

Ex No: 1	CREATING A WEBPAGE WITH IMAGE MAPS AND HOTSPOTS
Date:	

AIM:

To create a webpage with the following using HTML

- To embed an image in web page
- To fix the hot spots.
- Show all the related information when a hot spot is clicked in the map

ALGORITHM:

- Create a html file with map tag.
- Set the source attribute of the img tag to the location of the image and also set the use map attribute.
- Specify an area with name, shape and href set of the appropriate value.
- Repeat step3 as many hot spots you want to put in the map.
- Create html file for each and every hot spots the user will select the particular location it shows information about it.

PROGRAM:

```

/*mapping.html*/
<html>
<head>
<title>India Map</title>
</head>
<body bgcolor="PINK">
<font face="Monotype Corsiva" color="BLUE" size="6">
<marquee direction="left" behavior="alternate">INDIA MAP
</marquee>
</font>
<hr size="6" color="RED">
<map name="pagemap">
<area shape="rect" coords="194,151,247,219" href="map1.html">
<area shape="rect" coords="291,268,384,337" href="map2.html">
<area shape="rect" coords="100,337,197,384" href="map3.html">
<area shape="rect" coords="236,543,344,577" href="map4.html">

```

```

</map>

</body>
<font color="#ff0000" size="5">
<p><b>Hints:</b><i>Click on the Name of the Cities in the map to know its
description</i></p>
</html>

```

/*map1.html*/

```

<html>
<body bgcolor="SKYBLUE">
<font face="Monotype Corsiva" size="18" color="RED">
<center><b><i><tt>Delhi is the capital of our INDIA<br> and <br>More IT companies are
Camped at Delhi</tt></i></b></center>
<a href="mapping.html">Home Page</a>
</font>
</body>
</html>

```

/*map2.html*/

```

<html>
<body bgcolor="SKYBLUE">
<font face="Times New Roman" size="18" color="RED">
<center><b><i>Calcutta is the wealthy city in WEST BENGAL<br> and <br>it has Famous
"Sunderbans Forests"</i></b></center>
<a href="mapping.html">Home Page</a>
</font></body></html>

```

/*map3.html*/

```

<html>
<body bgcolor="DARKGREEN">
<font face="Times New Roman" size="12" color="RED">
<center>MUMBAI is the capital of Maharashtra<br> and <br>it has Famous India
Gate</center>
<a href="mapping.html">Home Page</a>
</font>
</body>
</html>

```

/*map4.html*/

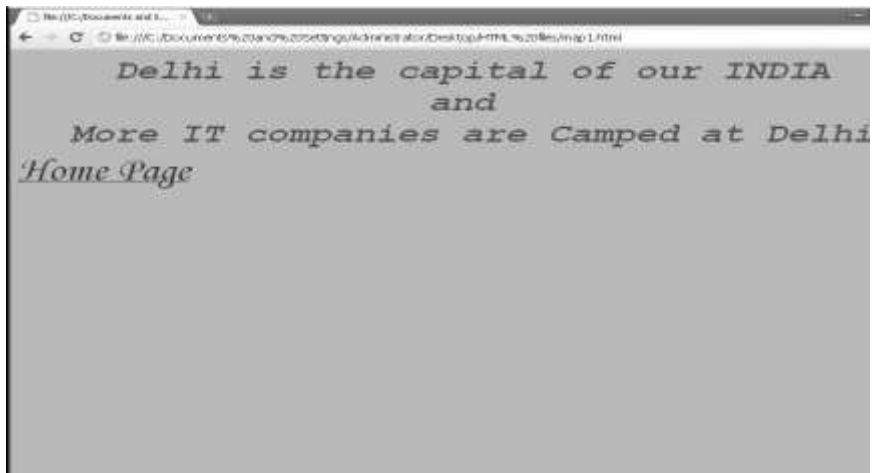
```

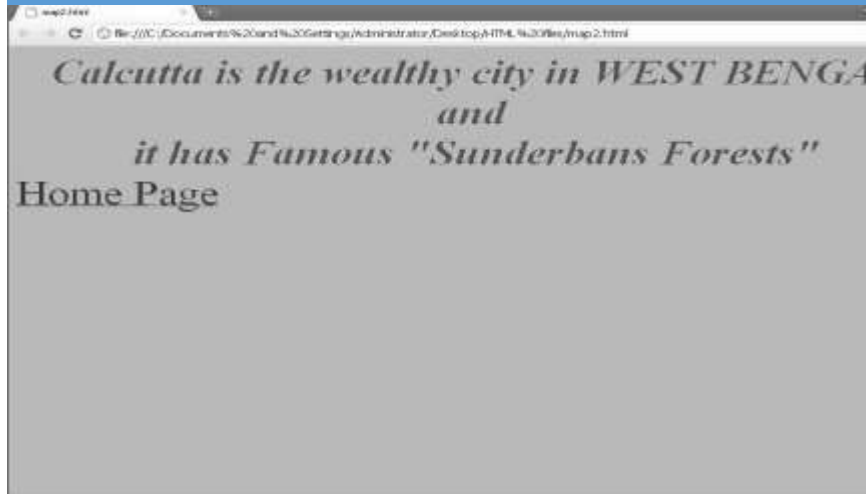
<html>
<body bgcolor="BLACK">
<font face="Times New Roman" size="12" color="RED">
<center>Chennai is hte capital of Tamil Nadu<br> and <br>More IT companies are camped at
Chennai</center>
<a href="mapping.html">Home Page</a>

```

</body></html>

OUTPUT:





RESULT: Thus creation of an webpage using cascading style sheet has been developed successfully.

Ex No:2	CREATING A WEBPAGE WITH CASCADING STYLE SHEET
Date:	

AIM:

To create a webpage with the following using html to embedded the style sheet

ALGORITHM:

Step1: Create html file with the style tag, inside head tag.

Step2: Set the style such as font-family, font-size, color, left etc, for the heading
h1,h2,...h6 and respectively.

Step3: Close the head tag.

Step4: Specify the heading and information required inside the body tag.

Step5: Close the opened tag.

PROGRAM:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
TRANSITION//EN" http://www.w3.org/TR/html1/DTD/html1.1.dtd>
```

```
<html xmlns="http://www.w3org/1999/xhtml">
```

```
<head>
```

```
<title>Embedded style sheet</title>
```

```
<style type="text/css">
```

```
h1
```

```
{  
font-family:arial;  
color:green;  
}  
h2  
{  
font-family:arial;  
color:red;  
left:20px  
}  
h3  
{  
font-family:arial;  
color:blue;  
}  
p  
{  
font-size:14pt;  
font-family:verdana  
}  
</style>  
</head>  
<body>  
<h1>
```

<center>This is created using embedded style sheet

</center>

</h1>

<h2>This line is alligned left and red colored;

</h2>

<p>

The embedded style sheet is the most commonly used style sheet

This paragrah is return in verdana font with font size of 14.

</p>

<h3>

This is a blue colored line

</h3>

</body>

</html>

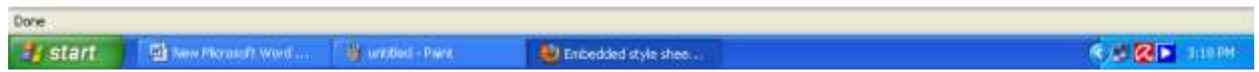
OUTPUT:



EMBEDDED STYLE SHEET

Computer science and Engineering

It is the best department.It has several lab facilities.



RESULT: Thus creation of an webpage using cascading style sheet has been developed successfully.

Ex No:3	CLIENT SIDE SCRIPTS IN DHTML
Date:	

AIM:

To develop a program for validating web form control using DHTML.

ALGORITHM:

Step1:Start the program.

Step2: Define the title within the tag.

Step3: Give the script type within the script tag.

Step4: Validate each and every column as the box with the if condition.

Step5: If empty value are given or the block term are next then it is verified with certain condition.

Step6:If values is empty then a message is been displayed.

Step7: Form is designed with GUI tool is description.

Step8: All buttons are processed accordingly.

Step9: Stop the program.

PROGRAM:**//Webforms.html**

```

<html>
<head>
<script type='text/javascript'>
function formValidator()
{
var firstname=document.getElementById('firstname');
var lastname=document.getElementById('lastname');
var addr=document.getElementById('addr');
var zip=document.getElementById('zip');
var Countries=document.getElementById('Countries');
var username=document.getElementById('username');
var email=document.getElementById('email');
var dd=document.getElementById('dd');
var mm=document.getElementById('mm');
var yyyy=document.getElementById('yyyy');
var comment=document.getElementById('comment');
var password=document.getElementById('password');
if(isAlphabet(firstname,"Please enter only letters for your First name"))
{
if(isAlphabet(lastname,"Please enter only letters for your Last name"))
{
if(isNumeric(dd,"Please enter a date"))
{
if(madeSelection(mm,"Please Choose"))
{
if(isNumeric(yyyy,"Please enter a year"))
{
if(isAlphanumeric(addr,"Enter Numbers and letters only for address"))
{
if(isNumeric(zip,"please enter a valid zip code"))
{
if(madeSelection(Countries,"Please Choose"))
{
if(lengthRestriction(username,6,8))
{
if(isAlphanumeric(password,"Enter Numbers and letters only for password"))

```

```

{
if(emailValidator(email,"Please enter a valid email address"))
{
if(notEmpty(comment,"Please fill the comment"))
{
document.write("<b><i>Thank's for submitting your details</i></b>");
alert("Successful Entry!!");
return true;
}}}}}}}}}}}}
return false;
}
function notEmpty(elem,helperMsg)
{
if(elem.value.length==0)
{
alert(helperMsg);
elem.focus();
return false;
}
return true;
}
function isNumeric(elem,helperMsg)
{
var numericExpression=/^[0-9]+$/;
if(elem.value.match(numericExpression))
{
return true;
}
else
{
alert(helperMsg);
elem.focus();
return false;
}
}
function isAlphabet(elem,helperMsg)
{
var alphaExp=/^[a-zA-Z]+$/;
if(elem.value.match(alphaExp))
{
return true;
}
else
{
alert(helperMsg);

```

```

elem.focus();
return false;
}
}
function isAlphanumeric(elem,helperMsg)
{
var alphaExp=/^[0-9, a-z a-z, 0-9, A-Z A-Z, - 0-9 . ]+$/;
if(elem.value.match(alphaExp))
{
return true;
}
else
{
alert(helperMsg);
elem.focus();
return false;
}
}
function lengthRestriction(elem,min,max)
{
var unput=elem.value;
if(unput.length>=min&&unput.length<=max)
{
return true;
}
else
{
alert("Please enter between "+min+" and "+max+" characters");
elem.focus();
return false;
}
}
function madeSelection(elem,helperMsg)
{
if(elem.value=="Please Choose")
{
alert(helperMsg);
elem.focus();
return false;
}
else
{
return true;
}
}

```

```

function emailValidator(elem,helperMsg)
{
var emailExp=/^[0-9 a-z . a-z 0-9]+\@[a-z]+\.[a-z]{2,4}$/;
if(elem.value.match(emailExp))
{
return true;
}
else
{
alert(helperMsg);
elem.focus();
return false;
}
}
</script>
<h1><center><b><font color="#347235">Please Enter Your
Details</font></b></center></h1>
</head>
<body bgcolor="LIGHTGREEN">
<hr>
<form onsubmit='return formValidator()' height="50%">
<table height="50%" border="3pt" align="center">
<tr><td><b><font color="#347235">First Name:</font></b></td><td><input
type='text' id='firstname'/></td></tr><br />
<tr><td><b><font color="#347235">Last Name:</font></b></td><td><input
type='text' id='lastname'/></td></tr><br />
<tr><td><b><font color="347235">Date of
Birth(dd/mm/yyyy):</font></b></td><td><input type='text' id='dd' />
<select id='mm'>
<option>Please Choose</option>
<option value="1">Jan</option>
<option value="2">Feb</option>
<option value="3">Mar</option>
<option value="4">Apr</option>
<option value="5">May</option>
<option value="6">Jun</option>
<option value="7">Jul</option>
<option value="8">Aug</option>
<option value="9">Sep</option>
<option value="10">Oct</option>
<option value="11">Nov</option>
<option value="12">Dec</option>
</select>
<input type='text' id='yyyy' /></td></tr><br />

```

```

<tr><td><b><font color="#347235">Address:</font></b></td><td><input
type='text' id='addr' /></td></tr><br />
<tr><td><b><font color="#347235">Zip code:</font></b></td><td><input
type='text' id='zip' /></td></tr><br />
<tr><td><b><font color="#347235">Countries:</font></b></td><td><select
id='Countries'>
<option>Please Choose</option>
<option value="United Kingdom">United Kingdom</option>
<option value="Afghanistan">Afghanistan</option>
<option value="America">America</option>
<option value="India">India</option>
<option value="Tanzania">Tanzania</option>
<option value="Zimbabwe">Zimbabwe</option>
<option value="Switzerland">Switzerland</option>
</select></td></tr><br />
<tr><td><b><font color="#347235">Username(6-8
characters):</font></b></td><td><input type='text' id='username' /></td></tr><br />
<tr><td><b><font color="#347235">Password:</font></b></td><td><input
type='password' id='password' /></td></tr><br />
<tr><td><b><font color="#347235">Email:</font></b></td><td><input
type='text' id='email' /></td></tr><br />

<tr><td><b><font
color="#347235">Comment:</font></b></td><td><textarea id='comment'cols="20"
rows="5" name="Address"></textarea></td></tr><br />
<tr><td><input type='submit' value='Check Form' /></td></tr>
</table>
</form>
</body></html>

```

OUTPUT:

C:\Documents and Settings\student\Desktop\Web Forms ex\webforms.html - Windows Internet Explorer

C:\Documents and Settings\student\Desktop\Web Forms ex\webforms.html

File Edit View Favorites Tools Help

C:\Documents and Settings\student\Desktop\Web Fo...

Please Enter Your Details

First Name:	<input type="text"/>
Last Name:	<input type="text"/>
Date of Birth(dd/mm/yyyy):	<input type="text"/> Please Choose
Address:	<input type="text"/>
Zip code:	<input type="text"/>
Countries:	<input type="text"/> Please Choose
Username(6-8 characters):	<input type="text"/>
Password:	<input type="text"/>
Email:	<input type="text"/>

Done

start Z Notepad C:\WINDOWS\... bin DOM.SAX Web Forms ex C:\Documents a... 1:24 AM

C:\Documents and Settings\student\Desktop\Web Forms ex\webforms.html - Windows Internet Explorer

C:\Documents and Settings\student\Desktop\Web Forms ex\webforms.html

File Edit View Favorites Tools Help

C:\Documents and Settings\student\Desktop\Web Fo...

Please Enter Your Details

First Name:	Vadivelan
Last Name:	<input type="text"/>
Date of Birth(dd/mm/yyyy):	<input type="text"/> Please Choose
Address:	<input type="text"/>
Zip code:	<input type="text"/>
Countries:	<input type="text"/> Please Choose
Username(6-8 characters):	<input type="text"/>
Password:	<input type="text"/>
Email:	<input type="text"/>
Comment:	<input type="text"/>

Check Form

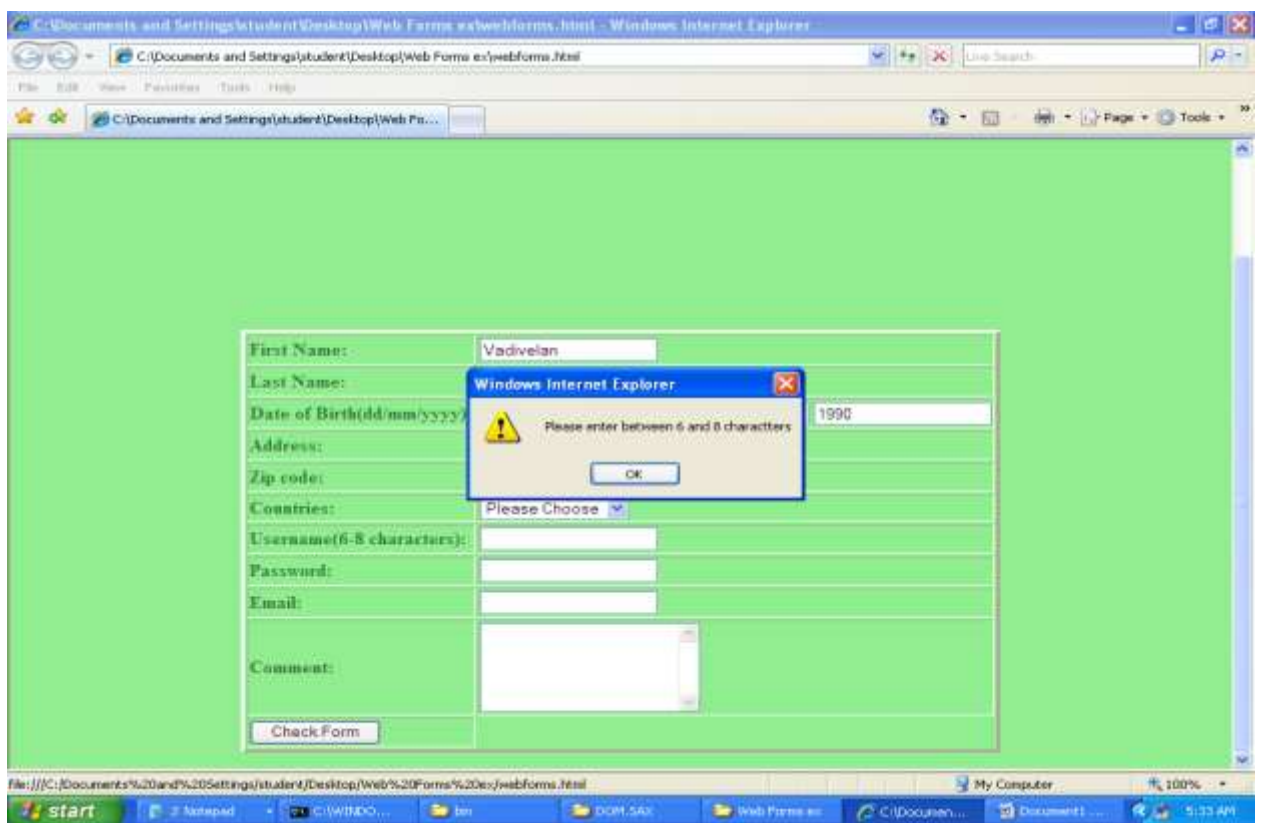
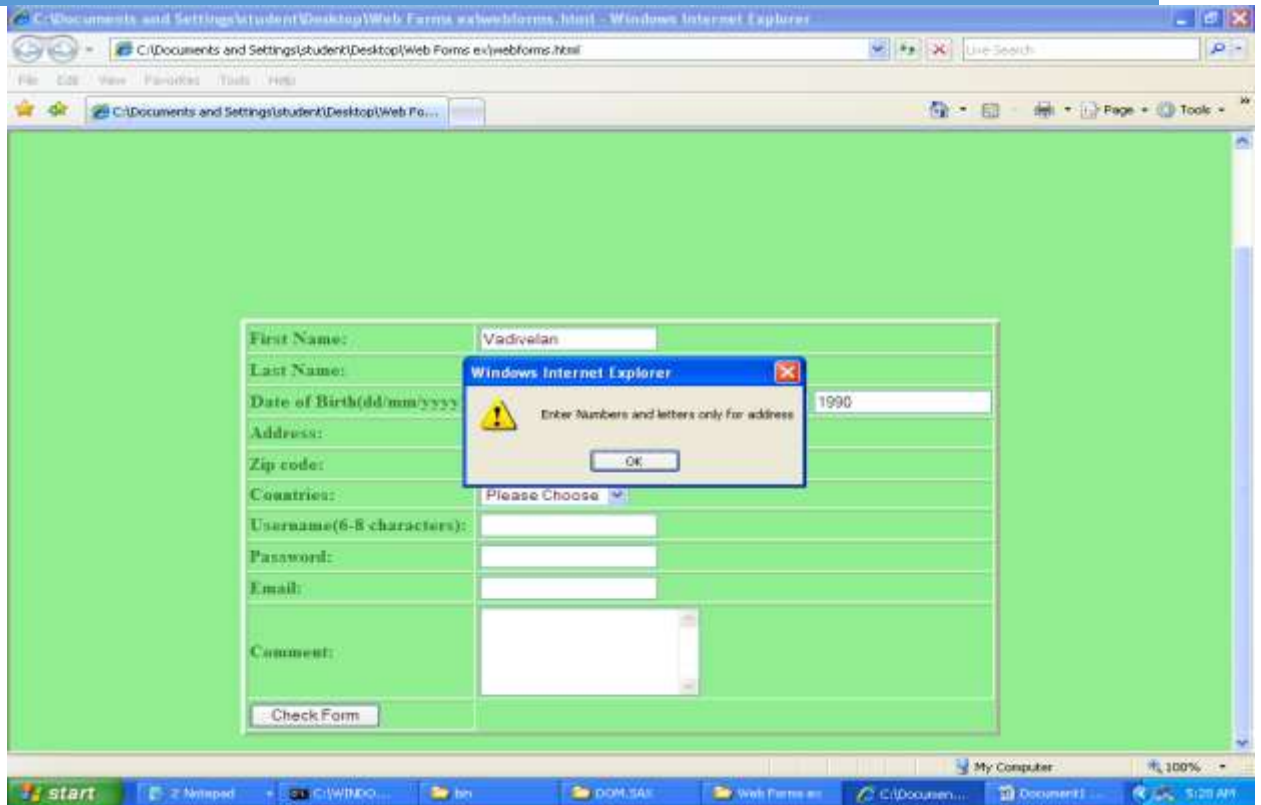
Windows Internet Explorer

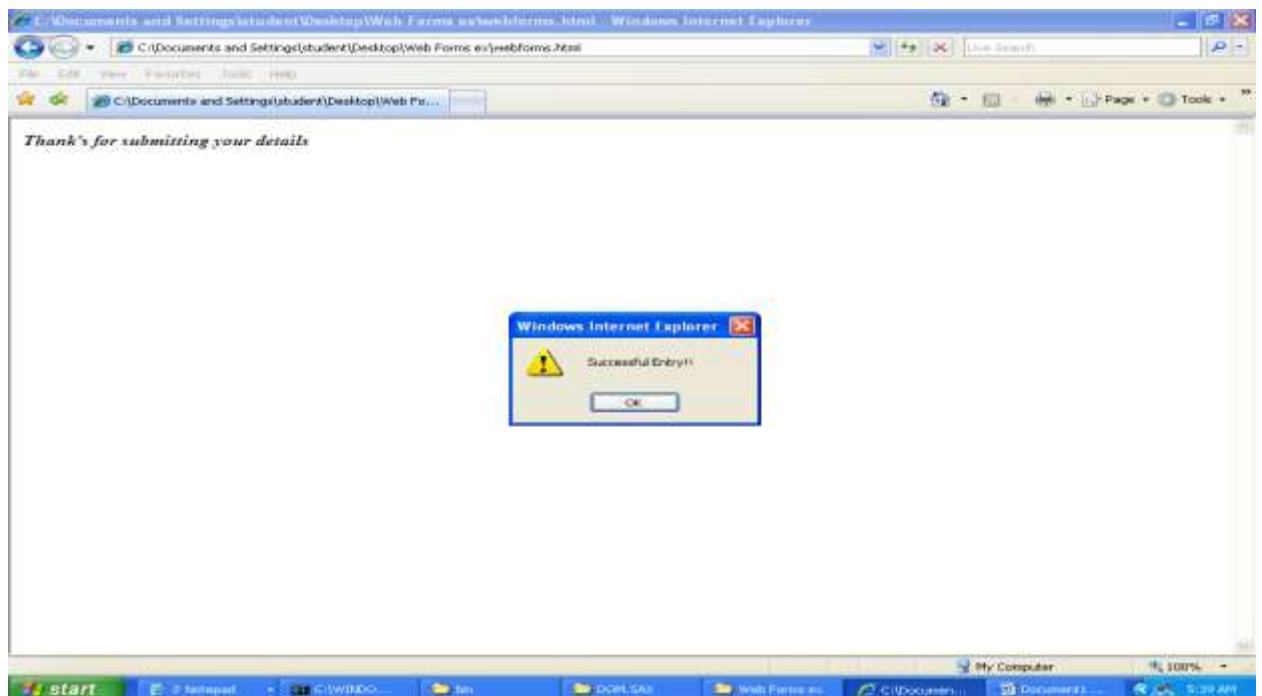
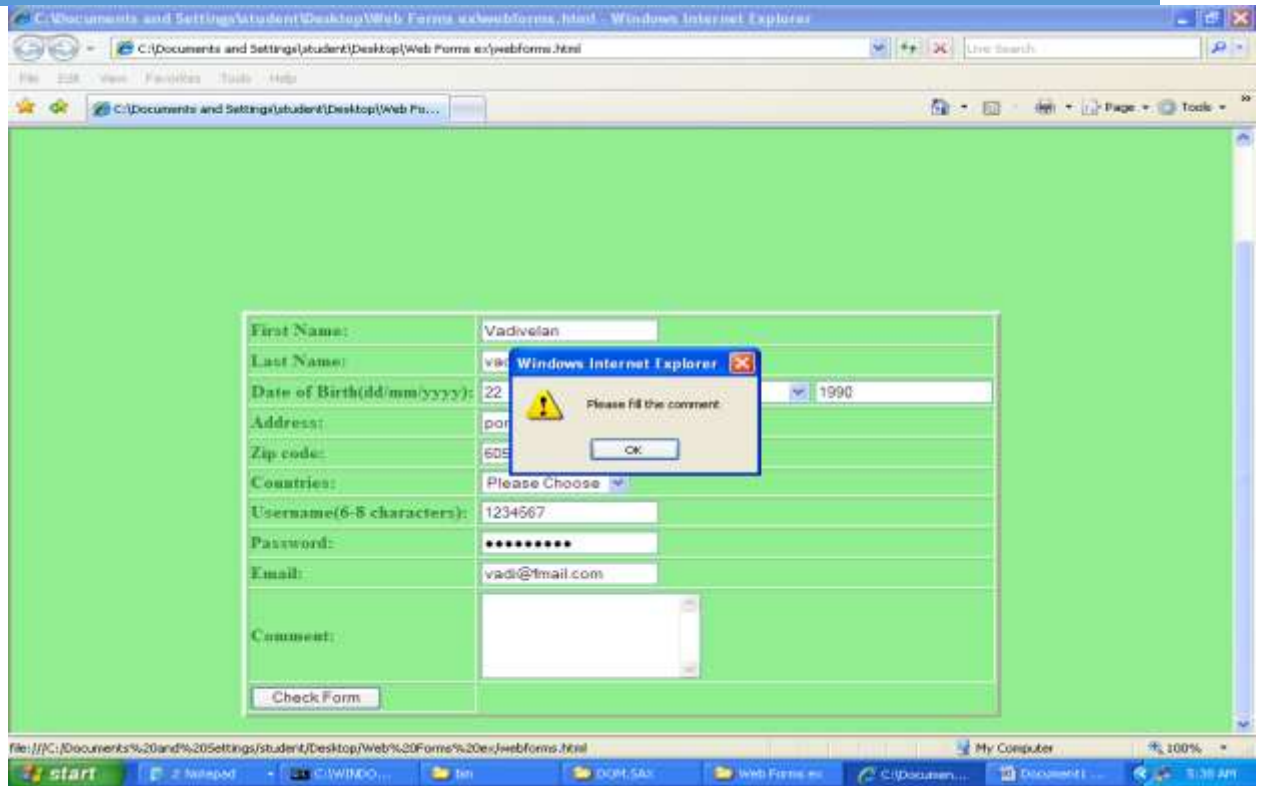
Please enter only letters for your Last name

OK

File: C:\Documents%20and%20Settings\student\Desktop\Web%20Forms%20ex\webforms.html

start Z Notepad C:\WINDOWS\... bin DOM.SAX Web Forms ex C:\Documents... Document... 5:25 AM





RESULT: Thus developing client side scripts for validating web form controls using DHTML has been verified.

Ex No :4	CREATION OF COLOR PALETTE
Date:	

AIM:

To write a java program to create applet with the following features create a color with matrix of buttons. Set the background and foreground of the written text area by selecting a color from color palette. Order to select foreground or background use check control or radio button. To set background images.

ALGORITHM:

Step1: Import all necessary package and classes.

Step2: Define a class that extends applet and implements action listener and item listener.

Step3: Declare an array of button to set colors, font check boxes for foreground and background.

Step4: Add the array of buttons in the init function.

Step5: Stop the program.

PROGRAM:

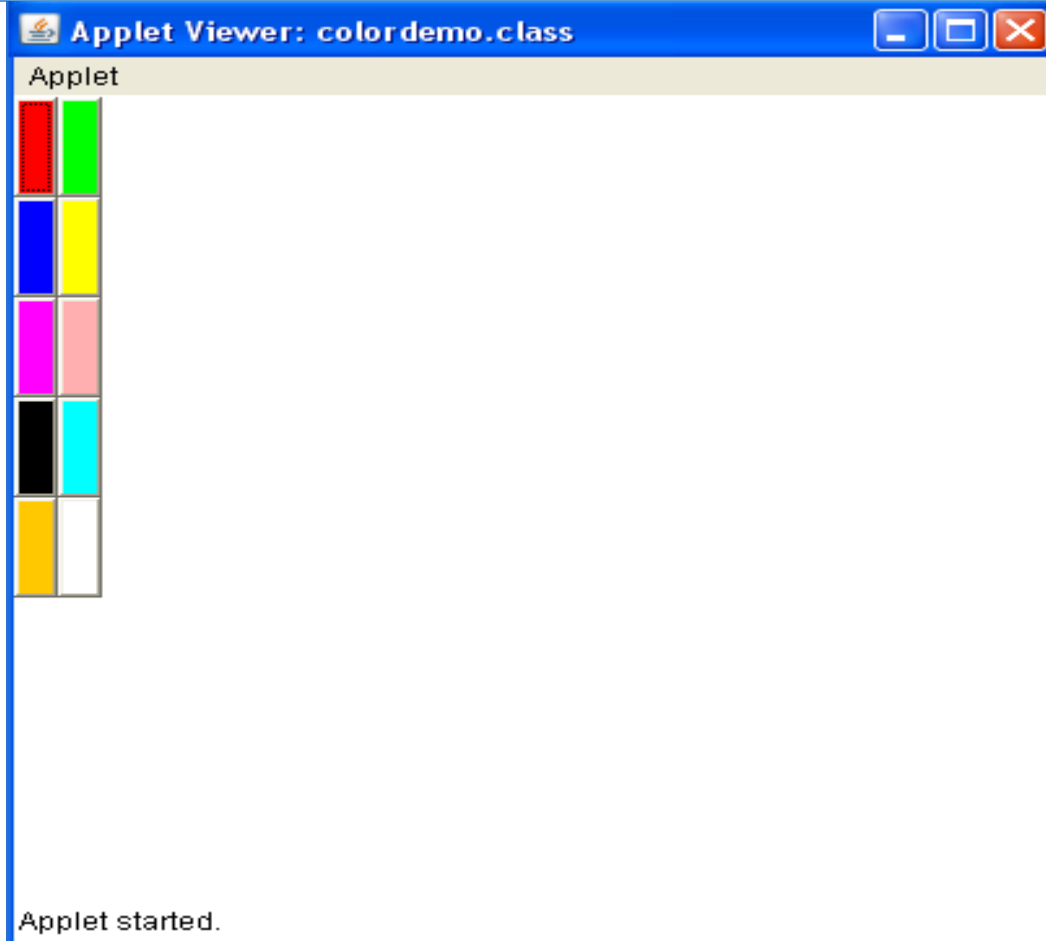
```
//colordemo.java
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<html>
<applet code="colordemo.class" height="400" width="400">
</applet>
</html>
```

```

*/
public class colordemo extends Applet implements ActionListener
{
    Button colorpalette[]=new Button[10];
    Color
fillcolor[]={ Color.red,Color.green,Color.blue,Color.yellow,Color.magenta,Color.pink,Color.blac
k,Color.cyan,Color.orange,Color.white };
    Panel colorpanel;
    int i;
    public void init()
    {
        setLayout(new BorderLayout());
        colorpanel=new Panel();
        colorpanel.setLayout(new GridLayout(8,10));
        for(i=0;i<10;i++)
        {
            colorpalette[i]=new Button(" ");
            colorpalette[i].setBackground(fillcolor[i]);
            colorpalette[i].addActionListener(this);
            colorpanel.add(colorpalette[i]);
        }
        add(colorpanel,BorderLayout.WEST);
    }
    public void actionPerformed(ActionEvent ae)
    {
    }
}

```

OUTPUT:



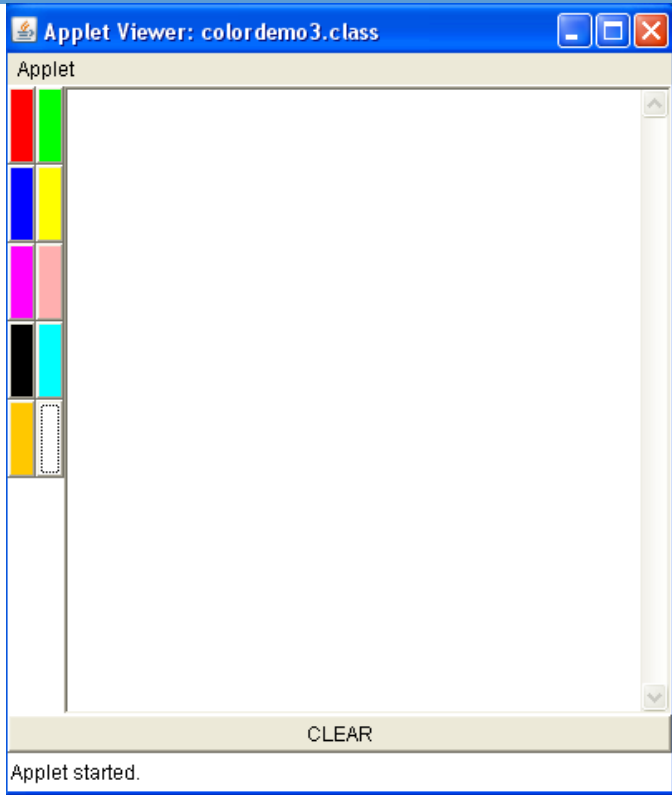
```
//colordemo2.java
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<html>
<applet code="colordemo2.class" height="400" width="400">
</applet>
</html>
*/
public class colordemo2 extends Applet implements ActionListener
{
    Button colorpalette[]=new Button[10];
    Color
fillcolor[]={ Color.red,Color.green,Color.blue,Color.yellow,Color.magenta,Color.pink,Color.black,Color.cyan,Color.orange,Color.white};
    Panel colorpanel;
    int i;
```

```

public void init()
{
    setLayout(new BorderLayout());
    colorpanel=new Panel();
    colorpanel.setLayout(new GridLayout(8,10));
    for(i=0;i<10;i++)
    {
        colorpalette[i]=new Button(" ");
        colorpalette[i].setBackground(fillcolor[i]);
        colorpalette[i].addActionListener(this);
        colorpanel.add(colorpalette[i]);
    }
    add(colorpanel,BorderLayout.WEST);
}
public void actionPerformed(ActionEvent ae)
{
    for(i=0;i<10;i++)
    {
        if(ae.getSource()==colorpalette[i])
        {
            ta.setBackground(colorpalette[i].getBackground());
        }
    }
}
}

```

OUTPUT:



RESULT: Thus the developing of java program to create applets in colop palette with matrix of buttons has been verified successfully.

Ex No : 5	COLOR PALETTE TO SET BACKGROUND & FOREGROUND COLOR
Date:	

AIM:

To write a java program to Set the background and foreground of the written text area by selecting a color from color palette. Order to select foreground or background use check control or radio button. To set background images.

ALGORITHM:

Step1: Import all necessary package and classes.

Step2: Define a class that extends applet and implements action listener and item listener.

Step3: Declare an array of button to set colors, font check boxes for foreground and background.

Step4: Add the array of buttons in the init function.

Step5: In the action performed() method, do the following

Step6: Get the action command in the string, color.

Step7: If the foreground in checked the set the following color to the color.

Step8: If the background is changed then the bgcolor to the selector color.

Step9: Stop the program.

PROGRAM:

```

//colordemo0.java
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<html>
<applet code="colordemo0.class" height="500" width="500">
</applet>
</html>
*/
public class colordemo0 extends Applet implements ActionListener,ItemListener
{
    Button colorpalette[]=new Button[10];
    Color
fillcolor[]={ Color.red,Color.green,Color.blue,Color.yellow,Color.magenta,Color.pink,Color.black,Color.cyan,Color.orange,Color.white };
    Panel colorpanel;
    int i;
    Checkbox back,fore;
    CheckboxGroup cbg;
    boolean flag=true;
    String val="Hai";
    TextArea ta;
    Button clear;
    public void init()
    {
        cbg=new CheckboxGroup();
        back=new Checkbox("BACKGROUND COLOR",cbg,true);
        fore=new Checkbox("FOREGROUND COLOR",cbg,false);
        setLayout(new BorderLayout());
        colorpanel=new Panel();
        colorpanel.setLayout(new GridLayout(5,6));
        colorpanel.add(back);
        colorpanel.add(fore);
        back.addItemListener(this);
        fore.addItemListener(this);
        for(i=0;i<10;i++)
        {
            colorpalette[i]=new Button(" ");
            colorpalette[i].setBackground(fillcolor[i]);
            colorpalette[i].addActionListener(this);
            colorpanel.add(colorpalette[i]);
        }
    }
}

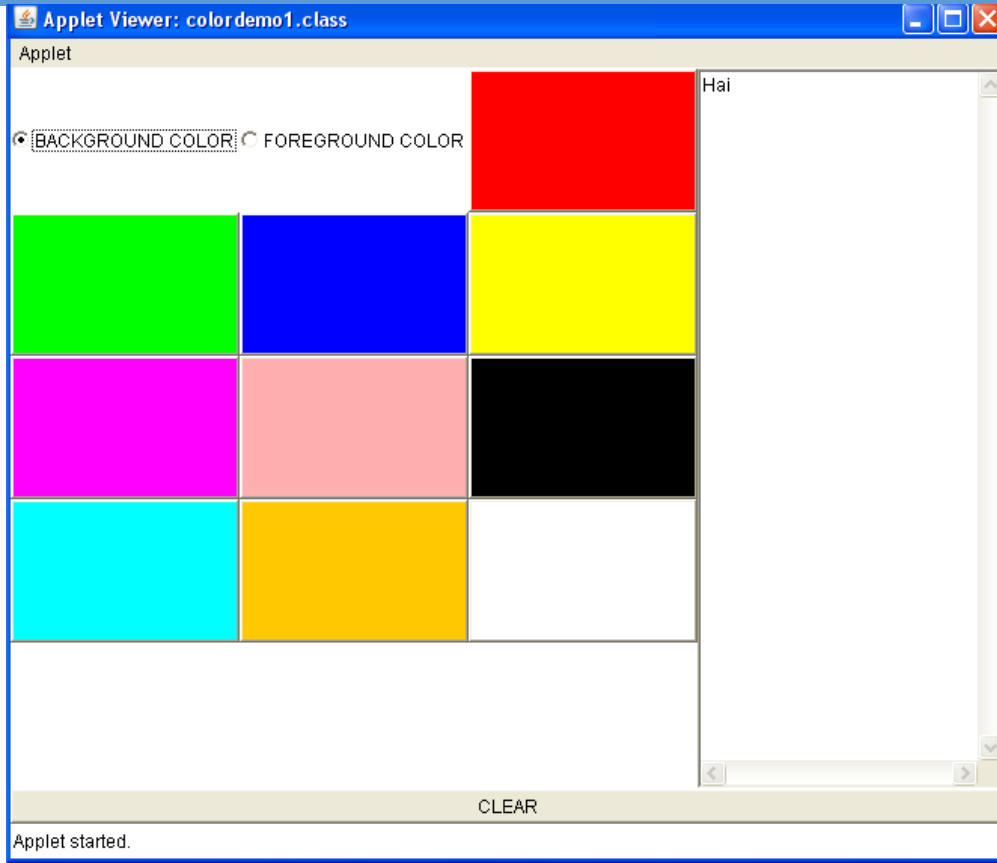
```

```

    }
    add(colorpanel, BorderLayout.WEST);
    ta=new TextArea(val,10,200);
    add(ta, BorderLayout.CENTER);
    clear=new Button("CLEAR");
    clear.addActionListener(this);
    add(clear, BorderLayout.SOUTH);
    }
    public void itemStateChanged(ItemEvent ie)
    {
    if(cbg.getSelectedCheckbox().getLabel()=="BACKGROUND COLOR")
    flag=true;
    else
    flag=false;
    }
    public void actionPerformed(ActionEvent ae)
    {
    if(ae.getSource()==clear)
    ta.setBackground(Color.white);
    for(i=0;i<10;i++)
    {
    if(ae.getSource()==colorpalette[i])
    {
    if(flag==true)
    ta.setBackground(colorpalette[i].getBackground());
    else
    ta.setForeground(colorpalette[i].getBackground());
    }
    repaint();
    }
    }
    }
}

```

OUTPUT:



RESULT: Thus the developing of java program to create applets in color palette with matrix of buttons has been verified successfully.

Ex No :6	INVOKING SERVLET FROM HTML FORMS
Date:	

AIM:

To write a html program for invoking servlet using html.

ALGORITHM:

Step1: In html program, define the html, head and title tag.

Step2: Then the title is Student Information Form and close the title and head tag.

Step3: Define the body tag inside the body tag create form and table simultaneously.

Step4: The table consists of following information Roll no, Student name, Address, Phone no and total marks.

Step5: In the servlet program, import the summary package and create a own servlet class extends with generic servlet.

Step6: In the service method defined to request and response.

Step7: Create the object and for print writer and get writer() value.

Step8: The enumeration object get the servlet request parameter.

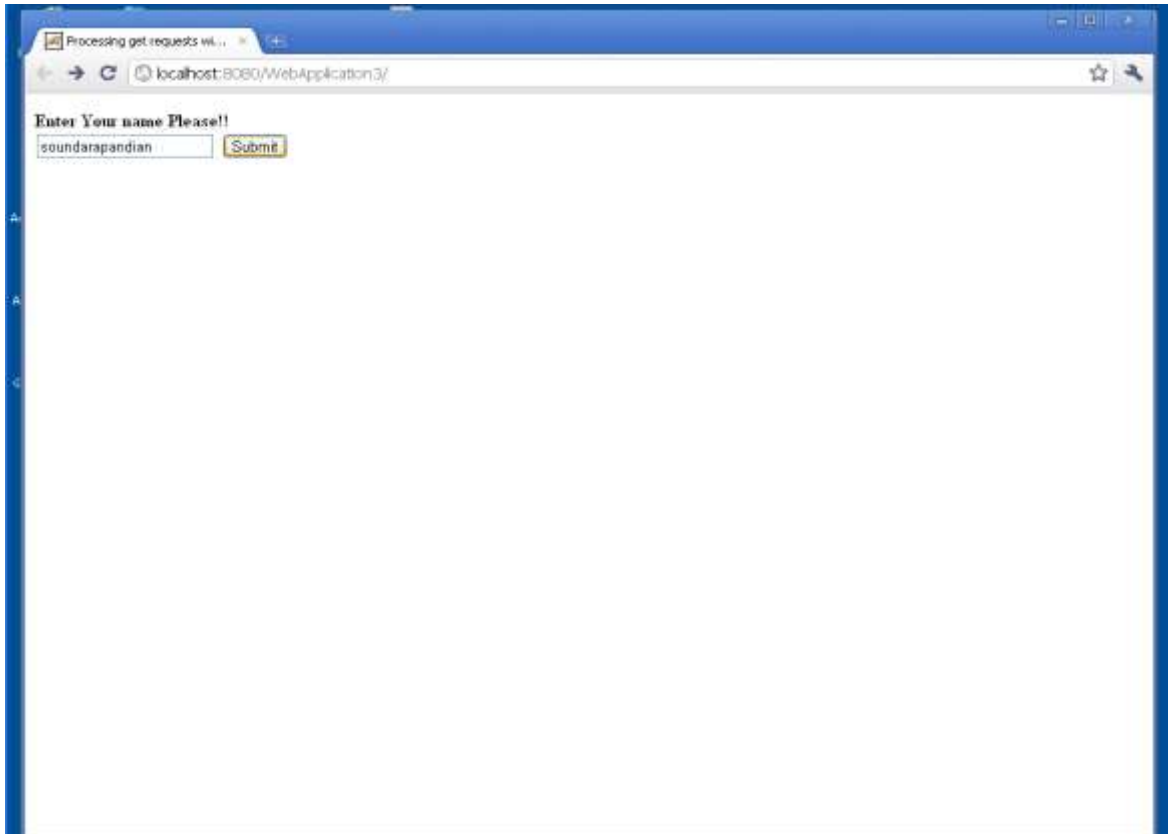
Step9: Create objects for string method and it is displayed another object value received get parameter of name received and displayed the value received value.

PROGRAM:

```
//index.jsp
<html>
<head>
<title>Processing get requests with data</title>
</head>
<body>
<form action = "Servlet3" method = "get">
<b><p><label>Enter Your name Please!!
<br />
<input type = "text" name = "firstname" />
<input type = "submit" value = "Submit" />
</label></p></b>
</form>
</body>
</html>
```

```
//Servlet3.java
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class Servlet3 extends HttpServlet {
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
throws ServletException, IOException {
String firstName = request.getParameter( "firstname" );
response.setContentType( "text/html" );
PrintWriter out = response.getWriter();
// send XHTML document to client
// start XHTML document
out.println( "<html>" );
// head section of document
out.println( "<head>" );
out.println("<title>Processing get requests with data</title>" );
out.println( "</head>" );
// body section of document
out.println( "<body>" );
out.println( "<h1>Hello " + firstName + ",<br />" );
out.println( "Welcome to Servlets!</h1>" );
out.println( "</body>" );
// end XHTML document
```

```
out.println( "</html>" );  
out.close(); // close stream to complete the page  
}  
public String getServletInfo() {  
    return "Short description";  
}  
}
```

OUTPUT:



RESULT: Thus the invocation of swrvlet from HTML from has been developed successfully.

Ex No :7	ONLINE EXAMINATION
Date:	

AIM:

To write a java servlet program to conduct online examination and to display student mark list available in a database.

ALGORITHM:

Step1: Create a html file with form tag.

Step2: The form tag action="http://localhost:8080/example/servlet/exam".

Step3: Create a two textbox(name & seat number).

Step4: The 5 question are defined into true or false model and close the all tags.

Step5: Import the necessary packages and declare class, class name in exam.

Step6: Declare the connection, statement and result set object.

Step7: Use the deposit () for check the connection in JDBC:ODBC driver.

Step8: The data are inserting into corresponding table.

Step9: The execute update () are update the database.

Step10: Display the table in after html file compilation.

PROGRAM:

```
//index.jsp
<html>
```



```

<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Welcome to Online Examination!!!!</title>
</head>
<body>
Welcome to Online Examination!!!!
<form action="exam" method="get">
<label><p> Enter Your name Please!!<br/> <input type="text"
name="name"/>
<br/>
<input type="submit" name="SUBMIT"/>
</p></label>
</form>
</body>
</html>
//exam.java
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class exam extends HttpServlet {
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
throws ServletException, IOException {
response.setContentType("text/html");
PrintWriter out=response.getWriter();
String name=request.getParameter("name");
out.println("<html>");
out.println("<head>");
out.println("<title>Online Examination</title>");
out.println("</head>");
out.println("<body bgcolor=PINK>");
out.println("<h2 align=center>Online Examination</h2><hr>");
out.println("<h3 align=center> Welcome Mr."+name+"</h3><hr>");
out.println("<h4><u>Terms and Conditions:</u></h4>");
out.println("<ul type=disc>");
out.println("<li>The Paper consists a set of five questions.</li>");
out.println("<li>Every question consists of two options.</li>");
out.println("<li>All must be answered</li></ul><hr>");
out.println("<center><h5><u>Your Questions</u></h5></center>");
out.println("<hr>");
out.println("<form method=get action=exam2>");

```

```

        out.println("<p>1.Operating System is a .....</p>");
        out.println("<input type=radio name=q1 value=0>Hardware");
        out.println("<br>");
        out.println("<input type=radio name=q1 value=1>Software");
        out.println("<hr>");
        out.println("<p>2.Developer of C Language is .....</p>");
        out.println("<br>");
        out.println("<input type=radio name=q2 value=0>Dennis Richee");
        out.println("<br>");
        out.println("<input type=radio name=q2 value=1>James Thompson");
        out.println("<hr>");
        out.println("<p>3.Which of the following is a multitasking,multi
user,multiprocessing);
        out.println("OS.....</p>");
        out.println("<br>");
        out.println("<input type=radio name=q3 value=0>MS DOS");
        out.println("<br>");
        out.println("<input type=radio name=q3 value=1>Windows NT");
        out.println("<hr>");
        out.println("<p>4.Father of Computers is .....</p>");
        out.println("<br>");
        out.println("<input type=radio name=q4 value=1>Charles babbage");
        out.println("<br>");
        out.println("<input type=radio name=q4 value=0>Charles Dickson");
        out.println("<hr>");
        out.println("<p>5.What is the current generation of computers ?</p>");
        out.println("<br>");
        out.println("<input type=radio name=q5 value=0>Fifth");
        out.println("<br>");
        out.println("<input type=radio name=q5 value=1>Sixth");
        out.println("<hr>");
        out.println("<input type=submit value=Done>");
        out.println("</form>");
        out.println("</body>");
        out.println("</html>");

```

```

    }
    public String getServletInfo() {
        return "A Servlet of the user";
    }
}

```

//exam2.java

```

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;

```

```

import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class exam2 extends HttpServlet {
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
throws ServletException, IOException {
    int count=0,j;
    response.setContentType("text/html");
    PrintWriter out=response.getWriter();
    String q1=request.getParameter("q1");
    String q2=request.getParameter("q2");
    String q3=request.getParameter("q3");
    String q4=request.getParameter("q4");
    String q5=request.getParameter("q5");
    if(q1.equals("1"))
    {
        count=count+1;
    }
    if(q2.equals("1"))
    {
        count=count+1;
    }
    if(q3.equals("1"))
    {
        count=count+1;
    }
    if(q4.equals("1"))
    {
        count=count+1;
    }
    if(q5.equals("1"))
    {
        count=count+1;
    }
    out.println("<html>");
    out.println("<head><title>Examination Results</title></head>");
    out.println("<body>");
    out.println("<h2 align=center>Online Examination</h2><hr>");
    out.println("<h3>Number of Questions answered correctly:</h3>"+count);
    if(count>=3)
    {
        out.println("<hr><h3>Congrats!!! You Have Passed!!!</h3><hr>");
        out.println("<h4><b>Try Other Tests!!</b></h4>");
    }
}

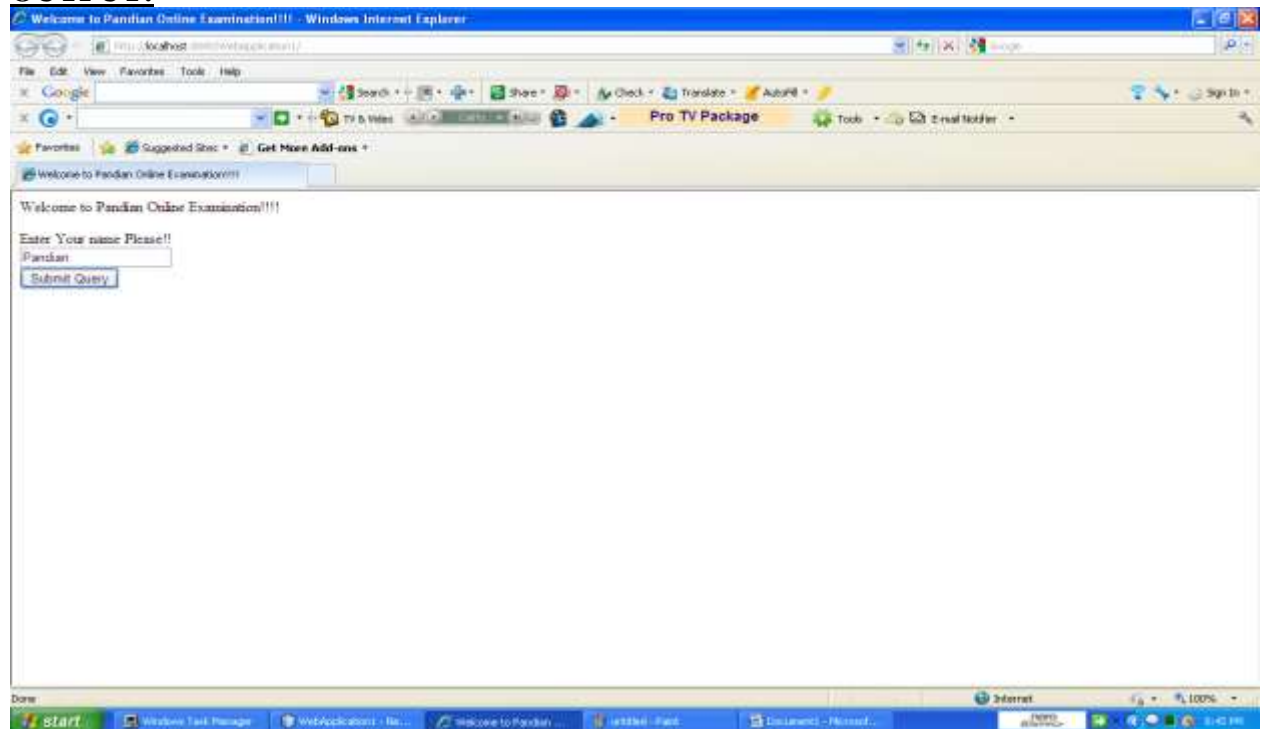
```

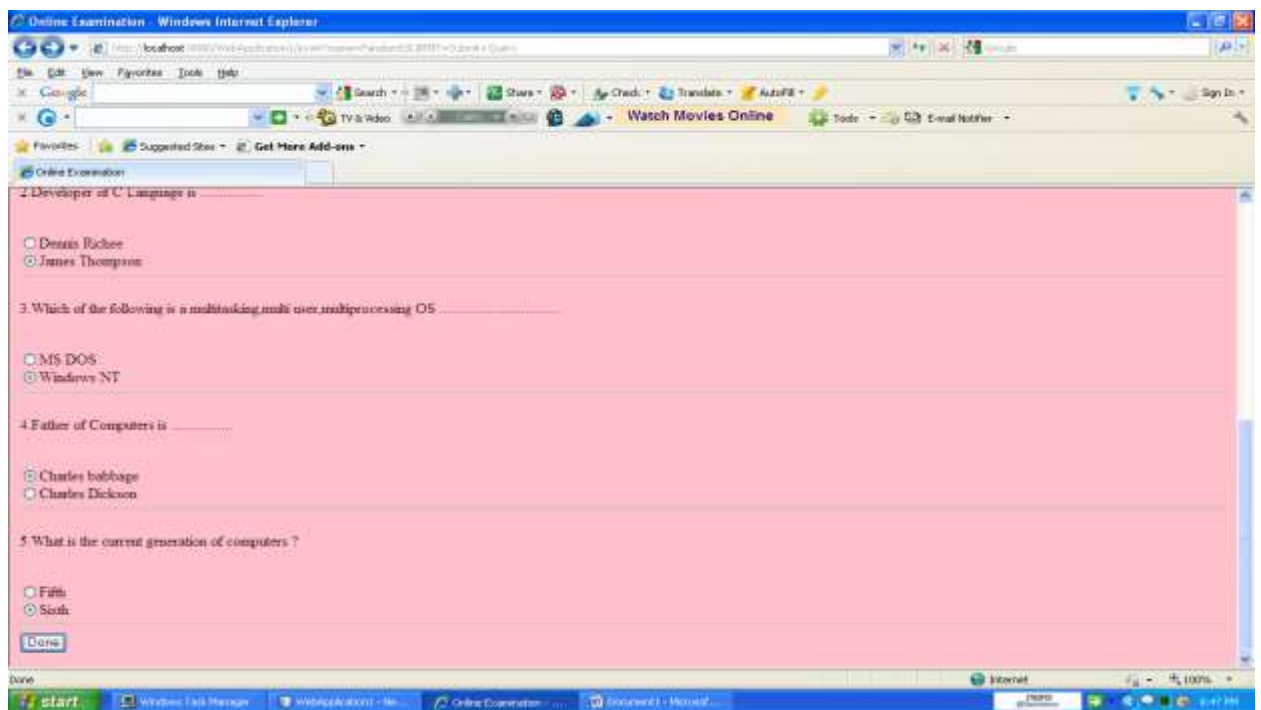
```

else
{
    out.println("<hr><h3>Sorry!!! You Have Failed!!!</h3><hr>");
    out.println("<h4><b>Try Again:</b></h4>");
}
out.println("</body>");
out.println("</html>");
}
public String getServletInfo() {
    return "A Servlet of the User";
}
}

```

OUTPUT:





RESULT: Thus the development of program in java to create three tire application using servlet has been verified successfully.

Ex No : 8	XML SCHEMA FOR STUDENT DETAILS
Date:	

AIM:

To write a program for implementing student information using XML & XSL.

ALGORITHM:

Step1:The XML document reference to the XSL document.

Step2:The create the student information in the student tag and insert the same information about the student.

Step3:Close all opened tags.

Step4:In XSL document create a html file include the student information in table format.

Step5:Close the necessary tags.

PROGRAM:

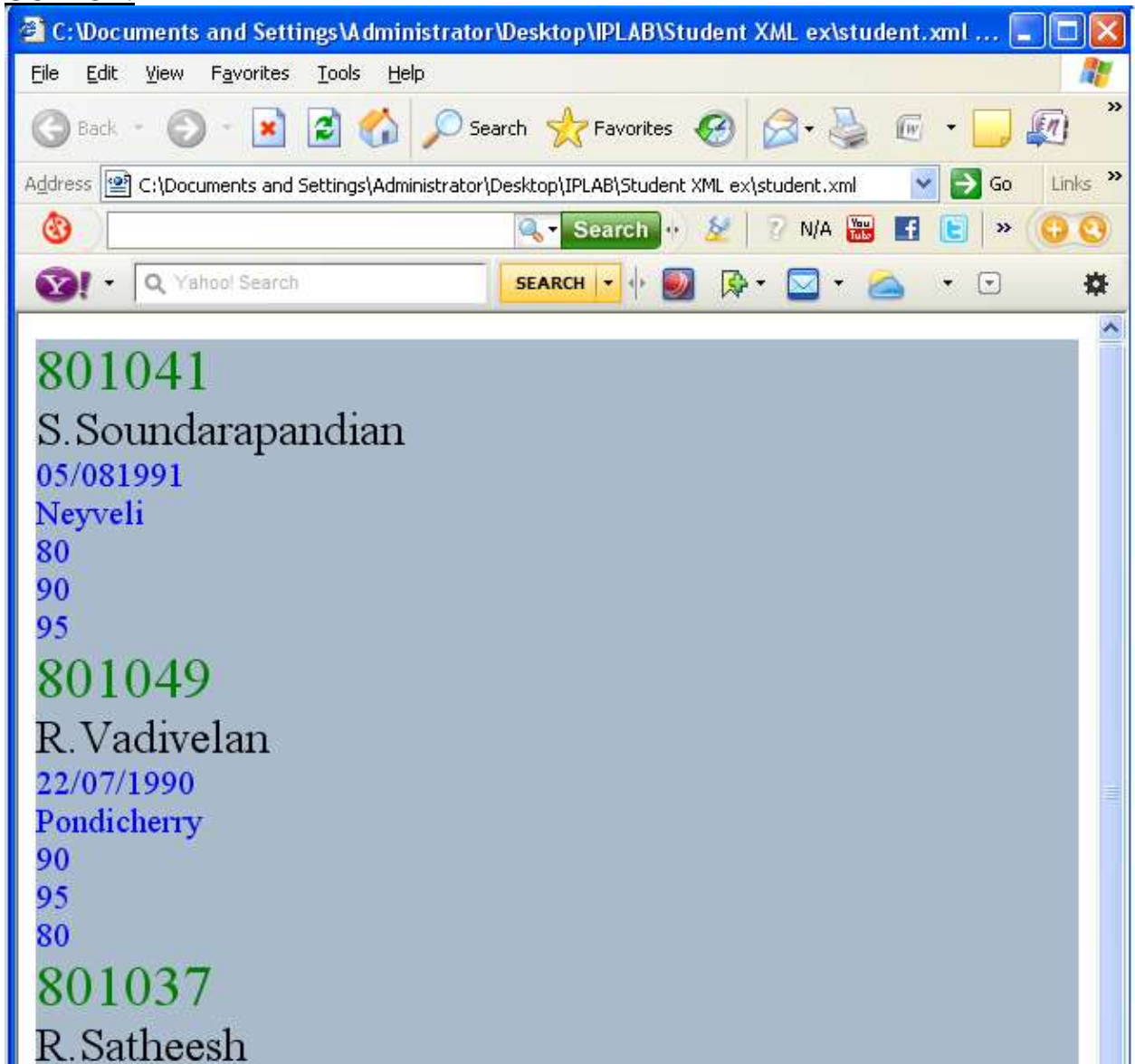
```
//student.xml
<?xml version="1.0"?>
<?xml-stylesheet type="text/css" href="student.css"?>
<!DOCTYPE student SYSTEM "student.dtd">
<students>
<student>
<sno>801041</sno>
<sname>S.Soundarapandian</sname>
```

```

<dob>05/081991</dob>
<address>Neyveli</address>
<m1>80</m1>
<m2>90</m2>
<m3>95</m3>
</student>
<student>
<sno>801049</sno>
<sname>R.Vadivelan</sname>
<dob>22/07/1990</dob>
<address>Pondicherry</address>
<m1>90</m1>
<m2>95</m2>
<m3>80</m3>
</student>
<student>
<sno>801037</sno>
<sname>R.Satheesh</sname>
<dob>21/01/1991</dob>
<address>Kanyakumari</address>
<m1>80</m1>
<m2>90</m2>
<m3>95</m3>
</student>
</students>
//student.css
Student { background.color:#aabbcc;width:100%;}
Sno { display:block; color:GREEN; font.size:25pt; }
Sname { display:block; color:BLACK; font.size:20pt; }
Dob { display:block; color:BLUE; font.size:15pt; }
Address { display:block; color:BLUE; font.size:15pt; }
m1 { display:block; color:BLUE; font.size:15pt; }
m2 { display:block; color:BLUE; font.size:15pt; }
m3 { display:block; color:BLUE; font.size:15pt;}
//student.dtd
<?xml version="1.0"?>
<!ELEMENT students (student+)>
<!ELEMENT student (sno,sname,dob,address,m1,m2,m3)>
<!ELEMENT sno (#PCDATA)>
<!ELEMENT sname (#PCDATA)>
<!ELEMENT dob (#PCDATA)>
<!ELEMENT address (#PCDATA)>
<!ELEMENT m1 (#PCDATA)>
<!ELEMENT m2 (#PCDATA)>
<!ELEMENT m3 (#PCDATA)>

```

OUTPUT:



RESULT: Thus the creation of XSL document using Xml has been verified successfully.

Ex No : 9	XML - DOM PARSING
Date:	

AIM:

To write a program for implementing student information XML document is well formed or not using DOM.

ALGORITHM:

Step1:In java program, import the necessary package and declare the class parsing-DOM demo.

Step2:In that main function display enter the name of the XML document and get the input from the buffer reader object.

Step3:Then create a file object and it assign to buffered reader value.

Step4:The input source is filename, then it is well formed otherwise it is not well formed.

Step5:In XML program create the student information and the java file complete and display the output.

```

//DOMCountEle.java
import org.w3c.dom.*;
import javax.xml.parsers.*;
import java.io.*;
public class DOMCountEle
{
    public static void main(String[] args)
    {
        try
        {
            BufferedReader bf=new BufferedReader(new InputStreamReader(System.in));
            System.out.print("Enter File name: ");
            String xmlfile=bf.readLine();
            File file=new File(xmlfile);
            if(file.exists())
            {
                DocumentBuilderFactory factory=DocumentBuilderFactory.newInstance();
                //create the Builder and parse the file
                Document doc=factory.newDocumentBuilder().parse(xmlfile);
                System.out.print("Enter the Element Name: ");
                String element=bf.readLine();
                NodeList nodes=doc.getElementsByTagName(element);
                System.out.println("XML Document Contains "+nodes.getLength()+"
Elements.");
            }
            else
            {
                System.out.println("File Not Found!");
            }
        }
        catch(Exception ex)
        {
            System.out.println(ex);
        }
    }
}
//employee-deails.xml
<?xml version="1.0"?>
<Employee-Detail>

```

```
<Employee>
<Emp-Id>801041</Emp-Id>
<Emp-Name>S.Soundarapandian</Emp-Name>
<Emp-E-mail>s.soundarapandian@hotmail.com</Emp-E-mail>
</Employee>
<Employee>
<Emp-Id>801049</Emp-Id>
<Emp-Name>R.Vadivelan</Emp-Name>
<Emp-E-mail>vadivelanvrs@gmail.com</Emp-E-mail>
</Employee>
<Employee>
<Emp-Id>801037</Emp-Id>
<Emp-Name>R.Satheesh</Emp-Name>
<Emp-E-mail>satheeshcse37@gmail.com</Emp-E-mail>
</Employee>
</Employee-Detail>
```

OUTPUT:

Enter the File name: employee-detail.xml

Enter the Element name: Emp-Id

XML Document Contains 3 Elements

RESULT: Thus the creation of program using DOM has been verified successfully.

Ex No : 10	SAX PARSING
Date:	

AIM:

To write a program for student information XML document is well-formed or not using SAX.

ALGORITHM:

Step1: In java program, input the necessary packages and declare the classname is parsing- SAXDEMO.

Step2: In the main function display the “Enter the XML document name” message and it get the buffered reader object.

Step3: A new file is created and it exists then it is well formed, otherwise it is not well formed.

Step4: Else file is not passed message will be displayed.

Step5: In XML document create the student information.

PROGRAM:

```
//SAX.java
import java.io.*;
import org.xml.sax.*;
import org.xml.sax.helpers.*;
public class SAX
```

```

    {
    public static void main(String[] args) throws IOException
    {
    try
    {
    System.out.println("Enter the Name of XML Document");
    BufferedReader input=new BufferedReader(new
InputStreamReader(System.in));
    String filename=input.readLine();
    File fp=new File(filename);
    if(fp.exists())
    {
    try
    {
    XMLReader reader=XMLReaderFactory.createXMLReader();
    reader.parse(filename);
    System.out.println(filename +"is Well-Formed.");
    }
    catch(Exception e)
    {
    System.out.println(filename +"is not Well-Formed.");
    System.exit(0);
    }
    }
    else
    {
    System.out.println(filename +" File is not Present! ");
    }
    }
    catch(IOException ex)
    {
    ex.printStackTrace();
    }
    }
    }

```

OUTPUT:

Enter the name of XML Document
employee-detail.xml
employee-detail.xml is well-Formed

RESULT: Thus developing a document using SAX has been verified
successfully completed.
