**\*\*\*IT-352 WEB TECHNOLOGIES LAB\*\*\***

*LAB CYCLE PROGRAMS:*

*AUTHOR:* **@Y19IT15.**

*NAME:* ***CHINDU KISHORE.***

**LAB PROGRAM-🡪1:**

1.Develop your **Bio-data** form using forms and basic XHTML tags.

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"><html><head><META http-equiv="Content-Type" content="text/html; charset=utf-8"><style></style></head><body><u></u>

<div>

<h1 id="m\_title">Bio Data Form</h1>

<div id="m\_form-outer">

<p id="m\_description">

To know the details of students

</p>

<form id="m\_survey-form" method="GET" target="\_blank">

<div class="m\_rowTab">

<div class="m\_labels">

<label id="m\_name-label" for="m\_name">Name <span id="m\_star">\*</span> : </label>

</div>

<div class="m\_rightTab">

<input type="text" name="name" id="m\_name" class="m\_input-field">

</div>

</div>

<div class="m\_rowTab">

<div class="m\_labels">

<label id="m\_fname-label" for="m\_fname">Father&#39;s Name: </label>

</div>

<div class="m\_rightTab">

<input type="text" name="fname" id="m\_fname" class="m\_input-field">

</div>

</div>

<div class="m\_rowTab">

<div class="m\_labels">

<label id="m\_mname-label" for="m\_mname">Mother&#39;s Name: </label>

</div>

<div class="m\_rightTab">

<input type="text" name="mname" id="m\_mname" class="m\_input-field">

</div>

</div>

<div class="m\_rowTab">

<div class="m\_labels">

<label id="m\_email-label" for="m\_email">Email: </label>

</div>

<div class="m\_rightTab">

<input type="email" name="email" id="m\_email" class="m\_input-field">

</div>

</div>

<div class="m\_rowTab">

<div class="m\_labels">

<label id="m\_number-label" for="m\_age">Age: </label>

</div>

<div class="m\_rightTab">

<input type="number" name="age" id="m\_number" class="m\_input-field">

</div>

</div>

<div class="m\_rowTab">

<div class="m\_labels">

<label id="m\_dob-label" for="m\_dob">D.O.B: </label>

</div>

<div class="m\_rightTab">

<input type="date" name="dob" id="m\_iddob" class="m\_input-field">

</div>

</div>

<div class="m\_rowTab">

<div class="m\_labels">

<label for="m\_department">Department: </label>

</div>

<div class="m\_rightTab">

<select id="m\_dropdown" name="department" class="m\_dropdown">

<option disabled value>Select an option</option>

<option value="it">IT</option>

<option value="cse">CSE</option>

<option value="ece">ECE</option>

<option value="mech">MECH</option>

<option value="civil">CIVIL</option>

<option value="other">Other</option>

</select>

</div>

</div>

<div class="m\_rowTab">

<div class="m\_labels">

<label for="m\_gender">Gender: </label>

</div>

<div class="m\_rightTab">

<ul style="list-style:none">

<li class="m\_radio"><label><input name="radio-buttons" value="male" type="radio" class="m\_userRatings">Male</label></li>

<li class="m\_radio"><label><input name="radio-buttons" value="female" type="radio" class="m\_userRatings">Female</label></li>

<li class="m\_radio"><label><input name="radio-buttons" value="others" type="radio" class="m\_userRatings">Others</label></li>

</ul>

</div>

</div>

<div class="m\_rowTab">

<div class="m\_labels">

<label for="m\_address">Address: </label>

</div>

<div class="m\_rightTab">

<textarea id="m\_comments" class="m\_input-field" style="height:50px" name="address"></textarea>

</div>

</div>

<div class="m\_rowTab">

<div class="m\_labels">

<label for="m\_pincode">Pincode: </label>

</div>

<div class="m\_rightTab">

<input type="number" name="pincode" id="m\_idpincode" class="m\_input-field" style="height:20px"></div>

</div>

<div class="m\_rowTab">

<div class="m\_labels">

<label for="m\_language">Language Known: </label>

</div>

<div class="m\_rightTab">

<ul id="m\_language" style="list-style:none">

<li class="m\_checkbox"><label><input name="prefer" value="english" type="checkbox" class="m\_userRatings">English</label></li>

<li class="m\_checkbox"><label><input name="prefer" value="tamil" type="checkbox" class="m\_userRatings">Tamil</label></li>

<li class="m\_checkbox"><label><input name="prefer" value="hindi" type="checkbox" class="m\_userRatings">Hindi</label></li>

<li class="m\_checkbox"><label><input name="prefer" value="malayalam" type="checkbox" class="m\_userRatings">Malayalam</label></li>

<li class="m\_checkbox"><label><input name="prefer" value="kannada" type="checkbox" class="m\_userRatings">Kannada</label></li>

<li class="m\_checkbox"><label><input name="prefer" value="telugu" type="checkbox" class="m\_userRatings">Telugu</label></li>

<li class="m\_checkbox"><label><input name="prefer" value="others" type="checkbox" class="m\_userRatings">Others</label></li>

</ul>

</div>

</div>

<button id="m\_submit" type="submit">Submit</button>

<button id="m\_reset" type="reset">Reset</button>

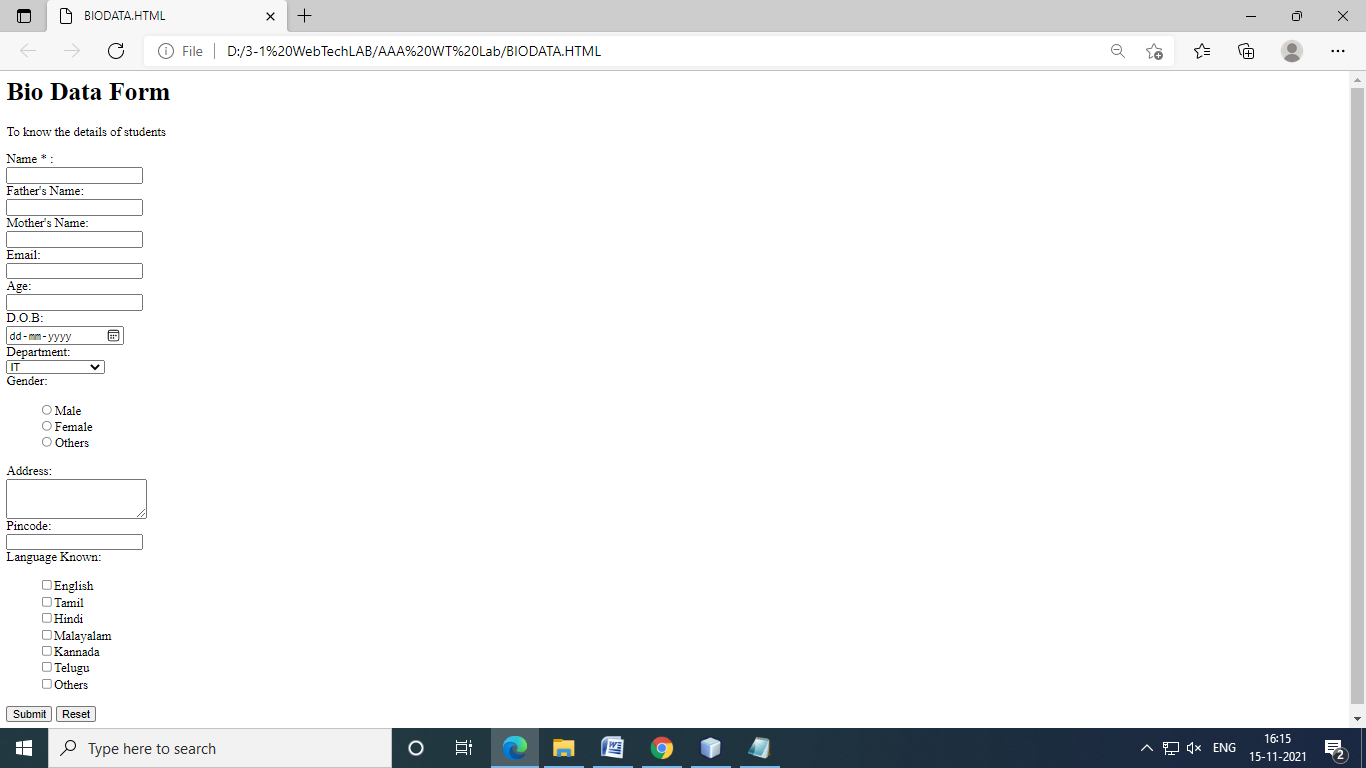
</form>

</div>

</div>

</body></html>

OUTPUT:



**LAB PROGRAM-🡪2A:**

1. Write the CSS rules for a simple static website.
   * 1. Rule for a background image is **left top** of the page, **tiling horizontally**. The image should **remain in place** when the user scrolls up or down.
     2. All paragraphs text **1.5 times** larger than the base font of the system and colors it **red** (inline, embedded and external style sheet).
     3. Rule for all H1 & H2 elements a **padding of 0.5em**, a grooved border style and a **margin of 0.5em**. (Box Model)

<!DOCTYPE html>

<html>

<head>

<title>Coustom Demo Website</title>

<style>

body

{

min-height:500vh;

min-width:1000vw;

background-image:url("nature3.jpg");

background-attachment:fixed;

background-position:left-top;

background-repeat:repeat-x;

}

</style>

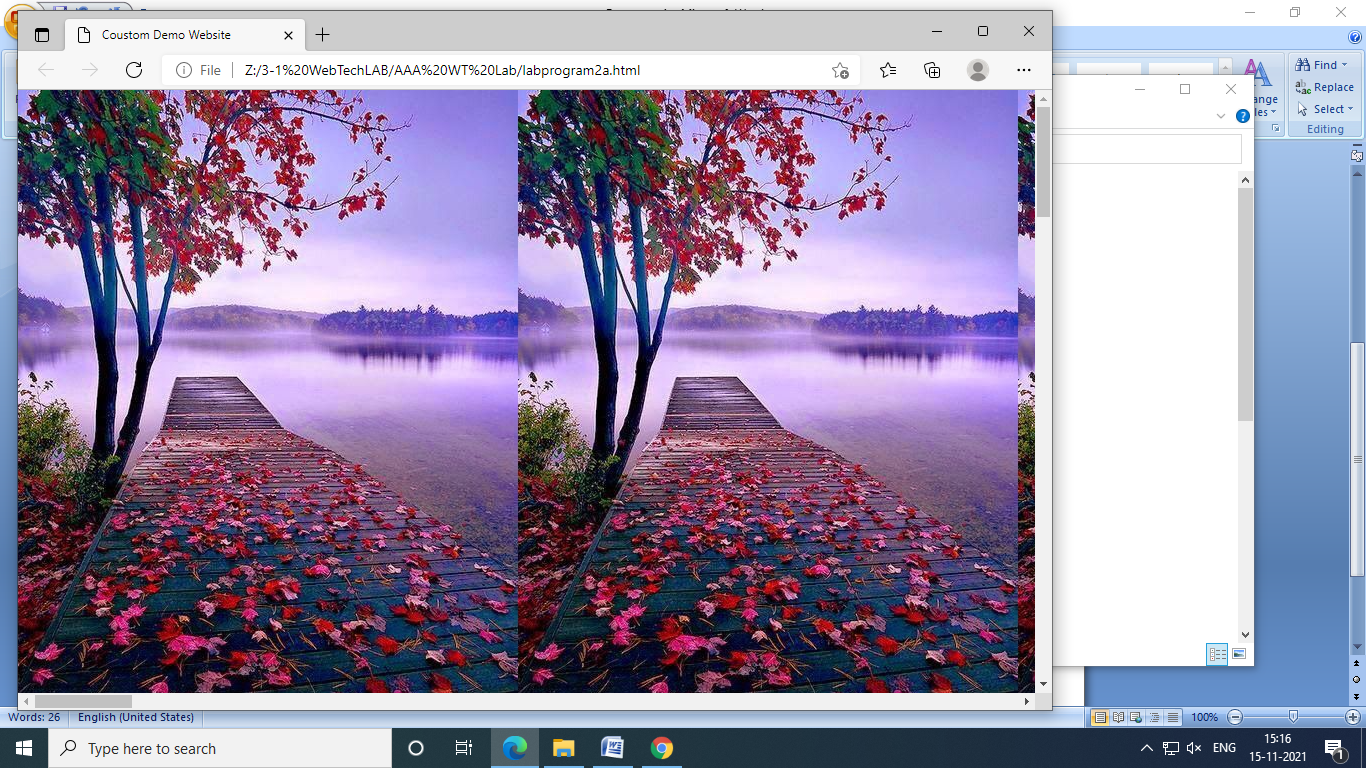
</head>

<body>

</body>

</html>

OUTPUT:



**LAB PROGRAM-🡪2B:**

<!DOCTYPE html>

<html>

<head>

<title> All Types of style sheets </title>

<link rel="stylesheet" href="external.css">

<style>

\*

{

padding:0;

margin:0;

box-sizing:border-box;

}

body

{

background-image:url("lion.jpg");

height:50vh;

width:50vh;

background-repeat:no-repeat;

background-size:cover;

background-direction:center;

display:flex;

justify-content:center;

align-items:left-top;

flex-direction:column;

color:black;

}

#internal

{

color:red;

font-size:24px;

font-size:24px;

}

span

{

color:red;

background-color:black;

}

</style>

</head>

<body>

<div id="content">

This is <span>base</span> font.

<p style="color:red; font-size:24px;">This is done Using <span>In-line</span> style sheet and the font-size is 1.5 times than the base font.</p>

<p id="internal">This is done Using <span>Internal</span> style sheet and the font-size is 1.5 times than the base font.</p>

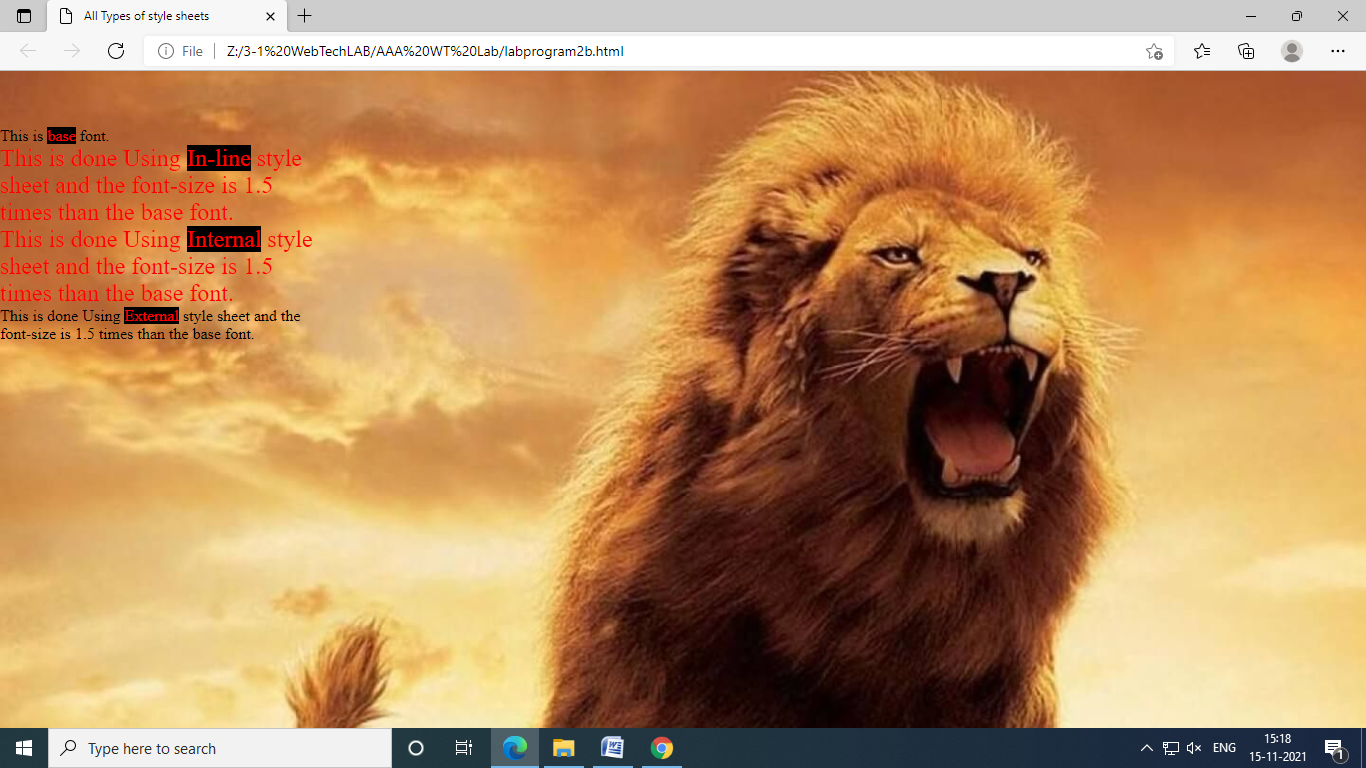
<p id="external">This is done Using <span>External</span> style sheet and the font-size is 1.5 times than the base font.</p>

</div>

</body>

</html>

OUTPUT:



**LAB PROGRAM-🡪2C:**

<!DOCTYPE html>

<html>

<head>

<title>Playing With Box-Model</title>

<style>

\*

{

padding:0;

margin:0;

box-sizing:border-box;

overflow:hidden;

}

body

{

background-image:url("nature3.jpg");

background-position:center;

background-repeat:no-repeat;

background-size:cover;

min-height:100vh;

display:flex;

justify-content:center;

align-items:center;

flex-direction:column;

}

h1,h2

{

color:#5618b7;

padding:0.5em;

border:4px groove red;

margin:0.5em;

}

</style>

</head>

<body>

<h1>KISHORE</h1>

<h1>KISHORE</h1>

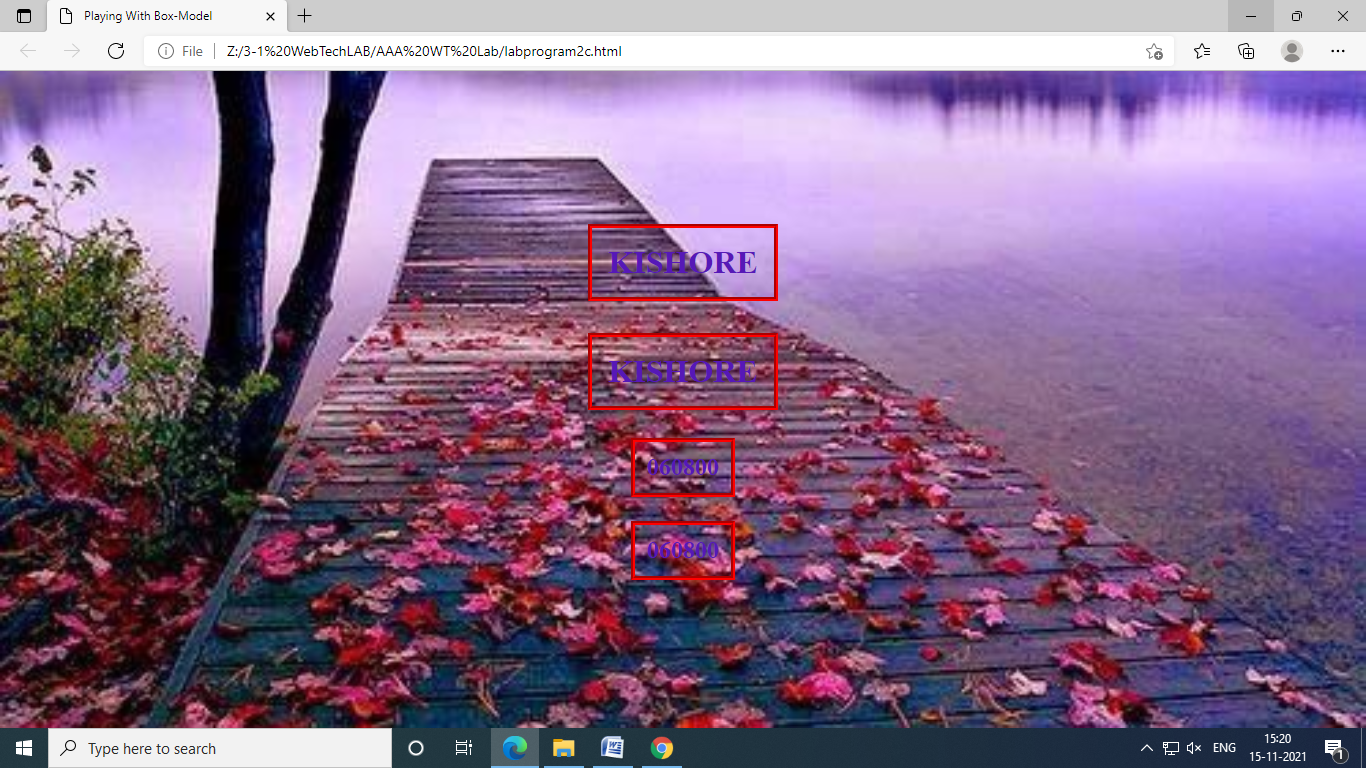
<h2>060800</h2>

<h2>060800</h2>

</body>

</html>

OUTPUT:



**LAB PROGRAM-🡪3:**

1. Write a script which reads a four digit integer entered by the user in a prompt dialog and encrypt it as replace each digit by the sum of that digit plus 7 modulus 10 and then swap the first digit with the third, and swap the second digit with the fourth. Then output the encrypted data . Write a separate script that inputs an encrypted integer and decrypts it to from the original number

<html>

<head> <!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>PROGRAM-3<script language="JAVASCRIPT" type="TEXT/JAVASCRIPT">

var n=parseInt(window.prompt("Enter a number:",0000));

document.writeln("The given number is "+n+"<br><br>");

var a=new Array(4);

i=0;

while(n>0)

{

a[i++]=parseInt(parseInt(n%10)+7)%10;

n/=10;

}

b=a[0];

a[0]=a[2];

a[2]=b;

b=a[1];

a[1]=a[3];

a[3]=b;

sum=0;

for(var i=3;i>=0;i--)

sum=sum\*10+a[i];

document.writeln("The encrypted number is ");

if(a[3]==0)

document.writeln("0");

document.writeln(sum);

for(i=0;i<4;i++)

a[i]=parseInt((a[i]+3)%10);

b=a[0];

a[0]=a[2];

a[2]=b;

b=a[1];

a[1]=a[3];

a[3]=b;

sum=0;

for(var i=3;i>=0;i--)

sum=sum\*10+a[i];

document.writeln("<br><br>The decrypted number is "+sum);

</script>

</title>

</head>

<body>

<script language="JAVASCRIPT" type="TEXT/JAVASCRIPT">

var n=parseInt(window.prompt("Enter a number:",0000));

document.writeln("The given number is "+n+"<br><br>");

var a=new Array(4);

i=0;

while(n>0)

{

a[i++]=parseInt(parseInt(n%10)+7)%10;

n/=10;

}

b=a[0];

a[0]=a[2];

a[2]=b;

b=a[1];

a[1]=a[3];

a[3]=b;

sum=0;

for(var i=3;i>=0;i--)

sum=sum\*10+a[i];

document.writeln("The encrypted number is ");

if(a[3]==0)

document.writeln("0");

document.writeln(sum);

for(i=0;i<4;i++)

a[i]=parseInt((a[i]+3)%10);

b=a[0];

a[0]=a[2];

a[2]=b;

b=a[1];

a[1]=a[3];

a[3]=b;

sum=0;

for(var i=3;i>=0;i--)

sum=sum\*10+a[i];

document.writeln("<br><br>The decrypted number is "+sum);

</script>

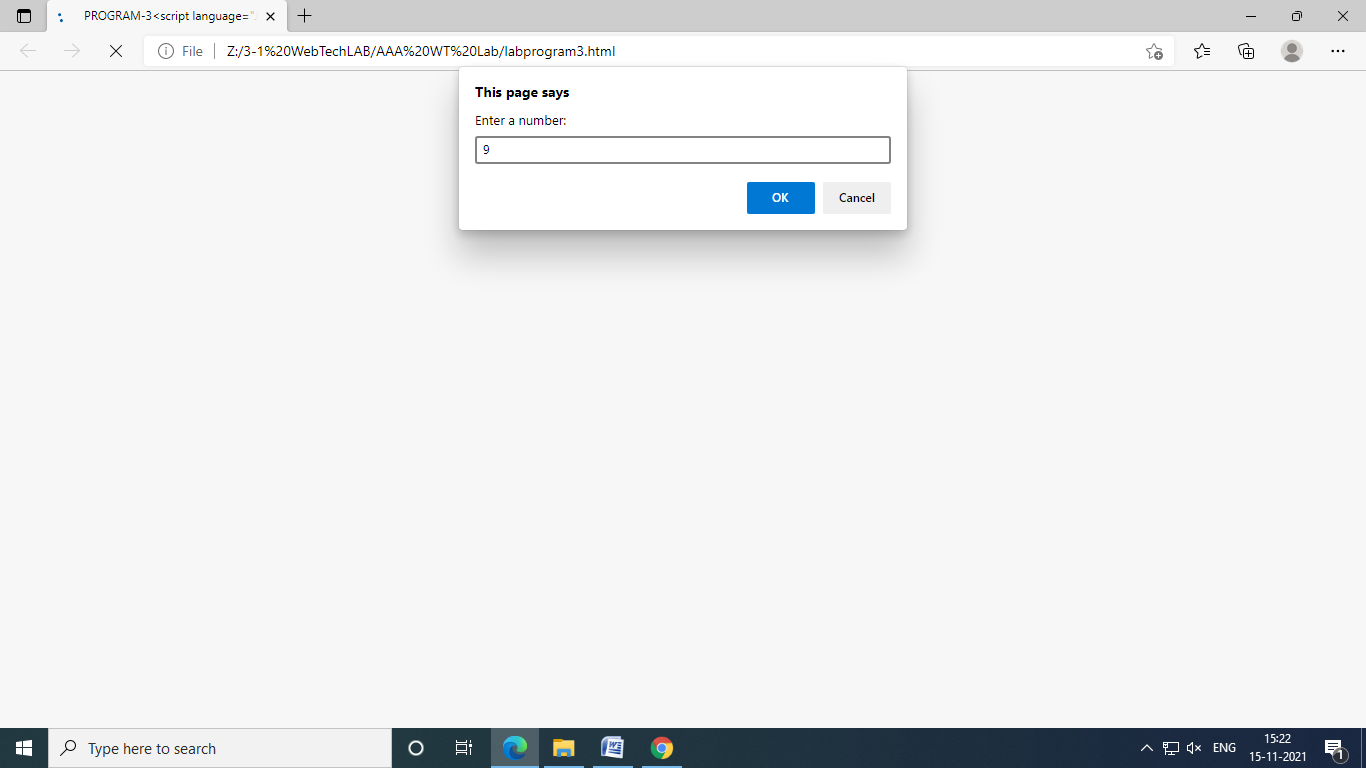
</body>

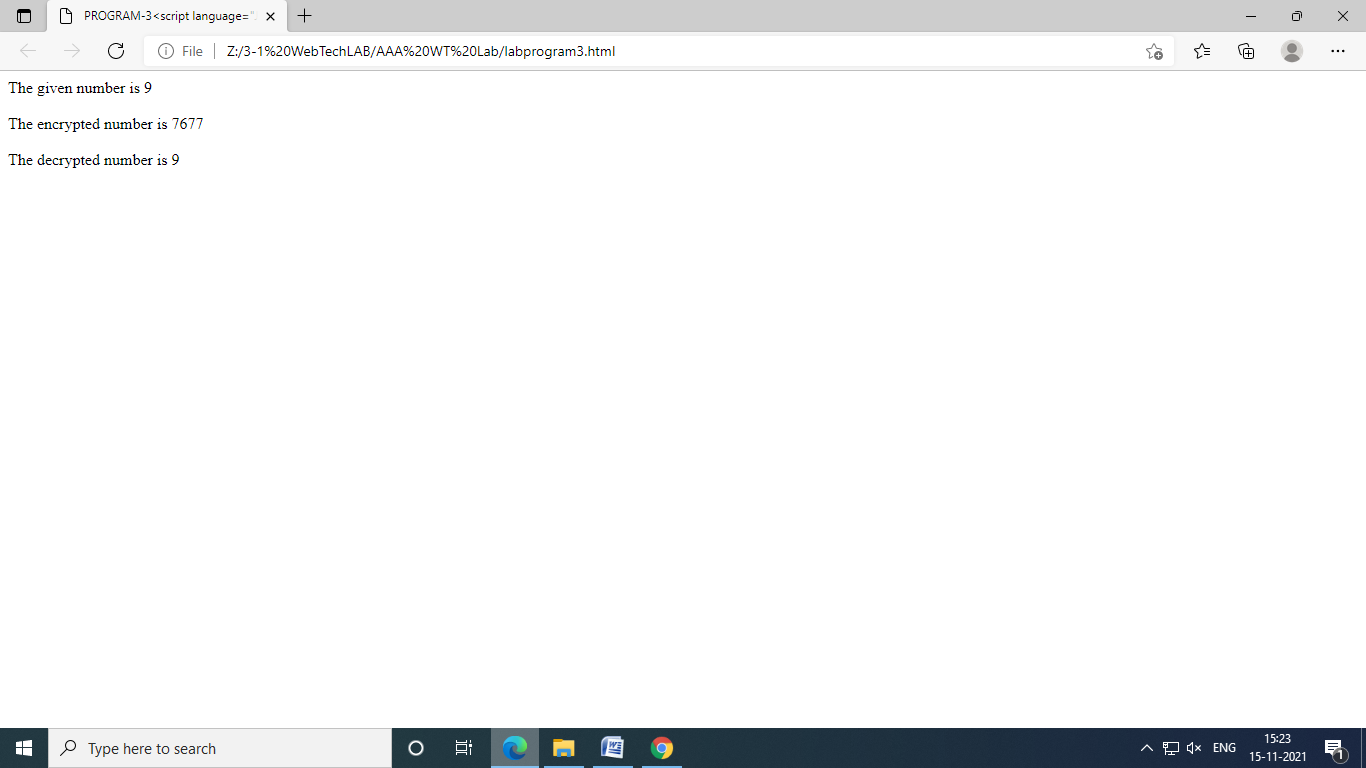
<html>

<title>PROGRAM-3</title>

</html>

OUTPUT:





**LAB PROGRAM-🡪4A:**

1. Write the following scripts using JavaScript
2. Search a number from a given list of numbers using **Binary Search**technique.
3. Sort given set of numbers using **Bubble Sorting** technique.

<!DOCTYPE html>

<html>

<head>

<title>Binary Search Tecgnique

</title>

<style>

\*

{

padding:0;

margin:0;

box-sizing:border-box;

}

#container

{

display:flex;

align-items:center;

flex-direction:column;

padding:10px;

}

h1

{

text-align:center-top;

}

</style>

</head>

<body>

<div id="container">

<h1>Implementation of Binary Search Technique</h1>

<h3 id="array"></h3>

<h3 id="output"></h3>

</div>

</body>

<script>

let output=document.getElementById("output");

let array=document.getElementById("array");

function binarySearch(arr, l, r, x){

if (r >= l) {

let mid = l + Math.floor((r - l) / 2);

// If the element is present at the middle

// itself

if (arr[mid] == x)

return mid;

// If element is smaller than mid, then

// it can only be present in left subarray

if (arr[mid] > x)

return binarySearch(arr, l, mid - 1, x);

// Else the element can only be present

// in right subarray

return binarySearch(arr, mid + 1, r, x);

}

// We reach here when element is not

// present in array

return -1;

}

// Elements in the array are the list of elements to search within

let arr = [ 12, 14, 17, 25, 36, 49,72];

document.write("Total number of elements in the array are : "+arr.length);

document.write(" ; The Following are the elements : ");

for(let t=0;t<arr.length;t++)

{

document.write("array["+(t)+"]" + ":" + arr[t] + " ");

}

//Change the required element in the place of X

let x = 72;

array.innerText=`You have searched for ${x}`;

let n = arr.length;

let result = binarySearch(arr, 0, n-1, x);

if(result==-1)

{

output.innerText="The required element is not present in the list";

}

else

{

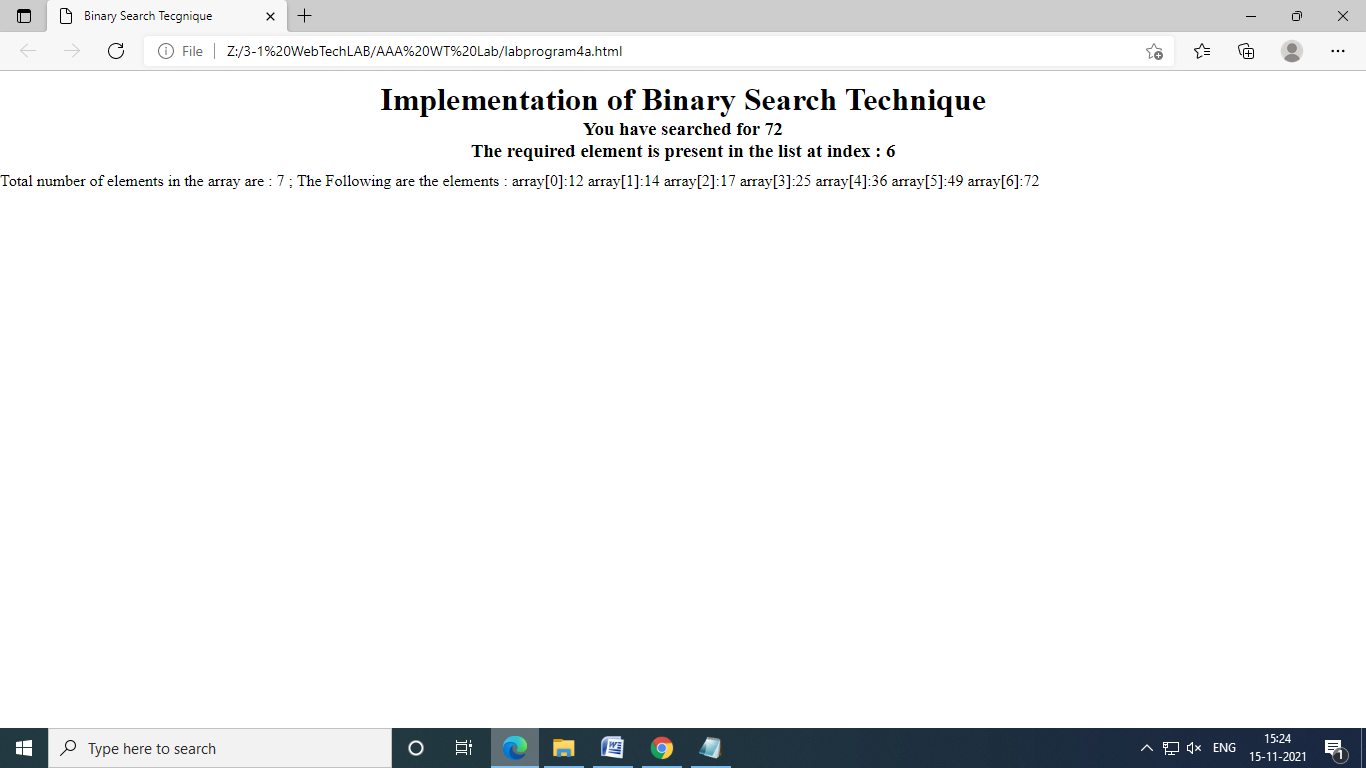
output.innerText=`The required element is present in the list at index : ${result}`;

}

</script>

</html>

OUTPUT:



**LAB PROGRAM-🡪4B:**

<!DOCTYPE html>

<head>

<title>

Bubble Sorting Technique

</title>

<style>

h1{

text-align:center;

}

</style>

</head>

<body>

<h1>Bubble Sroting Technique</h1>

</body>

<script>

let arr=[111,109,102,557,797,336];

let len=arr.length;

let i,j;

const bubble\_sort = (arr) =>

{

for(i=0;i<len;i++){

for(j=0;j<=(len-i);j++){

if(arr[j]>arr[j+1]){

let temp=arr[j];

arr[j]=arr[j+1];

arr[j+1]=temp;

}

}

}

document.write(arr);

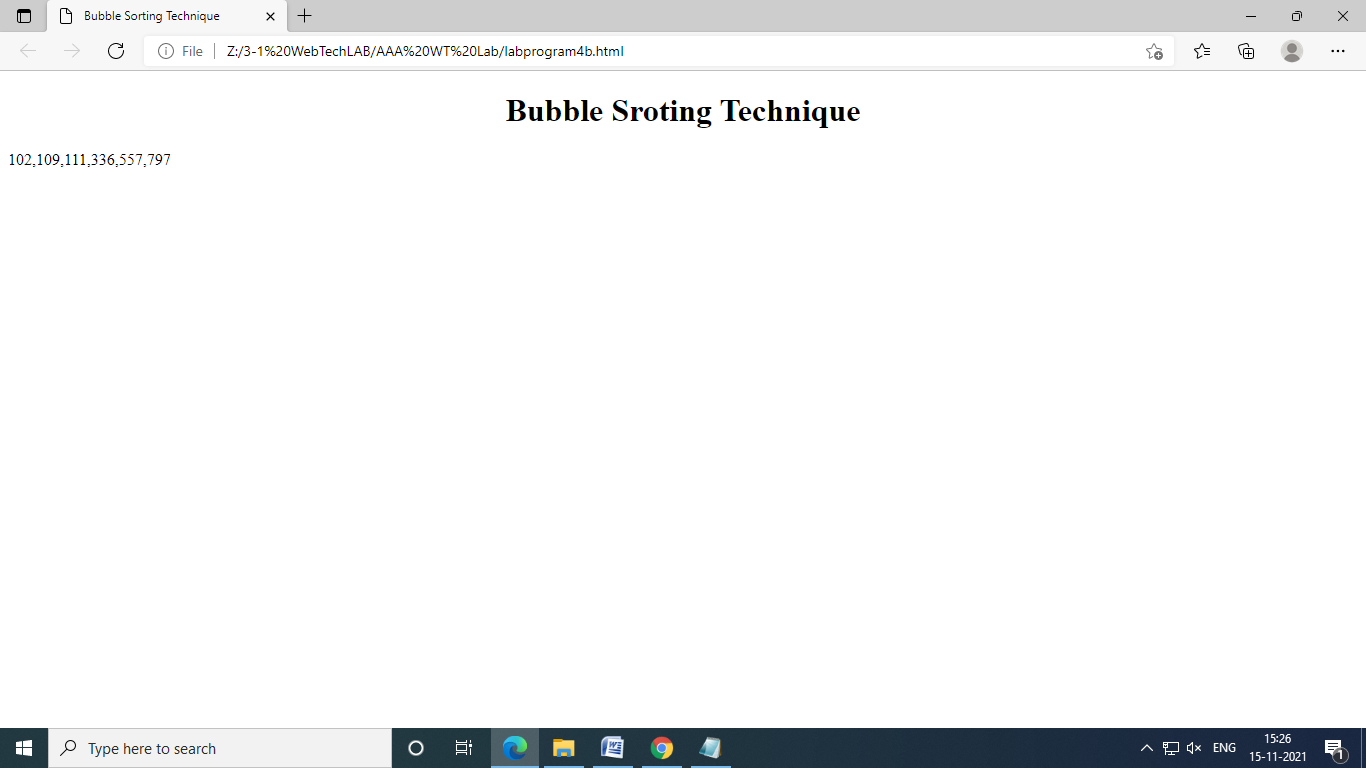
}

bubble\_sort(arr);

</script>

</html>

OUTPUT:



**LAB PROGRAM-🡪5A:**

1. Write scripts for the following using recursive functions
2. Write a function to power which takes two arguments **m** and **n**, and returns **mn** (consider m and n are integers**).**
3. Factorial & finding nth Fibonacci

<!DOCTYPE html>

<head>

<title>

Finding power of two numbers

</title>

</head>

<body>

<h1>Finding the power of two integers </h1>

<h1>m,n are two integers m<sup>n</sup></h1>

<input id="first" type="number" placeholder="Enter value of m">

<input id="second" type="number" placeholder="Enter value of n">

<button onclick="power()">Eval</button>

<h1>Result is : <span id="res"></span></h1>

</body>

<script>

const power = () =>

{

let res=document.getElementById("res");

let first\_number=document.getElementById("first");

let second\_number=document.getElementById("second");

let m=first\_number.value;

let n=second\_number.value;

let result=Math.pow(m,n);

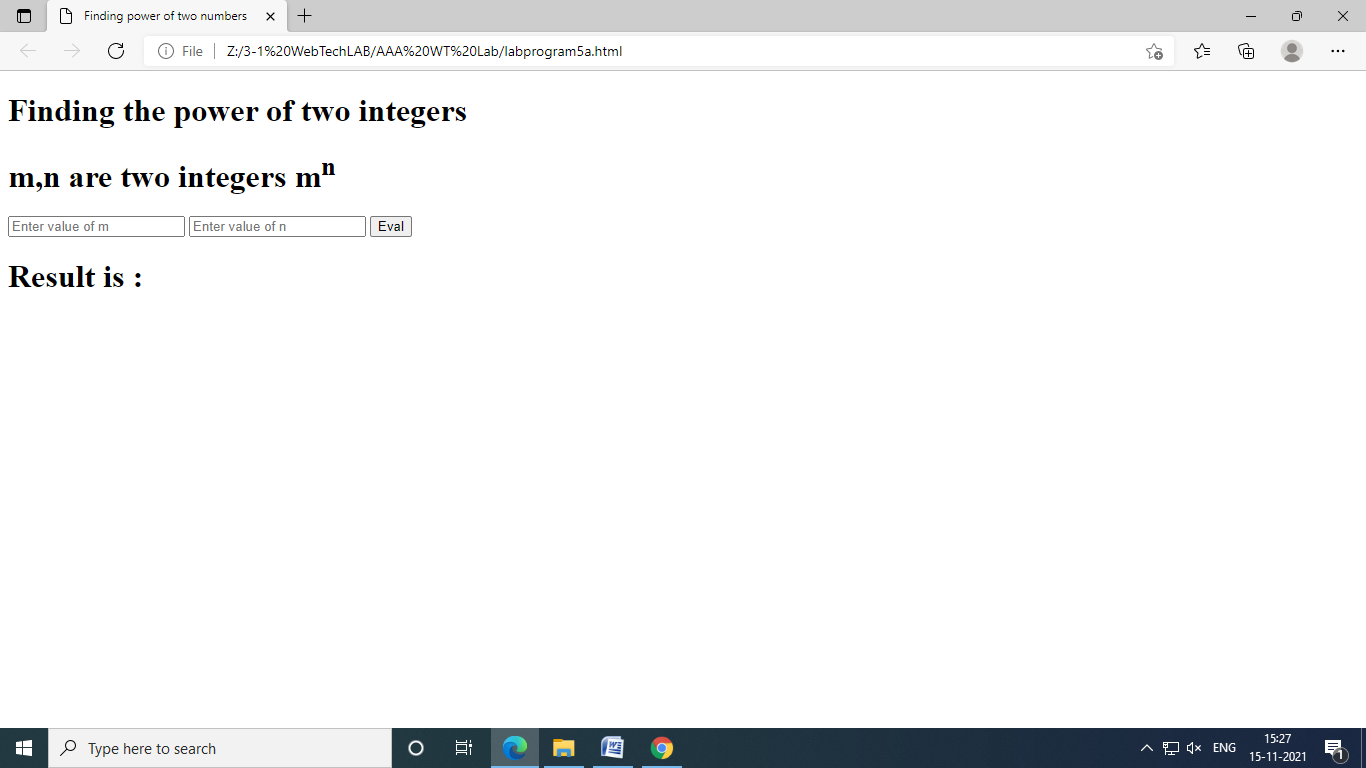
res.innerText=result;

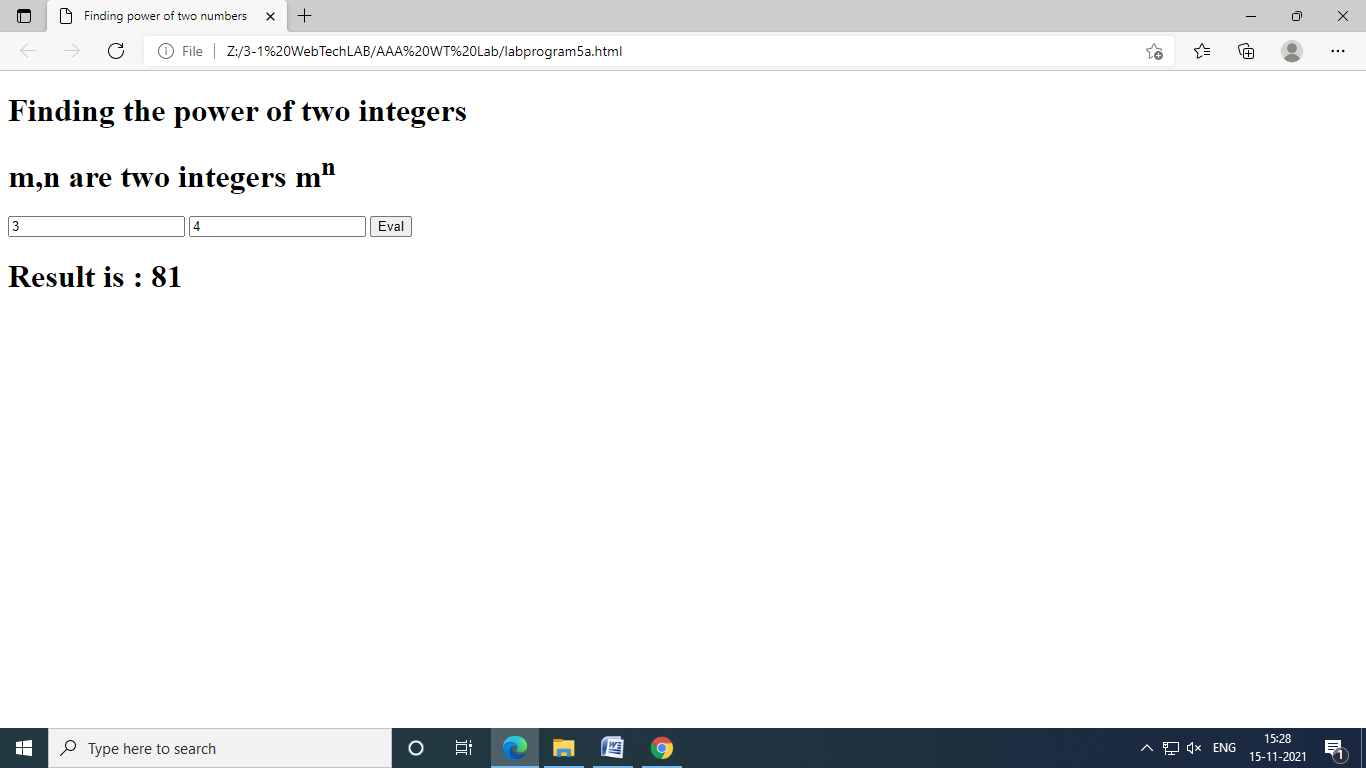
}

</script>

</html>

OUTPUT:





**LAB PROGRAM-🡪5B:**

<script>

var n;

var n;

n=parseInt(window.prompt("Enter a number"));

document.writeln("<h4>The Factorial numbers are<br></h4>");

document.write("<table style='border:1px solid red;'>");

for(var i=0;i<=n;i++)

document.write("<tr><td>"+i+"!</td><td>"+factorial(i)+"</td></tr>");

document.write("</table>");

document.writeln("<br><br><h4>The "+n+" fibonacci numbers are: </h4>");

for(i=0;i<n;i++)

{

f=fib(i);

document.writeln(f+" ");

}

function fib(n)

{

if(n==1||n==0)

return 1;

else

return fib(n-1)+fib(n-2);

}

function factorial(n)

{

if(n<=1)

return 1;

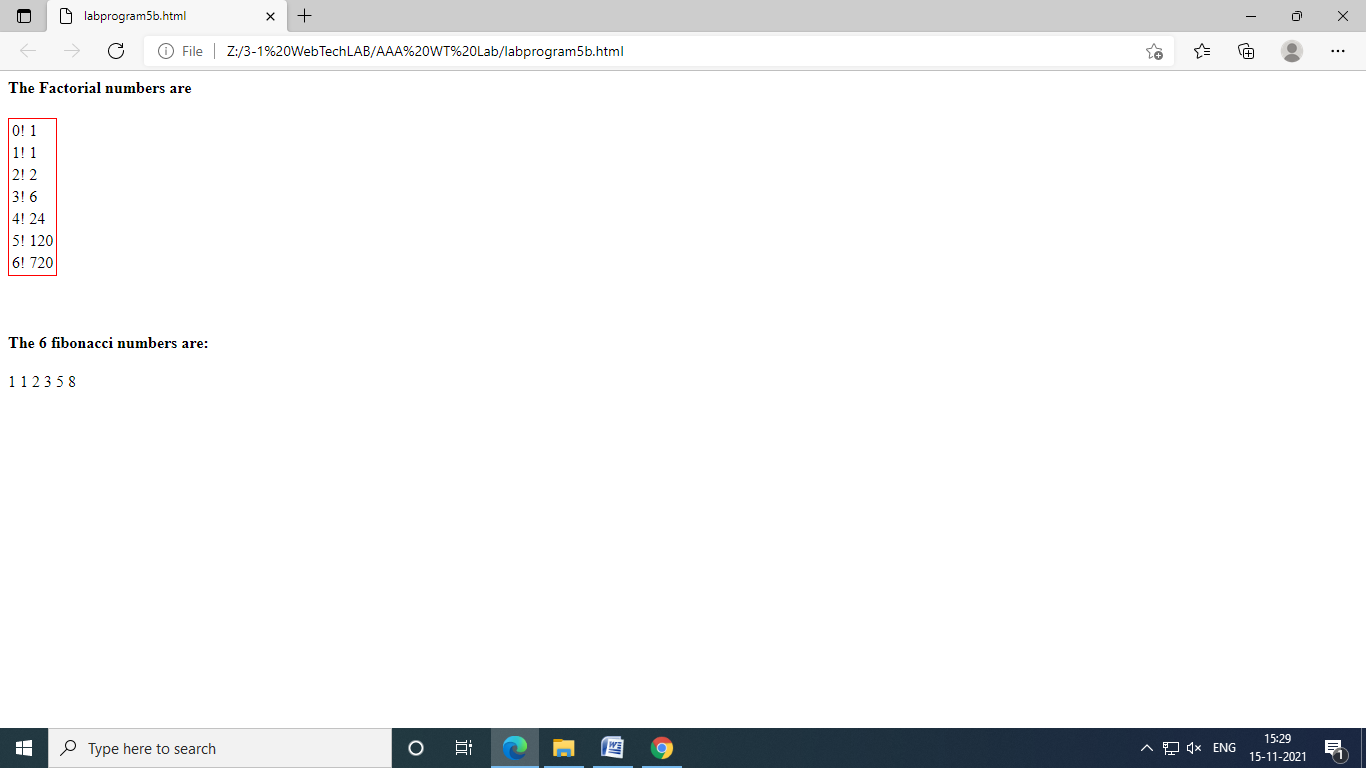
else

return n\*factorial(n-1);

}

</script>

OUTPUT:



**LAB PROGRAM-🡪6:**

1. Write a script which reads two matrices from the prompt dialog, and do the matrix multiplication.

<html>

<head>

<script>

var m1,n1,m2,n2;

m1=parseInt(window.prompt("enter no of rows of first matrix","m1"));

n1=parseInt(window.prompt("enter no of colomns of first matrix","n1"));

m2=parseInt(window.prompt("enter no of rows of second matrix","m2"));

n2=parseInt(window.prompt("enter no of colomns of second matrix","n2"));

if(n1!=m2)

window.alert("Two matrices r not compatable for multiplication");

else

{

a=new Array(m1);

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

a[i]=new Array(n1);

b=new Array(m2);

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

b[i]=new Array(n2);

c=new Array(m1);

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

c[i]=new Array(n2);

read(a);

read(b);

for(i=0;i<m1;i++)

for(j=0;j<n2;j++)

{

c[i][j]=0;

for(k=0;k<n1;k++)

c[i][j]+=a[i][k]\*b[k][j];

}

display("first matrix",a);

display("second matrix",b);

display("result matrix",c);

}

function read(x)

{

for(i=0;i<x.length;i++)

for(j=0;j<x[i].length;j++)

{

x[i][j]=parseInt(window.prompt("enter["+i+1+","+j+1+"]element","0"));

}

}

function display(str,y)

{

document.write("<h2>"+str+"</h2>");

for(i=0;i<y.length;i++)

{

for(j=0;j<y[i].length;j++)

document.write(y[i][j]+" ");

document.writeln("<br/>");

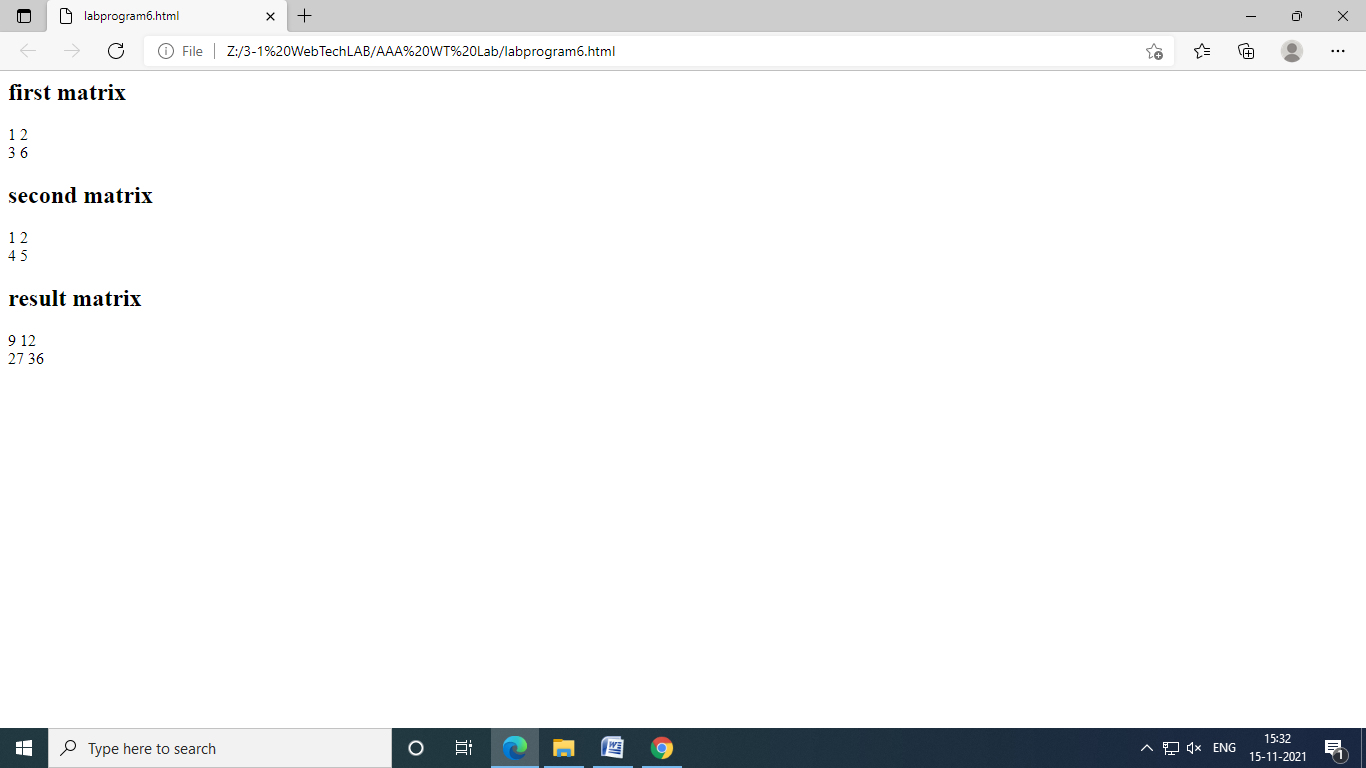
}

}

</script>

</html

OUTPUT:



**LAB PROGRAM-🡪7A:**

1. Write Scripts for the following events
2. Display digital clock using *onload* event.
3. To move text along with mouse pointer. The text will be inputted by a text field.

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

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

<title>Digital Clock</title>

<style>

h1

{

text-align:center;

}

h1:nth-child(1)

{

color:red;

}

h1:nth-child(2)

{

border:2px solid red;

background-color:teal;

font-size: 5.5em;

}

</style>

</head>

<body>

<h1>Digital Clock</h1>

<h1><span id="hours"></span><span id="minutes"></span><span id="seconds"></span><span id="meridian"></span></h1>

</body>

<script>

const showTime = () => {

let hours=document.getElementById("hours");

let minutes=document.getElementById("minutes");

let seconds=document.getElementById("seconds");

let meridian=document.getElementById("meridian");

var time=new Date();

let hr = time.getHours();

let min=time.getMinutes();

let sec=time.getSeconds();

if(hr<10){

hr=`0${hr}`;

}

if(hr>12)

{

hr=hr-12;

}

if(min<10){

min=`0${min}`;

}

if(sec<10){

sec=`0${sec}`;

}

if(hr<12){

meridian.innerText=` AM`;

}

if(hr>=12){

meridian.innerText=` PM`;

}

hours.innerText=hr;

minutes.innerText=`:${min}`;

seconds.innerText=`:${sec}`;

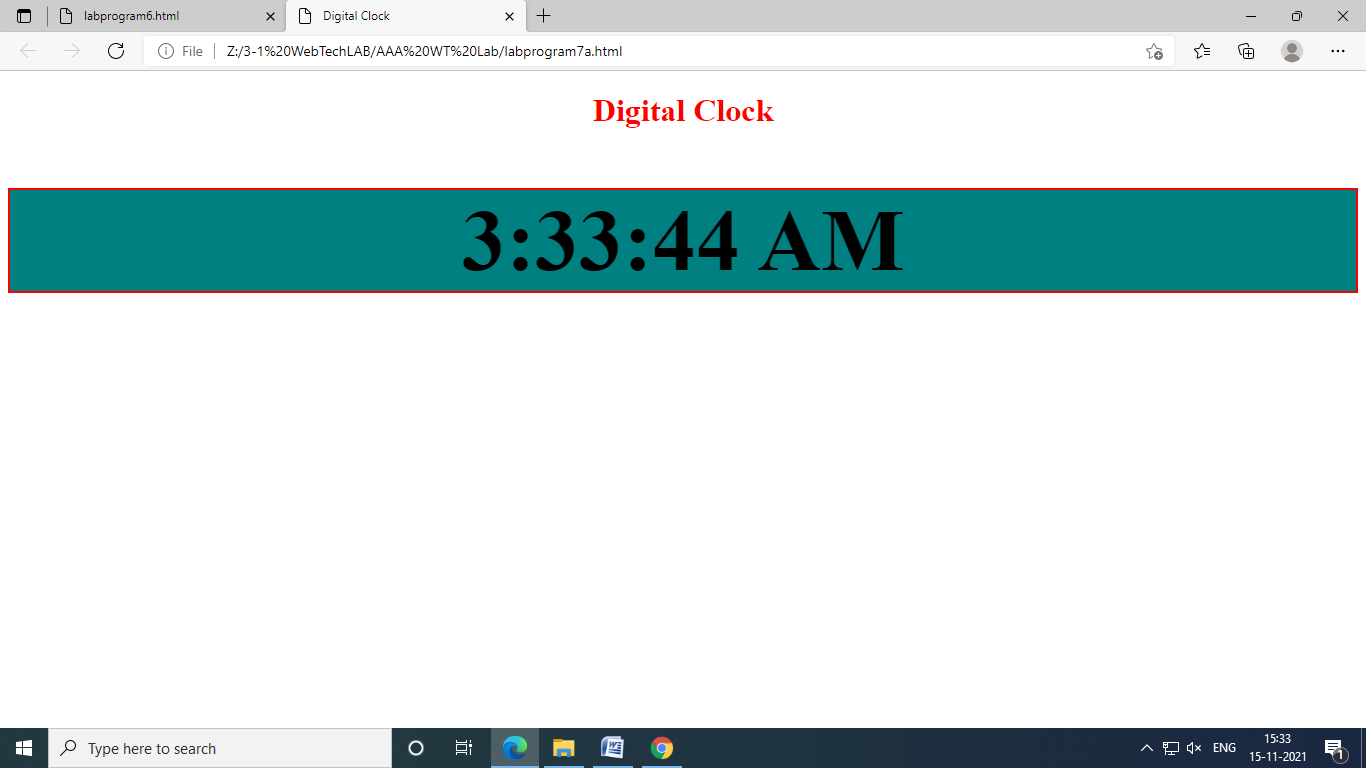
}

setInterval(showTime,1000);

</script>

</html>

OUTPUT:



**LAB PROGRAM-🡪7B:**

<html>

<head>

<script>

functionmovetext()

{

var v=document.f.t.value;

para.innerText=v;

para.style.left=event.offsetX;

para.style.top=event.offsetY;

}

</script>

</head>

<body onmousemove="movetext()"></body>

<form name="f">

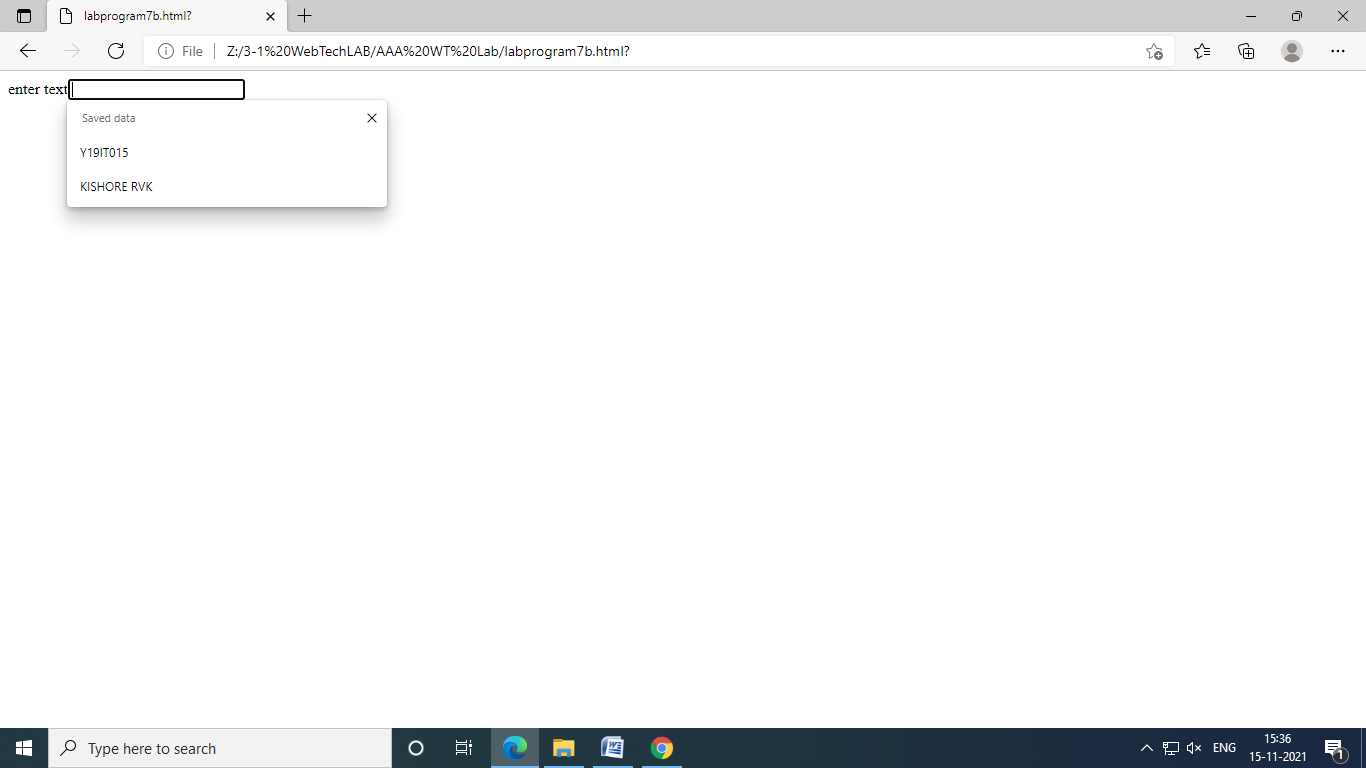
enter text<input type="text" id="t">

<p id="para" style="position:relative"></p>

</form>

</html>

OUTPUT:



**LAB PROGRAM-🡪8:**

1. Design a simple web page containing user’s login information. Validate that form by using following conditions.
   1. Name should be entered.
   2. Password must be greater than or equal to 6 characters.
   3. Ignore the submit event.
   4. Prompt the user whether he wants to clear the data or not. If yes then only clear the login information from the text fields.

<html>

<head>

<script language="JAVASCRIPT" type="TEXT/JAVASCRIPT">

function validate()

{

event.returnvalue=false;

if(document.f1.t1.value=="")

window.alert("Enter the username");

else if(document.f1.p1.value.length<8)

window.alert("The password should be minimum 8 characters");

else

window.alert("You logged in successfully");

}

function Clear()

{

v=window.confirm("Do you want to clear the data");

if(v)

{

document.f1.t1.value=" ";

document.f1.p1.value="";

}

}

</script>

<form name="f1" onsubmit="validate()">

<table>

<tr>

<td>Username :</td>

<td><input type="text" name="t1" size="20" /></td>

</tr>

<tr>

<td>Password :</td>

<td><input type="password" name="p1" size="20" /></td>

</tr>

<tr>

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

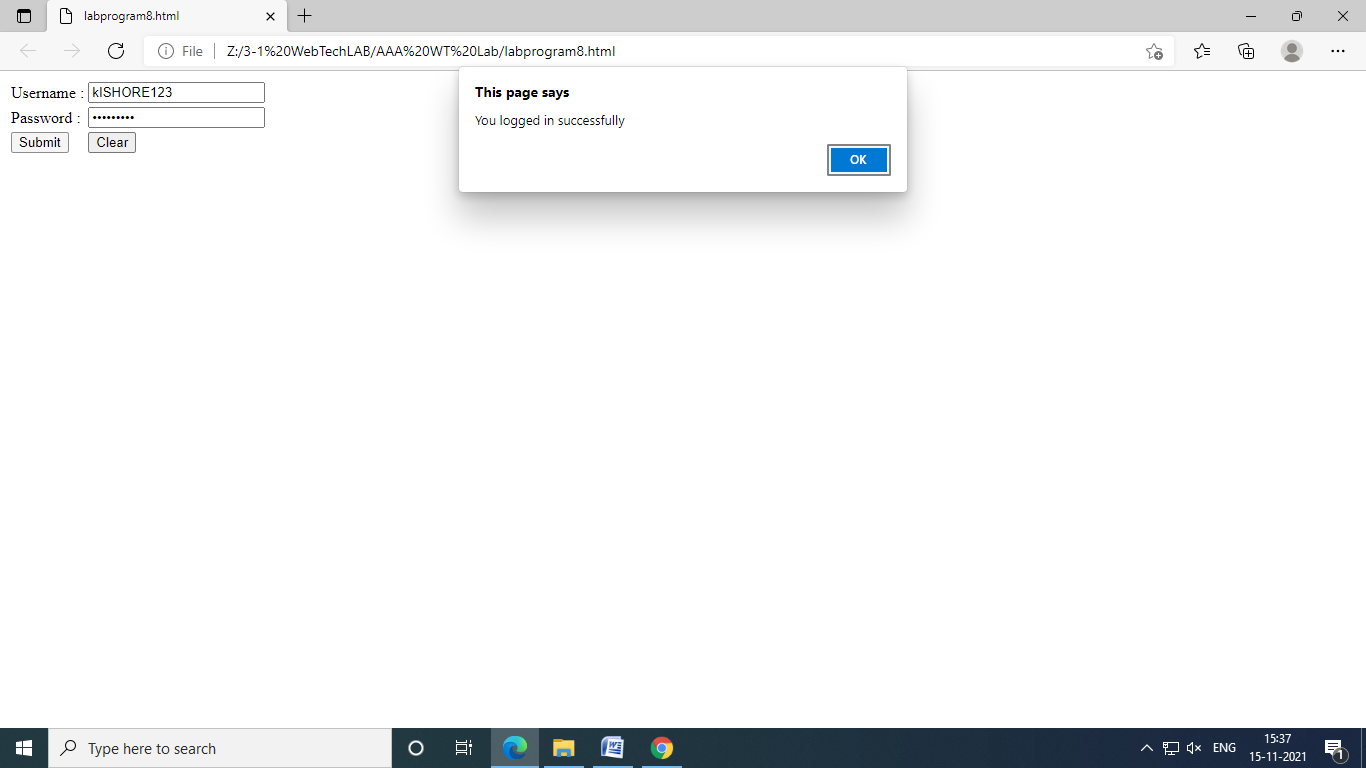
<td><input type="button" value="Clear" onclick="Clear()" /></td></tr>

</table>

</form>

</html>

OUTPUT:



**LAB PROGRAM-🡪9:**

1. Write a script to dynamically change the color and font size of the *textarea*. Select color & size using select box.

<html>

<head>

<title>color change</title>

</head>

<body>

<form>

color<select name="colors" id="t1">

<option value="green">green</option>

<option value="red">red</option>

<option value="blue">blue</option>

<option value="black">black</option>

</select><br><br>

size<select name="size" id="t2">

<option value="30%">30%</option>

<option value="40%">40%</option>

<option value="150%">150%</option>

</select><br><br>

<textarea name="message" rows="10" cols="10" id="t3">

KISHORE RVK

</textarea><br>

<br>

<button type="button" onclick="ok()">ok</button><br><br>

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

</form>

</body>

<script>

function ok()

{

x=document.getElementById("t3");

y=document.getElementById("t1");

z=document.getElementById("t2");

x.style.color=y.value;

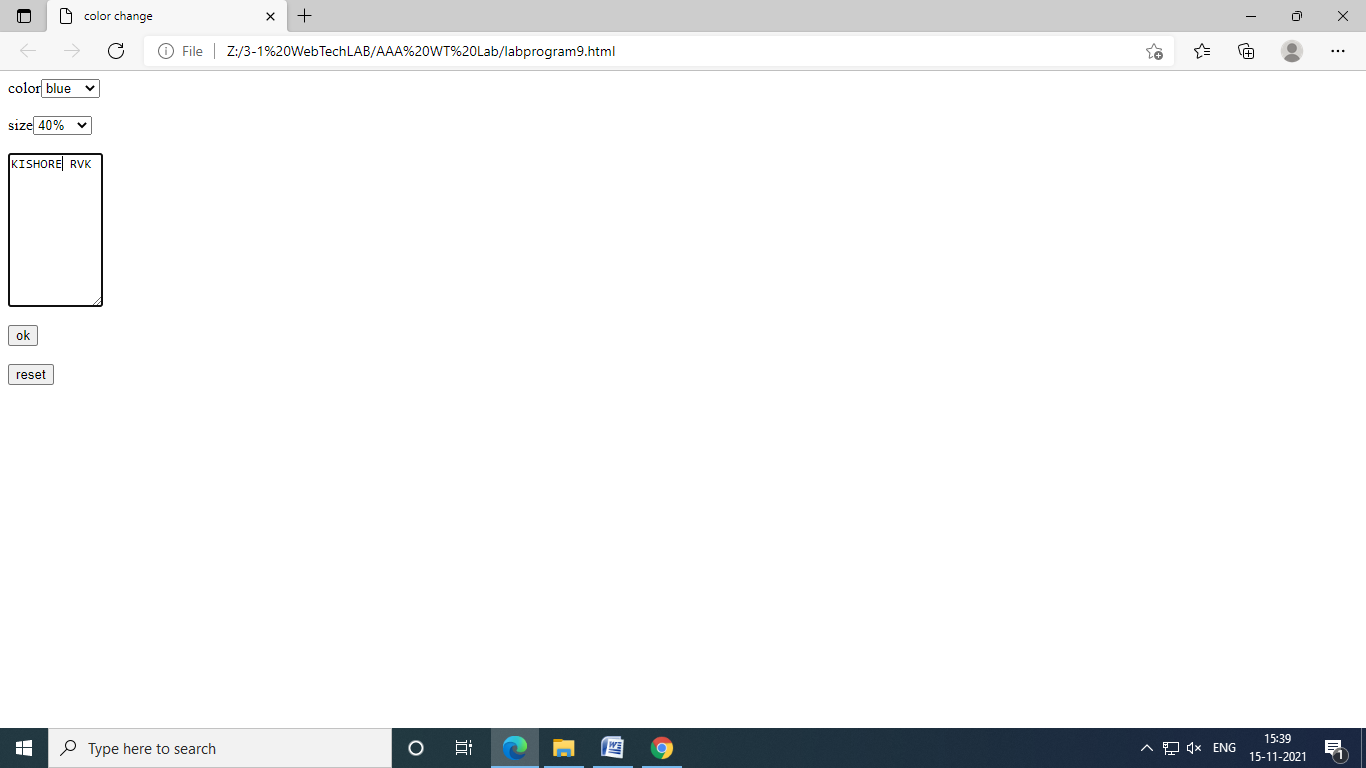
x.style.fontSize=z.value;

}

</script>

</html>

OUTPUT:



**LAB PROGRAM-🡪10:**

1. Write java script which uses random number generation to create sentences. Use four arrays of strings called **article**, **noun**, **verb**, and **preposition**. Create a sentence by selecting a word at random from each array in the following order.

**Article, noun, verb, preposition, article, and noun.**

As each word is picked, concatenate it to the previous words in the sentence. The words should be separated by spaces. When the final sentence is output, it should start with capital letter and end with period. The program should generate 20 sentences and output them to an HTML TEXTAREA.

The arrays should be filled as follows.

**Article** array contains “**the**”, “**a**”, “**one**”, “**some**”, and “**any**”.

**Noun** array contains “**boy**”, “**girl**”, “**dog**”, “**town**”, and “**car**”.

**Verb** array contains “**drove**”, “**jumped**”, “**ran**”, “**walked**”, and “**skipped**”.

**Preposition** array contains “**to**”, “**from**”, “**over**”, “**under**”, and “**on**”.

<html>

<head>

<title>

</title>

<body>

<form name="f1">

<textarea id="t1" rows="20" cols="35">

</textarea>

<input type="button" value="click me" onclick="generate()">

</form>

</body>

<script>

function generate()

{

var A=new Array("the","a","one","some");

var N=new Array("boy","girl","dog","town","car");

var V=new Array("drove","jumped","ran","walked","skipped");

var P=new Array("to","from","over","under","on");

var sent="";

var total="";

for(i=0;i<20;i++)

{

var w1=new String(A[Math.round(Math.random()\*4)]);

var w2=new String(N[Math.round(Math.random()\*4)]);

var w3=new String(V[Math.round(Math.random()\*4)]);

var w4=new String(P[Math.round(Math.random()\*4)]);

var w5=new String(A[Math.round(Math.random()\*4)]);

var w6=new String(N[Math.round(Math.random()\*4)]);

var l1=w1.substr(0,1)

var l2=w1.substr(1,3);

w1=l1.toUpperCase(l1)+l2

sent=w1+" "+w2+" "+w3+" "+w4+" "+w5+" "+w6+"."+"\n";

total=total+sent;

}

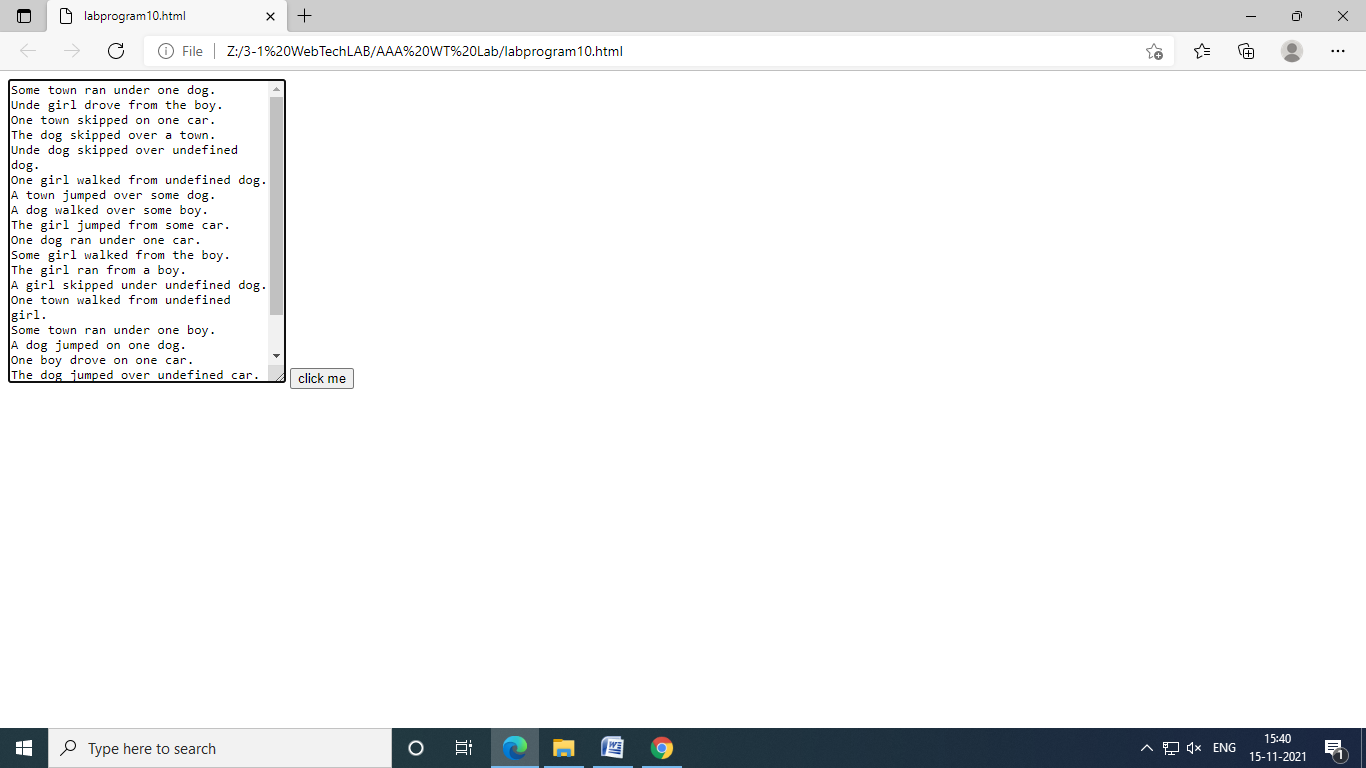
document.f1.t1.value=total;

}

</script>

</html>

OUTPUT:



**LAB PROGRAM-🡪11:**

1. Display all anchor elements in a given page into a XHTML table using DOM collections

<html>

<head>

<script>

function display()

{

var list=document.links;

varlen=list.length;

varlinkarr=new Array(len);

for(i=0;i<len;i++)

linkarr[i]=list[i].href;

document.writeln("<table border='3'>");

for(i=0;i<len;i++)

document.writeln("<tr><td>"+linkarr[i]+"</td></tr>");

document.writeln("</table>");

}

</script>

</head>

<body onload="display()">

<a href="www.gmail.com">gmail</a>

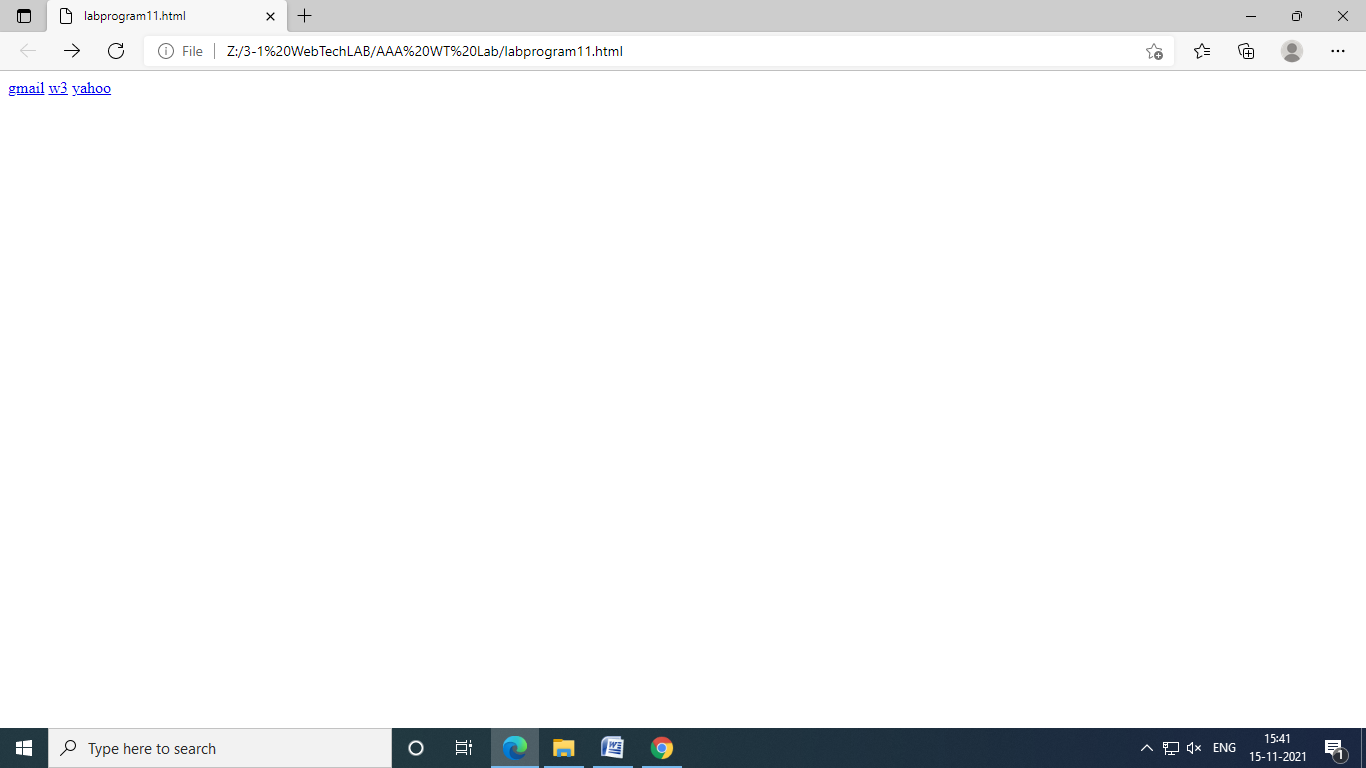
<a href="www.w3schools.com">w3</a>

<a href="www.yahoo.com">yahoo</a>

</body>

</html>

OUTPUT:

**

**LAB PROGRAM-🡪12:**

1. Display XML data in the XHTML table (Data Island).

<html>

<body>

<xml id="students">

<?xml version ="1.0"?>

<class>

<student>

<sno>123</sno>

<sname>abc</sname>

</student>

<student>

<sno>124</sno>

<sname>xyz</sname>

</student>

<student>

<sno>125</sno>

<sname>pqr</sname>

</student>

</class>

</xml>

<table datasrc="#students" border="2">

<tr>

<td><div DATAFLD="sno"></div></td>

<td><div DATAFLD="sname"></div></td>

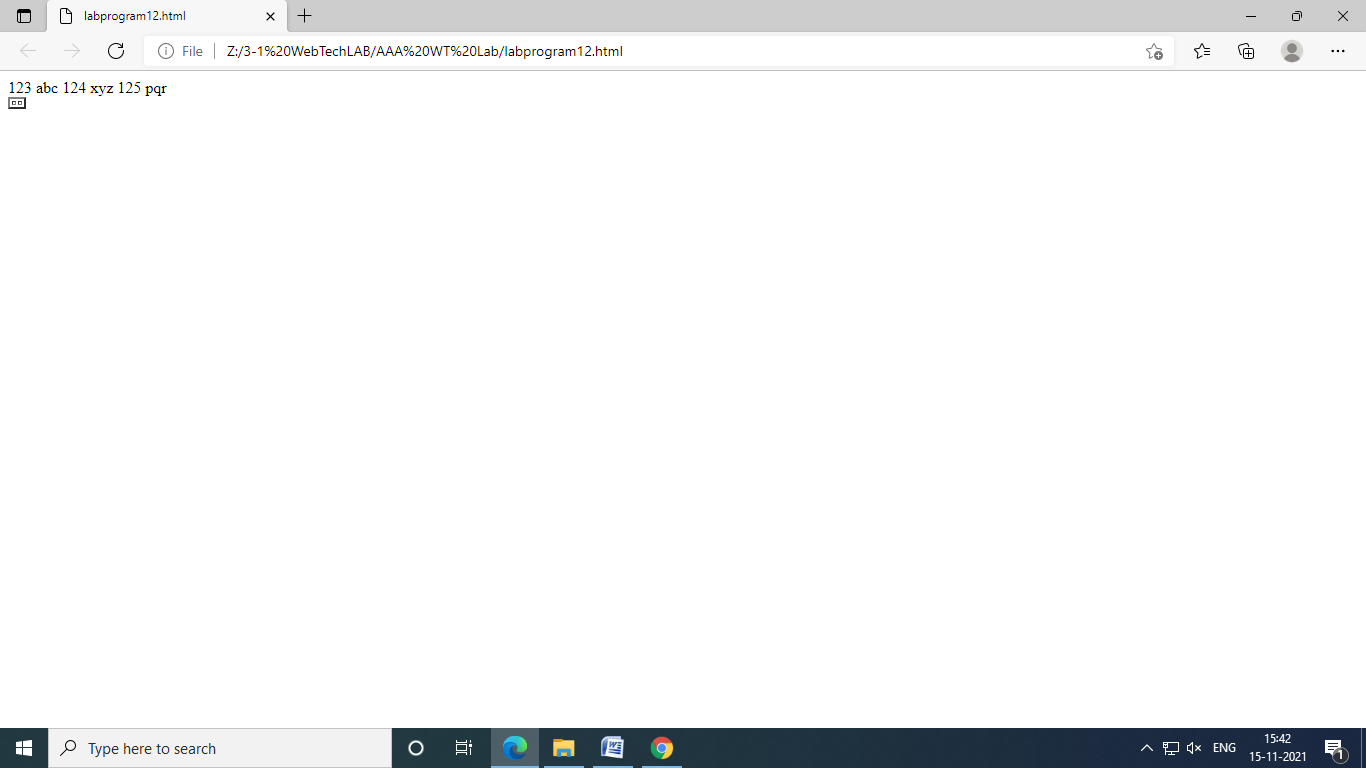
</tr>

</table>

</body>

</html>

OUTPUT:



**LAB PROGRAM-🡪13A:**

1. Validate XML documents using DTD & XML Schema (XSD).

<?xml version="1.0"?>

<!DOCTYPE student

[

<!ELEMENT student (sname,address,marks)>

<!ELEMENTsname (#PCDATA)>

<!ELEMENT address (#PCDATA)>

<!ELEMENT marks (#PCDATA)>

]>

<student>

<sname>ram</sname>

<address>gnt</address>

<marks>70</marks>

</student>

**LAB PROGRAM-🡪13B:**

<?xml version="1.0"?>

<student xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"xsi:noNamespaceSchemaLocation="student.xsd">

<sname>ram</sname>

<address>gnt</address>

<marks>70</marks>

</student>

LAB PROGRAM—13C:

<?xml version="1.0"?>

<xs:schemaxmlns="http://www.w3.org/2001/XMLSchema">

<xs:element name="student">

<xs:complex Type>

<xs:sequence>

<xs:element name="sname" type="xs:string">

<xs:element name="address" type="xs:string">

<xs:element name="marks" type="xs:integer">

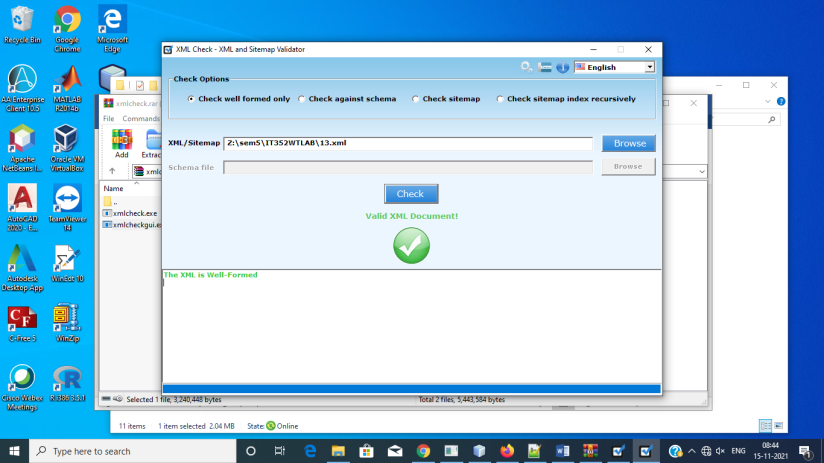
</xs:sequence>

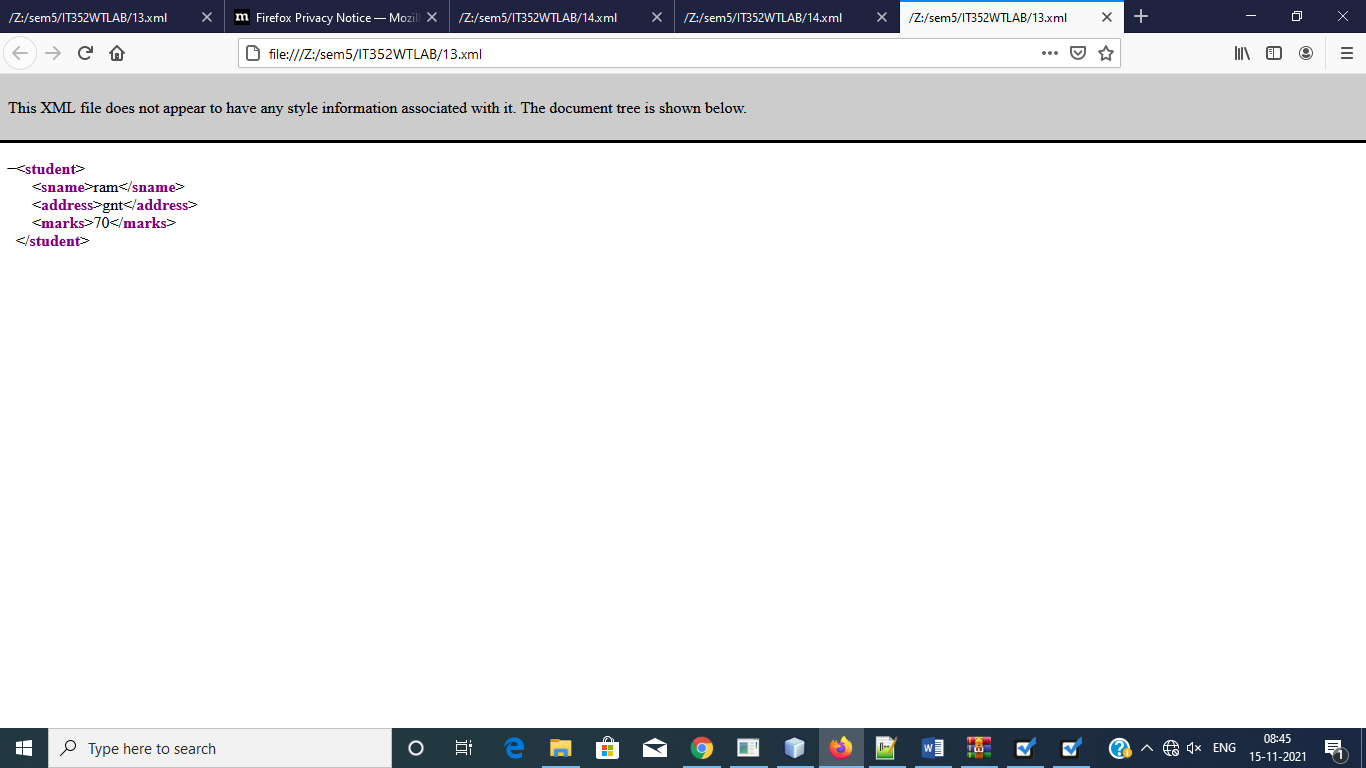
</xs:complex Type>

</xs:element>

</xs:schema>

OUTPUT:





**LAB PROGRAM-🡪14A:**

1. Display XML data in the XHTML table using XSL

<?xml version="1.0"?>

<?xml-stylesheet type="text/xsl" href="labprogram14b.xsl"?>

<stu\_info>

<stu Gender="m">

<name>abc</name>

<rollno>101</rollno>

<city>gnt</city>

</stu>

<stu Gender="f">

<name>xyz</name>

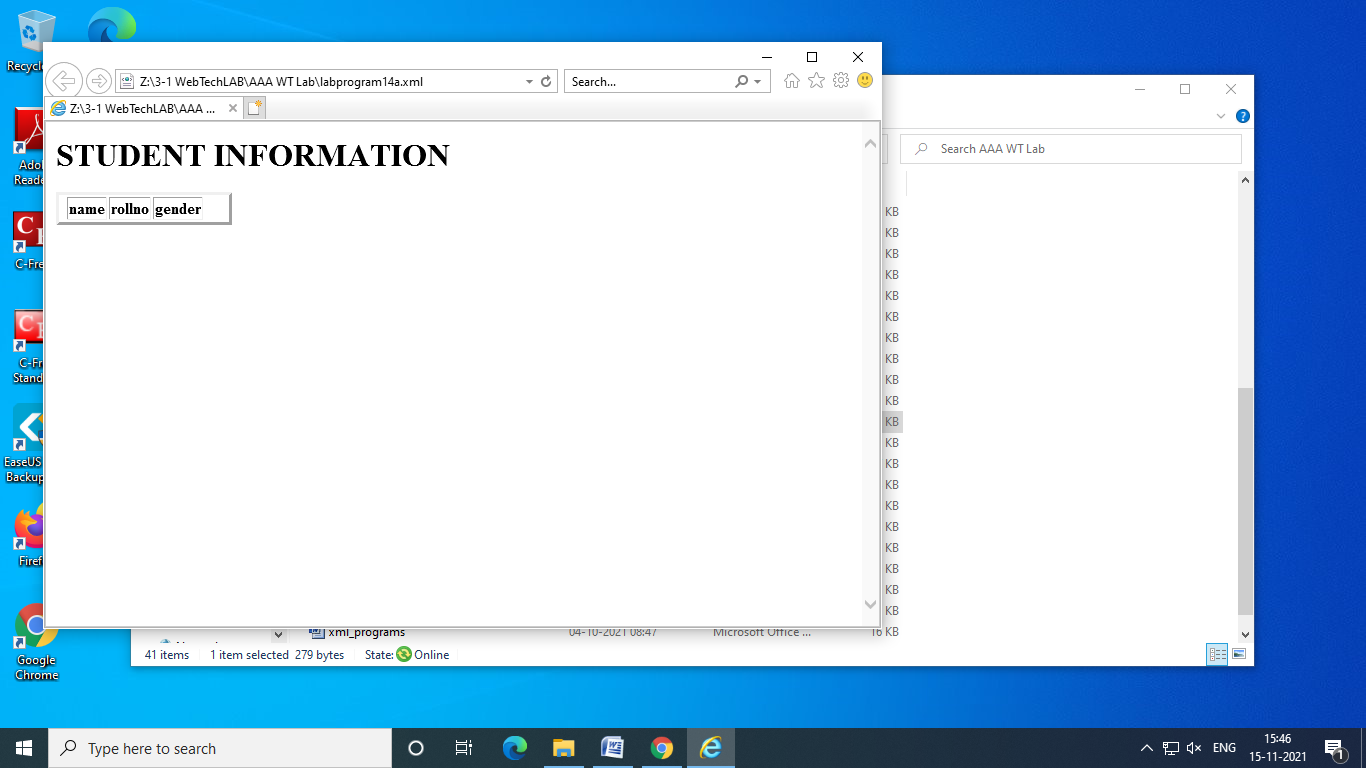
<rollno>102</rollno>

<city>hyd</city>

</stu>

</stu\_info>

OUTPUT:



**LAB PROGRAM-🡪14B:**

<?xml version="1.0"?>

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

<xsl:template match="/">

<html>

<body>

<h1>STUDENT INFORMATION</h1>

<table border="3">

<td>

<th>name</th>

<th>rollno</th>

<th>gender</th></td>

<xsl:for-each select="stu-info/stu">

<td>

<th><xsl:value-of select="name"/></th>

<th><xsl:value-of select="rollno"/></th>

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

</xsl:for-each>

</table>

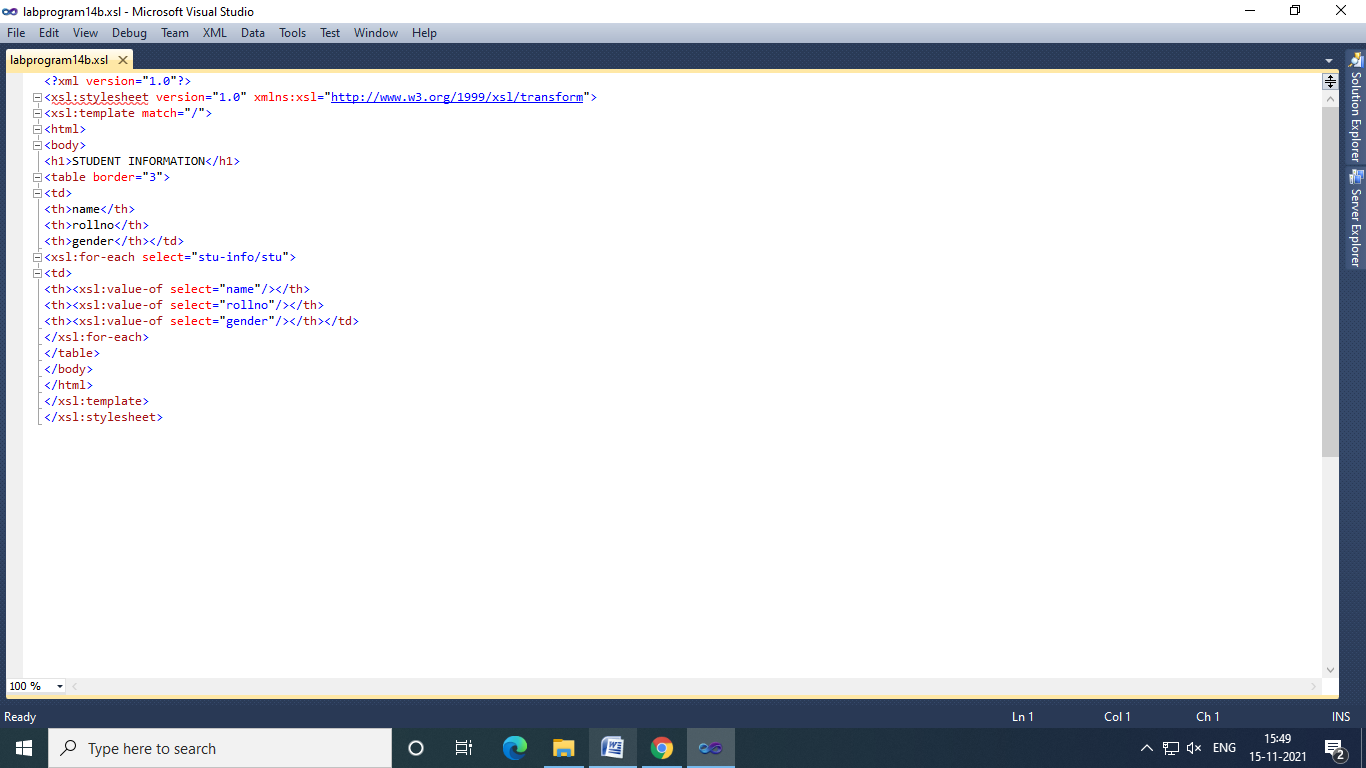
</body>

</html>

</xsl:template>

</xsl:stylesheet>

OUTPUT:



**LAB PROGRAMS-🡪15:**

1. Write a program to demonstrate Generic and HTTP servlets

GENERIC SERVLETS:

INDEX.JSP

<%--

Document : index

Created on : 15 Nov, 2021, 8:17:56 AM

Author : y19it15

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"

"http://www.w3.org/TR/html4/loose.dtd">

<html>

<body>

<center>

<form name="Form1" method="post" action="PostParameterServlet">

<table>

<tr>

<td><B>Employee</td>

<td><input type="textbox" name="ename"></td>

</tr>

<tr>

<td><B>Phone</td>

<td><input type="textbox" name="ephone"></td>

</tr>

</table>

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

</form>

</body>

</html>

PostParameterServlet.java

import java.io.IOException;

import java.io.PrintWriter;

import java.util.Enumeration;

import javax.servlet.GenericServlet;

import javax.servlet.ServletException;

import javax.servlet.ServletRequest;

import javax.servlet.ServletResponse;

/\*\*

\*

\* @author y19it15

\*/

public class PostParameterServlet extends GenericServlet{

public void service(ServletRequest request, ServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

PrintWriter pw = response.getWriter();

Enumeration e = request.getParameterNames();

try{

String ename = request.getParameter("ename");

String ephone = request.getParameter("ephone");

pw.print("Employee:"+ename);

pw.print("<br>phone number:"+ephone);

pw.close();

}finally{

pw.close();

}

}

}

WEB.XML:

<?xml version="1.0" encoding="UTF-8"?>

<web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd">

<servlet>

<servlet-name>PostParameterServlet</servlet-name>

<servlet-class>PostParameterServlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>PostParameterServlet</servlet-name>

<url-pattern>/PostParameterServlet</url-pattern>

</servlet-mapping>

<session-config>

<session-timeout>

30

</session-timeout>

</session-config>

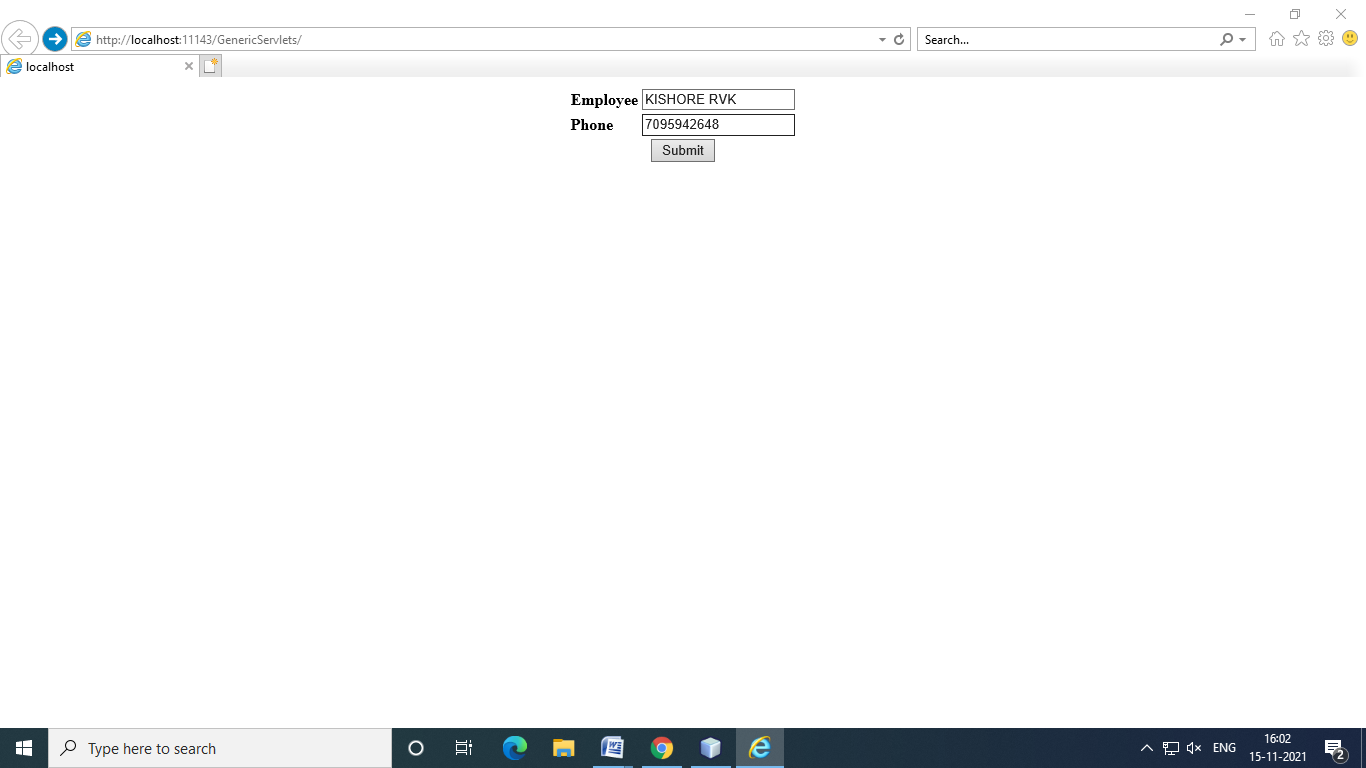
<welcome-file-list>

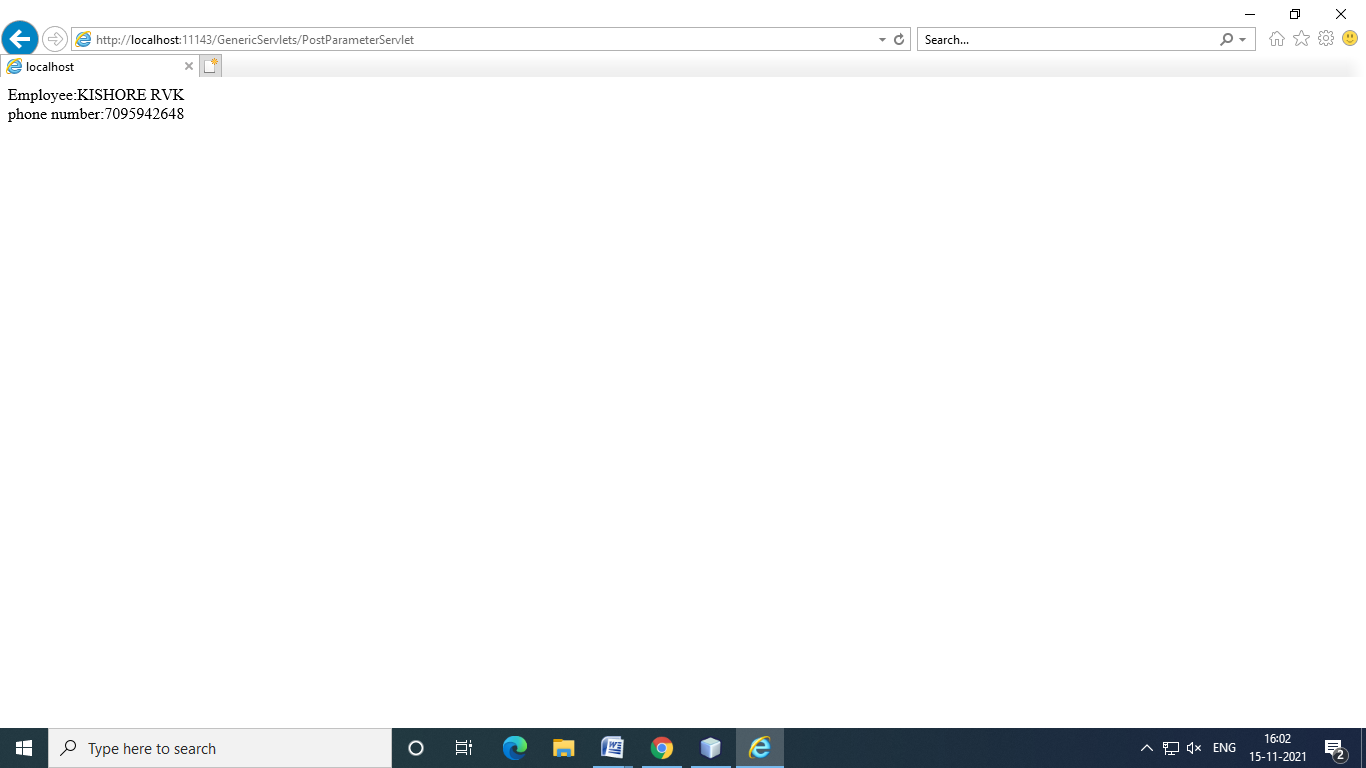
<welcome-file>index.jsp</welcome-file>

</welcome-file-list>

</web-app>

OUTPUT:





HTTP SERVLETS:

INDEX.JSP:

<%--

Document : index

Created on : 15 Nov, 2021, 9:25:32 AM

Author : y19it15

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"

"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Change Background color</title>

</head>

<body>

<center>

<form name="Form1" action="ColorGetServlet" method="get">

<B>Color:</B>

<select name="color" size="1">

<option value="Red">Red</option>

<option value="Green">Greeen</option>

<option value="Blue">Blue</option>

<option value="Pink">pink</option>

</select><br><br>

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

</form>

</center>

</body>

</html>

ColorGetServlet.java:

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

/\*\*

\*

\* @author y19it15

\*/

public class ColorGetServlet extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter();

String Color = null;

try {

String color = request.getParameter("color");

response.setContentType("text/html");

PrintWriter pw = response.getWriter();

String str ="<b style='color:"+color+";'>The selected color is: ";

pw.println(str);

pw.println(color);

} finally {

out.close();

}

}

}

Web.xml:

<?xml version="1.0" encoding="UTF-8"?>

<web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd">

<servlet>

<servlet-name>ColorGetServlet</servlet-name>

<servlet-class>ColorGetServlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>ColorGetServlet</servlet-name>

<url-pattern>/ColorGetServlet</url-pattern>

</servlet-mapping>

<session-config>

<session-timeout>

30

</session-timeout>

</session-config>

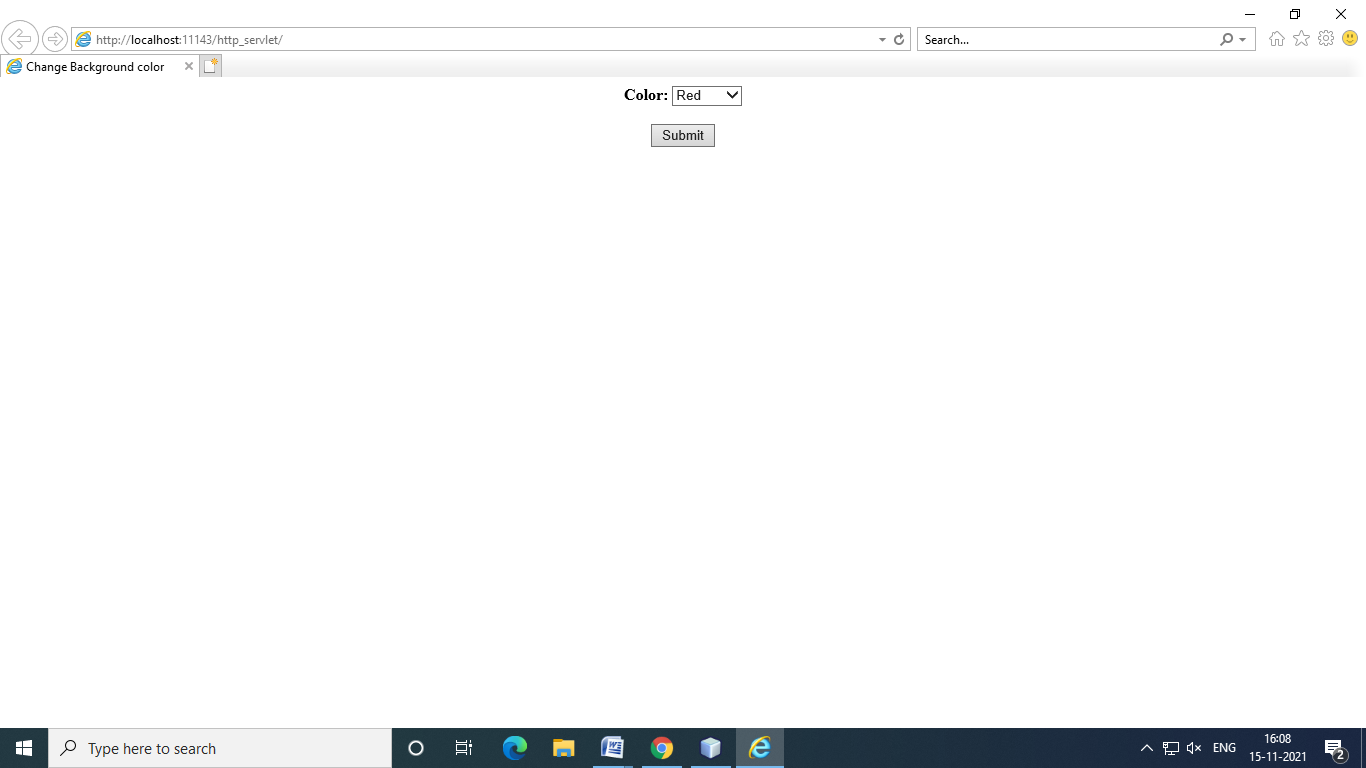
<welcome-file-list>

<welcome-file>index.jsp</welcome-file>

</welcome-file-list>

</web-app>

OUTPUT:

**LAB PROGRAMS-🡪16:**

1. Write a program to demonstrate cookies.

Index.jsp

<%--

Document : index

Created on : 9 Nov, 2021, 10:59:38 AM

Author : y19it15

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"

"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

<body>

<center>

<form name="form1" action="PostParameterServlet" method="post">

Name:<input type="text" name="ename"/><br>

Phone:<input type="text" name="ephone"/>

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

</form>

</center>

</body>

</html>

Servlet1.java

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.Cookie;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

/\*\*

\*

\* @author y19it15

\*/

public class servelt1 extends HttpServlet {

/\*\*

\* Processes requests for both HTTP <code>GET</code> and <code>POST</code> methods.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter();

try {

String n = request.getParameter("uname");

out.print("Welcome "+n);

Cookie ck = new Cookie("username",n);

response.addCookie(ck);

out.print("<form action='servlet2' method='post'>");

out.print("<input type='submit' value='go'>");

out.print("</form>");

} finally {

out.close();

}

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/\*\*

\* Handles the HTTP <code>GET</code> method.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Handles the HTTP <code>POST</code> method.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Returns a short description of the servlet.

\* @return a String containing servlet description

\*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

Servlet2.java

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.Cookie;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

/\*\*

\*

\* @author y19it15

\*/

public class servlet2 extends HttpServlet {

/\*\*

\* Processes requests for both HTTP <code>GET</code> and <code>POST</code> methods.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter();

try {

Cookie ck[] = request.getCookies();

out.print("Hello "+ck[0].getValue());

} finally {

out.close();

}

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/\*\*

\* Handles the HTTP <code>GET</code> method.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Handles the HTTP <code>POST</code> method.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Returns a short description of the servlet.

\* @return a String containing servlet description

\*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

Web.xml

<?xml version="1.0" encoding="UTF-8"?>

<web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd">

<servlet>

<servlet-name>servelt1</servlet-name>

<servlet-class>servelt1</servlet-class>

</servlet>

<servlet>

<servlet-name>servlet2</servlet-name>

<servlet-class>servlet2</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>servelt1</servlet-name>

<url-pattern>/servelt1</url-pattern>

</servlet-mapping>

<servlet-mapping>

<servlet-name>servlet2</servlet-name>

<url-pattern>/servlet2</url-pattern>

</servlet-mapping>

<session-config>

<session-timeout>

30

</session-timeout>

</session-config>

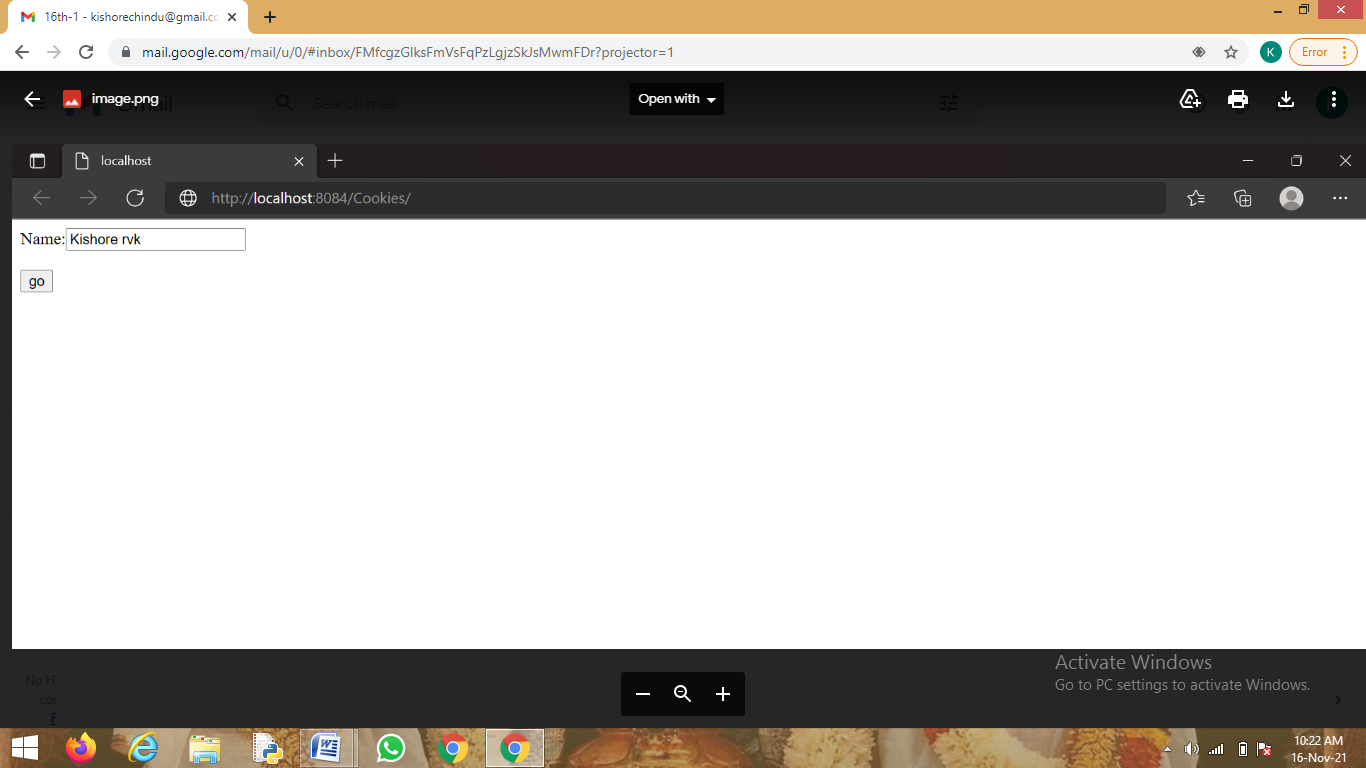
<welcome-file-list>

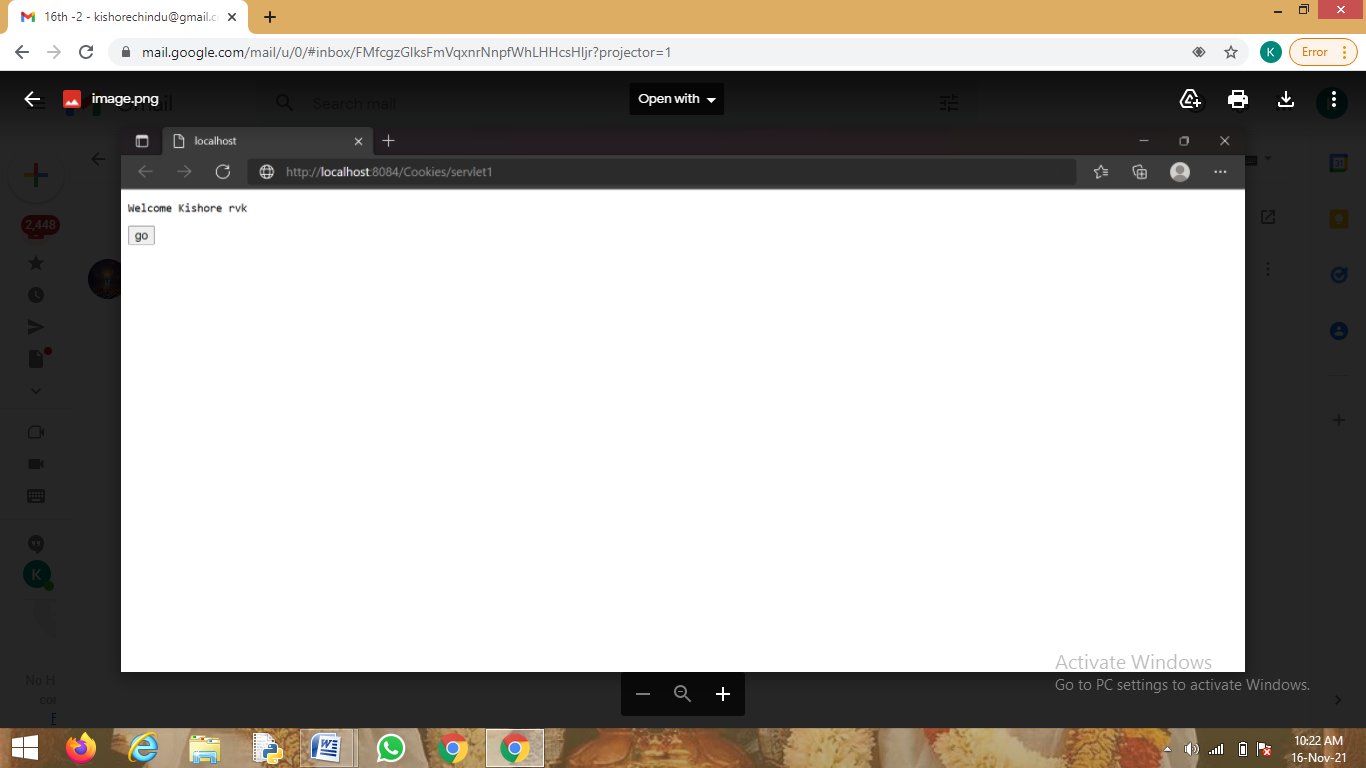
<welcome-file>index.jsp</welcome-file>

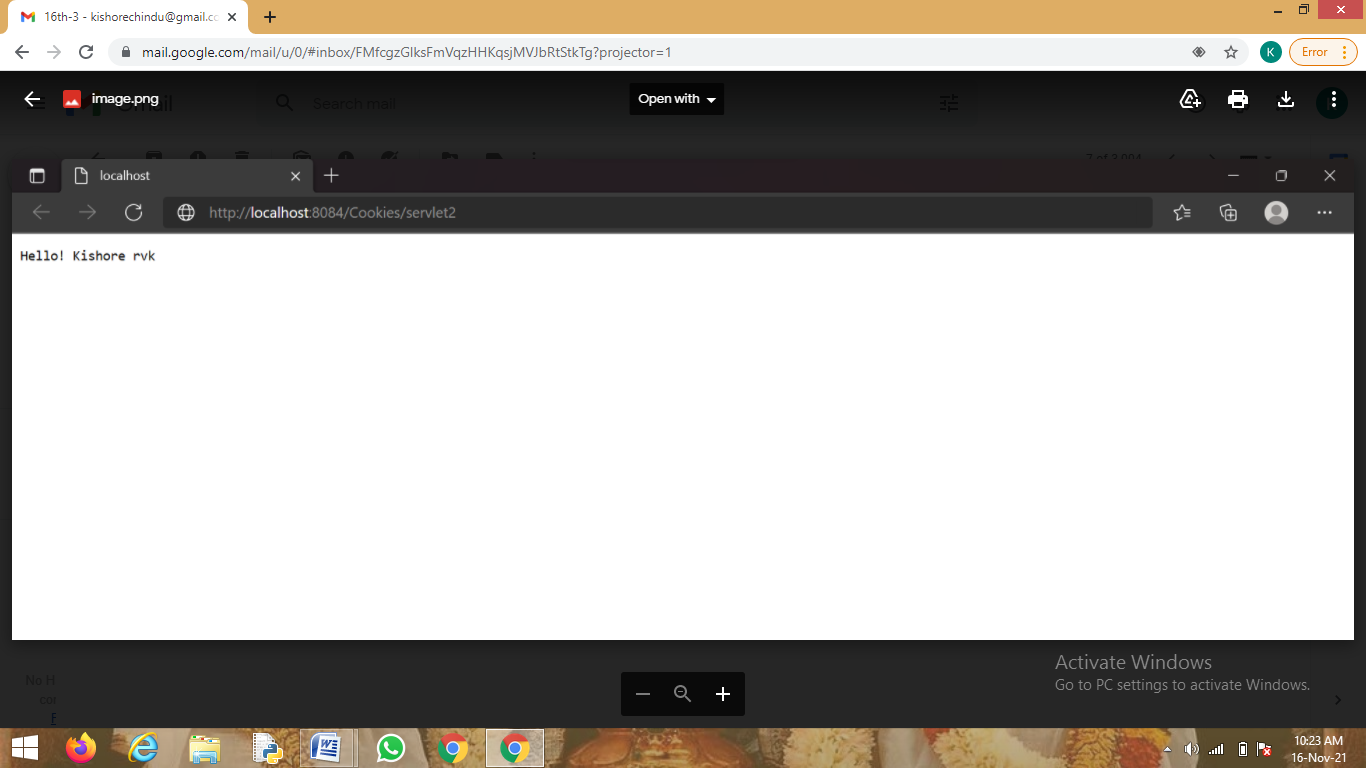
</welcome-file-list>

</web-app>

**OUTPUT:**







**LAB PROGRAM-🡪17:**

**17. Write a program to demonstrate Sessions using HTTP Servlets.**

Index.jsp

<%--

Document : index

Created on : 15 Nov, 2021, 3:29:08 PM

Author : y19it15

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"

"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

<body>

<form action="servelt1" method="get">

Name:<input type="text" palceholder="Enter Your name" name="uname">

<br> <input type="submit" value="go">

</form>

</body>

</html>

Servelt.java

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.Cookie;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

/\*\*

\*

\* @author y19it15

\*/

public class servelt1 extends HttpServlet {

/\*\*

\* Processes requests for both HTTP <code>GET</code> and <code>POST</code> methods.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter();

try {

String n = request.getParameter("uname");

out.print("Welcome "+n);

HttpSession session = request.getSession();

session.setAttribute("username", n);

out.print("<br><br><a href='servlet2'>Visit</a>");

} finally {

out.close();

}

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/\*\*

\* Handles the HTTP <code>GET</code> method.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Handles the HTTP <code>POST</code> method.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Returns a short description of the servlet.

\* @return a String containing servlet description

\*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

Servlet.2.java

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

/\*\*

\*

\* @author y19it15

\*/

public class servlet2 extends HttpServlet {

/\*\*

\* Processes requests for both HTTP <code>GET</code> and <code>POST</code> methods.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter();

try {

HttpSession session = request.getSession(false);

String s = (String)session.getAttribute("username");

out.print("Hello! "+s);

} finally {

out.close();

}

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/\*\*

\* Handles the HTTP <code>GET</code> method.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Handles the HTTP <code>POST</code> method.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Returns a short description of the servlet.

\* @return a String containing servlet description

\*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

Web.xml

<?xml version="1.0" encoding="UTF-8"?>

<web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd">

<servlet>

<servlet-name>servelt1</servlet-name>

<servlet-class>servelt1</servlet-class>

</servlet>

<servlet>

<servlet-name>servlet2</servlet-name>

<servlet-class>servlet2</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>servelt1</servlet-name>

<url-pattern>/servelt1</url-pattern>

</servlet-mapping>

<servlet-mapping>

<servlet-name>servlet2</servlet-name>

<url-pattern>/servlet2</url-pattern>

</servlet-mapping>

<session-config>

<session-timeout>

30

</session-timeout>

</session-config>

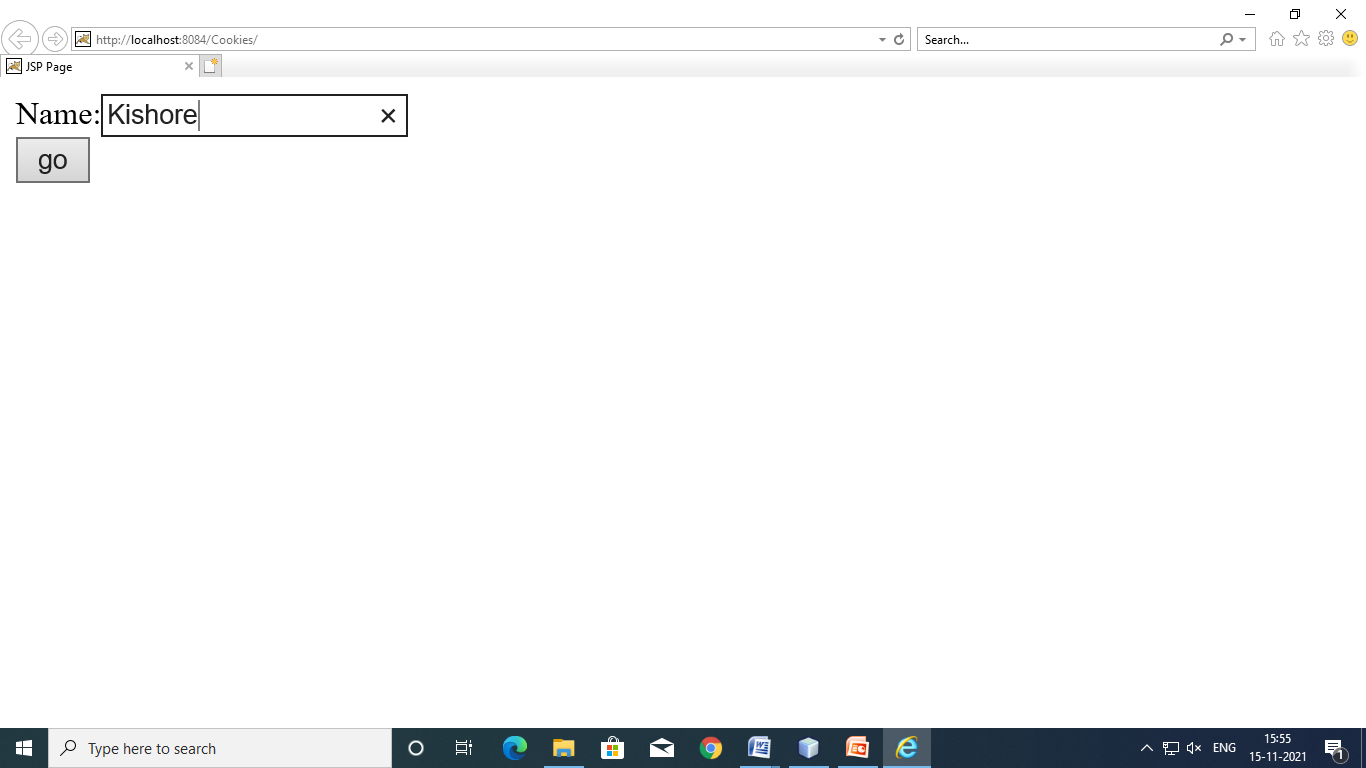
<welcome-file-list>

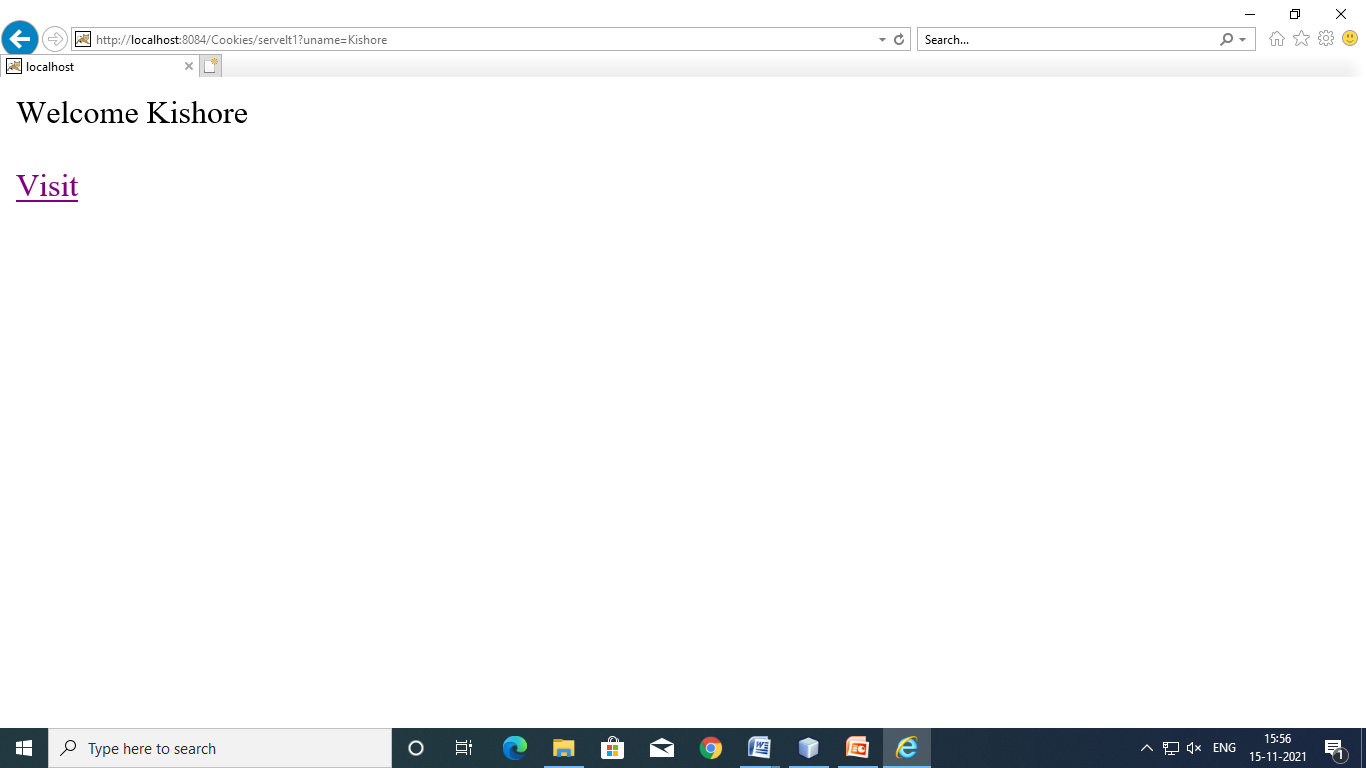
<welcome-file>index.jsp</welcome-file>

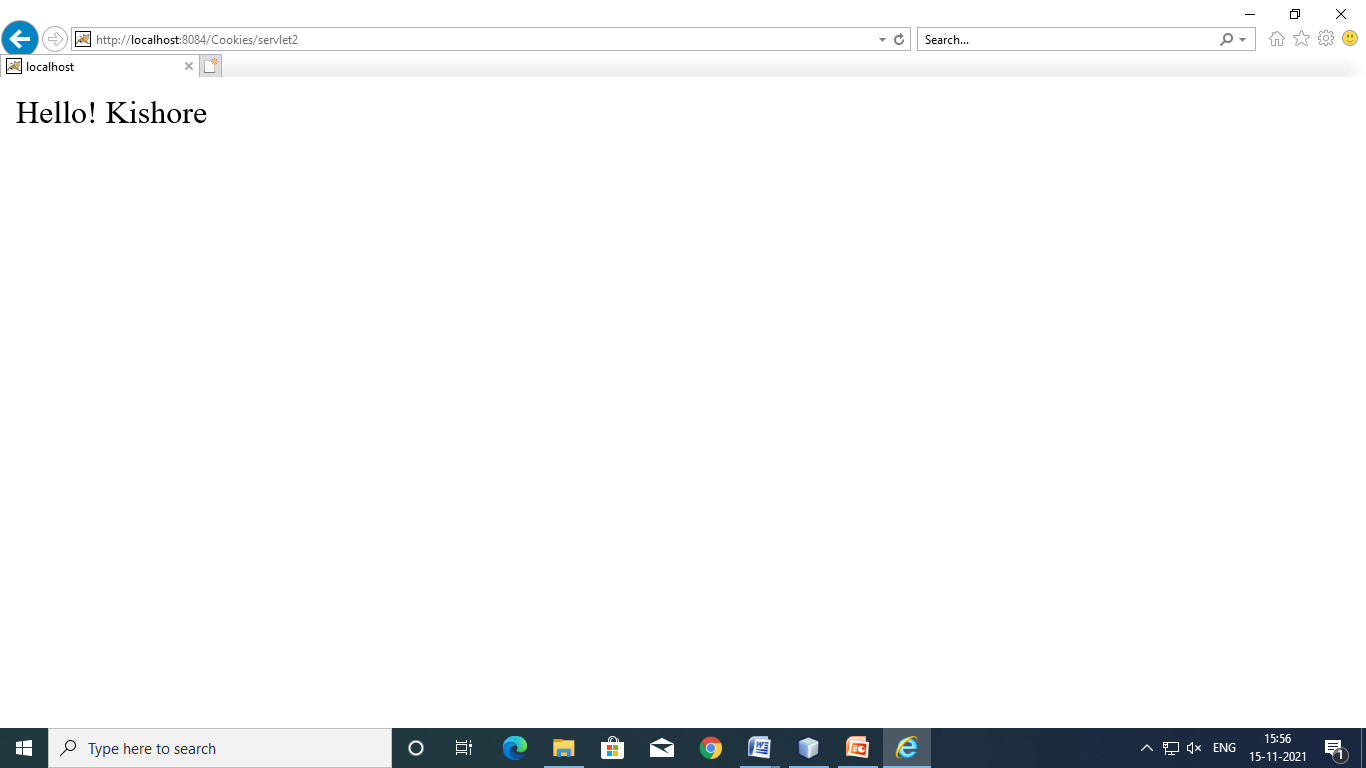
</welcome-file-list>

</web-app>

**OUTPUT:**







**LAB PROGRAM-🡪18:**

18.Create a Java web application to validate user login details by implementing the following.

* Create one JSP (ex: login.jsp ) in which you will submit your “user name” and “password”.
* And create one servlet (ex: ControllerServlet) that will check if the password entered by the user is correct or not. If the username & password entered by the user is correct then your controller servlet will redirect the client request to another servlet (ex: ValidUserServlet) which will display welcome message. If the password entered by the user is wrong then the request will be forwarded to your login.JSP page (Use Send Redirect in servlet).

Code

Index.jsp

<%--

Document : index

Created on : 15 Nov, 2021, 4:00:30 PM

Author : y19it15

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"

"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Login Page</title>

</head>

<body>

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

User Name:<input type="text" palceholder="enter your name" name="uname"><br>

Password:<input type="password" name="upass"><br>

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

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

</form>

</body>

</html>

LoginServlet.java

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

/\*\*

\*

\* @author y19it15

\*/

public class LoginServlet extends HttpServlet {

/\*\*

\* Processes requests for both HTTP <code>GET</code> and <code>POST</code> methods.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter();

try {

String name=request.getParameter("uname");

String pass=request.getParameter("upass");

if(pass.equals("rvrjcit"))

{

out.print("Welcome "+name);

}

else

{

out.print("<center style='color:red;'>Sorry <br> Please enter correct password</center>");

request.getRequestDispatcher("index.jsp").include(request, response);

}

} finally {

out.close();

}

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/\*\*

\* Handles the HTTP <code>GET</code> method.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Handles the HTTP <code>POST</code> method.

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/\*\*

\* Returns a short description of the servlet.

\* @return a String containing servlet description

\*/

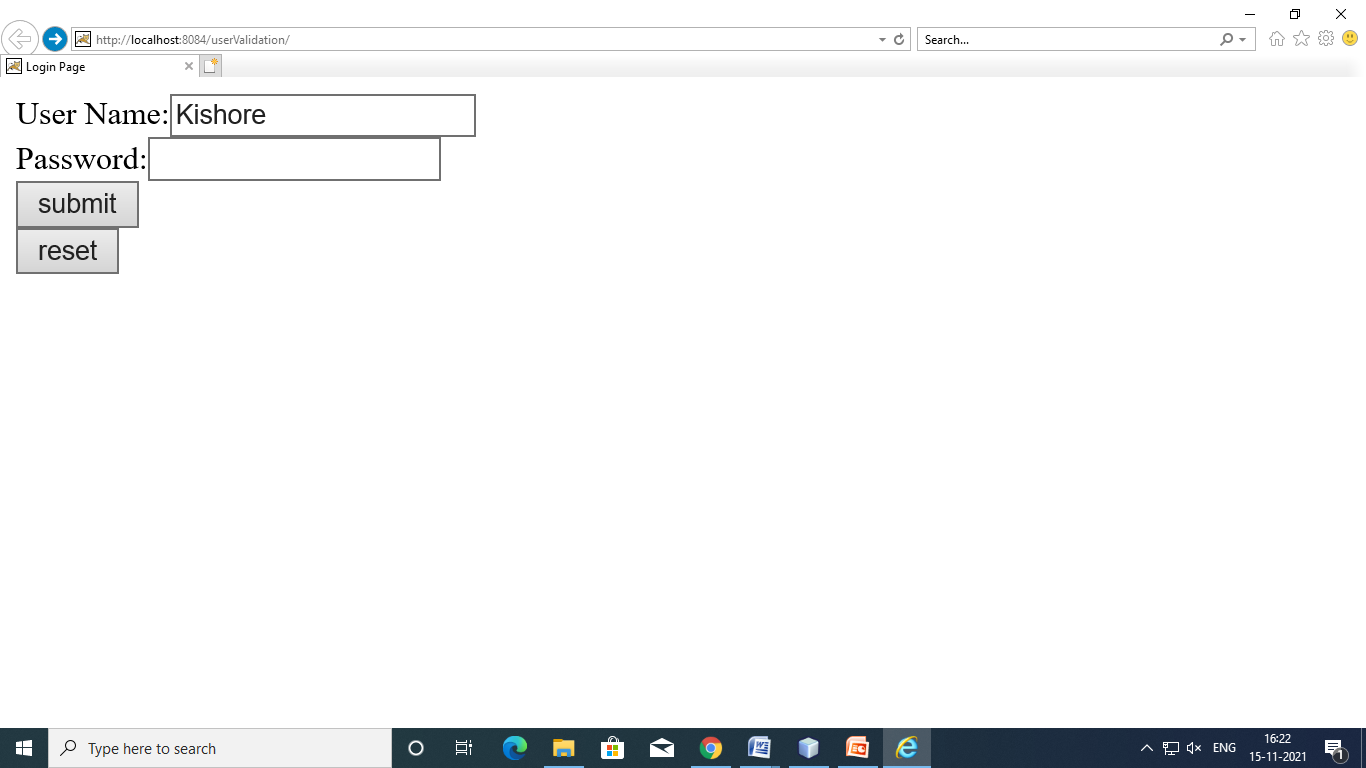
@Override

public String getServletInfo() {

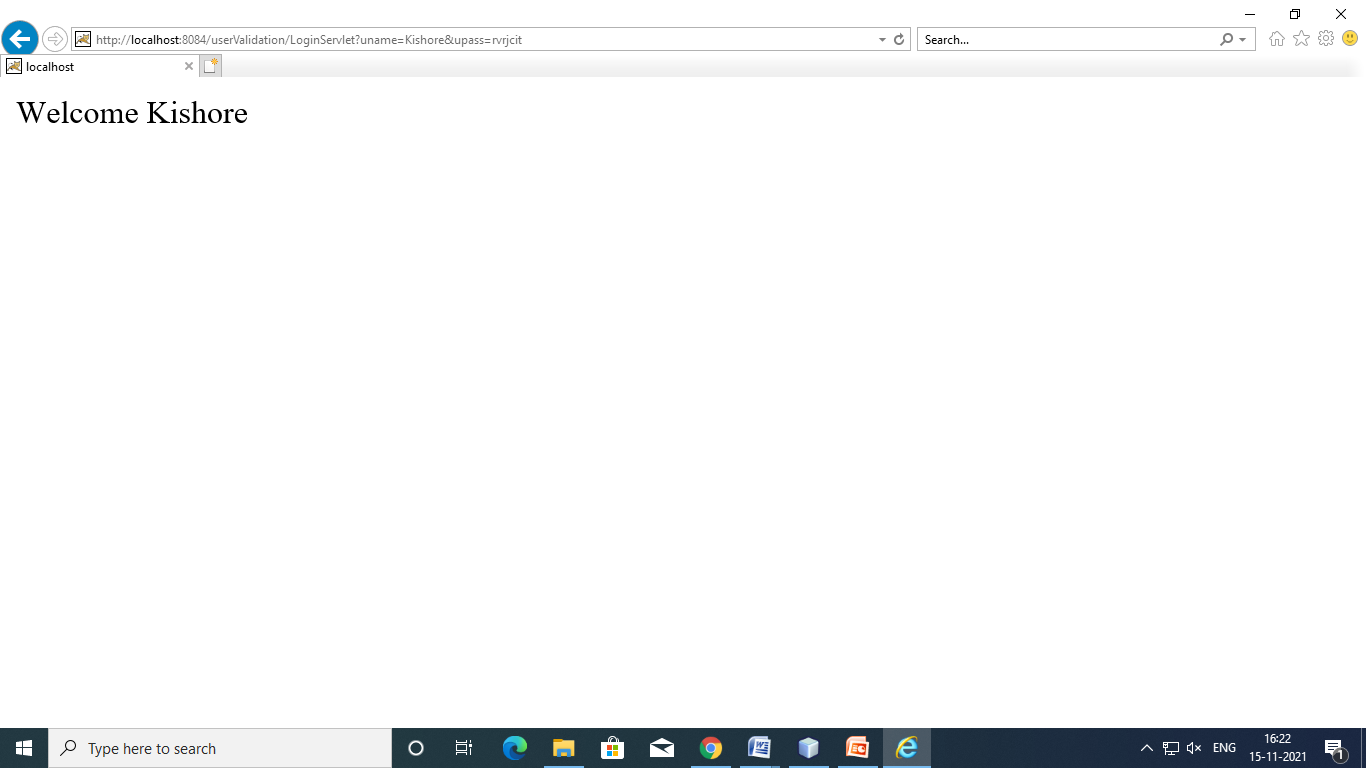
return "Short description";

}// </editor-fold>

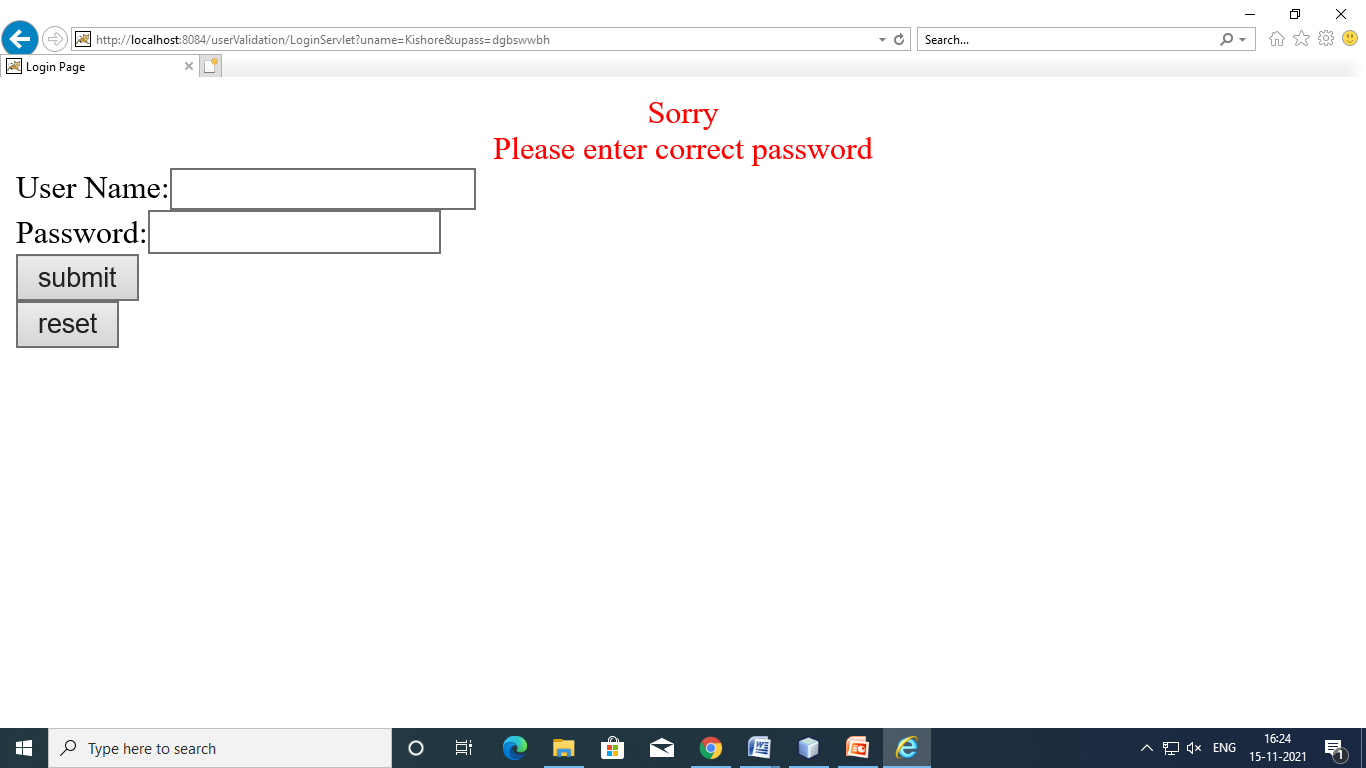
}

**OUTPUT:**

On Successful Validation



Unsuccessful Validation



**LAB PROGRAM-🡪19:**

**19. Make a “Course registration” form using JSP, that collects a first name, last name, contact no, email address & course name.**

* **Create 4 text boxes for first name, last name, contact no, email address.**
* **Use 3 check boxes for multiple selection.**
* **Send the registration information to a servlet that displays it. (use HTTP POST method)**

Code

Index.jsp

<%--

Document : index

Created on : 15 Nov, 2021, 10:26:14 AM

Author : y19it15

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"

"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

<body>

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

fname:<input type="text" name="fname"><br><br>

lname:<input type="text" name="lname"><br><br>

contact:<input type="number" name="contact"><br><br>

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

select courses

<input type="checkbox" name="rvr" value="os">os<br><br>

<input type="checkbox" name="rvr" value="ds">ds<br><br>

<input type="checkbox" name="rvr" value="oops">c++<br><br>

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

</form>

</body>

</html>

**REGISTER.JAVA**

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.\*;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

public class Register extends HttpServlet {

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter();

try {

String f=request.getParameter("fname");

String l=request.getParameter("lname");

String c=request.getParameter("contact");

String e=request.getParameter("email");

String a[]=request.getParameterValues("rvr");

out.println(f);

out.println(l);

out.println(c);

out.println(e);

for(inti=0;i<a.length;i++)

{

out.println(a[i]);

}

} finally {

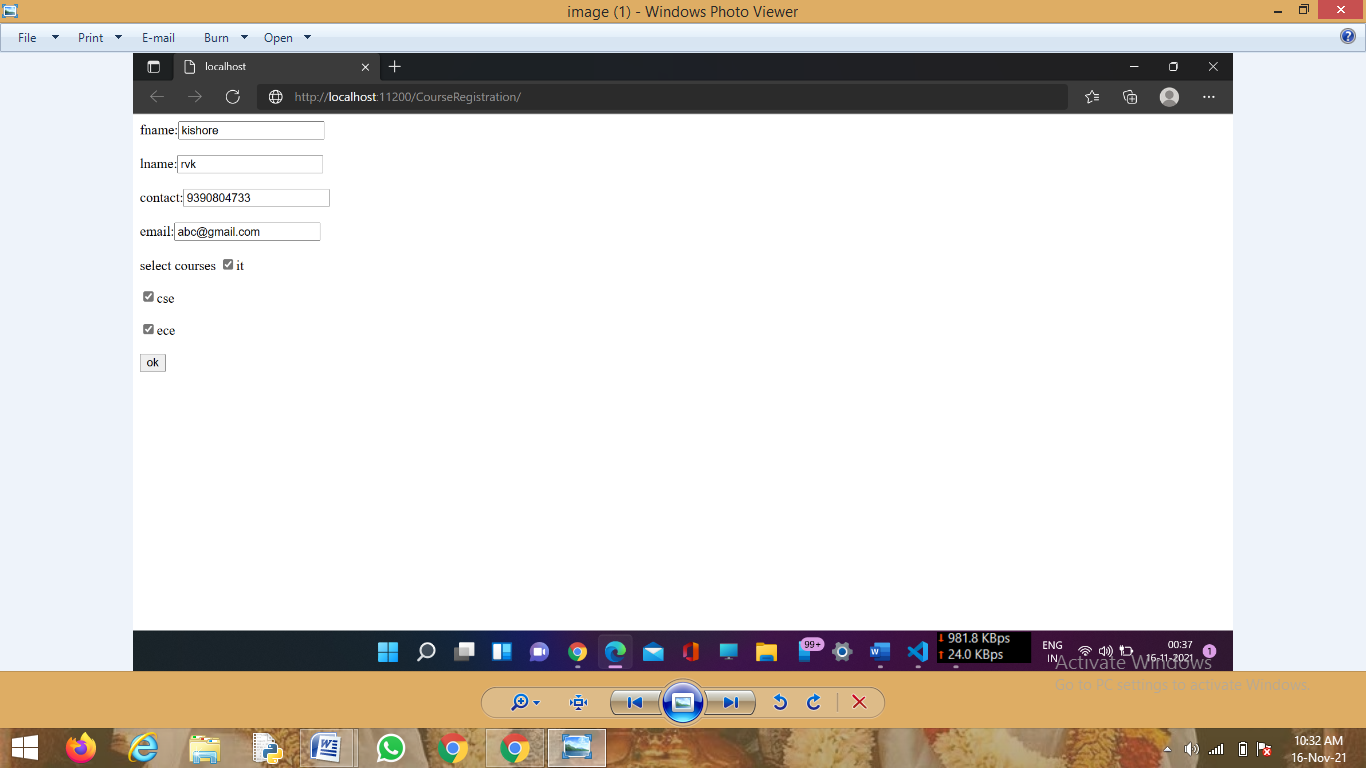
out.close();

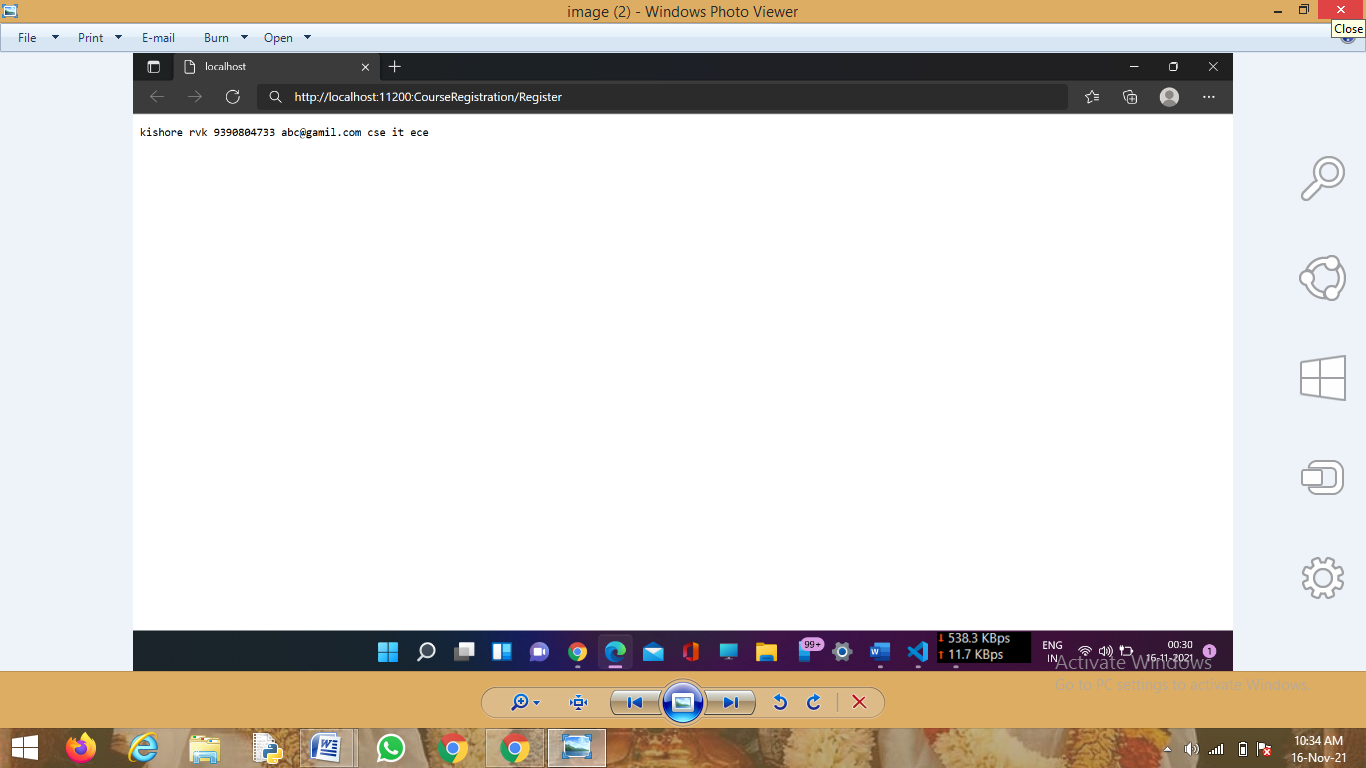
}

}

}

**OUTPUT**:





**20.Create a bean to maintain student information and display the information in JSP page**

Code

INSERT.JSP

<%--

Document : INSERT

Created on : 15 Nov, 2021, 4:00:30 PM

Author : y19it15

-%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

<body>

<%

String u=request.getParameter("name");

String i=request.getParameter("id");

String p=request.getParameter("per");

String a=request.getParameter("addr");

%>

<jsp:useBean id="std" class="p1.Student" scope="session">

<jsp:setProperty name="std" property="stdname" value="<%=u%>"/>

<jsp:setProperty name="std" property="stdid" value="<%=i%>"/>

<jsp:setProperty name="std" property="stdper" value="<%=p%>"/>

<jsp:setProperty name="std" property="stdaddr" value="<%=a%>"/>

<jsp:getProperty name="std" property="stdname"/>

<jsp:getProperty name="std" property="stdid"/>

<jsp:getProperty name="std" property="stdper"/>

<jsp:getProperty name="std" property="stdaddr"/>

</jsp:useBean>

</body>

</html>

STD.JSP

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

<body>

<h1>Hello World!</h1>

<form method="post" action="insert.jsp">

<h1>User login page</h1>

name<input type="text" name="name"><br>

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

percentage<input type="text" name="per"><br>

address<input type="text" name="addr"><br>

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

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

</form>

</body>

</html>

STUDENT.JAVA

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package p1;

/\*\*

\*

\* @author :y19it15

\*/

public class Student {

public String stdname;

public String stdid;

public String stdper;

public String stdaddr;

public Student()

{

}

public void setStdname(String n)

{

stdname=n;

}

public void setStdid(String i)

{

stdid=i;

}

public void setStdper(String p)

{

stdper=p;

}

public void setStdaddr(String a)

{

stdaddr=a;

}

public String getStdname()

{

return stdname;

}

public String getStdid()

{

return stdid;

}

public String getStdper()

{

return stdper;

}

public String getStdaddr()

{

return stdaddr;

}

}

**OUTPUT:**

