PANIMALAR ENGINEERING COLLEGE

An Autonomous Institution, Affiliated to Anna University, Chennai

A CHRISTIAN MINORITY INSTITUTION

(JAISAKTHI EDUCATIONAL TRUST)

APPROVED BY ALL INDIA COUNCIL FOR TECHNICAL EDUCATION

Bangalore Trunk Road, Varadharajapuram

Poonamallee, Chennai – 600 123.

DEPARTMENT OF INFORMATION TECHNOLOGY

REGISTER NUMBER : 211419205038

NAME OF THE STUDENT : DEVENDRAN V

IT8511
WEB TECHNOLOGY LABORATORY

III YEAR – V SEMESTER

PANIMALAR ENGINEERING COLLEGE

An Autonomous Institution, Affiliated to Anna University, Chennai

A CHRISTIAN MINORITY INSTITUTION

(JAISAKTHI EDUCATIONAL TRUST)

APPROVED BY ALL INDIA COUNCIL FOR TECHNICAL EDUCATION

Bangalore Trunk Road, Varadharajapuram

Poonamallee, Chennai – 600 123.

BONAFIDE CERTIFICATE

This is a Certified Bonafide Record Book of Mr./~~Ms.~~ : **DEVENDRAN V**

REGISTER NO **: 211419205038** submitted for Anna University

practical examination held on _____ in **IT8511 WEB TECHNOLOGY LABORATORY** during Examinations.

HEAD OF THE DEPARTMENT

FACULTY IN-CHARGE

EXTERNAL EXAMINER

INTERNAL EXAMINER

TABLE OF CONTENTS

DEVENDRAN V
211419205038

EXNO: 1 IMPLEMENTATION OF EMBED IMAGE USING VARIOUS HOT SPOT

DATE:21.08.21

AIM:

To implement and embed the map for various hot spot and display the details of hot spot by using HTML.

ALGORITHM:

1. Start the process.
2. To specify the html language used to embed an image into the browser by using <img> element.
3. Open the new window to display the clicked hot spot details from the map.
4. By using src attribute in <img> tag to load the map image into the web browser and give the name and id for it.
5. To identify various hot spot in the loaded image with different shape as circle/rect/poly/point in <area> tag in <map>using map name and map id.like,

```
<map name="Map" id="Map">
<area shape="circle" coords="110, 447, 11" href="tamilnadu.html" />
</map>
```

6. To highlight the clicked hot spot in the loaded image using <area> tag.
7. To display the clicked hot spot details from the loaded image by using “href” attribute of <area> tag.
8. Stop the process

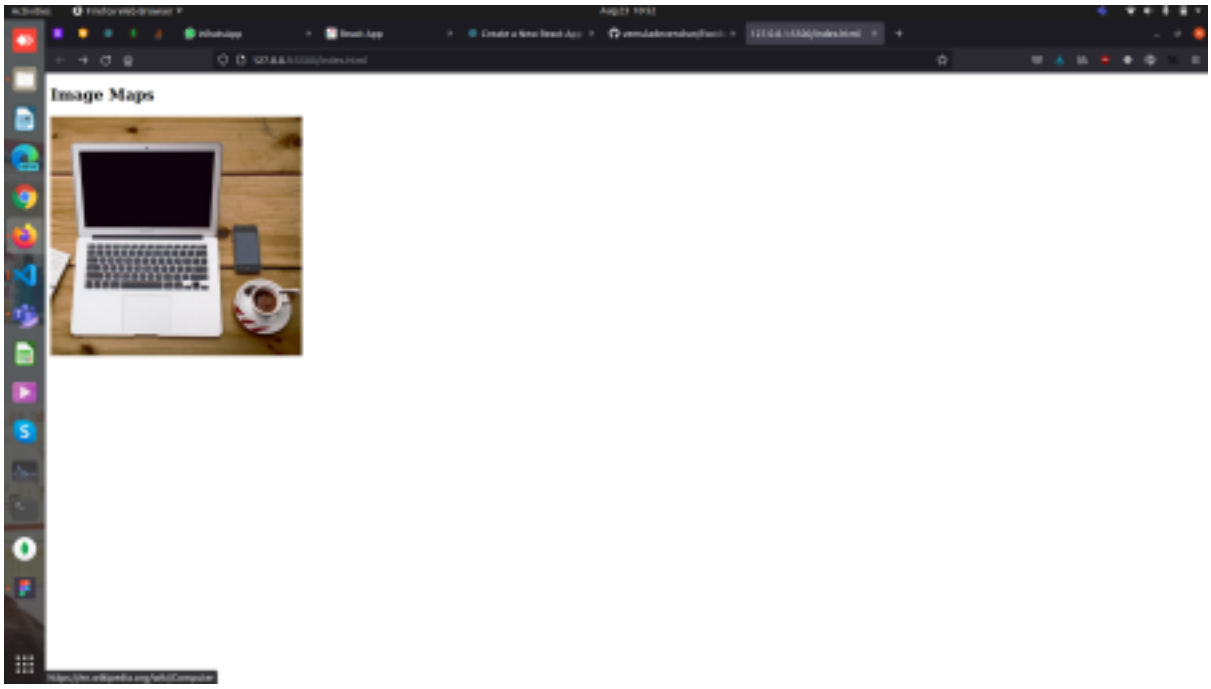
PROGRAM:

```
<!DOCTYPE html>
<html>
<body>
<h2>Image Maps</h2>
<img src="workplace.jpg" alt="Workplace" usemap="#workmap"
width="400" height="379">
<map name="workmap">
<area shape="rect" coords="34,44,270,350" alt="Computer"
href="https://en.wikipedia.org/wiki/Computer">
<area shape="rect" coords="290,172,333,250" alt="Phone"
href="https://en.wikipedia.org/wiki/Telephone">
<area shape="circle" coords="337,300,44" alt="Cup of coffee"
href="https://en.wikipedia.org/wiki/Coffee">
</map>
</body>
```

DEVENDRAN V
211419205038

</html>

OUTPUT :



RESULT:

The implementation of embedded an image for various hot spot and display the details of hot spot by using HTML was written , executed and verified successfully.

DEVENDRAN V
211419205038

EXNO:2 CREATE A WEBPAGE FOR REGSITRATION FORM
DATE:28.08.21

AIM:
To create a web page which includes registration form for any application containing various attributes like table tag, form tag and List tag display the related information.

ALGORITHM:
1. Start the process.

2. To specify the html language used to embed an image into the browser by using specified Element .
3. Open the new window to display the details of registration form .
4. By using form tag to display the registration form.

5. To identify various table tag in the registration form by using border, color, cellspacing and cellpading in the form.
6. To display the web browser in multiple section we use frame tag.
7. Stop the process.

PROGRAM:
FORM VALIDATION:

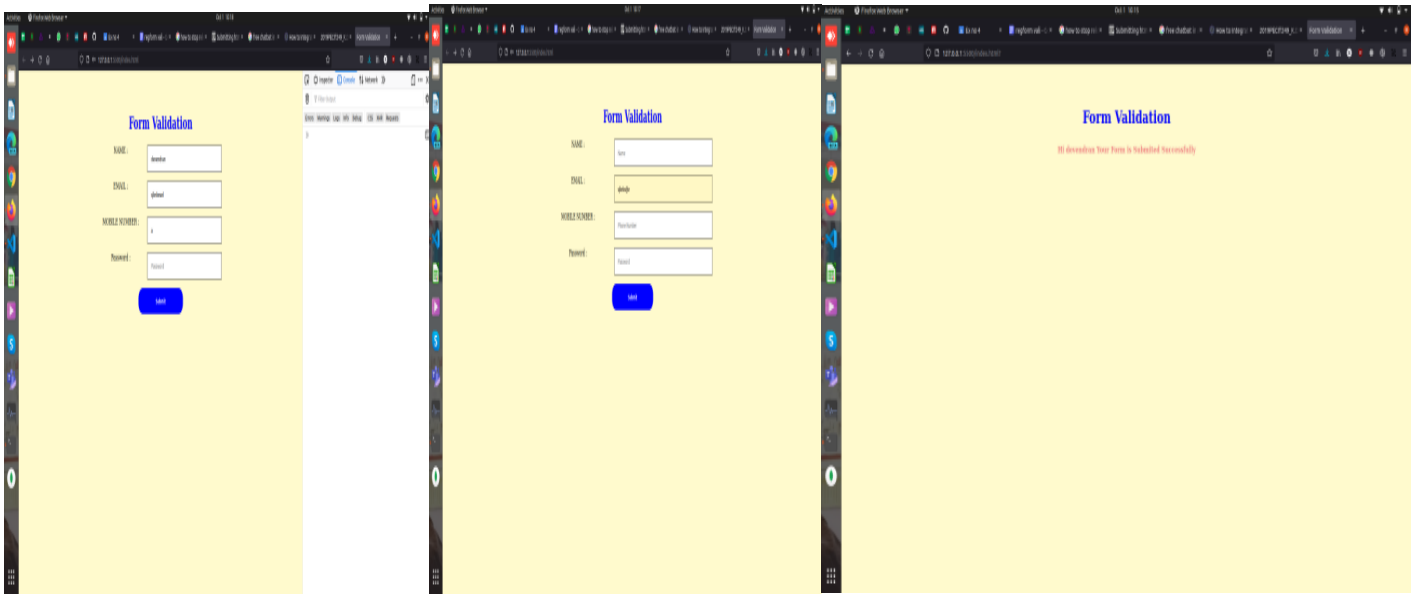
```
<!DOCTYPE html>
        <html lang="en">

        <head>
                <meta charset="UTF-8">
                <meta http-equiv="X-UA-Compatible" content="IE=edge">
                <meta name="viewport" content="width=device-width,
                initial-scale=1.0"> <title>Form Validation</title>
        </head>

        <body>

                <style>
                body {
                display: flex;
                justify-content: center;
                background-color: lemonchiffon;
                }

                .topic {
                text-align: center;
                font-weight: bold;
                color: blue;
                }
```

```
        .main {
        width: 50%;
        padding: 3rem;
        }

        .form-input {
        border: 1px solid black;
        padding: 1rem;
        width: 50%;
        }

        .input-wrapper {
        display: flex;
        flex-wrap: wrap;
        margin: 1rem;
        }

        label {
        width: 40%;
        text-align: center;
}

        .submit-btn {
        border: none;
        padding: 1rem;
        width: 200px;
        color: white;
        background-color: blue;
        border-radius: 20px;
        font-weight: bold;
        }
        </style>

        <!-- script file -->

        <script>


        function checkForm(event) {
        event.preventDefault();
        const name = document.getElementById('name').value;

        if (name) {
        console.log('hiiiii')
        document.getElementById('form').style.display = 'none'
        document.getElementById('submit').innerText = `Hi ${name} Your Form is
        Submited Successfully` }

        }
```

```
        </script>


        <div class="main">
        <h1 class="topic">
        Form Validation
        </h1>


        <form id="form" onsubmit="checkForm(event)">
        <div class="input-wrapper">
        <label for="name">NAME : </label>
        <input type="text" placeholder="Name" id="name" minlength="1"
        class="form-input" required> </div>
        <div class="input-wrapper">
        <label for="email">EMAIL : </label>
        <input type="email" id="email" placeholder="Email" class="form-input" required>
        </div>
        <div class="input-wrapper">
        <label for="mobile">MOBILE NUMBER : </label>
    <input type="tel" id="mobile" placeholder="Phone Number" minlength="10"
        maxlength="10" class="form-input" required>
        </div>
        <div class="input-wrapper">
        <label for="password">Password : </label>
        <input type="password" id="password" placeholder="Password" minlength="5"
        class="form-input" required> </div>
        <div style="text-align: center;">
        <button value="button" class="submit-btn">Submit</button>
        </div>
        </form>

        <p id="submit" style="margin-top: 30px; text-align: center; font-weight: bold;

        color: lightcoral;"> </p> </div>

    </body>

    </html>
```

OUTPUT :

DEVENDRAN V
211419205038



RESULT:

Thus the program for creating a web page for registration form was verified and executed successfully.

DEVENDRAN V
211419205038

EXNO: 4 BASIC JAVASCRIPT PROGRAM

DATE**:** 11:09:21

AIM:

To create a script that calculates the basic JavaScript program.

ALOGIRTHM:

1. Start the process.
2. Create the form for getting the input “n” using <form> tag.
3. In form, we create the input text box to get one input for calculate factorial and generate the Fibonacci series using window. Prompt () object element and one button element to calculate.
4. create the output and display the result of factorial and Fibonacci series numbers generation of given inputs.
5. Define the function name as “calc ()” to process the given input for calculate the operation using JavaScript function.
6. Result should be assigned into corresponding html document in web browser.
7. Stop the process.

1. Write a script that calculates the square and cube of numbers from 5 to 15 using alert dialog.

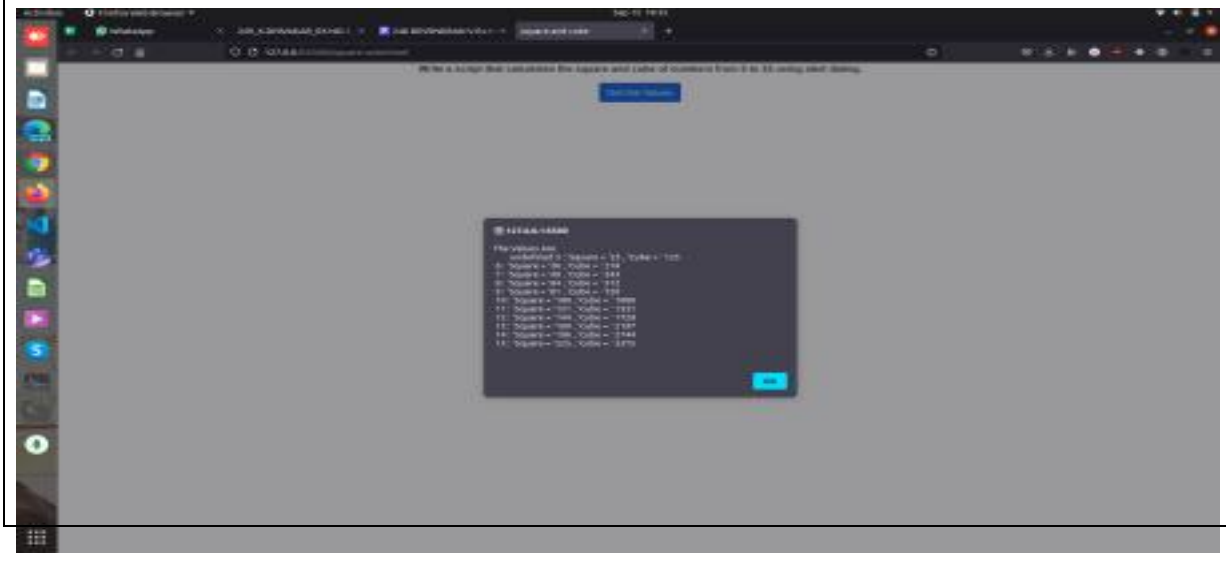
1. **PROGRAM:**

```
<!DOCTYPE html>

<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <!-- CSS only -->
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.m in.css" rel="stylesheet"

integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJ
ByhZMI3AhiU" crossorigin="anonymous">

    <title>square and cube</title>
</head>

<script>
```

DEVENDRAN V
211419205038

```
function squareAndCube(params) {
        let result;
        for (let index = 5; index <= 15; ++index) {
            result += ` ${index} : 'Square = '${Math.pow(index, 2)} , 'Cube = ' ${Math.pow(index, 3)}
\n`;
        }
        window.alert(`The Values Are
        ${result}
        `);



</script>

<body>
    <div>
        <p class="fw-bold text-center">
            Write a script that calculates the square and cube of numbers from 5 to 15 using alert
            dialog.
        </p>
        <div class="text-center">
        <button class="btn btn-primary" onclick="squareAndCube()"> Get the Values
            </button>
        </div>
    </div>
</body>
</html>
```

OUTPUT:



DEVENDRAN V
211419205038

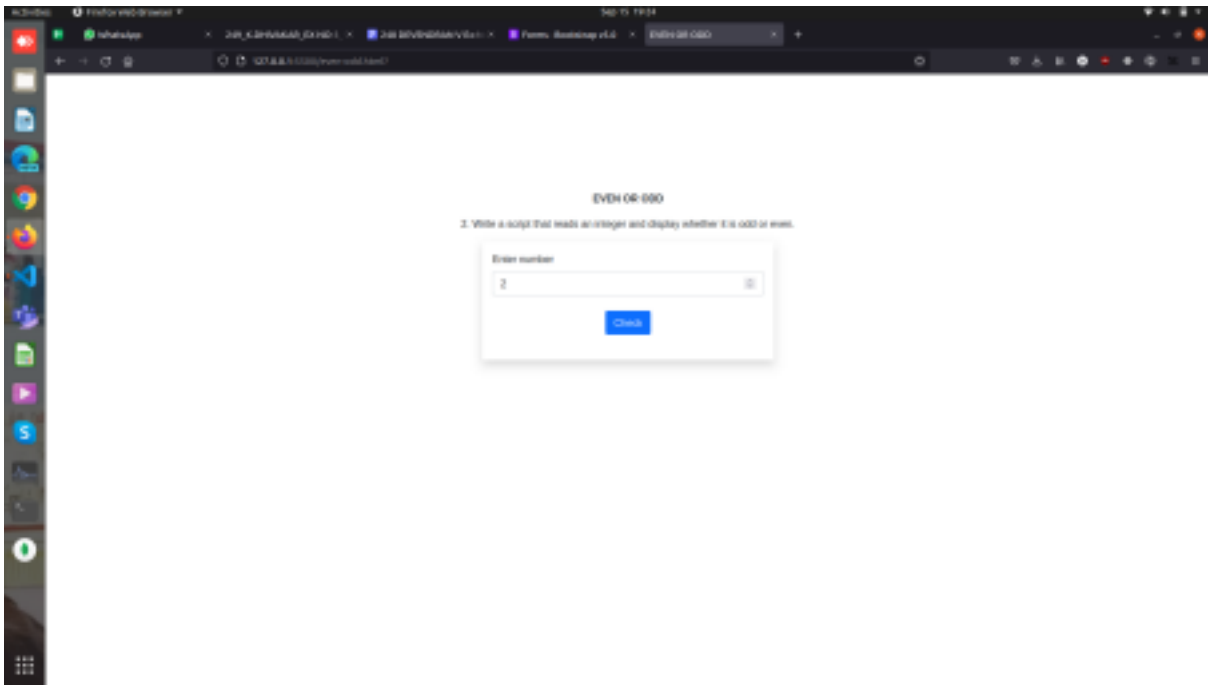
2. Write a script that reads an integer and display whether it is odd or even.

PROGRAM :

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

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <!-- CSS only -->
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.m in.css" rel="stylesheet"

integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJ
ByhZMI3AhiU" crossorigin="anonymous">
    <title>EVEN OR ODD</title>
</head>

<body>

    <script>


        function evenOrOdd() {

            const number =
document.getElementById('exampleInputPassword1').value;

            if (number % 2 === 0) {
                document.getElementById('result').innerText = `${number} : is even number`;
                return;
            }
            document.getElementById('result').innerText = `${number} : is Odd number`;
            return;
        }
    </script>

    <p class="fw-bold text-center" style="margin-top: 10%;">EVEN OR ODD</p>
```

```
    <p class="text-center"> 2. Write a script that reads an integer and display whether it is odd or even.
</p>

    <div class="d-flex justify-content-center">
        <div class="col-12 col-md-6 col-lg-3 shadow p-3">
            <label for="exampleInputPassword1" class="form-label">Enter number</label>
            <input type="number" class="form-control"
id="exampleInputPassword1" required>
            <div class="text-center mt-4">
            <button class="btn btn-primary" onclick="evenOrOdd()"> Check
                </button>
            </div>
            <p class="text-center mt-4 fw-bold text-success"
id="result"></p>
        </div>
    </div>

</body>

</html>
```

OUTPUT :



DEVENDRAN V
211419205038

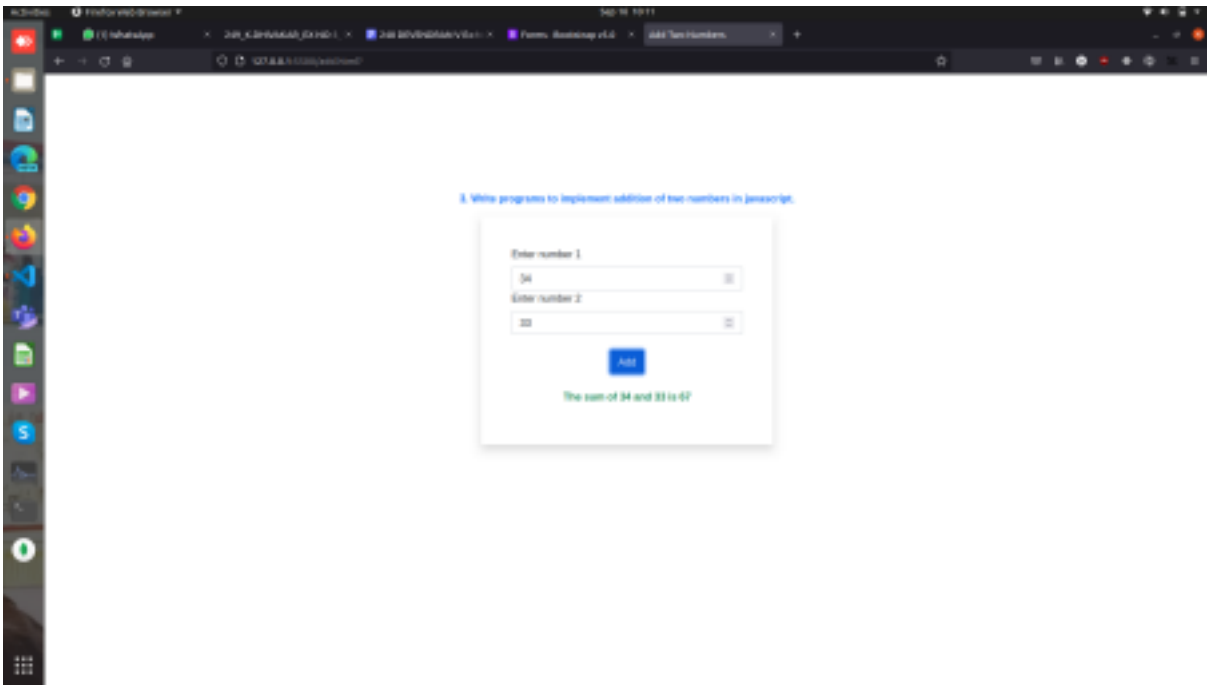
3. Write programs to implement addition of two numbers in javascript.

PROGRAM :

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

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <!-- CSS only -->
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.m in.css" rel="stylesheet"

integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJ
ByhZMI3AhiU" crossorigin="anonymous">
    <title>Add Two Numbers</title>
</head>

<body>
    <script>
        function add(event) {
                                        const a = document.getElementById('number-1').value;
                                        const b = document.getElementById('number-2').value;
            if (!a && !b) {
                return;
            }
            const result = parseInt(a) + parseInt(b);
            document.getElementById('result').innerText = `The sum of ${a} and ${b} is ${result}`
        }
    </script>
    <div>
        <p class="fw-bold text-center text-primary" style="margin-top: 10%;">
            3. Write programs to implement addition of two numbers in javascript.
        </p>
        <div class="d-flex justify-content-center">
                        <div class="col-12 col-md-6 col-lg-3 shadow p-5">
                                <label for="number-1" class="form-label">Enter number
1</label>
```

```
                <input type="number" class="form-control" id="number-1"> <label for="number-2"
                class="form-label">Enter number
2</label>
                <input type="number" class="form-control" id="number-2"> <div class="text-center mt-
                4">
                    <button class="btn btn-primary"
onclick="add()">Add</button>
                </div>
                                        <p class="text-center mt-4 fw-bold text-success"
id="result"></p>
                                                                                    </div>
            </div>
        </div>

</body>

</html>
```

OUTPUT :



DEVENDRAN V
211419205038

4.Write a script that asks the user to enter two integers and displays the larger number followed by the words “is larger” in an information message dialog. If the numbers are equal, display the message “These numbers are equal.”

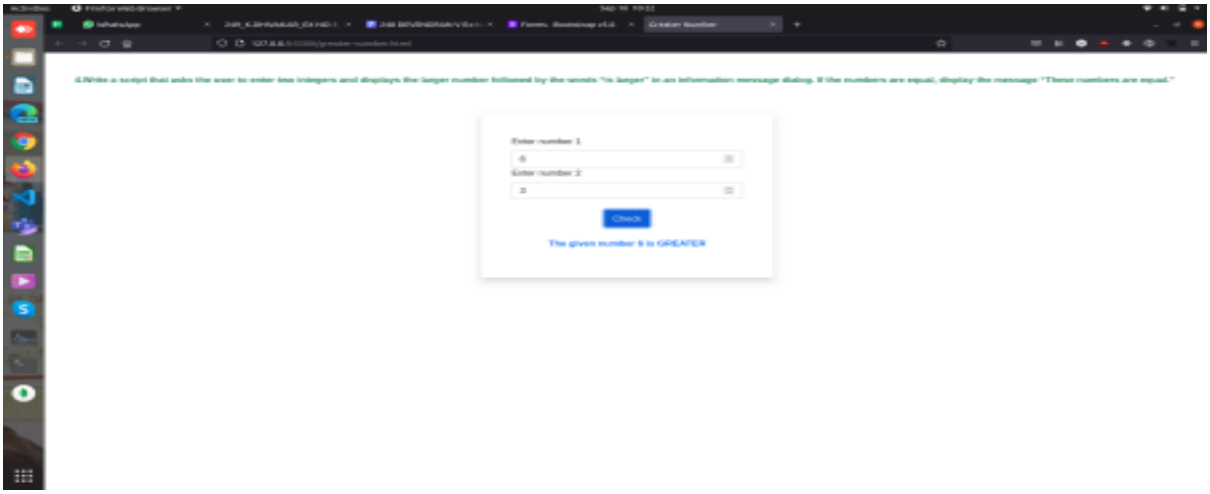
PROGRAM :

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

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.m in.css" rel="stylesheet"

integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJ
ByhZMI3AhiU" crossorigin="anonymous">

    <title>Greater Number</title>
</head>

<body>

    <script>
        function greaterNumber() {
            const a =
parseInt(document.getElementById('number-1').value);
            const b =
parseInt(document.getElementById('number-2').value);
            if (!a && !b) {
                return;
            }

            if(a === b){
                document.getElementById('result').innerText = `The given number ${a} and ${b} are
equal`;
                return;
            }

            if(a > b){
```

```
            document.getElementById('result').innerText = `The given number ${a} is GREATER`;
            return;
        }

        document.getElementById('result').innerText = `The given number ${b} is GREATER`;

    }
</script>

<p class="fw-bold text-success p-5">
    4.Write a script that asks the user to enter two integers and displays the larger number followed by
    the words “is larger” in an information message dialog. If the numbers are equal, display the message
    “These numbers are equal.”
</p>
<div class="d-flex justify-content-center">
    <div class="col-12 col-md-6 col-lg-3 shadow p-5">
                                <label for="number-1" class="form-label">Enter number 1</label>
        <input type="number" class="form-control" id="number-1"> <label for="number-2" class="form-label">Enter number
2</label>
        <input type="number" class="form-control" id="number-2"> <div class="text-center mt-4">
            <button class="btn btn-primary" onclick="greaterNumber()">Check</button>
        </div>
        <p class="text-center mt-4 fw-bold text-primary" id="result"></p>
    </div>
</div>

</body>

</html>
```

OUTPUT :

DEVENDRAN V
211419205038



5.Write a script that inputs three integers from the user and displays the sum, average, and product of the numbers in an alert dialog.

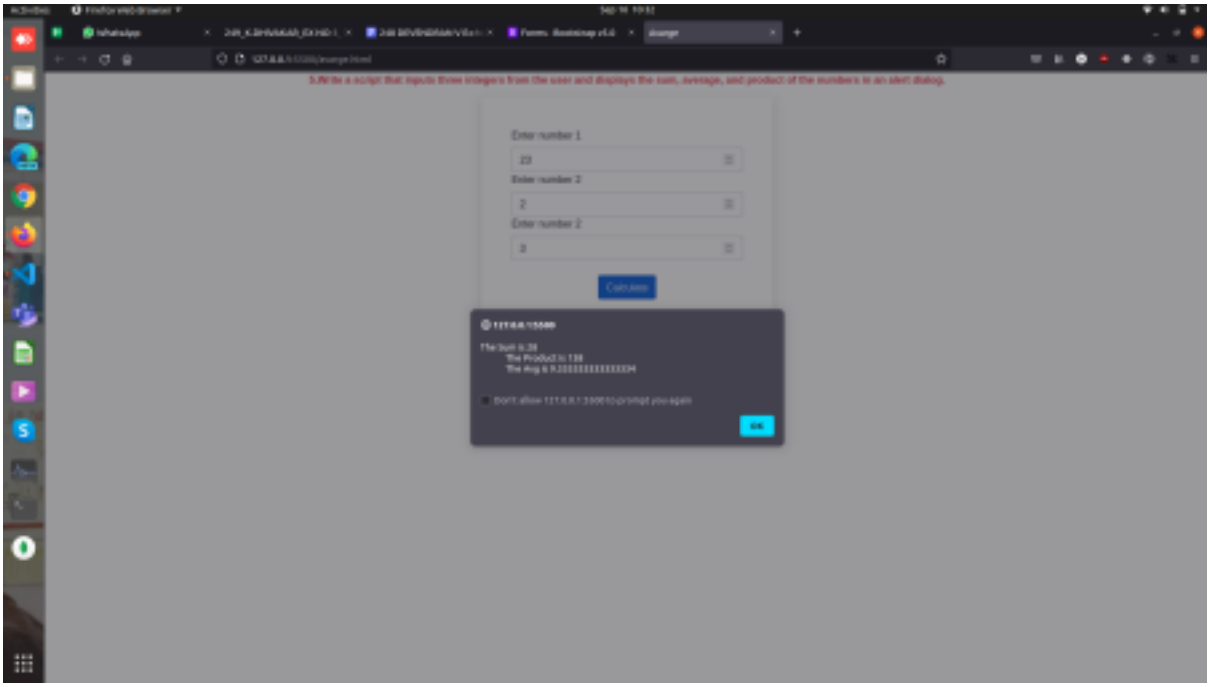
PROGRAM :

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

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.m in.css" rel="stylesheet"

integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJ
ByhZMI3AhiU" crossorigin="anonymous">
    <title>Avarge</title>
</head>

<body>

    <script>

        function avarge() {
            const a =
parseInt(document.getElementById('number-1').value);
            const b =
```

```
parseInt(document.getElementById('number-2').value);
            const c =
parseInt(document.getElementById('number-3').value);

            if (!a && !b && !c) {
                return;
            }
            const sum = a + b + c;
            const product = a * b * c;
            const avg = (a + b + c) / 3;

            window.alert(`The Sum is ${sum}
            The Product is ${product}
            The Avg is ${avg}
            `)
        }

    </script>

    <p class="fw-bold text-center text-danger px-5">
        5.Write a script that inputs three integers from the user and displays the sum, average, and
product of the
        numbers in an alert dialog.
    </p>
    <div class="d-flex justify-content-center">
        <div class="col-12 col-md-6 col-lg-3 shadow p-5">
            <label for="number-1" class="form-label">Enter number 1</label>
            <input type="number" class="form-control" id="number-1"> <label for="number-2"
class="form-label">Enter number 2</label>
            <input type="number" class="form-control" id="number-2"> <label for="number-3"
            class="form-label">Enter number
2</label>
            <input type="number" class="form-control" id="number-3"> <div class="text-center mt-
            4">
                <button class="btn btn-primary"
onclick="avarge()">Calculate</button>
            </div>
        </div>
    </div>

</body>
```

```
</html>
```

OUTPUT :

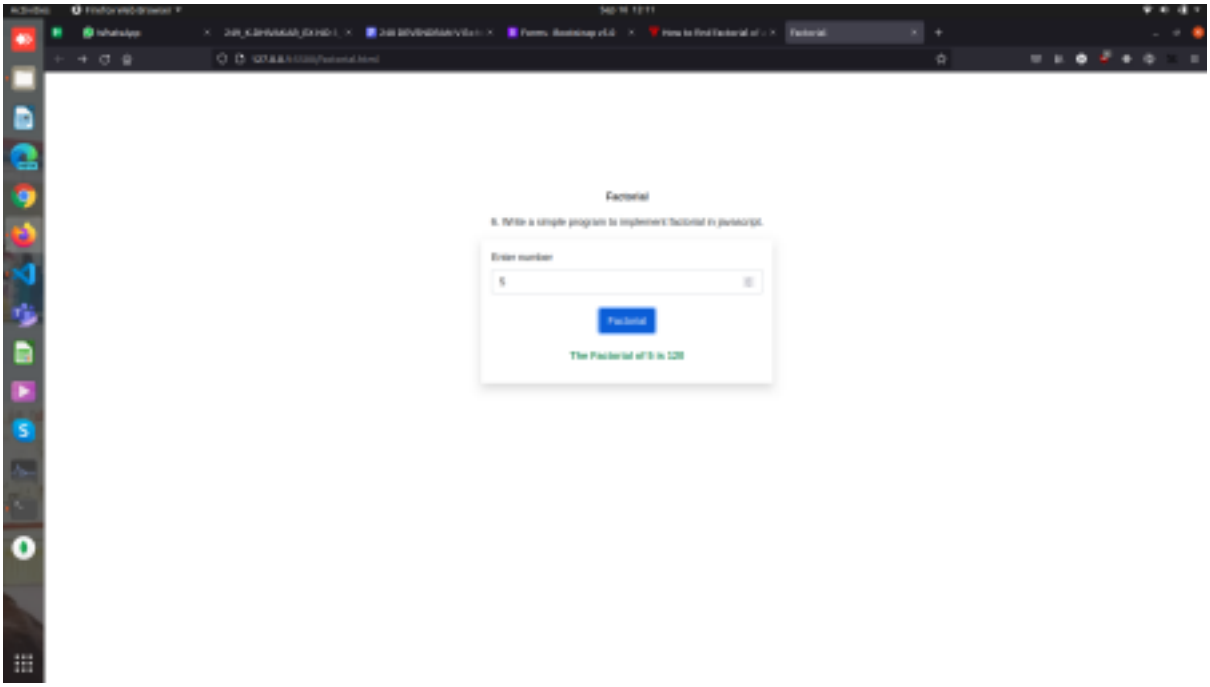


6. Write a simple program to implement factorial in javascript. PROGRAM :

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

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <!-- CSS only -->
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.m in.css" rel="stylesheet"

integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJ
ByhZMI3AhiU" crossorigin="anonymous">

    <title>Factorial</title>
</head>

<body>

    <script>
```

```
        function factorial() {
            const number =
        parseInt(document.getElementById('exampleInputPassword1').value); let result = 1;
            if (number === 0) {
                return;
            }

            for (let i = 1; i <= number; i++) {
                result = result * i;
            }

            document.getElementById('result').innerText = `The Factorial of ${number} is ${result}`;

        }
    </script>

    <p class="fw-bold text-center" style="margin-top:
10%;">Factorial</p>
    <p class="text-center"> 6. Write a simple program to implement factorial in javascript.
    </p>

    <div class="d-flex justify-content-center">
        <div class="col-12 col-md-6 col-lg-3 shadow p-3">
            <label for="exampleInputPassword1" class="form-label">Enter number</label>
            <input type="number" class="form-control"
id="exampleInputPassword1">
            <div class="text-center mt-4">
            <button class="btn btn-primary" onclick="factorial()"> Factorial
                </button>
            </div>
            <p class="text-center mt-4 fw-bold text-success"
id="result"></p>
        </div>
    </div>

</body>

</html>
```

OUTPUT :



7. Write a java script that inputs from the user the radius of a circle and displays the circle's diameter, circumference and area.

PROGRAM :

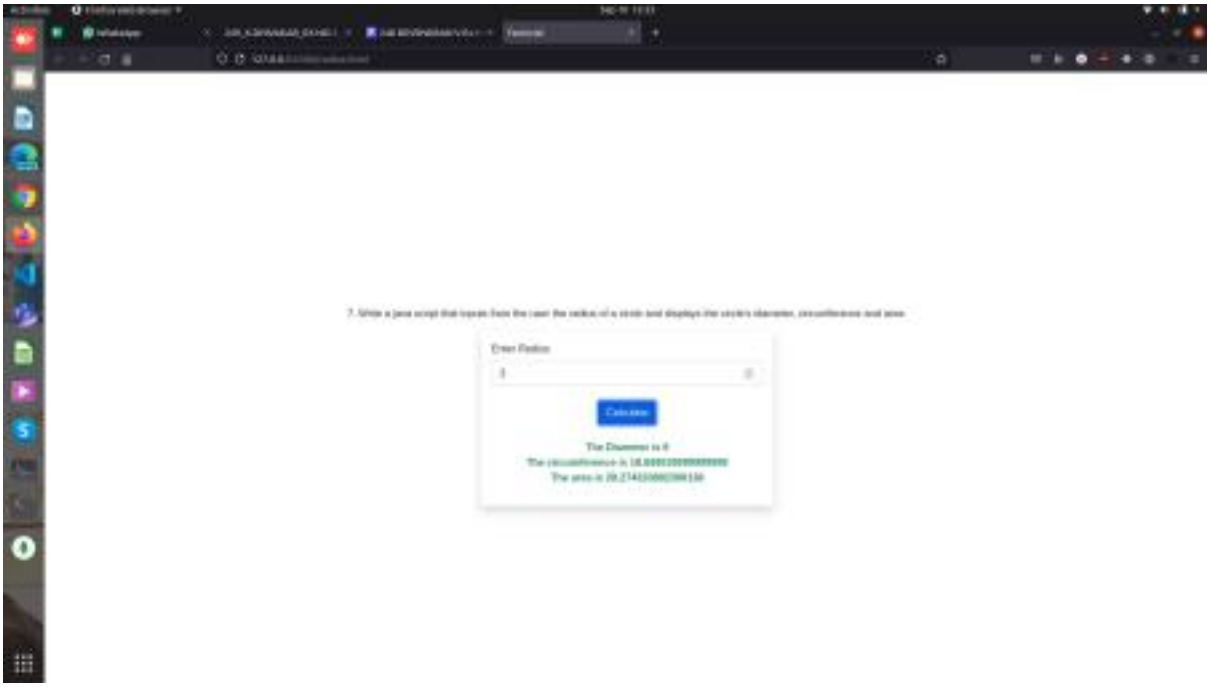
```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <!-- CSS only -->
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.m in.css" rel="stylesheet"

integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJ
ByhZMI3AhiU" crossorigin="anonymous">

    <title>Factorial</title>
</head>

<body>
```

```
    <script>


        function calculate() {
            const number =
            parseInt(document.getElementById('exampleInputPassword1').value); if(!number){
                return;
            }
            const diameter = 2 * number;
            const circumference = 2 * 3.14159 * number;
            const area = Math.PI * number * number;
            document.getElementById('result').innerText = `The Diameter is ${diameter}
            The circumference is ${circumference}
            The area is ${area}`


        }
    </script>
```

 7. Write a java script that inputs from the user the radius of a circle and displays the
circle’s diameter, circumference and area.

```
    <div class="d-flex justify-content-center">
        <div class="col-12 col-md-6 col-lg-3 shadow p-3">
            <label for="exampleInputPassword1" class="form-label">Enter Radius</label>
            <input type="number" class="form-control"
id="exampleInputPassword1">
            <div class="text-center mt-4">
            <button class="btn btn-primary" onclick="calculate()"> Calculate
                </button>
            </div>
            <p class="text-center mt-4 fw-bold text-success"
id="result"></p>
        </div>
    </div>
</body>

</html>
```

OUTPUT :



8. Write a script that calculates the sum of integers from 1 to 10 and displays the result.

PROGRAM :

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

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <!-- CSS only -->
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.m in.css" rel="stylesheet"

integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJ
ByhZMI3AhiU" crossorigin="anonymous">

    <title>Sum of integers</title>
</head>

<body>

    <script>
```

```
        function calculate() {
            const number =
    parseInt(document.getElementById('exampleInputPassword1').value); let result = 0;
            if (!number) {
                return;
            }
            for (let i = 1; i <= number; i++) {
                result = result + i;
            }

            document.getElementById('result').innerText = `The sum of number from 1 to ${number}
is ${result}`;

        }
    </script>

    <p class="fw-bold text-center text-danger" style="margin-top: 20%;"> 8. Write a script that calculates the
sum of integers from 1 to 10 and displays the result.
    </p>

    <div class="d-flex justify-content-center">
        <div class="col-12 col-md-6 col-lg-3 shadow p-3">
            <label for="exampleInputPassword1" class="form-label">Enter number</label>
            <input type="number" class="form-control"
id="exampleInputPassword1">
            <div class="text-center mt-4">
            <button class="btn btn-primary" onclick="calculate()"> calculate
                </button>
            </div>
            <p class="text-center mt-4 fw-bold text-success"
id="result"></p>
        </div>
    </div>

</body>

</html>
```

OUTPUT :

9.Write a script that calculates the squares and cubes of the numbers from 0 to 10 and display the resulting values in an HTML.

PROGRAM :

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

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <!-- CSS only -->
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.m in.css" rel="stylesheet"

integrity="sha384-F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJ
ByhZMI3AhiU" crossorigin="anonymous">

    <title>Table</title>
</head>

<style>
    table,
    tr,
```

```
    th,
    td {
        border: 1px solid black;
    }
</style>

<body>
    <script>
        function calculate() {

            document.write(` <div class="d-flex
justify-content-center">`)
            document.write(` <div class="col-12 col-md-6 col-lg-3 shadow p-3 text-center" style="display:
block ruby">`)

            document.write('<table>');
            document.write('<th> Number </th> <th> Square </th> <th> Cube </th>');
            for (var i = 1; i <= 10; i++)
                document.write(`<tr><td> ${i} </td> <td> ${Math.pow(i, 2)} </td>
<td>${Math.pow(i, 3)}</td></tr>`);
            document.write("</table>");
            document.write(`</div>`)
            document.write(`</div>`)
        }
    </script>
    <p class="fw-bold text-center text-danger" style="margin-top: 20%;"> 9.Write a script that calculates the
squares and cubes of the numbers from 0 to 10 and display the resulting
        values in an HTML.
    </p>
    <script>
        calculate();
    </script>
</body>

</html>
```

OUTPUT :

DEVENDRAN V
211419205038

RESULT :

This above all question was executed and written successfully.

DEVENDRAN V
211419205038

EX NO: 5 CLIENT SIDE SCRIPT FOR VALIDATING WEB FORM CONTROLS USING DHTML

DATE: 18.09.21

AIM:

To write a java script program to validating web form controls using DHTML **.**

ALGORITHM :

1. Start the process.
2. Design the web form which use firstname, lastname,address1,address2,city, state,zip,home phone,email,etc. And one button to submit the value .
3. Define the style information using style sheet relationship for whole page by <style rel="stylesheet" id="mainStyle" type="text/css">.....< /style>
4. Using the <script > tag to specify the type of script using in web page.
5. Define the function named as "formValidator ()" to get form object which has all the web form controls and check the given control is empty or not using notEmpty() function .
6. By using notEmpty() function ,to get the control as object and check its value is „0" then return empty, otherwise not.
7. Perform the process for checking the given value is numeric or not by using isNumeric(elem) function.
8. Validating all the form control input to verify the value as alphabet or alphanumeric by using isAlphaNumeric(elem)and isAlphabet(elem) function correspondingly.
9. Checking the entered email values in the email field is proper format or not by using emailValidator(elem).
10. Validate the choice field value by using madeSelection(elem) function to verify atleast one value is selected or not .
11. Password field is validated to restrict the length by using lengthRestriction(elem).
12. Call the formValidator()function in form submission on onsubmit event to validate all fields correspondingly.
13. Stop the process.

PROGRAM:

Registration.html:

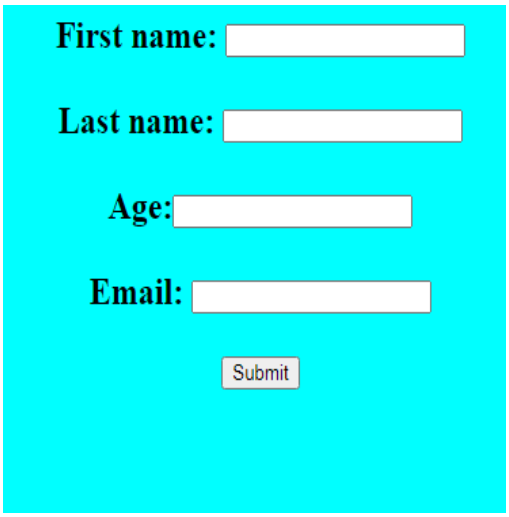
```
<html>

<head>

<script type=text/javascript>
function validateForm()

{

var x=document.forms["myForm"]["fname"].value;
var y=document.forms["myForm"]["lname"].value;
if (x=="")

  {
```

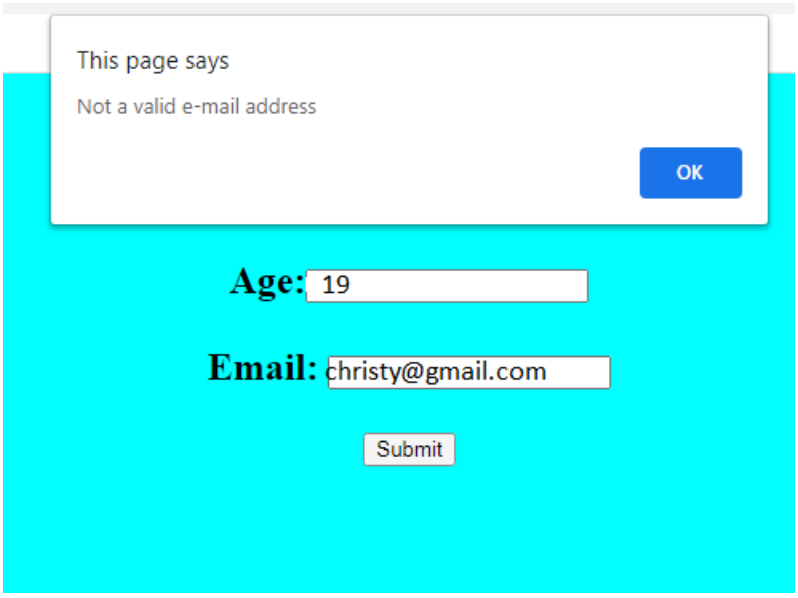
```
  alert("First name must be filled out");
  return false;
  }
if (y=="")
  {
  alert("Last name must be filled out");
  return false;
  }
if(document.forms["myForm"]["age"].value=="")
      {
          alert("Enter Some Age");
          document.myForm.age.focus() ;
          return false;
        }
if((document.myForm.age.value>50))
           {
           alert("Invalid Age");
           return false;
     }
var x=document.forms["myForm"]["email"].value;
var atpos=x.indexOf("@");
var dotpos=x.lastIndexOf(".");
if (atpos<1 || dotpos<atpos+2 || dotpos+2>=x.length)
  {
  alert("Not a valid e-mail address");
  return false;
}
 alert("LOGIN SUCESSFULLY");
}
</script>
</head>
<body BGCOLOR="aqua">
<form name="myForm"action="welcome.HTML"onsubmit="return validateForm();"
method="post">
```

DEVENDRAN V
211419205038

```
<h2><center>First name: <input type="text" name="fname"><br><br>
Last name: <input type="text" name="lname"><br><br>

Age:<input type=text name=age><br><br>

Email: <input type="text" name="email"><br><br>

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

</form>

</body>

 </html>
```

OUTPUT:





DEVENDRAN V
211419205038

RESULT:

The implementation of client side scripting for validating web form usin dhtml by using JavaScript inHTML was written, executed and verified successfully.

DEVENDRAN V
211419205038

EX.NO:6 **SERVLET IMPLEMENTATION THROUGH HTML.**
DATE:09.10.21

AIM:

To write a java program to invoke servlet from html by using web browser.

ALGORITHM:

1. Start the process.
2. Create the class htmlservlet which extends the base class HttpServlet for accessing the features ofHttpServlet.
3. Create the request and response parameter to handle the request and response of Httpservlet by usingservice() mthod.
4. To display the servlet information such as initialization parameter, size and type of the requested data,protocol and version of the request, name of the query parameter, remote user name, path of the servlet,etc. by using service() method.
5. Create the class AddCookieServlet which extends the base class HttpServlet for accessing the featuresof HttpServlet.
6. To get parameter from Http request then create and add cookie to Http response by using public voiddoPost(HttpServletRequest req,HttpServletResponse res) throws ServletException, IOException methodlike String data = req.getParameter(“data”); Cookie cookie = new Cookie(“MyCookie”,data); res.addCookie(cookie);
7. To set the Cookie maximum age by cookie.setMaxAge(3600).
8. To invoke the htmlservlet file from html by
9. Stop the process.

PROGRAM:
Cook1.java:

```
import java.io.*;
import
javax.servlet.*;
import
javax.servlet.annotation.WebServlet;
import javax.servlet.http.*;
@WebServlet(urlPatterns =
{"/cook1"}) public class cook1 extends
HttpServlet {
public void doPost(HttpServletRequest request, HttpServletResponse
response){try{ response.setContentType("text/html");
PrintWriter out = response.getWriter();
String
n=request.getParameter("userName");
out.print("Welcome "+n);
Cookie ck=new Cookie("uname",n);//creating cookie
objectresponse.addCookie(ck);//adding cookie in the
response ck.setMaxAge(3600);
//creating submit button
out.print("<form action=“cook2“

method=“post“>”);out.print(“<input

type=“submit“ value=“go“>”);
```

```
out.print("</form>");
out.close();
}catch(Exception e)
{ System.out.println(e);
}}}
```

Cook2.java:

```
import java.io.*;
import
javax.servlet.*;
import
javax.servlet.annotation.WebServlet;
import javax.servlet.http.*;
@WebServlet(urlPatterns = {"/cook2"})
public class cook2 extends HttpServlet {
public void doPost(HttpServletRequest request, HttpServletResponse
response){try{
response.setContentType("text/htm
l");PrintWriter out =
response.getWriter();Cookie
ck[]=request.getCookies();
out.print("Hello
"+ck[0].getValue()); out.close(); }
catch(Exception e){System.out.println(e);} }}
```

OUTPUT:

Name:
go

Name: monisha
go

Welcome
go

Hello

RESULT:

The implementation servlet through html by using JavaScript in HTML was written, executed andverified successfully.

DEVENDRAN V
211419205038

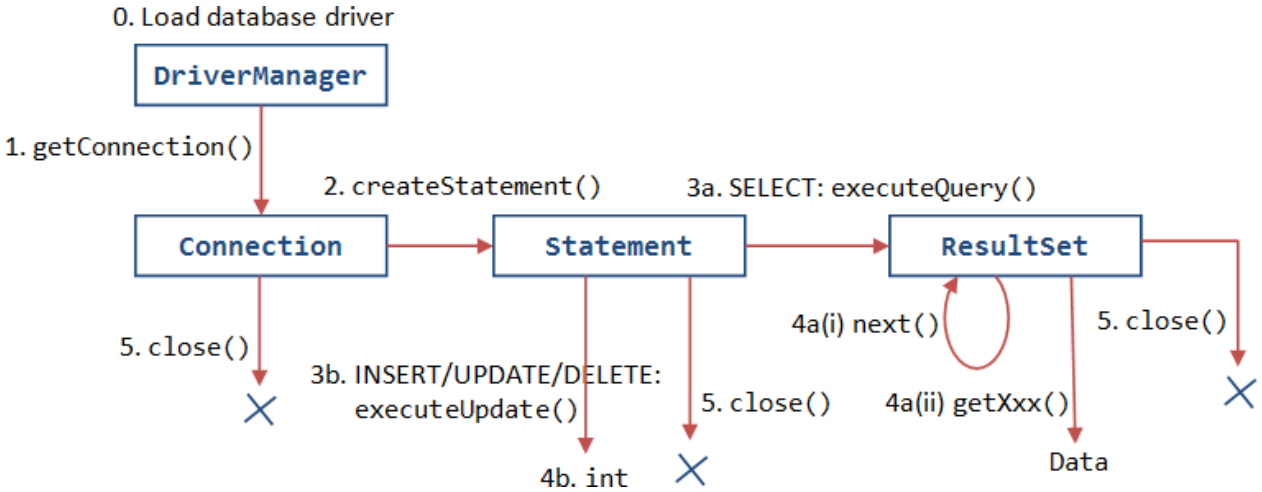
EX.NO:07 **SERVLET DATABASE**

DATE: 16.10.21

AIM:

To write a java program to invoke servlet database from html web browser.

JDBC Cycle :
Database Connectivity :



1. Allocate a database 'Connection' object
// The format is: "jdbc:mysql://hostname:port/databaseName", "username", "password"

Example,

Connection conn = DriverManager.getConnection(
jdbc:mysql://localhost:3306/ebookshop?allowPublicKeyRetrieval=true&useSSL=false&serverTimezone=UTC", "myuser", "xxxx"); // For MySQL

2. Allocate a 'Statement' object in the Connection
Statement stmt = conn.createStatement();
String sqlStr = "select * from books where author = "
+ "'" + request.getParameter("author") + "'" // Single-quote SQL string
+ " and qty > 0 order by price desc";
out.println("<h3>Thank you for your query.</h3>");
out.println(" Your SQL statement is: " + sqlStr + " "); // Echo for debugging
ResultSet rset = stmt.executeQuery(sqlStr); // Send the query to the server

3. Process the query result set
int count = 0;
while(rset.next()) {

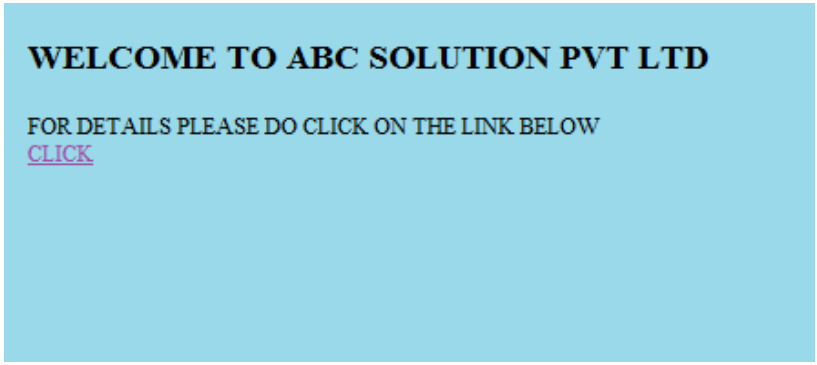
```
    // Print a paragraph <p>...</p> for each record
    out.println("<p>" + rset.getString("author")
        + ", " + rset.getString("title")
        + ", $" + rset.getDouble("price") + "</p>");
    count++;
  }
```

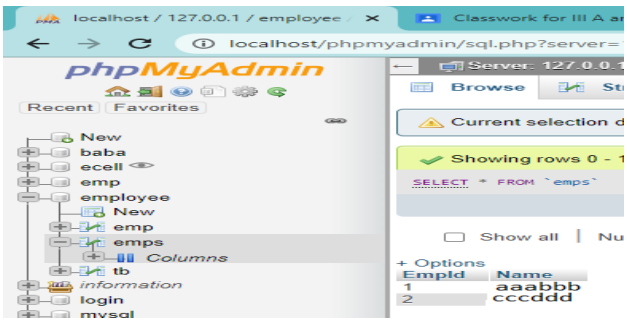
PROGRAM:

```
import java.sql.*;
import java.io.*;
import
javax.servlet.*;
importjavax.servlet.annotation.WebServ
let;importjavax.servlet.http.*;
@WebServlet(urlPatterns = {"/en"})
public class en extends HttpServlet {
public void doGet (HttpServletRequest request,HttpServletResponse
response)throws ServletException, IOException
 {
 // Set the HTTP content type in response header
response.setContentType("text/html; charset=\"UTF-8\"");
// Obtain a PrintWriter object for creating the body of the response
PrintWriter out =
 response.getWriter();try {
Class.forName("com.mysql.jdbc.Driver").newInstance
(); Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/tests?zeroDateTimeBehavior=conve
rtToNull","root","");
 // JDBC calls to con methods go
here ...Statement stmt =
con.createStatement();
 stmt.execute("INSERT INTO test1 VALUES " + "('1',
'Clary')");stmt.execute("INSERT INTO test1 VALUES " +
"('2', 'Joe')"); System.out.println("Row count is " +
stmt.getUpdateCount()); ResultSet rs =
stmt.executeQuery("select * from test1");
out.println("<table boder=1 width=50% height=50%>”);
```

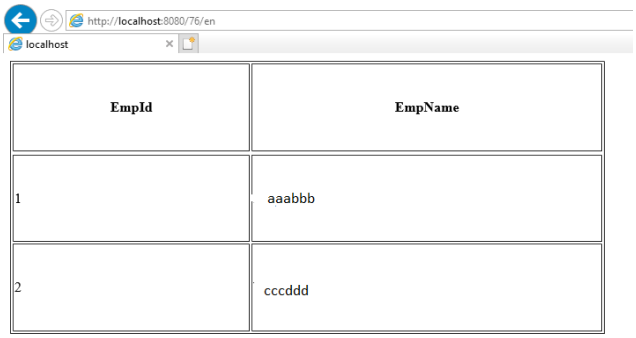
```
out.println("EmpIdEmpName")

;while (rs.next()) {

String nm =
 rs.getString("empname");String n
 = rs.getString("empid");
 out.println("<tr><td>" + n + "</td><td>" + nm + "</td></tr>");
 }
out.println("</table>");
 out.println("</html></body>”);
}
catch (Exception e)
 { out.print(e);
 }
 finally
{
out.close();
 } } }
```

OUTPUT:





DEVENDRAN V
211419205038



EmpId	EmpName
1	aaabbb
2	cccddd

RESULT:

To write a java program to invoke servlet database from html web browser was executed andsuccessfully executed

DEVENDRAN V
211419205038

EXP NO:08 COUNT THE NUMBER OF ELEMENT LINK USING DOM

DATE:23.10.21

AIM:

To implement the process for count the number of links in an given XML file by using DOM

.

ALGORITHM :

1. Start the Process.
2. Import all the necessary JAXP Classes for count the number of links in an given XML file.
3. Import all the necessary DOM Classes for count the number of links in an given XML file.
4. Import all the necessary JDK Classes for count the number of links in an given XML file.
5. JAXP-style initialization of DocumentBuilder for XML parser that builds DOM from document.
6. Parse an XML document from file given by first argument into a DOM Document object.
7. Process the Document object using the Java API version of the W3C DOM
8. Stop the process.

PROGRAM :

```
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.parsers.DocumentBuilder;

                                    // DOM classes

import org.w3c.dom.Document;
import org.w3c.dom.NodeList;
                                    // JDK classes

import java.io.File;
```

```
/** Count the number of link elements in an RSS 0.91 document */
class DOMCountLinks
{
    static public void main(String args[])
    {
        try
        {
    // JAXP-style initialization of DocumentBuilder  (XML parser that builds DOM from document)
    DocumentBuilderFactory docBuilderFactory = DocumentBuilderFactory.newInstance();
        DocumentBuilder parser = docBuilderFactory.newDocumentBuilder();

    // Parse XML document from file given by first argument     into a DOM Document object
        Document document = parser.parse(new File(args[0]));

      // Process the Document object using the Java API version of     // the W3C DOM
    NodeList dept = document.getElementsByTagName("bname");
    System.out.println("Input document has " + dept.getLength() +" 'link' elements.");
        }
        catch (Exception e)
        {
        e.printStackTrace();
        }
        return;
    }
}
```

collxml.cml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<?xml-stylesheet type="text/xsl" href="cdxsl.xsl"?>
<bank>
    <branch>
        <baname>State Bank Of India </baname>
        <bname>Egmore</bname>
        <manager>Mr.M.Venkat</manager>
```

```
        <ifsccode>SBIN0001516</ifsccode>
        <bcode>001516</bcode>
        <link>www.sbi.com</link>
</branch>
<branch>
        <baname>State Bank Of India </baname>
        <bname>Guindy</bname>
        <manager>Mr.J.Jeffrey</manager>
        <ifsccode>SBIN0001515</ifsccode>
        <bcode>001515</bcode>
        <link>www.sbi.com</link>
</branch>
<branch>
        <baname>State Bank Of India </baname>
        <bname>Annasalai</bname>
        <manager>Mr.A.Thomson</manager>
        <ifsccode>SBIN0001514</ifsccode>
        <bcode>001514</bcode>
        <link>www.sbi.com</link>
</branch>
<branch>
        <baname>State Bank Of India </baname>
        <bname>Saidapet</bname>
        <manager>Mr.N.Vijay</manager>
        <ifsccode>SBIN0001513</ifsccode>
        <bcode>001513</bcode>
        <link>www.sbi.com</link>
</branch>
```

```
<branch>
        <baname>State Bank Of India </baname>
        <bname>Porur</bname>
        <manager>Mr.R.Kumaran</manager>
        <ifsccode>SBIN0001512</ifsccode>
        <bcode>001512</bcode>
        <link>www.sbi.com</link>
</branch>       </college>
```

To run:

```
C:\jdk1.4.1\bin>javac DOMCountLinks.java
C:\jdk1.4.1\bin>java DOMCountLinks.java  collxml.xml
```

OUTPUT :

Input document has 5 'link' elements.

RESULT :

The implementations of the process for count the number of links in an given XML file by using DOM was successfully written , executed and verified

DEVENDRAN V
211419205038

EXP NO:09 PROGRAMS USING XML – SCHEMA – XSLT/XSL

DATE:23.10.21

AIM:

To generate simple xhtml document using XSL and XML by XSL Transformations.

ALGORITHM:

1. Start the Process.
2. Import all the necessary JAXP Classes to generate XHTML for the given XML and XSL file.
3. Import all the necessary JDK Classes to generate XHTML for the given XML and XSL file.
4. Initialize the TransformerFactory object by TransformerFactory tFactory =

   TransformerFactory.newInstance();

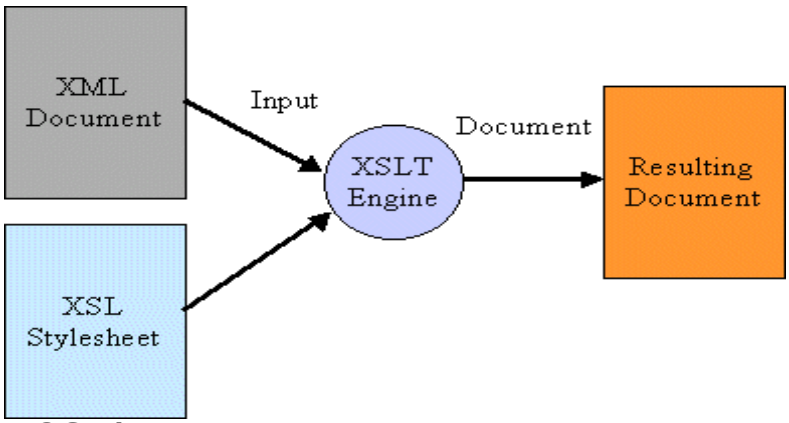
5. Initialize the Transformer object by the following to get stream source as xsl file.
   Transformer transformer = tFactory.newTransformer(new StreamSource(new File(args[0])));

6. By using Transformer object named as "transformer" to transform the XSL and XML in to XHTML by the following.
transformer.transform(new StreamSource(new File(args[1])), new StreamResult(System.out));

7. Apply the XSL transform contained in the file named by the first command-line argument to the XML document named by the second argument and write the resulting document to standard output.
8. Stop the Process.

XSLT :

XSL Transformations is a language for describing how to transform an XML document (explicitly or implicitly represented as a tree) into another.



PROGRAM :

```
import javax.xml.transform.TransformerFactory;
import javax.xml.transform.Transformer;

import javax.xml.transform.stream.StreamSource;
import javax.xml.transform.stream.StreamResult;
```

```
// JDK classes
import java.io.File;


/** Apply the XSL transform contained in the file named by
the first command-line argument to the XML document
named by the second argument and write the resulting
document to standard output. */

class XSLTransform {

public static void main(String args[]) {
```

cdxsl.xsl

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<!-- Edited by XMLSpy® -->

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

<xsl:template match="/">

<html>

<body>

<center>

<h2><center><u>PLACEMENT DETAILS</u></center></h2>

<table border="1">

<tr bgcolor="#9acd32">

<th>Department Name</th>

<th>Student Name</th>

<th>Company Name</th>

<th>CGPA</th>

</tr>

<xsl:for-each select="college/dept">

<xsl:if test="cgpa>9.1">

<!--<xsl:sort select="Title" /> -->

<tr>

</tr>

</xsl:if>

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

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

```
<td><xsl:value-of select="company"/></td>
<td><xsl:value-of select="cgpa"/></td>
</xsl:for-each>
</table>
</center>
</body>
</html>
</xsl:template>
</xsl:stylesheet>
```

cdxml.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<?xml-stylesheet type="text/xsl" href="cdxsl.xsl"?>
<college>
<dept>
<dname>Information Technology </dname>
<studentName>Rajesh</studentName>
<registerNo>211402205091</registerNo>
<cgpa>8.9</cgpa>
<company>Infosys</company>
<year>2011</year>
</dept>
<dept>
<dname>Information Technology </dname>
<studentName>Makesh</studentName>
<registerNo>211402205081</registerNo>
<cgpa>9.2</cgpa>
<company>CTS</company>
<year>2011</year>
</dept>
<dept>
<dname>Information Technology </dname>
```

<studentName>Ganesh Kumar</studentName>

<registerNo>211402205051</registerNo>

<cgpa>9.3</cgpa>

<company>IBM</company>

<year>2011</year>

</dept>

<dept>

<dname>Information Technology </dname>

<studentName>Sankaran</studentName>

<registerNo>211402205056</registerNo>

<cgpa>9.1</cgpa>

<company>MindTree</company>

<year>2011</year>

</dept>

<dept>

<dname>Information Technology </dname>

<studentName>Shiva</studentName>

<registerNo>211402205034</registerNo>

<cgpa>8.5</cgpa>

<company>TCS</company>

<year>2011</year>

</dept>

<dept>

<dname>Information Technology </dname>

<studentName>Sakthi</studentName>

<registerNo>211402205011</registerNo>

<cgpa>8.2</cgpa>

<company>Infosys</company>

<year>2011</year>

</dept>

<dept>

<dname>Information Technology </dname>

<studentName>Karthik</studentName>

<registerNo>211402205023</registerNo>

<cgpa>8.1</cgpa>

<company>IBM</company>

<year>2011</year>

</dept>

</college>

TO RUN :

E:\ sri>javac XSLTransform.java

E:\sri>java XSLTransform cdxsl.xsl cdxml.xm

OUTPUT :

DEVENDRAN V
211419205038

PLACEMENT DETAILS

Department Name	Student Name	Company Name	CGPA
CSE	Makesh	Infosys	9.12
IT	Ganesh Kumar	TCS	9.52

RESULT :

The Generation Of Simple Xhtml Document By XSLtransformation Using Xsl And Xml was written , executed and verified .

DEVENDRAN V
211419205038

EXPNO:10 PROGRAM USING SAX

DATE: 30.10.21

AIM :

To write a simple SAX program to count the number of "item" elements in the given XML file.

ALGORITHM:

1. Start the Process.
2. Import all the necessary JAXP Classes to count the number of item elements in the given XML file
3. Import all the necessary SAX Classes to count the number of item elements in the given XML file.
4. Initialize the URL to feed the xml file into SAX program.
5. Initialize XMLReader and set up event handlers for count the number of item elements in xml file.
6. JAXP-style initialization of SAX parser is initialized by
   SAXParserFactory saxFactory = SAXParserFactory.newInstance();
   XMLReader parser = saxFactory.newSAXParser().getXMLReader();
7. SAX-style processing of RSS document given at FEED_URL by
   parser.setContentHandler(new CountElementsHelper());
   parser.parse(FEED_URL);
8. Helper class containing SAX event handler methods for Process the start of an element and end of an element.
9. Stop the Process.

PROGRAM :

```
//Count the Number of Item elements in given xml file.

// JAXP classes
import javax.xml.parsers.SAXParserFactory;

import javax.xml.parsers.SAXParser;

// SAX classes

import org.xml.sax.XMLReader;
import org.xml.sax.Attributes;
import org.xml.sax.SAXException;

import org.xml.sax.helpers.DefaultHandler;

/** Count the number of item elements in an XML document */
class SAXCountLinks

{

/** Source for RSS feed */
static String FEED_URL = "E:/sri/ExampleContentFeed.xml";
```

```
                    /** Initialize XMLReader and set up event handlers */
static public void main(String args[])

 {

        try

        {               // JAXP-style initialization of SAX parser
            SAXParserFactory saxFactory = SAXParserFactory.newInstance();
```

```
            XMLReader parser = saxFactory.newSAXParser().getXMLReader();
                        // SAX-style processing of RSS document at FEED_URL
        parser.setContentHandler(new CountElementsHelper());
        parser.parse(FEED_URL);
        }
        catch (Exception e)
        {
                e.printStackTrace();
        }
        return;
 }                  /** Helper class containing SAX event handler methods */

private static class CountElementsHelper extends DefaultHandler
{
                                /** Number of 'p' elements seen so far */
int numElements;
                        /** Constructor (allows for superclass initialization) */
CountElementsHelper()
  {
        super();
  }
                                /** Perform initialization for this instance */
public void startDocument() throws SAXException
  {
        numElements = 0;
        return;
  }
                                /** Process the start of an element */
public void startElement(String namespaceURI, String localName,String qName, Attributes atts)
throws SAXException
  {
        if (qName.equals("item"))
            {
                                                                        }
```

```
return;                    numElements++;
    }

                    /** Done with document; output final count */
public void endDocument() throws SAXException
  {
        System.out.println("Input document has " + numElements + " 'item' elements.");
return;}
  }
}
```

ExampleContentFeed.xml

```
<?xml version="1.0"?>
<rss version="0.91">
<channel>
        <title>www.example.com</title>
        <link>http://www.example.com/</link>
        <description>   www.example.com is not a site that changes often...      </description>
        <language>en-us</language>
        <item>
        <title>Announcing a Sibling Site!</title>
        <link>http://www.example.org/</link>
        <description>
                        Were you aware that example.com is not the only site in the example
family?
        </description>
        </item>
        <item>
                <title>We're Up!</title>
                <link>http://www.example.net/</link>
                <description>
                        Our new RSS feed is up. Visit us today!
                </description>
        </item>
</channel>
</rss>
```

DEVENDRAN V
211419205038

TO RUN :

E:\sri>javac SAXCountLinks.java
E:\sri>java SAXCountLinks

OUTPUT:

Input document has 2 'item' elements.

RESULT:

The simple SAX program to count the number of “item” elements in the given XML file was written, executed and verified**.**

DEVENDRAN V
211419205038

EXP NO:11 PROGRAMS USING AJAX

DATE:06.10.21

AIM :

To write a simple AJAX program to implement the counter servlet by using Java Servlet and Javascript with XML.

ALGORITHM:

1. Start the process.
2. Create the class countservlet which extends the base class HttpServlet for accessing the features of HttpServlet.
3. Create the request and response parameter to handle the request and response of Httpservlet by using doGet method.
4. Initialize the visit variable to increment for every time it is loaded.
5. Initialize the content type and PrintWriter object by
   res.setContentType("text/html");
   PrintWriter srout=res.getWriter();
6. Call the javascript code named as "countupdate.js" to perform count operation based on the number of visits by using same servlet.
7. Create the request and response parameter to handle the request and response of Httpservlet by using doPost method to display the result in XML output.
8. Stop the Process**.**

PROGRAM :

countservlet.java

```
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;

public class countservlet extends HttpServlet

{

        private static int visits=0;

        public void doGet(HttpServletRequest req,HttpServletResponse res) throws
ServletException,IOException

        {

                visits++;
                res.setContentType("text/html");
                PrintWriter srout=res.getWriter();

                srout.println("<html><head><title> Example program for Ajax</title><script
type='text/javascript' src='/countupdate.js'></script>"+

                "</head><body onload='init()'><p>Welcome to the world :\n Number of
visits:</p><p id='visits'>0 </p></body></html>");

                srout.close();

        }
```

```
    public void doPost(HttpServletRequest req,HttpServletResponse res) throws
ServletException,IOException
    {

        res.setContentType("application/xml");
        PrintWriter srout=res.getWriter();
        srout.println("<?xml version='1.0' ?><count>"+visits+"</count>");
        srout.close();

    }
}
```

countupdate.js

```
function init()
{
    window.setInterval("getvisits()",300);
    return;
}
function getvisits()
{
    var con=new XMLHttpRequest();
    if(con)
    {
        con.open("POST","/countservlet",true);
        con.onreadystatechange=function update(){updatevisits(con);}
        con.send("");
    }
    return;
}

function updatevisits(con)
{
    if(con.readyState==4 && con.status==200)
    {
        visits=document.getElementById("visits");
        var count=con.responseXML.documentElement;
        visits.childNodes[0].data=count.childNodes[0].d
```

DEVENDRAN V
211419205038

TO RUN :

1. javac countservlet.java
2. copy countservlet.class file into E:\jwsdp-1.3\webapps\ROOT\WEB-INF\classes directory
3. edit web.xml by the following

```
<servlet>
<servlet-name>Countservlet</servlet-name>
<servlet-class>countservlet</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>Countservlet</servlet-name>
<url-pattern>/countservlet</url-pattern>
</servlet-mapping>
```

RESULT :

Thus the implementation of the counter servlet using AJAX was written and executed successfully

DEVENDRAN V
211419205038

EX NO: 3 **WEB PAGE DESIGN USING INLINE STYLES, EMBEDDED STYLES AND CASCADING STYLES SHEETS**

DATE: 04.09.21

AIM:

To design a web site for implement the concepts of Inline styles, Embedded styles

and Cascading Styles Sheets using CSS.

ALGORITHM:

1. Start the process.

2. To represent the styles of elements individually for handling the students details in web page by using Inline styles sheet like, <p style=”font-family:Times;color:yellow;font-size:10pt>Student Namely;/p>

3. Declare and use Embedded styles sheets to apply the styles to whole document Elements for handling administration details of college web page in head section <style type=”text/css”> h1 { font-size : 20pt;color:yellow}</style>

4. Create the deptstyles.css file to declare the styles to present the element value for Handling the department details in college website and it is externally load into the website by <link rel=”stylesheet” type=”text/css” href=”deptstyles.css”>

5. Create a web page to handle placement details such as student name, company name, deptname, etc by using embedded style sheet.

6. Create a web page to handle academic details such as circular, exam time table, Admission details, etc by using cascading style sheet.

3. a. Create a web page that uses inline and external style sheet which has Descendant selector string and Universal selector String .

PROGRAM:

<html>

<head>

<title>Taj mahal</title>

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

</head>

<body background="back1.jpg" >

<h1 style="color:white;font-size:38;">TAJ MAHAL</h1>

<img src="Taj-Mahal.jpg" width="400" height="300">

The Taj Mahal Crown of the Palace', is an ivory-white marble mausoleum on the southern bank of the river Yamuna in the Indian city of Agra. It was commissioned in 1632 by the Mughal emperor Shah Jahan (reigned from 1628 to 1658) to house the tomb of his favourite wife, Mumtaz Mahal; it also houses the tomb of Shah Jahan himself. The tomb is the centrepiece of a 17-hectare (42-acre) complex, which includes a mosque and a guest house, and is set in formal gardens bounded on three sides by a crenellated wall.

The Taj Mahal incorporates and expands on design traditions of Indo-Islamic and earlier Mughal architecture. Specific inspiration came from successful Timurid and Mughal buildings including the Gur-e Amir (the tomb of Timur, progenitor of the Mughal dynasty, in Samarkand),[14] Humayun's Tomb which inspired the Charbagh gardens and hasht-behesht (architecture) plan of the site, Itmad-Ud-Daulah's Tomb (sometimes called the Baby Taj), and Shah Jahan's own Jama Masjid in Delhi. While earlier Mughal buildings were primarily constructed of red sandstone, Shah Jahan promoted the use of white marble inlaid with semi-precious stones. Buildings under his patronage reached new levels of refinement </br>

The Taj Mahal is built on a parcel of land to the south of the walled city of Agra. Shah Jahan presented Maharajah Jai Singh with a large palace in the centre of Agra in exchange for the land.[40] An area of roughly 1.2 hectares (3 acres) was excavated, filled with dirt to reduce seepage, and levelled at 50 metres (160 ft) above riverbank. In the tomb area, wells were dug and filled with stone and rubble to form the footings of the tomb. Instead of lashed bamboo, workmen constructed a colossal brick scaffold that mirrored the tomb. The scaffold was so enormous that foremen estimated it would take years to dismantle.

<h1 style="color:white;font-size:38;" > <u>CONSTRUCTION</u> </h1>

<P style="color:white;font-size:20;">The Taj Mahal was constructed using materials from all over India and Asia. It is believed over 1,000 elephants were used to transport building materials. It took the efforts of 22,000 labourers, painters, embroidery artists and stonecutters to shape the Taj Mahal.[42] The translucent white marble was brought from Makrana, Rajasthan, the jasper from Punjab, jade and crystal from China. The turquoise was from Tibet and the Lapis lazuli from Afghanistan, while the sapphire came from Sri Lanka and the carnelian from Arabia. In all, twenty-eight types of precious and semi-precious stones were inlaid into the white marble.[citation needed]

According to the legend, Shah Jahan decreed that anyone could keep the bricks taken from the scaffold, and thus it was dismantled by peasants overnight.[43] A 15-kilometre (9.3 mi) tamped-earth ramp was built to transport marble and materials to the construction site and teams of twenty or thirty oxen pulled the blocks on specially constructed wagons.[44] An elaborate post-and-beam pulley system was used to raise the blocks into desired position. Water was drawn from the river by a series of purs, an animal-powered rope and bucket mechanism, into a large storage tank and raised to a large distribution tank. It was passed into three subsidiary tanks, from which it was piped to the complex.</P>

<ul style="color:white;font-size:26;" >
one of the wonder

tourist spot

symbol of love

<h1><u style="color:gold;font-size:38;"> TOURISM </u></h1>

The Taj Mahal attracts a large number of tourists. UNESCO documented more than 2 million visitors in 2001,[60] which had increased to about 7–8 million in 2014.[3] A two-tier pricing system is in place, with a significantly lower entrance fee for Indian citizens and a more expensive one for foreigners. In 2018, the fee for Indian citizens was 50 INR, for foreign tourists 1,100 INR.[61] Most tourists visit in the cooler months of October, November and February. Polluting traffic is not allowed near the complex and tourists must either walk from parking areas or catch an electric bus. The Khawasspuras (northern courtyards) are currently being restored for use as a new visitor centre.[62][63] In 2019, in order to address overtourism, the site instituted fines for visitors who stayed longer than three hours

<ul style="color:white;font-size:18;">
entry:50

camera not allowed

keep the surrounding clean

 wear mask,head glows and legs schocks

<img src="back.jpg" width="660" height="300">

<h1><u style="color:white;font-size:38;"> Controversies </u></h1>

As of 2017, several court cases about Taj Mahal being a Hindu temple have been inspired by P. N. Oak's theory.[82][83] In August 2017, Archaeological Survey of India (ASI) stated there was no evidence to suggest the monument ever housed a temple.[84] Bharatiya Janata Party's Vinay Katiyar in 2017 claimed that the 17th century monument was built by Mughal emperor Shah Jahan after destroying a Hindu temple called "Tejo Mahalaya" and it housed a Shiva linga. This claim had also been made by another BJP member Laxmikant Bajpai in 2014. The BJP government's union minister of culture Mahesh Sharma stated in November 2015 during a session of the parliament, that there was no evidence that it was a temple. The theories about Taj Mahal being a Shiva temple started circulating when Oak released his 1989 book "Taj Mahal: The True Story". He claimed it was built in 1155 AD and not in the 17th century, as stated by the ASI.

<img src="taj.jpg"width="500" height="300">

</body>

</html>

OUTPUT:

3. b Create a web page that uses Embedded and external style sheet which has Class selector string , Pseudo selector String and ID Selector String.

PROGRAM:

```
<html>
<head>
<body bgcolor="rosybrown">
<title>RED FORT</title>
<link rel="stylesheet" type="text/css" href="wild.css">
<style type="text/css">
#first_paragraph{
color:navy;
font-size:22;
}
a:link{
font-size:35;
}
a:visited{
color:darkred;
font-weight:25;
}
a:active{
color:orange;
font-weight:25;
}
.elephant
{
color:maroon;
```

```
background-color:papayawhip;

}

h1{

color:indigo;

}

</style>

</head>

<body background="nature.jpg" text="black">

<center><h1 style="color:red;">RED FORT</h1></center>

<center><img src="RED.jpg" width="450" height="300"></center>

<p style="color:navy;font-size:25;"> The Red Fort is a historic fort in Old Delhi, Delhi in India that served as the main residence of the Mughal Emperors. Emperor Shah Jahan commissioned construction of the Red Fort on 12 May 1638, when he decided to shift his capital from Agra to Delhi. Originally red and white, its design is credited to architect Ustad Ahmad Lahori, who also constructed the Taj Mahal. The fort represents the peak in Mughal architecture under Shah Jahan, and combines Persianate palace architecture with Indian traditions.

</p>

<center><h1  style="color:red;"> HISTROY </h1></center>

<ol style="font-size:20;"><P>Emperor Shah Jahan commissioned construction of the Red Fort on 12 May 1638, when he decided to shift his capital from Agra to Delhi. Originally red and white, Shah Jahan's favourite colours,[7] its design is credited to architect Ustad Ahmad Lahori, who also constructed the Taj Mahal.[8][9] The fort lies along the Yamuna River, which fed the moats surrounding most of the walls.[10] Construction began in the sacred Islamic month of Muharram, on 13 May 1638.[11]:?01? Supervised by Shah Jahan, it was completed on 6 April 1648.[12][13][14] Unlike other Mughal forts, the Red Fort's boundary walls are asymmetrical to contain the older Salimgarh Fort.[11]:?04? The fortress-palace was a focal point of the city of Shahjahanabad, which is present-day Old Delhi. Shah Jahan's successor, Aurangzeb, added the Moti Masjid (Pearl Mosque) to the emperor's private quarters, constructing barbicans in front of the two main gates to make the entrance to the palace more circuitous.[11]:?08?</P><h1><b><u> Architecture </u></b></h1>

<h2 class="READFORT ARCH">

<P></P><center><img src="ARCH.jpg" width="450" height="300">

<p id="first_paragraph"> to learn more
```

<a href=https://en.wikipedia.org/wiki/Red_Fort"

id="REDFORT ARCH" target="_self">

REDFORT ARCH

</br>

<h1><u> Major structures: </u> </h1>

<h2 class="RED FORT">

<P>Delhi Gate

Main article: Delhi Gate (Red Fort)

The Delhi Gate is the southern public entrance and is similar in layout and appearance to the Lahori Gate. Two life-size stone elephants on either side of the gate face each other</P>

<img src=" MA.jpg" width="450" height="300">

to learn more

<a href="https://en.wikipedia.org/wiki/Red_Fort"

id="deer" target="_self">

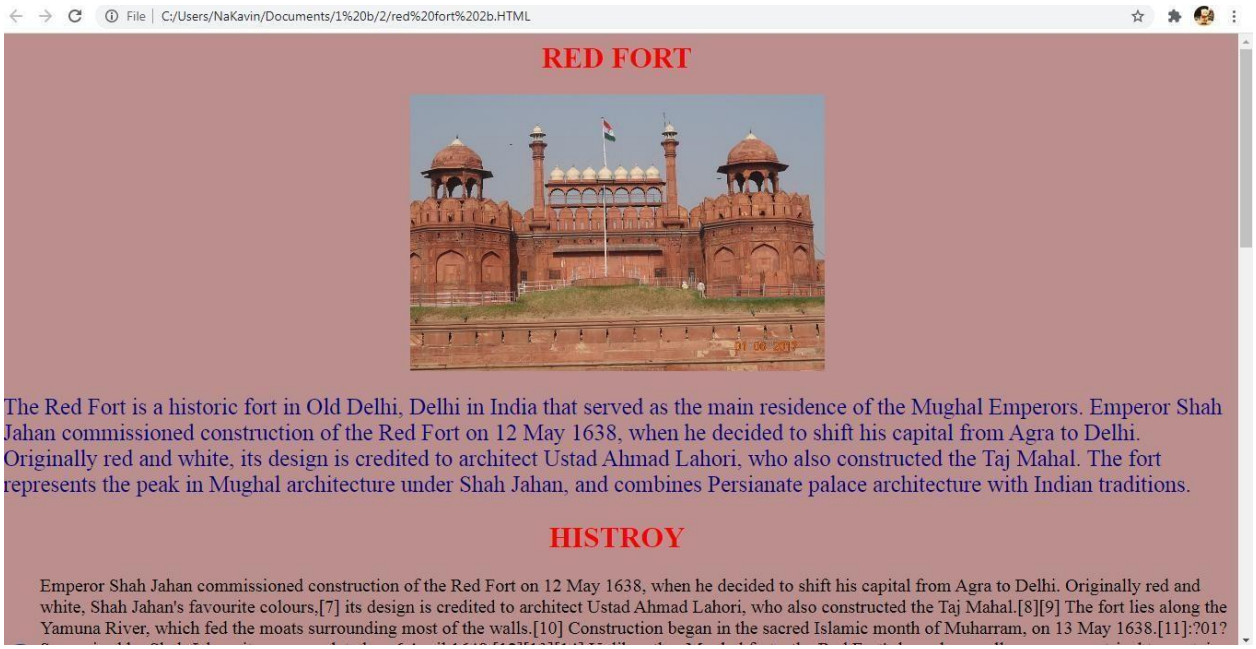
REDFORT

</br>

</body>

</html>

OUTPUT:



3. C Create a web page with all types of Cascading style sheets using any three selector String.

PROGRAM:

```
<html>

<head>

<title>Brihadeeswara Temple</title>

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

<style type="text/css">

#first_paragraph{

color:white;

font-size:35;

}

a:link{

font-size:25;

}
```

```
a:visited{
color:darkred;
font-weight:25;
}
a:active{
color:white;
font-weight:25;
}
h1,h2,h3{color:white;font-family:times new roman;text-decoration:blink;}
h3{
font-size:33;
}
</style>
</head>
<body background="than.jpg" text="black">
<center><h1 style="white;font-size:40;">PERUVUDAIYĀR KŌVIL</h1></center>
<center><img src="1.jpg" width="450" height="300"></center>
<p style="color:white;font-size:25;">
Peruvudaiyār Kōvil (Original Name) locally known as Thanjai Periya Kovil, also called as Brihadishvara temple, Rajarajeswaram, is a Hindu temple dedicated to Shiva located in South bank of Cauvery river in Thanjavur, Tamil Nadu, India. It is one of the largest South Indian temples and an exemplary example of a fully realized Tamil architecture. It is called as Dakshina Meru (Meru of south). Built by Tamil king Raja Raja Chola I between 1003 and 1010 AD, the temple is a part of the UNESCO World Heritage Site known as the "Great Living Chola Temples", along with the Chola dynasty era Gangaikonda Cholapuram temple and Airavatesvara temple that are about 70 kilometres (43 mi) and 40 kilometres (25 mi) to its northeast respectively. </p>
<center><h2 style="font-size:30;"> NOMENCLATURE  </h2></center>
<ol style="font-size:22;">
<p> Rajaraja Chola, who commissioned the temple, called it Rajarajeshwaram (Rajarājeśwaram), literally "the temple of the almighty of Rajaraja". A later inscription in the Brihannayaki shrine calls the temple's
```

deity Periya Udaiya Nayanar, which appears to be the source of the modern names Brihadisvara and Peruvudaiyar Kovil.

<h3><u> History </u></h3>

Statue of Rajaraja Chola I who built the temple over 1003-1010 CE.
A spectrum of Hindu temple styles continued to develop from the 5th to the 9th century over the Chalukya era rule as evidenced in Aihole, Badami and Pattadakal, and then with the Pallava era as witnessed at Mamallapuram and other monuments. Thereafter, between 850 and 1280 CE, Cholas emerged as the dominant dynasty. The early Chola period saw a greater emphasis on securing their geopolitical boundaries and less emphasis on architecture. In the 10th century, within the Chola empire emerged features such as the multifaceted columns with projecting square capitals. This, states George Michell, signaled the start of the new Chola style. This South Indian style is most fully realized both in scale and detail in the Brihadeshwara temple built between 1003 and 1010 by the Chola king Rajaraja I.[

<ol style="font-size:22;">

<img src="2.jpg" width="450" height="300">

 to learn more

<a href="https://en.wikipedia.org/wiki/Brihadisvara_Temple,_Thanjavur"

id="painted stork" target="_self">

big temple

</br>

<h3><u> ARCHITECTURE</u></h3>

The temple complex integrates a large pillared and covered veranda (prakara) in its spacious courtyard, with a perimeter of about 450 metres (1,480 ft) for circumambulation. Outside this pillared veranda there are two walls of enclosure, the outer one being defensive and added in 1777 CE by the French colonial forces with gun-holes with the temple serving as an arsenal. They made the outer wall high, isolating the temple complex area. On its east end is the original main gopuram or gateway that is barrel vaulted. It is less than half the size of the main temple's vimana. Additional structures were added to the original temple after the 11th century, such as a mandapa in its northeast corner and additional gopurams (gateways) on its perimeters to allow people to enter and leave from multiple locations.

Some of the shrines and structures were added during the Pandya, Nayaka, Vijayanagara and Maratha era, before the colonial era started, and these builders respected the original plans and symmetry rules. Inside

the original temple courtyard, along with the main sanctum and Nandi-mandapam are two major shrines, one for Kartikeya and for Parvati. The complex has additional smaller shrines.

<ol style="font-size:22;">

<img src="3.jpg" width="450" height="400">

 to learn more

<a href="https://en.wikipedia.org/wiki/Brihadisvara_Temple,_Thanjavur"

id="lesser whistling duck" target="_self">

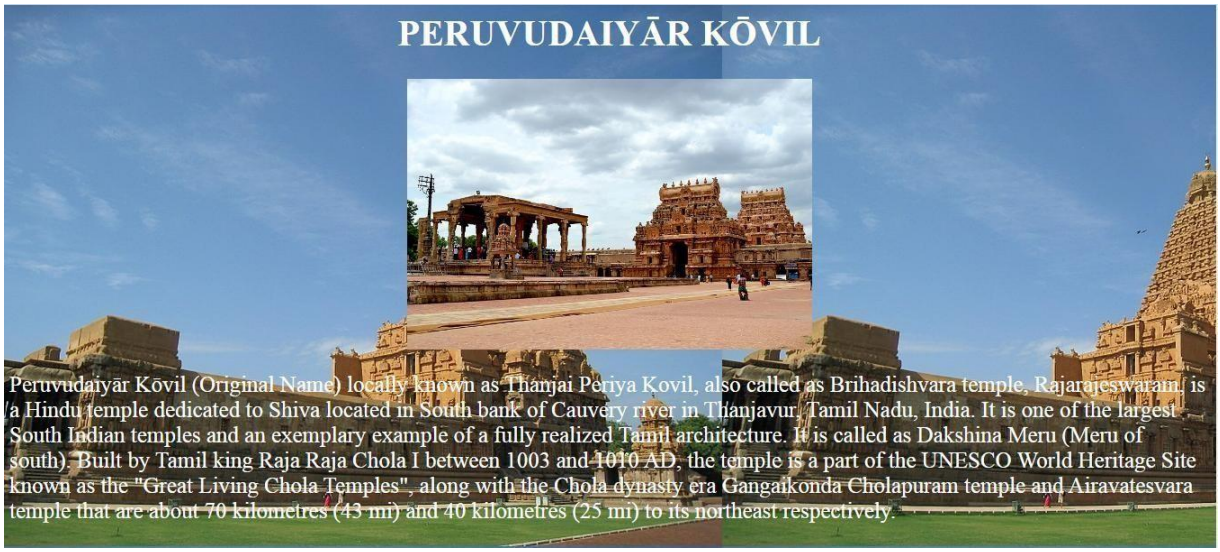
BIG temple

</br>

</body>

</html>

OUTPUT:



3. D Create a web page using frame sets that containing inline and internal style sheets and make it link to the external style sheets. Include information about a company providing Tours and Travels.

PROGRAM:

FRAME 1:

```
<html>

<head>

<title>TOURS AND TRAVELS</title>

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

</head>

<body  background="thomas.jpg" >

<center><img src="D:\web technology lab\EX2\comprof.jpg" width="500" height="300"></center>

<h1 style="font-size:35">  TRAVELS AND TOURISM </h1>

<div><p style="color:crimson;">Tourism is travel for pleasure or business; also the theory and practice of touring, the business of attracting, accommodating, and entertaining tourists, and the business of operating tours.</p>

<p style="color:crimson;">The World Tourism Organization defines tourism more generally, in terms which go "beyond the common perception of tourism as being limited to holiday activity only", as people "traveling to and staying in places outside their usual environment for not more than one consecutive year for leisure and not less than 24 hours, business and other purposes".</p>

</div>

<h1 style="font-size:35">  THOMAS COOK  </h1>

<p style="color:crimson;font-size:25;">

<div><p>Thomas Cook (India) Limited is a leading integrated travel and travel-related services company with operations in 29 countries, across five continents through its Indian and global subsidiaries and key investments. Headquartered in India, travel and travel-related financial services companies in the Asia Pacific region.</p>

<p>Thomas Cook (India) Ltd. is an Indian travel agency, headquartered in Mumbai, India.</p>

<p> It offers a range of travel services including Foreign Exchange, International and Domestic Holidays, Visa, Passport, Travel Insurance and MICE. Founded in 1881 by Thomas Cook, the founder of the
```

defunct British brand Thomas Cook & Son, who established its first office in India and eventually extended to over 233 locations, in 94 cities across India, Sri Lanka and Mauritius.

Type:Public Limited Company

Traded as NSE: THOMASCOOK;

BSE: 500413

Industry:Tourism Hospitality

Founded:1881; 140 years ago

Founder:Thomas Cook

Headquarters:Thomas Cook (India) Limited, Thomas Cook Building, Dr D.N. Road, Mumbai–400001, India

Products:Package Holidays Foreign Exchange

Website:www.thomascook.in

<h1 style="font-size:35"> GUIDELINES FOR TRAVELLERS </h1>

A tour guide or a tourist guide is a person who provides assistance, information on cultural, historical and contemporary heritage to people on organized sightseeing and individual clients at educational establishments, religious and historical sites such as; museums, and at various venues of tourist attraction resorts.

STEP-1 - Health Declaration

Mandatory Health Status to be declared on the Aarogya Setu App by every travellers before each travel.

STEP-2 - State Guidelines Check

Check the quarantine guidelines of the state they are travelling to.

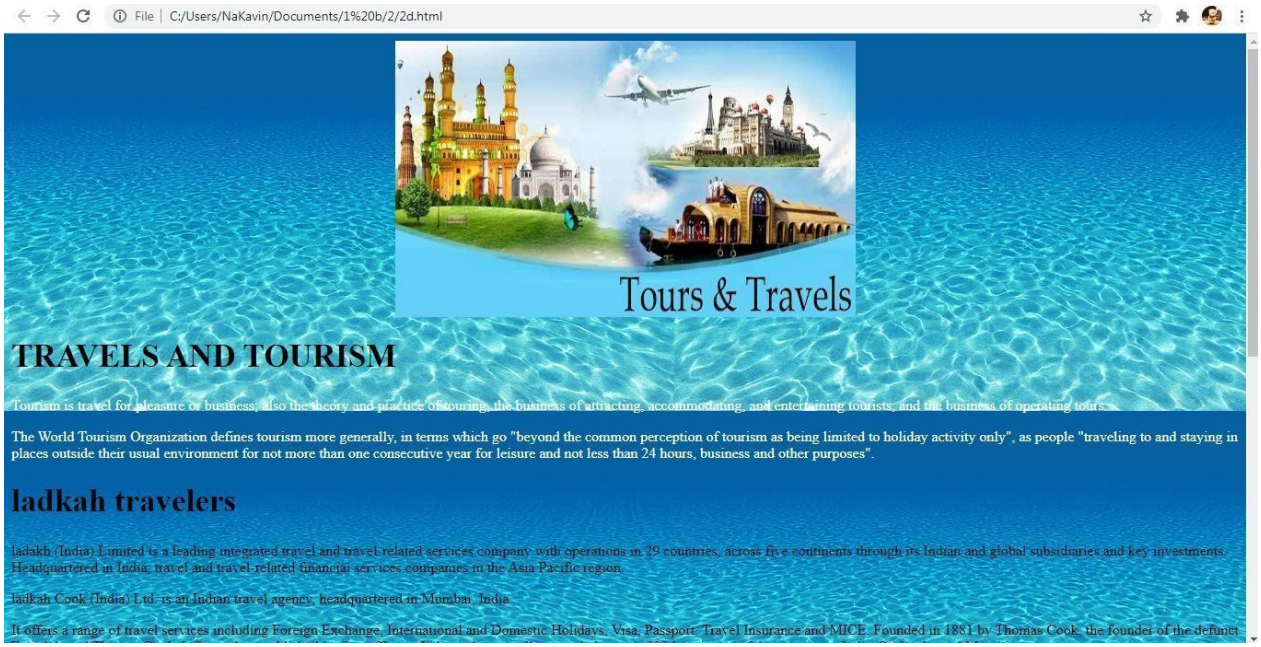
STEP-3 - Accommodation

Only approved accommodation facilities should be utilised.

STEP-4 - Transfers

Approved transfers to be used by travelers; preferably private cab transfer companies.

FRAME 1 OUTPUT:



FRAME 2:

<html>

<head>

<title> TOURS AND TRAVEL FORM </title>

</head>

<body background="11.jpg">

<form name="TRAIN" method="post">

```
<body style="color:white;">

<center>

<table border="2" cellpading=10 cellspacing=5>


<tr><td><center><b><u>  USER  PERSONAL DETAILS  </u></b></center></br></td></tr>


<p><tr><td>

    <b> <label> ENTER YOUR FULL NAME:  </label><center>

      <input type="text" size="40" name="name"><center></b></td></tr></p></br>

<p><tr><td>

       <b> <label>ENTER YOUR EMAIL ADDRESS:  </label><center>

       <input type="email" size="40" name="email"></center><br></td></tr></b></p>

<p><center><tr><td>

      <b> <label> ENTER YOUR PHONE NUMBER(in format of xxx-xxx-xxxx):
 </label><center>

    <input type="tel" size="40" name="telephone" pattern="[0-9]{3}-[0-9]{3}-[0-9]{4}" required>
</center><br></td></tr></b></p>

<tr><td>

<b>GENDER:</br>

       <input type="radio" id="gender" name="gender" value="MALE"/>MALE  <br>

       <input type="radio" id="gender" name="gender" value="FEMALE"/>FEMALE <br/>

       <input type="radio" id="gender" name="gender" value="OTHERS"/>OTHERS <br/>
</td></tr></b>


<tr><td><center><b><u>  TRIP INFORMATION  </u></b></center></br> </td></tr>


<p><center><tr><td>

<b>DEPARTURE DATE: <label for="start"></label><center>
```

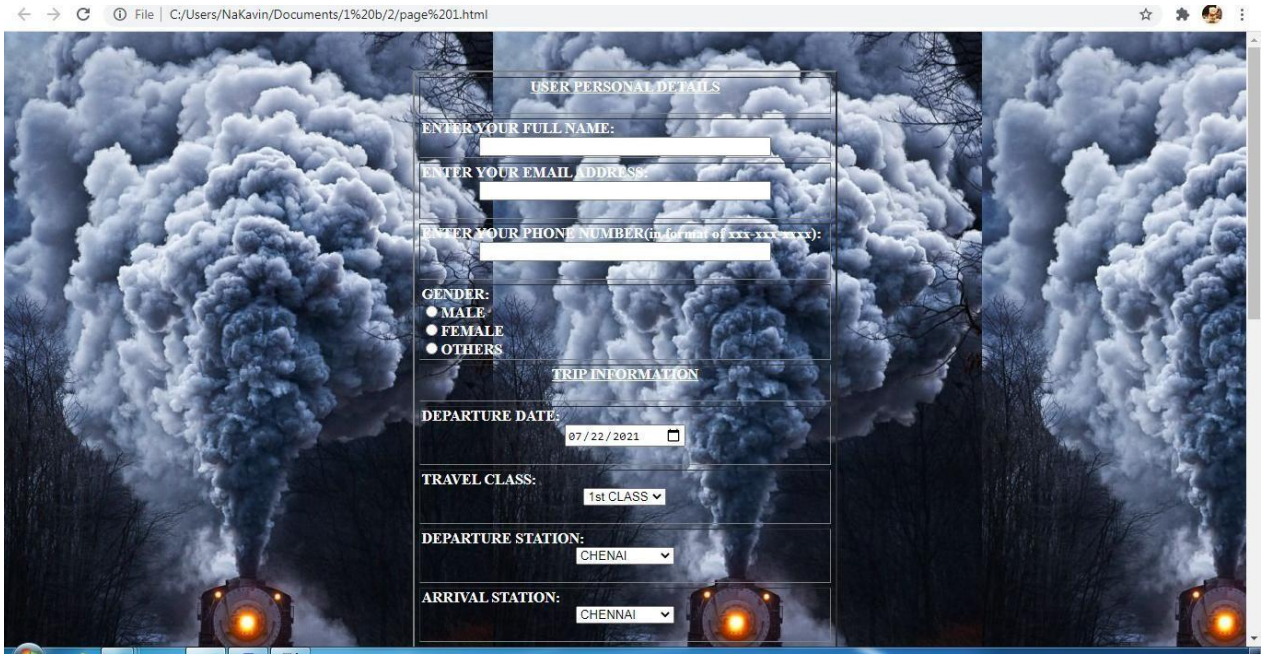
```
<input type="date" size="40" id="start" name="trip-start"
      value="2021-07-22"
      min="2021-01-01" max="2023-12-31"></center></br></b></td></tr></p>
<tr><td>
<b>TRAVEL CLASS:   <center><select name="dropdown">
<option value="1st CLASS">1st CLASS </option>
<option value="AC">AC </option>
<option value="SLEEPER">SLEEPER</option>
</select></center></br></td></tr></b>
<tr><td>
<b>DEPARTURE STATION:   <center><select name="dropdown">
<option value="CHENNAI">CHENAI</option>
<option value="BANGALORE">BANGALORE </option>
<option value="GOA">GOA</option>
<option value="HYDERABAD">HYDERABAD</option>
<option value="VELLORE">VELLORE</option>
<option value="LUCKNOW">LUCKNOW</option>
<option value="DELHI">DELHI</option>
<option value="MUMABI">MUMBAI</option>
<option value="VARANASI">VARANASI</option>
</select></center></br></td></tr></b>
<tr><td>
<b>ARRIVAL STATION:   <center><select name="dropdown">
<option value="CHENNAI">CHENNAI</option>
<option value="BANGALORE">BANGALORE </option>
```

```
<option value="GOA">GOA</option>
<option value="HYDERABAD">HYDERABAD</option>
<option value="VELLORE">VELLORE</option>
<option value="LUCKNOW">LUCKNOW</option>
<option value="DELHI">DELHI</option>
<option value="MUMBAI">MUMBAI</option>
<option value="VARANASI">VARANASI</option>
</select></center></br></td></tr></b>
<tr><td>
<b>ARRIVAL DATE:  <label for="start"></label><center>

<input type="date" id="start" name="trip-start"
      value="2021-07-22"
      min="2021-01-01" max="2023-12-31"></center></br></td></tr></b>
<tr><td>
<b>ARRIVAL TIME:  <label for="appt"></label><center>
<input type="time" id="appt" name="appt"
      min="09:00" max="18:00" required></center></br></td></tr></b>
<tr><td>
<b>TRAIN :   <center><select name="dropdown">
<option value="DOUBLE DECKER">DOUBLE DECKER </option>
<option value="CHENNAI EXPRESS">CHENNAI EXPRESS</option>
<option value="MUMBAI EXPRESS">MUMBAI EXPRESS</option>
<option value="YVSSL EXPRESS">YVSSL EXPRESS</option>
<option value="SSML EXPRESS">SSML EXPRESS</option>
</select></center></br></td></tr></b>
```

```
<tr><td>
<b>WEIGHT OF LUGGAGES: <center>
      <input type="number" size="80" name="name"></center><br></td></tr></b>
<tr><td>
<b>NUMBER OF ADULTS: <center>
      <input type="number" size="80" name="name"></center><br></td></tr></b>
<tr><td>
<b>NUMBER OF CHILDRENS: <center>
      <input type="number" size="80" name="name"></center><br></td></tr></b>

<tr><td>
<b>MEALS:   <center><input type ="checkbox">VEG
<input type="checkbox">NON VEG </center></br></td></tr></b>
<tr><td><center><b><u>  TRAVELLER'S DETAILS  </u></b></center></br></td></tr>
<tr><td>
<b>TRAIN NUMBER: <center>
      <input type="text" size="40" name="name"> </center></br></td></tr></b>
<tr><td>
<b>RAILWAY FREE PASS EXPIRATION DATE:  <label for="start"></label> <center>

<input type="date" id="start" name="trip-start"
      value="2021-07-22"
      min="2021-01-01" max="2023-12-31"> </center></br></td></tr></b>
<tr><td>
<b>ANY MESSAGE:  <center>
      <textarea cols="45" size="70"></textarea></center></br></td></tr></b>
```

```
</table>

</center>

<b>SUBMIT:<input type="button" size="20" name="SUBMIT" value="SUBMIT"</b>

<b>CANCEL:<input type="button" size="20" name="CANCEL" value="CANCEL"</b>

<b>RESET: <input type="reset"size="40" value=" RESET "> </b>

</form>

</body>

</html>
```

FRAME 2 OUTPUTS:



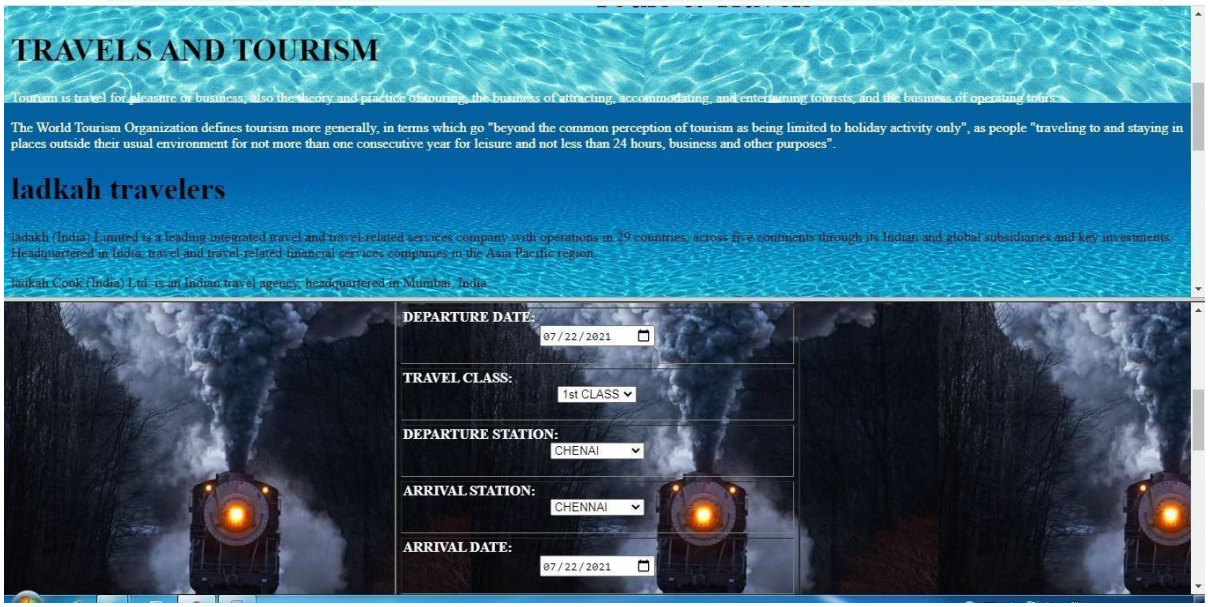
FRAME FINAL:

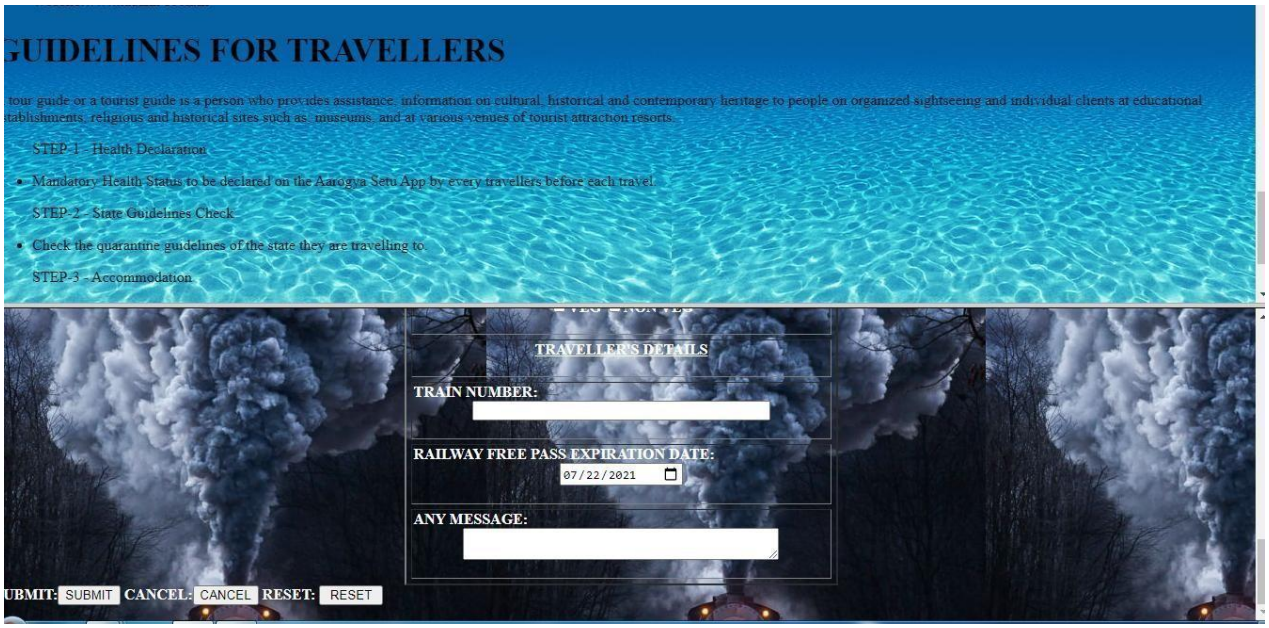
```
<!DOCTYPE html>

<html>
```

```
  <head>
    <title>HTML Frames</title>
  </head>

  <frameset rows = "50%,50%">
    <frame name = "top" src = "2d.html">
    <frame name = "main" src = "page 1.html">

    <noframes>
      <body>Your browser does not support frames.</body>
    </noframes>

  </frameset>

</html>
```

FRAME FINAL OUTPUT:


Tours & Travels
USER PERSONAL DETAILS
ENTER YOUR FULL NAME:
ENTER YOUR EMAIL ADDRESS:
ENTER YOUR PHONE NUMBER(in format of
GENDER:
MALE
FEMALE


TRAVELS AND TOURISM
Tourism is travel for pleasure or business; also the theory and practice of touring, the business of attracting, accommodating, and entertaining tourists, and the business of operating tours.
The World Tourism Organization defines tourism more generally, in terms which go "beyond the common perception of tourism as being limited to holiday activity only", as people "traveling to and staying in places outside their usual environment for not more than one consecutive year for leisure and not less than 24 hours, business and other purposes".
ladkah travelers
ladakh (India) Limited is a leading integrated travel and travel-related services company with operations in 29 countries, across five continents through its Indian and global subsidiaries and key investments. Headquartered in India, travel and travel-related financial services companies in the Asia Pacific region.
ladkah Cook (India) Ltd. is an Indian travel agency, headquartered in Mumbai, India.
DEPARTURE DATE:
07/22/2021
TRAVEL CLASS:
1st CLASS
DEPARTURE STATION:
CHENAI
ARRIVAL STATION:
CHENNAI
ARRIVAL DATE:
07/22/2021



RESULT:

The design of a web site for the concepts of Inline styles, Embedded styles and Cascading Styles Sheets using CSS was successfully written, executed and verified