	CSE 482L: Internet and Web Technology Lab	Lab Manual
	Faculty: naqib.hussain@northsouth.edu Instructor: shabbir.ahmed01@northsouth.edu TA: afsana01@northsouth.edu Updated: 2023-Oct-25	8

Goal:

- Illustrate XML / XSLT, and AJAX by plain JS
- Use AJAX by JQuery :Get and Post requests
- Implement search suggestion

Illustrate XML:

Following is an example of XML code `breakfast.xml`. Keep this file in the `htdocs` folder inside your XAMPP installation. Try to fetch this file using the AJAX code snippet given in the next table below.

File name: **breakfast.xml**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<breakfast_menu>
  <food>
    <name>Nun Ruti</name>
    <price>BDT35.95</price>
    <description>
      This nun roti is an ideal breakfast/evening snack. Serve it with
      omlet, Vegetables curry, chicken/beaf/muttun curry and enjoy.
    </description>
    <calories>650</calories>
  </food>
  <food>
    <name>Lamb kabab</name>
    <price>BDT270.95</price>
    <description>
      Serve juicy lamb kebabs on skewers at a barbecue or summer
      gathering
    </description>
    <calories>900</calories>
  </food>
</breakfast_menu>
```

Below is the code for sample AJAX application: Can you explain each line? Modify this code below and use it in any page of your project.

AJAX JS code snippet

```
<script type="text/javascript">
  var ajaxRequest = getXMLHttpRequest();
  if (ajaxRequest){
    ajaxRequest.onreadystatechange = ajaxResponse;
    ajaxRequest.open("GET","breakfast.xml");
    ajaxRequest.send(null);
  }
  function ajaxResponse(){
    if (ajaxRequest.readyState != 4{
      return;
    }
    else {
      if (ajaxRequest.status == 200){
       // do whatever you need to do with ajaxRequest.responseXML
      }
      else {
        alert("Request failed:  "+ ajaxRequest.statusText);
      }
    }
  }
</script>
```

How does the data look like? Is there any smart way to properly add some style to this data? Using a regular JS way you will need lots of code for loops. But there is an easier way.

XSLT Demo

Now modify and save `breakfast_with_style.xml` and `breakfast.xsl` file on same directory. Do the tasks on XAMPP.

    i.   Show how it looks when `breakfast.xml` can't point out `./breakfast.xsl` file

    ii.   Show how it looks when `breakfast.xml` can point out `./breakfast.xsl` file

    iii.   Delete `./breakfast.xsl` template and show how it looks.

File name: **breakfast_with_style.xml**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="./breakfast.xsl"?>
<breakfast_menu>
  <food>
    <name>Nun Ruti</name>
    <price>BDT35.95</price>
    <description>
      This nun roti is an ideal breakfast/evening snack. Serve it with
      omlet, Vegetables curry, chicken/beaf/muttun curry and enjoy.
    </description>
    <calories>650</calories>
  </food>
  <food>
    <name>Lamb kabab</name>
    <price>BDT270.95</price>
    <description>
      Serve juicy lamb kebabs on skewers at a barbecue or summer
      gathering
    </description>
    <calories>900</calories>
  </food>
</breakfast_menu>
```

File name: **breakfast.xsl**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<html xsl:version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"> <body
style="font-family:Arial;font-size:12pt;backgroundcolor:#EEEEEE">
<xsl:for-each select="breakfast_menu/food">
  <div style="background-color:teal;color:white;padding:4px">
    <span style="font-weight:bold"><xsl:value-of select="name"/></span>
    <xsl:value-of select="price"/>
  </div>
  <div style="margin-left:20px;margin-bottom:1em;font-size:10pt">
    <p><xsl:value-of select="description"/>
      <span style="font-style:italic">
        <xsl:value-of select="calories"/> calories per serving
      </span>
    </p>
  </div>
</xsl:for-each>
</body>
</html>
```

Simple Alert Message. Use this to test if JQuery is working fine. Then move on to next.

File name: **alert.html**

```html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Lab-8</title>
    <script src="jquery-3.3.1.js"></script>
  </head>
  <body>
    <div id="test"></div>
    <button id="loadBtn">Load</button>
    <script src="alert.js"></script>
  </body>
</html>
```

File name: **alert.js**

```
$(document).ready(function(){
    $('#loadBtn').click(function(){
      alert("Hello");
    });
});
```

Load a static data from a file `data.txt` by using JQuery `.load()` method . Open `alert.html` and save as `load.html`

File name: **load.html**

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Lab-8</title>
    <script src="jquery-3.3.1.js"></script>
  </head>
  <body>
    <div id="test"></div>
    <button id="loadBtn">Load</button>
    <script src="load.js"></script>
  </body>
</html>
```

Open `alert.js` and save it as `load.js` with the following modification

File name: **load.js**

```
$(document).ready(function(){
    $('#loadBtn').click(function(){
      $('#test').load('data.txt');
    });
});
```

What happens when you run it? Is there any error? What can be the cause of this error? Create a text file named `data.txt` under the same folder and test again. Enter some content in `data.txt` and test again.

Passing value by $.get() method. Make sure that sources (src) are pointing to the right targets.

File name: **get.html**

```html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Lab-8</title>
    <script src="jquery-3.3.1.js"></script>
  </head>
  <body>
    <div id="test">
    </div>
    <button id="btn">Load</button>
    <script src="js/get.js"></script>
  </body>
</html>
```

File name: **get.js**

```js
$(document).ready(function () {
    $('#btn').click(function () {
      $.get('model1.php?my_limit=1', function(data, status){
         $('#test').html(data);
         alert(status);
      });
    });
});
```

Question: Can you explain what is the purpose of $(document).ready() in above code get.js?

Loading data from Database

File name: **model1.php**

```php
<?php require_once("connection.php"); ?>
<?php
  $limit = $_GET['my_limit'];
  $sql = "SELECT * FROM books LIMIT $limit";
  $result = mysqli_query($conn, $sql);
  if (mysqli_num_rows($result) > 0) {
    while ($row = mysqli_fetch_assoc($result)) {
     echo "<p>";
     echo $row['title'];
     echo "</p>";
    }
  } else {
      echo "No Data";
  }
?>
```

Before running the get.php open http://localhost/phpmyadmin/ and create a table named
books in a database ajax. In the books table add columns: id (int), title (varchar)
and price (double). Make the id primary (auto increment). Now enter some book names and
their prices.

Now you need to connect to this database table. Create another file as below. Keep all these files under
htdocs directory inside your XAMPP.

File name: **connection.php** [i]

```php
<?php
$servername = "localhost";
$username = "root";
$password = "";
$database = "ajax";
$conn = mysqli_connect($servername, $username, $password, $database);
if (!isset($conn)) {
  die("Connection failed: " . mysqli_connect_error());
}
?>
```

Question: When you run the code what happens? What data are you passing by using the $.get()?

Change the values and see result.

How what modifications do you need to use a `$.post()` method?

Open the `model1.php` and save as `model2.php` then make the following modifications

File name: **model2.php**

```php
<?php require_once('connection.php'); ?>
<?php
if (isset($_POST['my_limit'])){
  $limit = $_POST['my_limit'];
  $sql = "SELECT * FROM books LIMIT $limit";
  $result = mysqli_query($conn, $sql);
  if (mysqli_num_rows($result) > 0) {
    while ($row = mysqli_fetch_assoc($result)) {
      echo "<p>";
      echo $row['title'];
      echo "</p>";
    }
  } else { echo "No Data"; }
}
?>
```

Open `get.html` and save it as `post.html` with the following modification.

File name: **post.html**

```html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Lab-8</title>
    <script src="jquery-3.3.1.js"></script>
  </head>
  <body>
    <div id="test">
    </div>
    <button id="btn">Load</button>
    <script src="post.js"></script>
  </body>
</html>
```

How will you modify the JavaScript file? Open `get.js` file and save as `post.js` with the following modification.

File name: **post.js**

```
$(document).ready(function () {
    $('#btn').click(function () {
      $.post("model2.php", {my_limit: "1" }, function (data, status) {
            $('#test').html(data);
      });
    });
});
```
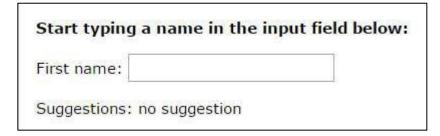

Question: What is the difference between a `$.post()` and a `$.get()`?

Lab Task

AJAX live search

Now modify the ajax code above to fetch a dynamic response. That means you will need to fetch from an URL targeting a .php file.

1. First create an html form as shown below:

Start typing a name in the input field below:

First name: [   ]

Suggestions: no suggestion

2. As the user starts typing on the input field, it should give suggestions based on a list of names:

"Anna","Brittany","Cinderella","Diana","Eva","Fiona","Gunda","Hege","Inga","Johanna","Kitty", "Linda","Nina","Ophelia","Petunia","Amanda","Raquel","Cindy","Doris","Eve","Evita","Sunniva" ,"Tove","Unni","Violet","Liza","Elizabeth","Ellen","Wenche","Vicky"

3. To respond to user's typing what event listener will you use?

You may use the following files. Can you explain what they are doing?

File name: **suggestion.php**

```php
<?php
$names = array("Jack", "Jones", "Martha", "Jony", "Glen");
if (isset($_POST['suggestion'])) {
  $suggestion = strtoupper($_POST['suggestion']);
  if (!empty($suggestion)) {
    foreach ($names as $n) {
      if (strpos($n, $suggestion) !== false) {
        echo $n;
        echo "<br />";
      }
    }
  }
}
?>
```

File name: **bring_suggestion.js**

```javascript
$(document).ready(function () {
    $('input').keyup(function () {
      var name = $('input').val();
      $.post("suggestion.php", {suggestion: name }, function (data, status) {
          $('#test').html(data);
      });
    });
});
```

End Notes

[i] This is a dangerous practice to use database `root` and a blank password. Create another DB user with proper password and use it to connect