**Question 1**

Design address book loads a list of entries, each containing a first and last name. Each time the user clicks a name, the address book uses Ajax functionality to load the person’s address from the server and expand the entry without reloading the page(use XML file)and it does this in parallel with allowing the user to click other names. The application allows the user to search the address book by typing a last name. As the user enters each keystroke, the application asynchronously displays the list of namesin which the last name starts with the characters the user has entered so far. The application also enables the user to add another entry to the address book by clicking the addEntrybutton. The application displays a form that enables live field validation. As the user fills out the form, the zip-code value is validated and used to generate the city and state. The telephone number is validated for correct format. When the Submit button is clicked, the application checks for invalid data and stores the values in a database on the server using PHP.

HTML:

<html>

<head>

<title>ws8\_1</title>

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>

<script src="ws8\_1.js"></script>

<style>

#box

{

background-color: lightgoldenrodyellow;

width: 50%;

border-radius: 2%;

}

a:link{

color: black !important;

}

a:hover{

color: black!important;

}

a:active{

color: black !important;

}

a:visited{

color: black !important;

}

</style>

</head>

<body onload="load()">

<h1>Address Book</h1>

<button onclick="search()">Search</button>

<button onclick="add()">Add an Entry</button>

<div id="search" method="post" >

<p>Search by last name:<input type="text" id = "search\_1" onkeyup="sendDetails()"></p>

<div id="contact\_list" align = "center"></div>

</div>

<div id = "add">

<p>First Name:<input type="text" name="first" class="add" required></p>

<p>Last Name:<input type="text" name="last" class="add" required></p>

<h4>Address:</h4>

<p>Street:<input type="text" name="street" class="add" required></p>

<p id = "city">City:<input type="text" name="city\_input" class="add" id = "city\_input" hidden></p>

<p id = "state">State:<input type="text" name="state\_input" class="add" id = "state\_input" hidden></p>

<p>Zipcode:<input type="number" name = "zipcode" id = "zipcode" class="add" onblur="getDetails()" required></p>

<div class="text-error "></div>

<p>Telephone:<input name="phone" placeholder="123-45-678" pattern="[0-9]{3}-[0-9]{2}-[0-9]{3}" class="add" required></p>

<button onclick="sendForm()">ADD</button>

</div>

</body>

</html>

Javascript:

var city;

var state;

var present\_value;

function load()

{

search();

}

function generate(parsed)

{

$("#contact\_list").html("");

for (var i = 0; i < parsed.length; i++)

{

if (parsed[i].last.includes(present\_value))

{

var box = $("<div></div>");

box.attr("id", "box");

box.attr("align", "center");

var a = $("<a></a>");

a.append($("<h3></h3>").text(parsed[i].first + " " + parsed[i].last));

a.attr("href", "#collapse" + i);

a.attr("data-toggle", "collapse");

var collapse = $("<div></div>");

collapse.addClass("collapse");

collapse.attr("id","collapse" + i);

var street = $("<p></p>").text( parsed[i].street+ " " + parsed[i].zipcode);

var city = $("<p></p>").text( parsed[i].city);

var state = $("<p></p>").text( parsed[i].state);

var number = $("<p></p>").text( parsed[i].t\_number);

collapse.append(street);

collapse.append(city);

collapse.append(state);

collapse.append(number);

box.append(a);

box.append(collapse);

$("#contact\_list").append(box);

}

}

}

function search()

{

document.getElementById('search').style.visibility = "visible";

document.getElementById('add').style.visibility = "hidden";

}

function add()

{

$("#contact\_list").html("");

document.getElementById('search').style.visibility = "hidden";

document.getElementById('add').style.visibility = "visible";

}

function getDetails()

{

var clientKey ="js-WbIEl7R6bwKV40RZXPKJtOkHPrmrVQnstd8tEc2EW0PrLZu78pCUN7Qls0f3Ikju";

var zipcode = document.getElementById('zipcode').value;

var xhr = new XMLHttpRequest(); var url = "https://www.zipcodeapi.com/rest/"+clientKey+"/info.json/" + zipcode + "/radians";

xhr.open("GET", url, true);

xhr.send();

xhr.onreadystatechange =function()

{

if(this.readyState == 4)

{

var output = this.status + this.responseText;

if (this.status == 200)

{

var data = JSON.parse(this.responseText);

document.getElementById('city').innerHTML += data.city;

document.getElementById('city\_input').value = data.city;

city = data.city;

document.getElementById('state').innerHTML += data.state;

document.getElementById('state\_input').value = data.state;

state = data.state;

}

else

{

document.getElementById('text-error').innerHTML = this.responseText;

}

}

};

}

function sendForm()

{

var inputs = document.getElementsByClassName("add");

console.log(inputs);

var data = "";

data += ("first="+inputs[0].value);

data += ("&last="+inputs[1].value);

data += ("&street="+inputs[2].value);

data += ("&city="+inputs[3].value);

data += ("&state="+inputs[4].value);

data += ("&zipcode="+inputs[5].value);

data += ("&phone="+inputs[6].value);

const searchExp = /\s/g;

data = data.replace(searchExp, '+');

console.log(data);

var xhr = new XMLHttpRequest();

xhr.open("POST", "ws8\_1.php", true);

xhr.setRequestHeader('Content-type', 'application/x-www-form-urlencoded');

xhr.send(data);

xhr.onreadystatechange = function()

{

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

{

console.log("Posted successfully");

console.log(this.responseText);

}

};

}

function sendDetails()

{

var input = document.getElementById("search\_1");

console.log(input.value);

var data = "search=" + input.value;

present\_value = input.value;

console.log(data);

var xhr = new XMLHttpRequest();

xhr.open("POST", "ws8\_1.php", true);

xhr.setRequestHeader('Content-type', 'application/x-www-form-urlencoded');

xhr.send(data);

xhr.onreadystatechange = function()

{

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

{

console.log("Posted successfully");

var parsed = JSON.parse(this.responseText);

generate(parsed);

}

};

}

PHP:

<?php

$valid = true;

if(isset($\_POST['first']))

{

$conn= mysqli\_connect("localhost", "ranga", "qwerty", "ws8\_1");

if(!$conn)

{

echo "Connectoin Failed:" . mysqli\_connect\_error() ;

}

else

{

$sql= "INSERT INTO address (first, last, street, city, state, zipcode, t\_number) VALUES ('$\_POST[first]','$\_POST[last]','$\_POST[street]', '$\_POST[city]', '$\_POST[state]', '$\_POST[zipcode]','$\_POST[phone]')";

if (mysqli\_query($conn, $sql))

{

echo "New record created successfully";

}

else

{

echo "Error: " . $sql. "<br>" . mysqli\_error($conn);

}

mysqli\_close($conn);

}

}

if (isset($\_POST['search']))

{

$conn= mysqli\_connect("localhost", "ranga", "qwerty", "ws8\_1");

if(!$conn)

{

echo "Connectoin Failed:" . mysqli\_connect\_error() ;

}

else

{

$retval = $conn->query("SELECT \* FROM address");

$table = [];

while($row = mysqli\_fetch\_assoc($retval))

{

array\_push($table, $row);

}

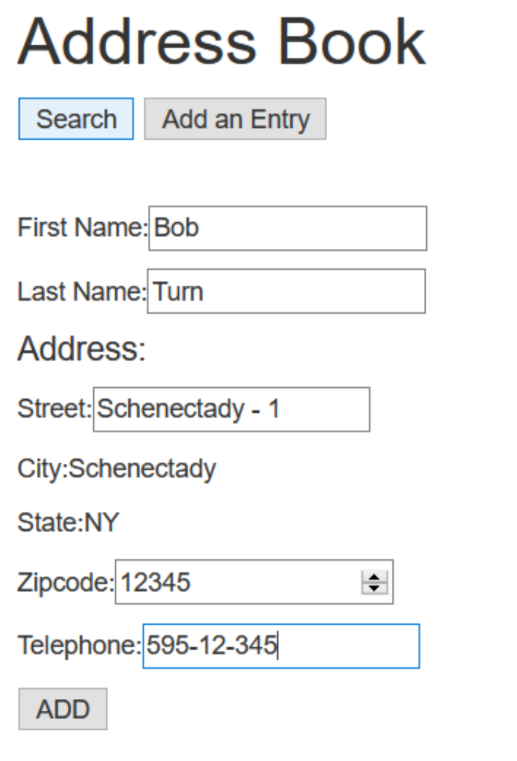
echo json\_encode($table);

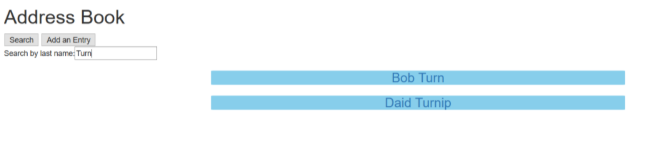
}

}

?>

Output:





**Question 2**

Create an AJAX-based product catalog that obtains its data from JSON files located on the server. The data should be separated into four JSON files. The first file should be a summary file, containing a list of products. Each product should have a title, an image filename for a thumbnail image and a price. The second file should contain a list of descriptions for each product. The third file should contain a list of filenames for the full-size product images. The last file should contain a list of the thumbnail image file names. Each item in a catalogue should have a unique ID that should be included with the entries for that product in every file. Next, create an AJAX-enabled web page that displays the product information in a table. The catalog should initially display a list of product names with their associated thumbnail images and prices. When the mouse hovers over a thumbnail image, the larger product image should be displayed. When the user moves the mouse away from that image, the original thumbnail should be redisplayed. You should provide a button that the user can click to display the product description

HTML:

<html>

<head>

<title>ws8\_2</title>

<style>

.product

{

margin-top: 40px;

border: 1px solid gray;

}

</style>

<script>

function display()

{

var xhr = new XMLHttpRequest();

xhr.open("GET","file1.json", true);

xhr.send();

xhr.onreadystatechange = function()

{

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

{

var obj = JSON.parse(this.responseText);

var cont = document.getElementById("display");

for(var i = 0; i<obj.length; i++)

{

var box = document.createElement("div");

box.setAttribute("class", "product");

box.id = obj[i].id;

box.innerHTML = "Item: " + obj[i].title + "<br>Price: " + obj[i].price;

var img = document.createElement("img");

img.setAttribute("src", obj[i].filename);

img.setAttribute("class", "thumb");

img.setAttribute("onmouseover", "big(this)");

img.setAttribute("onmouseout", "normal(this)");

box.appendChild(img);

var button = document.createElement("button");

button.setAttribute("onclick", "descr(this)");

button.innerHTML = "Show Description";

box.appendChild(document.createElement("br"));

box.appendChild(button);

cont.appendChild(box);

}

}

};

}

function big(img)

{

var xhr = new XMLHttpRequest();

xhr.open("GET","file3.json", true);

xhr.send();

xhr.onreadystatechange = function()

{

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

{

var obj = JSON.parse(this.responseText);

for(var i = 0;i<obj.length; i++)

{

if(obj[i].id == img.parentNode.id)

{

img.setAttribute("src", obj[i].bigname);

}

}

}

}

}

function normal(img)

{

var xhr = new XMLHttpRequest();

xhr.open("GET","file4.json", true);

xhr.send();

xhr.onreadystatechange = function()

{

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

{

var obj = JSON.parse(this.responseText);

for(var i = 0;i<obj.length; i++)

{

if(obj[i].id == img.parentNode.id)

{

img.setAttribute("src", obj[i].name);

}

}

}

}

}

function descr(button)

{

var xhr = new XMLHttpRequest();

xhr.open("GET","file2.json", true);

xhr.send();

xhr.onreadystatechange = function()

{

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

{

var obj = JSON.parse(this.responseText);

for(var i = 0;i<obj.length; i++)

{

if(obj[i].id == button.parentNode.id)

{

button.parentNode.innerHTML += "<br><br>" + obj[i].desc;

}

}

}

}

}

</script>

</head>

<body onload = "display()">

<h2> Product Catalog </h2>

<div id = "display"> </div>

</body>

</html>

FILE1.json:

[

{

"id": "1",

"title": "Item 1",

"price": "100",

"filename": "img1.png"

},

{

"id": "2",

"title": "Item 2",

"price": "200",

"filename": "img2.png"

},

{

"id": "3",

"title": "Item 3",

"price": "300",

"filename": "img3.png"

}

]

FILE2.json

[

{

"id": "1",

"desc": "This is the description for item 1"

},

{

"id": "2",

"desc": "This is the description for item 2"

},

{

"id": "3",

"desc": "This is the description for item 3"

}

]

FILE3.json:

[

{

"id": "1",

"bigname": "bigimg1.png"

},

{

"id": "2",

"bigname": "bigimg2.png"

},

{

"id": "3",

"bigname": "bigimg3.png"

}

]

FILE4.json:

[

{

"id": "1",

"name": "img1.png"

},

{

"id": "2",

"name": "img2.png"

},

{

"id": "3",

"name": "img3.png"

}

]

Output:

