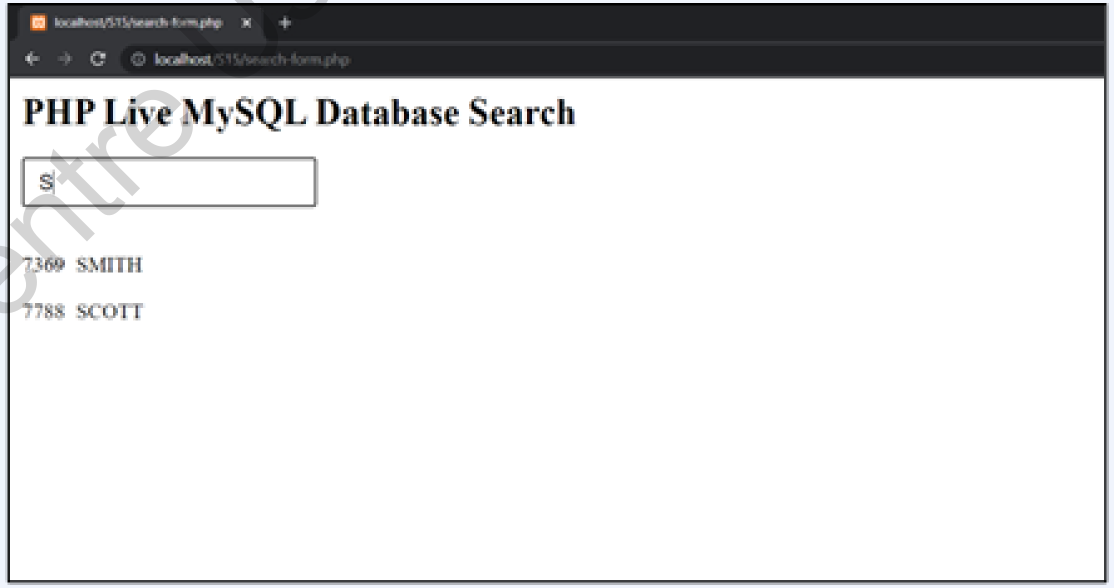
AJAX Live Search [4-4]



PHP Live MySQL Database Search

Results will be displayed here...

Figure: PHP Live Search Box



PHP Live MySQL Database Search

S

```
7369  SMITH
7788  SCOTT
```

Figure: Output of Server-side Code

Summary

- AJAX is used with PHP for asynchronous applications.
- Without AJAX, Web pages reload the entire page instead of updating the page, which results in server overhead and slower performances.
- AJAX combined with PHP and MySQL establishes a smooth and strong communication with the database and the server, improving the performance of applications.
- XML is widely used as a data format to send a request and receive a response.
- AJAX on Web pages are executed using an XMLHttpRequest object.
- AJAX improves user experience, supports multiple browsers, and has a good response time.