A logo of a institute of technology

Description automatically generatedA close up of a document

Description automatically generated

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

# 21ADA581 WEB APPLICATION DEVELOPMENT

NAME : USN : YEAR/SEM : SECTION : BRANCH :

1. Write a HTML program for the demonstration of Lists.
2. Write a HTML program for time-table using tables.
3. Write HTML for demonstration of cascading stylesheets.
4. Write a JavaScript to design a simple calculator to perform thefollowing operations:sum, product, difference and quotient.
5. Develop and demonstrate a HTML5 file that includes JavaScript script that uses functions for the following problems:

* Parameter: A string
* Output: The position in the string of the left-most vowel
* Parameter: A number
* Output: The number with its digits in the reverse order

1. a. Write a Javascript program for validating REGISTRATION FORM.

b. Write a Javascript program to validate USER LOGIN page.

1. Write an XML for student information and access second students data using DOM.
2. Write a Javascript program for implementing constructor
3. Write a Javascript to add items in a blank array and display the items
4. Write a Javascript program for implementing pattern matching.

|  |  |
| --- | --- |
| **Lab Exercises** | |
| 1. | Write a HTML program for the demonstration of Lists.   1. Unordered List 2. Ordered List 3. Definition List 4. Nested List |
| 2. | Write a HTML program for demonstrating Hyperlinks.   1. Navigation from one page to another. 2. Navigation within the page. |
| 3. | Write a HTML program for time-table using tables. |
| 4. | Write a HTML program to develop a static Home Page using frames. |
| 5. | Write a HTML program to develop a static Registration Form. |
| Write a HTML program to develop a static Login Page. |
| 6. | Write HTML for demonstration of cascading stylesheets.   1. Embedded stylesheets. 2. External stylesheets. 3. Inline styles. |
| 7. | Write a javascript program to validate USER LOGIN page. |
| Write a javascript program for validating REGISTRATION FORM |
| 8. | a. Write a program for implementing XML document for CUSTOMER DETAILS. |
| b. Write an internal Document Type Definition to validate XML for CUSTOMERDETAILS? |
| c. Write an external Document Type Definition to validate XML for CUSTOMERDETAILS? |
| 9. | Write an XML for person information and access the data using XSL. |
| 10. | Write an XML for student information and access second students data using DOM. |

## Write a HTML program for the demonstration of Lists.

* 1. **Unordered List**

## Ordered List

* 1. **Definition List**
  2. **Nested List Unordered List:**

<html>

<head>

<title> Creating Unorder List </title>

</head>

<body bgcolor=”pink”>

<h1 align=”center”> Creating Unorder List</h1>

<h1 align=”center”>List of Colleges in Karnataka</h1>

<ul type=”square”>

<li>CIT Gubbi</li>

<li>SIT Tumkur</li>

<li>NITTE MEENAKSHI Banglore</li>

</ul>

</body>

</html>

**Ordered List:**

<html>

<head>

<title> Creating Order List </title>

</head>

<body bgcolor=”pink”>

<h1 align=”center”> Creating Order List</h1>

<h1 align=”center”>List of branches in CIT GUBBI</h1>

<ol type=”A”>

<li>CSE</li>

<li>ISE</li>

<li>ECE</li>

<li>EEE</li>

<li>CIVIL</li>

<li>ME</li>

<li>AIDS</li>

</ol>

</body>

</html>

## Definition List:

<html>

<head>

<title>Creating Definition List</title>

</head>

<body bgcolor=”pink”>

<h1 align=”center”>Definition List</h1>

<dl>

</dl>

</body>

</html>

<dt>CSE<dd>Computer Science & Engineering

<dt>ECE<dd>Electronics & Communication Engineering

<dt>IT<dd>Information Technology

<dt>EEE<dd>Electrical & Electronics Engineering

<dt>CE<dd>Civil Engineering

## Nested List:

<html>

<head>

<title>Nested Lists</title>

</head>

<body bgcolor=”pink”>

<h1 align=”center”>List of Colleges in Karnataka</h1>

<ol>

<li>Karnataka</li>

<ul>

<li>CIT Gubbi</li>

<li>SIT Tumkur</li>

<li>SSIT Tumkur</li>

</ul>

<li>Bangalore</li>

<ul>

<li>BMS</li>

<li>NITTE</li>

</ul>

</ol>

</body>

</html>

**.**

# Write a HTML program for time-table using tables.

<html>

<head>

<title>Time Table</title>

</head>

<body bgcolor="skyblue">

<H1><FONT COLOR="DARKCYAN"><CENTER>V SEMESTER TIME TABLE<br>AI&DS</FONT></H1>

<table border="2" cellspacing="3" align="center">

<tr>

<td align="center">TIME/DAY</td>

<td>9:00AM-10:00AM

<td>10:00AM-11:00AM

<td>11:15AM-12:15PM

<td>12:15PM-1:15PM

<td>2:00PM-3:00PM

<td>3:00PM-4:00PM

<td>4:00PM-5:00PM

</tr>

<tr align="center">

<td>MONDAY</td>

<td>21AD581</td>

<td>21AD54</td>

<td>21ADL581</td>

<td>21AD51</td>

<td colspan="2">21CSL46</td>

<td></td>

</tr>

</tr>

<tr align="center">

<td>TUESDAY</td>

<td>21ADL581</td>

<td>21AD52</td>

<td>21AD51</td>

<td>21AD53</td>

<td colspan="2">21CS42</td>

<td></td>

</tr>

<tr align="center">

<td>WEDNESDAY</td>

<td>21AD51</td>

<td>21AD52</td>

<td>21AD54</td>

<td>21AD52</td>

<td colspan="2">21AD53</td>

<td></td>

</tr>

<tr align="center">

<td>THURSDAY</td>

<td>21AD53</td>

<td>21AD45</td>

<td>21AD54</td>

<td>21UH59</td>

<td>21AD51</td>

<td>PROCTOR MEETING</td>

<td>Scheduled Activity<br>4:30PM to 6:00PM</td>

</tr>

<tr align="center">

<td>FRIDAY</td>

<td>21AD54</td>

<td>21AD55</td>

<td>21CIP57</td>

<td>21AD53</td>

<td></td>

<td></td>

<td></td>

</tr>

<tr align="center">

<td>SATURDAY</td>

<td colspan="7"></td>

</tr>

</body>

</html>

## Write HTML for demonstration of cascading stylesheets.

1. **Embedded stylesheets.**

## External stylesheets.

1. **Inline styles.**

## Embedded stylesheets:

<!DOCTYPE html>

<html>

<head>

<title>Embedded Style sheets</title>

<style type="text/css">

body {

background-color: pink;

}

h1 {

color: orange;

text-align: center;

}

p {

font-family: "Times New Roman";

font-size: 20px;

}

</style>

</head>

<body>

<h1>Embedded Style Sheets</h1>

<p>This is a paragraph</p>

</body>

</html>

## External Stylesheets:

**extern.css:**

## body {

## background-color: #d0e4fe;

## }

## h1 {

## color: orange;

## text-align: center;

## }

## p {

## font-family: "Times New Roman";

## font-size: 20px;

## }

## extern.html:

<!DOCTYPE html>

<html>

<head>

<title>External Style Sheets</title>

<link rel="stylesheet" type="text/css" href="extern.css">

</head>

<body>

<h1>External Style Sheets</h1>

<p>This is a paragraph</p>

</body>

</html>

## Inline styles:

<!DOCTYPE html>

<html>

<head>

<title>External Style Sheets</title>

</head>

<body style="background-color: pink;">

<h1 style="color: orange; text-align: center;">Inline Style Sheets</h1>

<p style="font-family: 'Times New Roman'; font-size: 20px;">This is a paragraph</p>

</body>

</html>

## Write a JavaScript to design a simple calculator to perform the following operations: sum, product, difference and quotient.

<html>

<head>

<title>My calculator</title>

<script type="text/javascript"> function call(click\_id)

{

var v1=parseFloat(document.getElementById("ip1").value); var v2=parseFloat(document.getElementById("ip2").value); if(isNaN(v1) || isNaN(v2))

alert("enter a valid number"); else if(click\_id=="add")

document.getElementById("output").value=v1+v2; else if(click\_id=="sub") document.getElementById("output").value=v1-v2; else if(click\_id=="mul") document.getElementById("output").value=v1\*v2; else if(click\_id=="div") document.getElementById("output").value=v1/v2;

}

</script>

</head>

<body>

<center>

<h1> A SIMPLE CALCULATOR PROGRAM</h1>

<table style="background-color:yellow" align=="center">

<tr>

<td>

<form method="get" action="">

<div width=50% align="center">

<label>OP1<input type="text" id="ip1"/></label>

<label>op2<input type="text" id="ip2"/></label>

<lablel>total<input type="text" id="output"/></label>

</div>

<br>

<div width=50% align="center">

<input type="button" value="+" id="add" onclick="call(this.id)"/>

<input type="button" value="-" id="sub" onclick="call(this.id)"/>

<input type="button" value="\*" id="mul" onclick="call(this.id)"/>

<input type="button" value="/" id="div" onclick="call(this.id)"/>

<input type="reset" value="clear"/>

</div>

</form>

</td>

</tr>

</table>

</center>

</body>

</html>

## Develop and demonstrate a HTML5 file that includes JavaScript script that uses functions for thefollowing problems:

* 1. **Parameter: A string**

## Output: The position in the string of the left-most vowel

<!DOCTYPE html>

<html>

<head>

<title>vowel & reverse</title>

</head>

<body>

<script type="text/javascript">

var str = prompt("Enter the input", "");

if (!isNaN(str)) {

alert("No vowel found in the entered string");

var num = parseInt(str);

var rev = 0, remainder;

while (num !== 0) {

remainder = num % 10;

num = parseInt(num / 10);

rev = rev \* 10 + remainder;

}

alert("Reverse of " + str + " is " + rev);

} else {

str = str.toUpperCase();

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

var ch = str.charAt(i);

if (ch === 'A' || ch === 'E' || ch === 'I' || ch === 'O' || ch === 'U') {

break;

}

}

if (i < str.length) {

alert("The position of the leftmost vowel is " + (i + 1));

} else {

alert("No vowel found in the entered string");

}

}

</script>

</body>

</html>

## 6. Write a javascript program for validating REGISTRATION FORM.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Registration Form Validation</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 20px;

}

label {

display: block;

margin-bottom: 8px;

}

input {

width: 100%;

padding: 8px;

margin-bottom: 16px;

}

button {

padding: 10px;

background-color: #4CAF50;

color: white;

border: none;

cursor: pointer;

}

button:hover {

background-color: #45a049;

}

.error {

color: red;

margin-top: 5px;

}

</style>

</head>

<body>

<h2>Registration Form</h2>

<form id="registrationForm">

<label for="username">Username:</label>

<input type="text" id="username" name="username">

<label for="email">Email:</label>

<input type="email" id="email" name="email">

<label for="password">Password:</label>

<input type="password" id="password" name="password">

<div class="error" id="usernameError"></div>

<div class="error" id="emailError"></div>

<div class="error" id="passwordError"></div>

<button type="button" onclick="validateForm()">Register</button>

</form>

<script>

function validateForm() {

// Reset errors

document.getElementById("usernameError").innerHTML = "";

document.getElementById("emailError").innerHTML = "";

document.getElementById("passwordError").innerHTML = "";

// Get form values

var username = document.getElementById("username").value.trim();

var email = document.getElementById("email").value.trim();

var password = document.getElementById("password").value.trim();

// Validation checks

if (username === "") {

document.getElementById("usernameError").innerHTML = "Username is required";

}

if (email === "") {

document.getElementById("emailError").innerHTML = "Email is required";

} else if (!isValidEmail(email)) {

document.getElementById("emailError").innerHTML = "Invalid email format";

}

if (password === "") {

document.getElementById("passwordError").innerHTML = "Password is required";

} else if (password.length < 6) {

document.getElementById("passwordError").innerHTML = "Password must be at least 6 characters";

}

// Additional validation checks can be added

// If all checks pass, the form is considered valid

if (username !== "" && email !== "" && isValidEmail(email) && password !== "" && password.length >= 6) {

alert("Registration successful!");

// You can submit the form or perform other actions here

}

}

function isValidEmail(email) {

// Basic email format validation

var emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

return emailRegex.test(email);

}

</script>

</body>

</html>

## Write an XML for student information and access second students data using DOM.

**Schl.xml:**

<?xml version="1.0"?>

<school>

<class>

<class\_title>XML</class\_title>

<students>

<student>

<firstname>aaa</firstname>

<lastname>bbb</lastname>

</student>

<student>

<firstname>aaa</firstname>

<lastname>bbb</lastname>

</student>

</students>

</class>

</school>

## School.html:

<html>

<head>

<title>Accessing XML data</title>

<script type="text/javascript"> function getStudentData()

{

var xmldoc;

xmldoc=new ActiveXObject("Microsoft.XMLDOM"); xmldoc.load("school.xml"); nodeSchool=xmldoc.documentElement;

nodeClass=nodeSchool.firstChild; nodeStudents=nodeClass.lastChild; nodeStudent=nodeStudents.lastChild; nodeFirstname=nodeStudent.firstChild; nodeLastname=nodeFirstname.nextSibling;

message.innerHTML="Name:"+nodeFirstname.firstChild.nodeValue+" "+nodeLastname.firstChild.nodeValue;

}

</script>

</head>

<body bgcolor="pink">

<center>

<h1>Accessing XML Data</h1>

<div id="message"></div>

<input type="button" value="GET DATA" onClick="getStudentData()">

</center>

</body>

</html>

## Write a Javascript program for implementing contructor

<html>

<body>

<script>

class CompanyName

{

constructor()

{

this.company="NMIT";

}

}

class Employee extends CompanyName { constructor(id,name) {

super(); this.id=id; this.name=name;

}

}

var emp = new Employee(1,"Ram"); document.writeln(emp.id+" "+emp.name+" "+emp.company);

</script>

</body>

</html>

## Write a Javascript to add items in a blank array and display the items

<html>

<head>

<title>Arrays</title>

<style>

body {

padding-top: 50px

}

</style>

</head>

<body>

<input type="text" id="text1"></input>

<input type="button" id="button1" value="Add" onclick="add\_element\_to\_array();"></input>

<input type="button" id="button2" value="Display" onclick="display\_array();"></input>

<div id="Result"></div>

<script>

var x = 0;

var array = Array();

function add\_element\_to\_array() {

array[x] = document.getElementById("text1").value;

alert("Element: " + array[x] + " Added at index " + x);

x++;

document.getElementById("text1").value = "";

}

function display\_array() {

var e = "<hr/>";

for (var y = 0; y < array.length; y++) {

e += "Element " + y + " = " + array[y] + "<br/>";

}

document.getElementById("Result").innerHTML = e;

}

</script>

</body>

</html>

## 10. Write a Javascript program for implementing pattern matching.

<html>

<head>

<title>JavaScript Match Words Starts or Ends with a Pattern Using Regular Expression</title>

</head>

<body>

<script>

var regex = /(\bcar\w\*)/g;

var str = "Words beginning with car: cart, carrot, cartoon. Words ending with car: oscar, supercar.";

var replacement = '<b>$1</b>';

// Highlights the words beginning with car in bold

var result = str.replace(regex, replacement);

document.write(result);

</script>

</body>

</html>