Aim

To execute XML, PHP and database connectivity.

# Procedure

* 1. Save xml and xsl in .xml and .xsl files.
  2. Javascript can make AJAX calls to xml.
  3. XSL can be used to display xml content with style.

# Questions

## Payroll information of a company using XML, DTD and CSS

### index.html

*<!DOCTYPE html>*

<head>

<link rel="stylesheet" href="styles.css">

</head>

<html>

<body>

<div class="centering">

<form>

Choose a designation from the list below -

<select name="desig" id="desig">

<option val = '0'>Select designation</option>

<option val = 'CEO'>CEO</option>

<option val = 'Lead'>Lead</option>

<option val = 'SDE'>SDE</option>

</select>

</form>

<input type = "button" value = "Search Salary" onclick="ajaxlook()">

</form>

<br>

<p id="sal"></p>

<br>

</div>

<script>

function ajaxlook() {

var word = document.getElementById('desig').value; var xhttp = new XMLHttpRequest(); xhttp.onreadystatechange = function() {

if (this.readyState == 4 && this.status == 200) {

myFunction(this, word);

}};

xhttp.open("GET", "payroll.xml", true); xhttp.send();

}

function myFunction(xml, word) { var xmlDoc = xml.responseXML;

document.getElementById("sal").innerHTML = "Salary is " + xmlDoc.getElementsByTagName(word)[0].childNodes[1].innerHTML;

}

</script>

</body>

</html>

### styles.css

.centering {

display : flex ;

align-items : center ; justify-content : center ; flex-direction: column;

}

### Note.dtd

*<!DOCTYPE note [*

*<!ELEMENT payroll (CEO, Lead, SDE)>*

*<!ELEMENT CEO (#PCDATA)>*

*<!ELEMENT Lead (#PCDATA)>*

*<!ELEMENT SDE (#PCDATA)>*

]>

### payroll.xml

*<?xml version="1.0" encoding="UTF-8"?>*

*<!DOCTYPE note SYSTEM "Note.dtd">*

<payroll>

<CEO>

<salary>100000</salary>

</CEO>

<Lead>

<salary>75000</salary>

</Lead>

<SDE>

<salary>50000</salary>

</SDE>

</payroll>

### Output

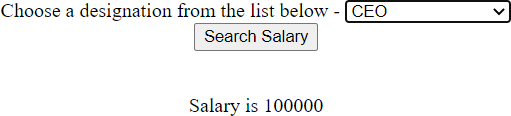


Figure 1: Question 1

## 2. Online bookstore using data islands - XML. HTML. DTD, CSS

### index.html

<html>

<head>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class = centering>

<h1>Online Bookstore</h1>

</div>

<xml>

*<!DOCTYPE Note SYSTEM "Note.dtd">*

<div class = centering>

<table border = 3>

<tr>

<th>Name</th>

<th>Author</th>

<th>Stock available</th>

<th colspan="2">Actions</th>

</tr>

<book>

<tr>

<td><name>Harry Potter</name></td>

<td><author>JK Rowling</author></td>

<td><stock>125 paperback</stock></td>

<td><button>Add to Cart</button></td>

<td><button>Buy Now</button></td>

</tr>

</book>

<book>

<tr>

<td><name>An Era of Darkness</name></td>

<td><author>Shashi Tharoor</author></td>

<td><stock>15 paperback</stock></td>

<td><button>Add to Cart</button></td>

<td><button>Buy Now</button></td>

</tr>

</book>

<book>

<tr>

<td><name>Playing it my Way</name></td>

<td><author>S. Tendulkar</author></td>

<td><stock>50 paperback</stock></td>

<td><button>Add to Cart</button></td>

<td><button>Buy Now</button></td>

</tr>

</book>

</table>

</div>

</xml>

</body>

</html>

### Note.dtd

*<!DOCTYPE Note [*

*<!ELEMENT book (name, author, stock)>*

*<!ELEMENT name (#PCDATA)>*

*<!ELEMENT author (#PCDATA)>*

*<!ELEMENT stock (#PCDATA)>*

]>

### styles.css

.centering {

display : flex ;

align-items : center ; justify-content : center ; flex-direction: column;

}

.books {

border: 1px solid black; padding: 2px;

}

### Output

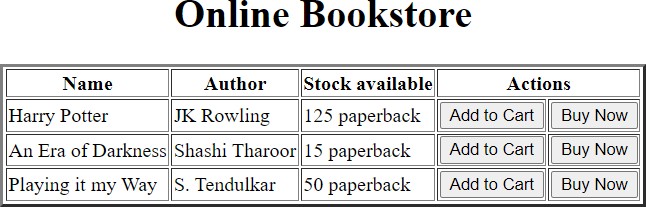


Figure 2: Question 2

## 3. Menu items using database connectivity

* + 1. **menu***sel.php*

<html>

<head>

<link [href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css"](https://cdn.jsdelivr.net/npm/bootstrap%405.2.2/dist/css/bootstrap.min.css) rel="styleshe

</head>

<body >

<h1 style = "margin: 10px">Menu Items</h1>

<br>

<form action="" method = "POST">

Choose which menu you would like to see:

<select name="ch" id="ch">

<option value="0">Select</option>

<option value="1">Veg Menu</option>

<option value="2">Non-Veg Menu</option>

<option value="3">Show all items</option>

</select>

<input type="submit" name="submit" value="Submit">

</form>

*<?php*

if (isset($\_POST['submit'])) {

$conn = mysqli\_connect ('localhost:3307', 'root','','test1');

$ch = $\_POST['ch']; if ($ch == 1)

$sql = 'SELECT \* from menu where veg = "yes"'; else

$sql = 'SELECT \* from menu';

$result = $conn -> query ($sql);

echo '

<table class="table table-striped" style="margin:10px; width: 250px">

<thead>

<tr>

<th>Item</th>

<th>Price</th>

</tr>

</thead>

<tbody>

';

while ($row = $result->fetch\_assoc()) { echo "

<tr>

}

}

echo "

</tbody>

</table>";

<th>".$row['item']."</th>

<th>".$row['price']."</th>

</tr> ";

*?>*

</body>

</html>

### Output

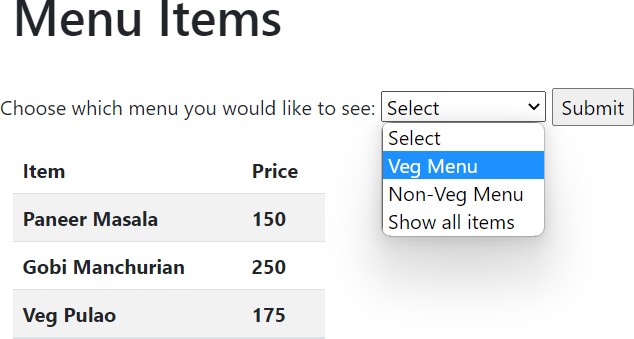


Figure 3: Question 3

## 4. PHP Sessions

### session.php

*<?php*

session\_start();

if (isset($\_SESSION))

echo "Session started";

else

die("Session error");

$\_SESSION['Name'] = "Timothy"; session\_unset();

echo "<br> Session status is ".session\_status(); session\_destroy();

echo "<br> Session status is ".session\_status();

*?>*

### Output



Figure 4: Question 4

## 5. PHP Cookies

### cookie.php

*<?php*

$cookie\_name = "user";

$cookie\_value = "IWP";

setcookie($cookie\_name, $cookie\_value, time() + (86400 \* 3), "/"); *// 86400 = 1 day*

if(isset($\_COOKIE[$cookie\_name])) {

echo "Cookie '" . $cookie\_name . "' is set!<br>"; echo "Value is: " . $\_COOKIE[$cookie\_name];

}

setcookie("user", "", time() - 3600); if (count ($\_COOKIE > 0) {

echo "Cookies are enabled.";

} else {

echo "Cookies are disabled.";

}

*?>*

### Output



Figure 5: Question 5

## 6. Extract .txt data with .php file

### hobbies.txt

* + 1. **index.html**

*<!DOCTYPE html>*

<html>

<body>

<h1>Load Data from text file </h1>

<button type="button" onclick="loadDoc()">Read hobbies.txt</button>

<p id = "demo"></p>

<script>

function loadDoc() {

var xhttp = new XMLHttpRequest();

xhttp.onreadystatechange = function () {

if (this.readyState == 4 && this.status == 200) { var allText = this.responseText;

document.getElementById("demo").innerHTML = xhttp.responseText;

}

};

xhttp.open("GET", "hobbies.txt", true); xhttp.send();

}

</script>

</body>

</html>

### Output



Figure 6: Question 6

## 7. AJAX using XML

### index.html

*<!DOCTYPE html>*

<html>

<body>

<h2>Restaurant List</h2>

<input type = "button" value = "View Restaurants" onclick="ajaxlook()">

<p id="r1"></p>

<p id="r2"></p>

<script>

function ajaxlook() {

var xhttp = new XMLHttpRequest(); xhttp.onreadystatechange = function() {

if (this.readyState == 4 && this.status == 200) { myFunction(this);

}

};

xhttp.open("GET", "rest.xml", true); xhttp.send();

}

function myFunction(xml) {

var xmlDoc = xml.responseXML;

console.log (xmlDoc.getElementsByTagName('restaurant'))

document.getElementById("r1").innerHTML = xmlDoc.getElementsByTagName('restaurant')[0].children[0

+ xmlDoc.getElementsByTagName('restaurant')[0].children[1].innerHTML +', ' + xmlDoc.getElementsByT

+ 'Ph: ' + xmlDoc.getElementsByTagName('restaurant')[0].children[3].innerHTML;

document.getElementById("r2").innerHTML = xmlDoc.getElementsByTagName('restaurant')[1].children[0

+ xmlDoc.getElementsByTagName('restaurant')[1].children[1].innerHTML +', ' + xmlDoc.getElementsByT

+ 'Ph: ' + xmlDoc.getElementsByTagName('restaurant')[1].children[3].innerHTML;

}

</script>

</body>

</html>

### rest.xml

*<?xml version="1.0" encoding="UTF-8"?>*

*<?xml-stylesheet type="text/xsl" href="style.xsl"?>*

<restaurants>

<restaurant>

<name>La Pasteria</name>

<address>13 Alonisou Street, Patras, 261 35</address>

<cuisine>Italian</cuisine>

<phoneno>2610325833</phoneno>

</restaurant>

<restaurant>

<name>Kalamarakia</name>

<address>21 Poseidonos Street, Patras, 264 45</address>

<cuisine>Greek</cuisine>

<phoneno[html]>2610428066</phoneno>

</restaurant>

</restaurants>

### style.xsl

*<?xml version="1.0" encoding="UTF-8"?>*

<xsl:stylesheet version="1.0" xmlns:xsl=["http://www.w3.org/1999/XSL/Transform"](http://www.w3.org/1999/XSL/Transform)>

<xsl:template match="/">

<html>

<body>

<h2>Restaurant List</h2>

<table border="1">

<tr bgcolor="#9acd32">

<th>Name</th>

<th>Adress</th>

<th>Cuisine</th>

<th>PhoneNo</th>

</tr>

<xsl:for-each select="restaurants/restaurant">

<tr>

<td><xsl:value-of select="name"/></td>

<td><xsl:value-of select="address"/></td>

<td><xsl:value-of select="cuisine"/></td>

<td><xsl:value-of select="phoneno"/></td>

</tr>

</xsl:for-each>

</table>

</body>

</html>

</xsl:template>

</xsl:stylesheet>

### Output

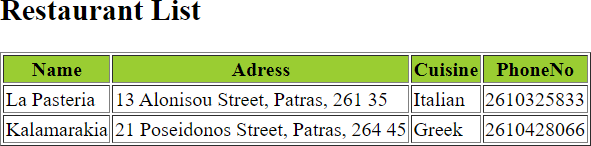


Figure 7: Question 7 - XML

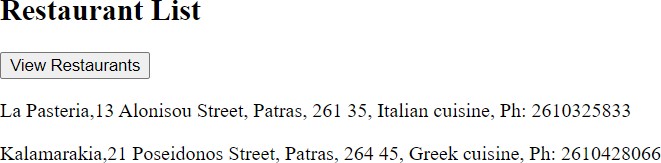


Figure 8: Question 7 - AJAX

# Results

Thus, we have utilised XML, PHP and database connectivity.