**Web Technology Lab(BCSL504)**

1. **Develop HTML page named as “newpaper.html” having variety of HTML semantic elements with background colors, text-colors & size for figure, table, aside, section, article, header, footer… etc.**

# WebLab\_Style5.css

/\* General body styling \*/ body {

font-family: Arial, sans-serif; margin: 0;

padding: 0;

background-color: #f4f4f4; /\* Light gray background for the page \*/

}

/\* Header styling \*/ header {

background-color: #2c3e50; /\* Dark blue background \*/ color: #fff; /\* White text color \*/

padding: 20px; text-align: center;

}

header h1 { margin: 0;

font-size: 2.5em; /\* Larger font size for the headline \*/ color:yellow;

}

header p {

margin: 10px 0 0;

font-size: 1.2em; /\* Slightly larger font size for the subheading \*/ color:red;

}

/\* Navigation styling \*/ nav {

background-color: #34495e; /\* Slightly lighter blue background \*/ padding: 10px;

text-align: center;

}

nav ul {

list-style-type: none; /\* Removes bullet points \*/ padding: 0;

}

nav ul li { display: inline;

margin: 0 10px;

}

nav ul li a {

color: #fff; /\* White text color for links \*/

text-decoration: none; /\* Removes underline \*/

}

/\* Main content styling \*/ main {

padding: 20px;

}

/\* Section styling \*/ section {

margin-bottom: 20px;

background-color: #fff; /\* White background \*/ padding: 15px;

border-radius: 8px; /\* Rounded corners \*/

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1); /\* Subtle shadow \*/

}

/\* Article styling \*/ article {

margin-bottom: 20px;

}

article header {

border-bottom: 2px solid #3498db; /\* Blue border for article headers \*/ margin-bottom: 10px;

padding-bottom: 10px;

}

article h2 {

font-size: 1.8em;

}

article p {

font-size: 1em;

}

/\* Figure styling \*/ figure {

margin: 20px 0;

}

figure img { width: 100%;

height: auto; /\* Maintains aspect ratio \*/

border-radius: 8px; /\* Rounded corners for images \*/

}

figcaption {

font-size: 0.9em;

color: #7f8c8d; /\* Light gray color \*/ text-align: center;

}

/\* Aside styling \*/ aside {

background-color: #ecf0f1; /\* Light gray background \*/ padding: 15px;

border-radius: 8px;

box-shadow: 0 0 5px rgba(0, 0, 0, 0.1); /\* Subtle shadow \*/

}

aside h3 {

font-size: 1.5em;

}

/\* Table styling \*/ table {

width: 100%;

border-collapse: collapse; margin-top: 20px;

}

table thead {

background-color: #3498db; /\* Blue background for table header \*/ color: #fff; /\* White text color \*/

}

table th, table td {

padding: 10px; text-align: left;

border-bottom: 1px solid #ddd; /\* Light gray border for table rows \*/

}

table tbody tr:hover {

background-color: #f2f2f2; /\* Light gray background on row hover \*/

}

/\* Footer styling \*/ footer {

background-color: #2c3e50; /\* Dark blue background \*/ color: #fff; /\* White text color \*/

padding: 5px; text-align: center; position: fixed; width: 100%;

bottom: 0;

}

# WebLab5.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Newspaper Layout</title>

<link rel="stylesheet" href="E:\WEBDEVELOPMENTBCSL504\WebLab\_style5.css">

</head>

<body>

<header>

<h1> The Times Of India </h1>

<p>Breaking News and More!..</p>

</header>

<nav>

<ul>

</ul>

</nav>

<li><a href="#home">Home</a></li>

<li><a href="#world">World</a></li>

<li><a href="#sports">Sports</a></li>

<li><a href="#entertainment">Entertainment</a></li>

<main>

<section>

<article>

<header>

<h2>Main Article Title: Cyber Security</h2>

<p>Author: Shashidhar T H</p>

</header>

<figure>

<!-- <img src="https://via.placeholder.com/800x400" alt="Placeholder Image"> -->

<img src="E:\WEBDEVELOPMENTBCSL504\CyberSecurity.jpg" alt="Placeholder Image">

<figcaption>Cyber Security.</figcaption>

</figure>

<p>Cybersecurity is the practice of protecting systems, networks, and programs from digital attacks. These cyberattacks are usually aimed at accessing, changing, or destroying sensitive information; extorting money from users through ransomware; or interrupting normal business processes...</p>

</article>

<article>

<header>

<h2>Another Article Title : Cloud Computing</h2>

<p>Author: Vaibhavi S H</p>

</header>

<p>Cloud computing lets client devices access rented computing resources, such as data, analytics and cloud applications over the internet. It relies on a network of remote data centers, servers and storage systems that are owned and operated by cloud service providers </p>

</article>

</section>

<aside>

<h3>Related Articles</h3>

<ul>

<li><a href="#article1">Related Article 1: Full Stack Web Developement</a></li>

<li><a href="#article2">Related Article 2: Full Stack Java Development</a></li>

<li><a href="#article3">Related Article 3: Full Stack UI Development</a></li>

</ul>

</aside>

<section>

<h2>Additional News</h2>

<table>

<thead>

<tr>

<th>Category</th>

<th>Details</th>

</tr>

</thead>

<tbody>

<tr>

<td>News</td>

<td>Details about recent news.</td>

</tr>

<tr>

<td>Updates</td>

<td>Latest updates on various topics.</td>

</tr>

</tbody>

</table>

</section>

</main>

<footer>

<p>&copy; 2024 Newspaper Inc. All rights reserved.</p>

</footer>

</body>

</html>

1. **Apply HTML, CSS and JavaScript to design a simple calculator to perform the following operations: sum, product, difference, remainder, quotient, power, square-root and square.**

# WebLab\_Style6.css

/\* General body styling \*/ body {

font-family: Arial, sans-serif; display: flex;

justify-content: center; align-items: center; height: 100vh; margin: 0;

background-color:orange; //#f0f0f0

}

/\* Calculator container styling \*/

.calculator {

background-color: #fff; border-radius: 10px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1); padding: 20px;

width: 300px; text-align: center;

}

/\* Input fields styling \*/ input[type="number"], input[type="text"] {

width: calc(100% - 20px); padding: 10px;

margin: 5px 0;

border: 1px solid #ddd; border-radius: 5px; font-size: 16px;

}

/\* Button styling \*/ button {

padding: 10px 15px; margin: 5px;

border: none; border-radius: 5px;

background-color: #3498db; color: white;

font-size: 16px; cursor: pointer;

}

button:hover {

background-color: #2980b9;

}

# WebLab6.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Simple Calculator</title>

<link rel="stylesheet" href="WebLab\_Style6.css">

<script type="text/javascript" src="WebLab6\_Java.js"></script>

</head>

<body>

<div class="calculator">

<h1>Simple Calculator</h1>

<form id="calculatorForm">

<input type="number" id="num1" placeholder="Enter first number" required>

<input type="number" id="num2" placeholder="Enter second number" required>

<br><br>

<button type="button" onClick="calculate('sum')">Sum</button>

<button type="button" onclick="calculate('difference')">Difference</button>

<button type="button" onclick="calculate('product')">Product</button>

<button type="button" onclick="calculate('quotient')">Quotient</button>

<button type="button" onclick="calculate('remainder')">Remainder</button>

<button type="button" onclick="calculate('power')">Power</button>

<button type="button" onclick="calculate('square')">Square</button>

<button type="button" onclick="calculate('sqrt')">Square Root</button>

<br><br>

<input type="text" id="result" placeholder="Result" readonly>

</form>

</div>

</body>

</html></html>

# WebLab6\_Java.js

// Function to perform calculator operations function calculate(operation) {

// Get input values

const num1 = parseFloat(document.getElementById('num1').value); const num2 = parseFloat(document.getElementById('num2').value); let result;

// Perform calculations based on the operation switch(operation) {

case 'sum':

result = num1 + num2; break;

case 'difference':

result = num1 - num2; break;

case 'product':

result = num1 \* num2; break;

case 'quotient':

result = num1 / num2; break;

case 'remainder':

result = num1 % num2; break;

case 'power':

result = Math.pow(num1, num2); break;

case 'square':

result = Math.pow(num1, 2); // Square of the first number break;

case 'sqrt':

result = Math.sqrt(num1); // Square root of the first number break;

default:

result = 'Invalid operation';

}

// Display the result document.getElementById('result').value = result;

}

1. **Develop JavaScript program (with HTML/CSS) for:**

# WebLab7a.html

1. Converting JSON text to JavaScript Object

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>JSON to JavaScript Object</title>

<style>

body {

font-family: Arial, sans-serif; margin: 0;

padding: 20px; background-color: #f4f4f4;

}

.container {

max-width: 600px; margin: 0 auto; background: #fff; padding: 20px; border-radius: 8px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

textarea { width: 100%;

height: 150px; margin-bottom: 10px; padding: 10px;

border: 1px solid #ddd; border-radius: 4px;

}

button {

display: block; width: 100%; padding: 10px; border: none; border-radius: 4px;

background-color: #007bff; color: #fff;

font-size: 16px; cursor: pointer;

}

button:hover {

background-color: #0056b3;

}

pre {

background: #f9f9f9; padding: 10px;

border: 1px solid #ddd; border-radius: 4px; white-space: pre-wrap;

}

</style>

</head>

<body>

<div class="container">

<h1>Convert JSON to JavaScript Object</h1>

<textarea id="jsonInput" placeholder="Enter JSON text here..."></textarea>

<button onclick="convertJsonToObject()">Convert to Object</button>

<pre id="jsonObjectResult"></pre>

</div>

<script>

function convertJsonToObject() {

const jsonInput = document.getElementById('jsonInput').value; try {

const jsonObject = JSON.parse(jsonInput); document.getElementById('jsonObjectResult').textContent = JSON.stringify(jsonObject, null, 2);

}

catch (error) {

document.getElementById('jsonObjectResult').textContent = 'Invalid JSON input.';

}

}

</script>

</body>

</html>



1. **Convert JSON results into a date.**

# WebLab7b.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>JSON Date Conversion</title>

<style>

body {

}

font-family: Arial, sans-serif;

.container {

padding: 20px;

}

textarea {

width: 100%; height: 100px;

}

button {

margin-top: 10px;

}

</style>

</head>

<body>

<div class="container">

<h1>Convert JSON Date</h1>

<textarea id="jsonDateInput" placeholder='Enter JSON with date string, e.g., {"date":"2024-08- 20T00:00:00Z"}'>

</textarea>

<button onclick="convertJsonDate()">Convert Date</button>

<p id="dateResult"></p>

</div>

<script>

function convertJsonDate()

{

const jsonDateInput = document.getElementById('jsonDateInput').value; try {

const jsonObject = JSON.parse(jsonDateInput); const date = new Date(jsonObject.date);

document.getElementById('dateResult').textContent = `Converted Date:

${date.toLocaleString()}`;

}

catch (e) {

document.getElementById('dateResult').textContent = 'Invalid JSON or Date';

}

}

</script>

</body>

</html>



1. **Converting From JSON To CSV and CSV to JSON.**

# WebLab7c.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>JSON to CSV and CSV to JSON</title>

<style>

body {

font-family: Arial, sans-serif;

}

.container {

padding: 20px;

}

textarea {

width: 100%; height: 150px;

}

button {

margin-top: 10px;

}

</style>

</head>

<body>

<div class="container">

<h1>JSON to CSV and CSV to JSON</h1>

<h2>Convert JSON to CSV</h2>

<textarea id="jsonToCsvInput" placeholder='Enter JSON text here...'></textarea>

<button onclick="jsonToCsv()">Convert JSON to CSV</button>

<pre id="csvResult"></pre>

<h2>Convert CSV to JSON</h2>

<textarea id="csvToJsonInput" placeholder='Enter CSV text here...'></textarea>

<button onclick="csvToJson()">Convert CSV to JSON</button>

<pre id="jsonResult"></pre>

</div>

<script>

function jsonToCsv() {

const jsonToCsvInput = document.getElementById('jsonToCsvInput').value; try{

const jsonArray = JSON.parse(jsonToCsvInput);

const keys = Object.keys(jsonArray[0]);

const csv = jsonArray.map(row =>keys.map(key => JSON.stringify(row[key], (key, value) => value === null ? '' : value)).join(',')

);

csv.unshift(keys.join(',')); document.getElementById('csvResult').textContent = csv.join('\r\n');

}

catch (e) {

document.getElementById('csvResult').textContent = 'Invalid JSON';

}

}

function csvToJson() {

const csvToJsonInput = document.getElementById('csvToJsonInput').value; try {

const [headerLine, ...lines] = csvToJsonInput.split('\n'); const headers = headerLine.split(',');

const json = lines.map(line => { const values = line.split(',');

const jsonObject = {};

});

headers.forEach((header, index) => {jsonObject[header] = values[index]; return jsonObject;});

document.getElementById('jsonResult').textContent = JSON.stringify(json, null, 2);

}

catch (e) {

document.getElementById('jsonResult').textContent = 'Invalid CSV';

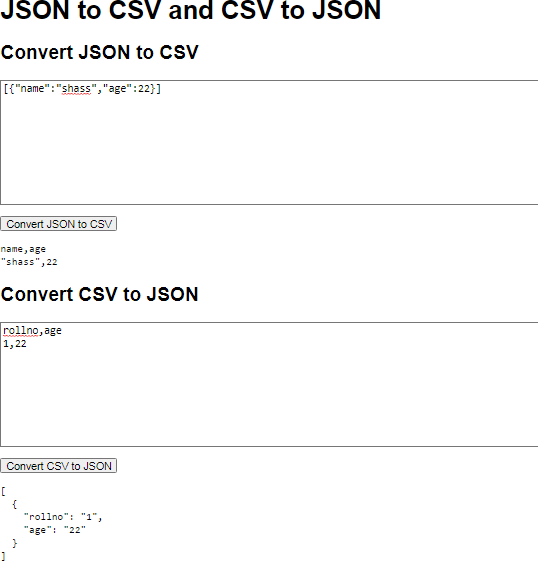
}

}

</script>

</body

</html>



1. **Create hash from string using crypto.createHash() method**

# WebLab7d.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Create Hash from String</title>

<style>

body { font-family: Arial, sans-serif; }

.container { padding: 20px; } input { width: 100%; } button { margin-top: 10px; }

</style>

</head>

<body>

<div class="container">

<h1>Create Hash from String</h1>

<input type="text" id="stringInput" placeholder="Enter string here...">

<button onclick="createHash()">Create Hash</button>

<p id="hashResult"></p>

</div>

<script>

async function createHash() {

const stringInput = document.getElementById('stringInput').value; const encoder = new TextEncoder();

const data = encoder.encode(stringInput);

const hashBuffer = await crypto.subtle.digest('SHA-256', data); const hashArray = Array.from(new Uint8Array(hashBuffer));

const hashHex = hashArray.map(b => b.toString(16).padStart(2, '0')).join(''); document.getElementById('hashResult').textContent = `Hash: ${hashHex}`;

}

</script>

</body>

</html>

