NITTE EDUCATION TRUST

NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY
An Autonomous Institution Approved by UGC/AICTE/Govt. of Karnataka
Accredited by NBA (Tier-I) and NAAC 'A+' Grade
Affiliated to Visvesvaraya Technological University, Belagavi
Post Box No. 6429, Yelahanka, Bengaluru-560064, Karnataka, INDIA

KNOWLEDGE • CHARACTER • UNITY

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

   b. Write a Javascript program to validate USER LOGIN page.
7. Write an XML for student information and access second students data using DOM.
8. Write a Javascript program for implementing constructor
9. Write a Javascript to add items in a blank array and display the items
10. Write a Javascript program for implementing pattern matching.

Lab Exercises	
1.	Write a HTML program for the demonstration of Lists. a. Unordered List b. Ordered List c. Definition List d. Nested List
2.	Write a HTML program for demonstrating Hyperlinks. a. Navigation from one page to another. b. 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. a. Embedded stylesheets. b. External stylesheets. c. 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.

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

a. Unordered List

b. Ordered List

c. Definition List

d. 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>
```

```
<html>
   <head>
      <title>Creating unordered List</title>
   </head>
   <body>
      <h1> Grocery List</h1>
      <ul>
         <li> Bread</li>
         <li> Eggs</li>
         <li> Milk</li>
         <li> Coffee</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>
```

```
<html>
  <head>
    <title>Creating ordered list</title>
  </head>
  <body>
    <h1>Creating a Orderlist</h1>
    <ol>
      <li>item1</li>
      <li>item2</li>
      <li>item3</li>
      <li>item4</li>
    </ol>
  </body>
</html>
```

Definition List:

<dd> (definition description)

(definition term)

<dl> (definition list)

```
<html>
<head>
<title>Creating Definition List</title>
</head>
<body bgcolor=”pink”>
        <h1 align=”center”>Definition List</h1>
        <dl>
              <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
        </dl>
    </body>
</html>
```

```
<html>
   <head>
      <title>Creating A Defination list</title>
   </head>
   <body>
      <h1>Defination List</h1>
      <dl>
         <dt>CSE <dd>Conputer Science and Engineering
         <dt>CSE <dd>Conputer Science and Engineering
         <dt>CSE <dd>Conputer Science and Engineering
         <dt>CSE <dd>Conputer Science and Engineering

      </dl>
   </body>
</html>
```

Nested List:

```
<html>
<head>
<title>Nested Lists</title>
</head>
<body bgcolor=”pink”>
<h1 align=”center”>List of Colleges in Karnataka</h1>
<ol>        ordeded list tag
<li>Karnataka</li>
<ul>        unorded list
<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>
```

.

2. 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>
```

```
<!DOCTYPE html>
<html>
<style>
table, th, td {
  border:1px solid black;
}
</style>
<body>

<h2>TH elements define table headers</h2>

<table style="width:100%">
  <tr>
    <th>Person 1</th>
    <th>Person 2</th>
    <th>Person 3</th>
</tr>
  <tr>
    <td>Emil</td>
    <td>Tobias</td>
    <td>Linus</td>
  </tr>
  <tr>
    <td>16</td>
    <td>14</td>
    <td>10</td>
  </tr>
</table>

<p>To understand the example better, we have
added borders to the table.</p>

</body>
</html>
```

```
 <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>
```

3. **Write HTML for demonstration of cascading stylesheets.**

    d. **Embedded stylesheets.**

    e. **External stylesheets.**

    f. **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>
```

4. 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>
```

```
<html>
    <style>
        body {
            font-family: 'Times New Roman', Times, serif;
            text-align: center;
            margin: 50px;
        }
        input {
            padding: 10px;
            margin: 5px;
        }
        button {
            padding: 10px;
            margin: 5px;
            cursor: pointer;
        }
    </style>
    <head>
        <title>Simple Calculator</title>
    </head>
    <body>
        <h2>Simple Calculator</h2>
        <input type="number" id="num1" placeholder="Enter number">
        <input type="number" id="num2" placeholder="Enter another number">

        <br>
        <button onclick="sum()">Sum</button>
        <button onclick="diff()">Difference</button>
        <button onclick="product()">Product</button>
        <button onclick="quotient()">Quotient</button>
        <br>
        <h3 span id="result">0</h3>
        <script>
            function sum() {
                setResult(getNum1() + getNum2());
            }
            function diff() {
                setResult(getNum1() - getNum2());
            }
            function product() {
                setResult(getNum1() * getNum2());
            }
            function quotient() {
                setResult(getNum1() / getNum2());
            }
            function setResult(value) {
                document.getElementById("result").innerText = value;
            }
            function getNum1() {
                return parseFloat(document.getElementById("num1").value);
            }

            function getNum2() {
                return parseFloat(document.getElementById("num2").value);
            }
        </script>
    </body>
</html>
```

5. **Develop and demonstrate a HTML 5 file that includes JavaScript script that uses functions for thefollowing problems:**

a. **Parameter: A string**

b. **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>
```

```
<!html>
<head>
    <title>Find the left most vowels</title>
    <script>
function findLeftMostVowelPosition(inputString) {
    // Convert the string to lowercase for case-insensitivity
    const lowercaseString = inputString.toLowerCase();

    // Define the vowels
    const vowels = ['a', 'e', 'i', 'o', 'u'];

    // Iterate through each character in the string
    for (let i = 0; i < lowercaseString.length; i++) {
        // Check if the character is a vowel
        if (vowels.includes(lowercaseString[i])) {
            // Return the position of the left-most vowel (1-indexed)
            return i + 1;
        }
    }

    // If no vowel is found, return -1
    return -1;
}

function displayResult(){
    const userInput = prompt("Enter the string")
    const result = findLeftMostVowelPosition(userInput)
    alert(`Left-Most Vowel Postion:${result}`)
}
    </script>
    </head>
    <body>
        <button onclick="displayResult()">Find the left most vowel</button>
    </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">
```

```
<html>
    <style>
        input{
            text-align: center;
            padding: 10px;
            margin: 5px;
            display: block;
        }
    </style>
    <head>
        <title>Register Form</title>

    </head>

    <body>
        <form >
        <h2>Regestion Form</h2>

        <input type="text" id="username"
placeholder="username" required>
        <input type="email" id="email"
placeholder="Email" required>
        <input type="password" id="password"
placeholder="password" required>
        <input type="password" id="
confirm-password" placeholder="
confirm-password" required>
        <input type="submit" value="Register">

    </form>

    </body>
</html>
```

```
    <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>
```

7. **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>
```

8. Write a Javascript program for implementing contructor

```
<!DOCTYPE html>

<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>
```

```
// Constructor function for creating a Person
object
function Person(name, age, gender) {
    this.name = name;
    this.age = age;
    this.gender = gender;

    // Method to display information about the
person
    this.displayInfo = function() {
        console.log(`Name: ${this.name}, Age: $
{this.age}, Gender: ${this.gender}`);
    };
}

// Creating instances of the Person object
using the constructor
var person1 = new Person("John Doe", 25, "
Male");
var person2 = new Person("Jane Smith", 30, "
Female");

// Displaying information about the persons
using the method
person1.displayInfo();
person2.displayInfo();
```

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

```
<!DOCTYPE html>
<html>

<head>

  <meta charset="utf-8" />

  <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;
   }
```

```
<html>
<head>
    <title>Add Elements in Blank Array</title>
</head>

<body>
    <input type="text" id="text">
    <input type="button" id="button1" value="Add" onclick="Add()">
    <input type="button" id="button2" value="Display" onclick="Display()">
    <div id="result"></div>
</body>

<script>
    let x = 0;
    let array = [];

    function Add() {
        array.push(document.getElementById("text").value);
        x++;
        alert(`Element ${array[x - 1]} is added at index ${x - 1}`);
        document.getElementById("text").value = ""; // Clear input field after adding
    }

    function Display() {
        let e = "";
        for (let y = 0; y < array.length; y++) {
            e += "Element " + y + " = " + array[y] + "<br/>";
        }
        document.getElementById("result").innerHTML = e;
    }
</script>
</html>
```

```
  </script>
</body>
</html>
```

10. Write a Javascript program for implementing pattern matching.

```
<!DOCTYPE html>
<html lang="en">

<head>

  <meta charset="utf-8">

  <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>
```