Ex No : 01	AWT Components
Date :	

Aim:

To Write a Program in Java to demonstrate the Use of Following Components:

Text Fields, Buttons, Scroll-bars, Choice, List and Checkbox.

Procedure:

Step 1 : Start the program

Step 2 : Import the necessary package of Java awt.*

Step 3 : Declare the class AWT controls and implements Action Listeners.

Step 4 : Include their components with Choice,Button ,Text fields, Checkbox.

Step 5 : Declare the paint functions and display the objects

Step 6 : Terminate the program.

Source code:

```
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
public class student extends Frame implements ActionListener
{
String msg;
Button b1=new Button("save");
Label l11=new Label("student details",Label.CENTER);
Label l1=new Label("NAME",Label.LEFT);
Label l2=new Label("AGE",Label.LEFT);
```

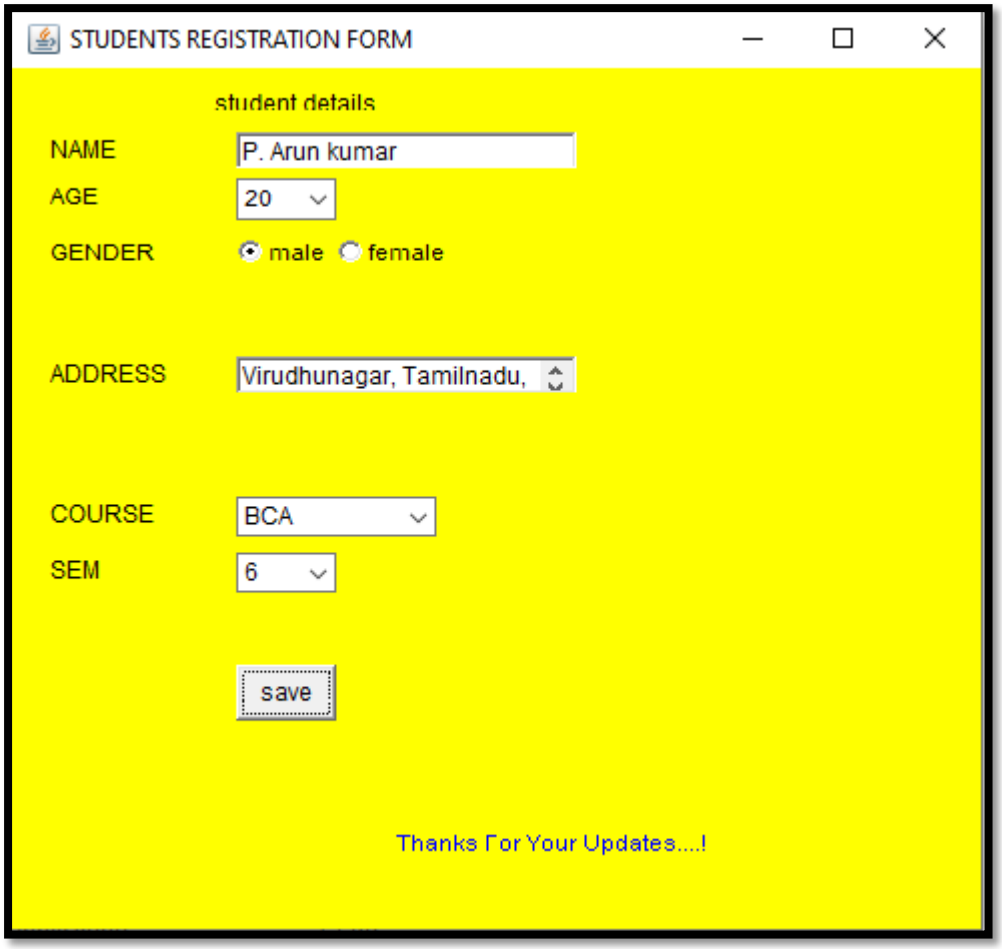
```
Label l3=new Label("GENDER",Label.LEFT);
Label l4=new Label("ADDRESS",Label.LEFT);
Label l5=new Label("COURSE",Label.LEFT) ;
Label l6=new Label("SEM",Label.LEFT);
Label l7=new Label("",Label.RIGHT);
TextField t1=new TextField ();
Choice c1=new Choice();
CheckboxGroup cbg=new CheckboxGroup();
Checkbox ck1=new Checkbox("male",false,cbg);
Checkbox ck2=new Checkbox("female",false,cbg);
TextArea t2=new TextArea("",180,90,TextArea.SCROLLBARS_VERTICAL_ONLY);
Choice course= new Choice();
Choice sem= new Choice();
Choice age= new Choice();
public student()
{
addWindowListener(new myWindowAdapter());
setBackground(Color.red);
setForeground(Color.black);
setLayout(null);
add(l11);
add(l1);
add(l2);
add(l3);
add(l4);
add(l5);
add(l6);
```

```
add(l7);
add(t1);
add(t2);
add(ck1);
add(ck2);
add(course);
add(sem);
add(age);
add(b1);
b1.addActionListener(this);
add(b1);
course.add("BSc cs");
course.add("MSc cs");
course.add("Bsc IT");
course.add("BCA");
course.add("MCA");
sem.add("1");
sem.add("2");
sem.add("3");
sem.add("4");
sem.add("5");
sem.add("6");
age.add("17");
age.add("18");
age.add("19");
age.add("20");
age.add("21");
```

```
l1.setBounds(25,65,90,20);
l2.setBounds(25,90,90,20);
l3.setBounds(25,120,90,20);
l4.setBounds(25,185,90,20);
l5.setBounds(25,260,90,20);
l6.setBounds(25,290,90,20);
l7.setBounds(25,260,90,20);
l11.setBounds(10,40,280,20);
t1.setBounds(120,65,170,20);
t2.setBounds(120,185,170,20);
ck1.setBounds(120,120,50,20);
ck2.setBounds(170,120,60,20);
course.setBounds(120,260,100,20);
sem.setBounds(120,290,50,20);
age.setBounds(120,90,50,20);
b1.setBounds(120,350,50,30);
}
public void paint(Graphics g)
{
g.drawString(msg,200,450);
}
public void actionPerformed(ActionEvent ae)
{
if(ae.getActionCommand().equals("save"))
{
msg="Thanks For Your Updates....!";
setForeground(Color.blue);
```

```
}
}
public static void main(String args[])
{
student stu = new student();
stu.setSize(new Dimension(500,500));
stu.setTitle("STUDENTS REGISTRATION FORM");
stu.setVisible(true);
}
}
class myWindowAdapter extends WindowAdapter
{
public void WindowClosing(WindowEvent we)
{
System.exit(0);
}
}
```

Output:



Result:

Thus the program to demonstrate the given components was designed and executed successfully.

Ex No : 02	Layouts
Date :	

Aim:

To create a Java program that demonstrate the use of various layouts like Flow layout, Border layout, Grid layout, Card layout with a JFrame

Procedure:

Step 1 : Start the program.

Step 2 : Define the main class layout example.

Step 3 : create a new JFrame with title layout example.

Step 4 : Set the default close operation to EXIT_ON_CLOSE.

Step 5 : Create a panel with different layout manager.

Step 6 : Add panels to the frame.

Step 7 : To run the source code.

Step 8 : To stop the program.

Source code:

1. Flow Layout

```
import javax.swing.*;
import java.awt.*;


public class FlowLayoutexample extends Frame
{
public static void main(String[] args)
{
```

```
JFrame frame=new JFrame("FlowLayoutexample");
frame.setLayout(new FlowLayout());

//add components
frame.add(new JButton("Button1"));
frame.add(new JButton("Button2"));
frame.add(new JButton("Button3"));

 frame.setSize(300,150);
 frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 frame.setVisible(true);
}
}
```

2. Border Layout

```
import javax.swing.*;
import java.awt.*;

public class BorderLayoutexample extends Frame
{
public static void main(String[] args)
{
JFrame frame=new JFrame ("BorderLayoutexample");
frame.setLayout(new BorderLayout());
//Add components
frame.add(new JButton("north"),BorderLayout.NORTH);
frame.add(new JButton("south"),BorderLayout.SOUTH);
frame.add(new JButton("East"),BorderLayout.EAST);
```

```
frame.add(new JButton("west"),BorderLayout.WEST);
frame.add(new JButton("center"),BorderLayout.CENTER);
frame.setSize(300,150);
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.setVisible(true);
}
}
```

3. Grid Layout

```
import javax.swing.*;
import java.awt.*;

public class GridLayoutexample extends Frame
{
public static void main(String[] args)
{
JFrame frame=new JFrame("GridLayoutexample");
frame.setLayout(new GridLayout());

//Add components
frame.add(new JButton("Button 1"));
frame.add(new JButton("Button 2"));
frame.add(new JButton("Button 3"));
frame.add(new JButton("Button 4"));

frame.setSize(300,150);
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.setVisible(true);
```

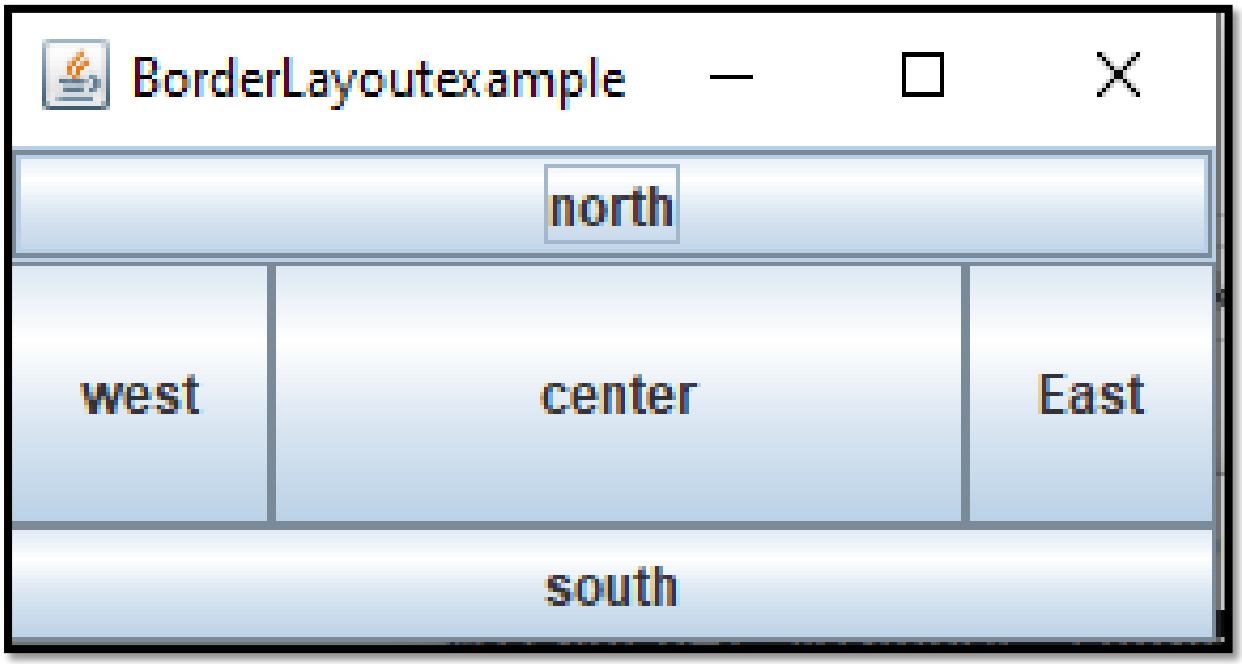
```
}
}
```

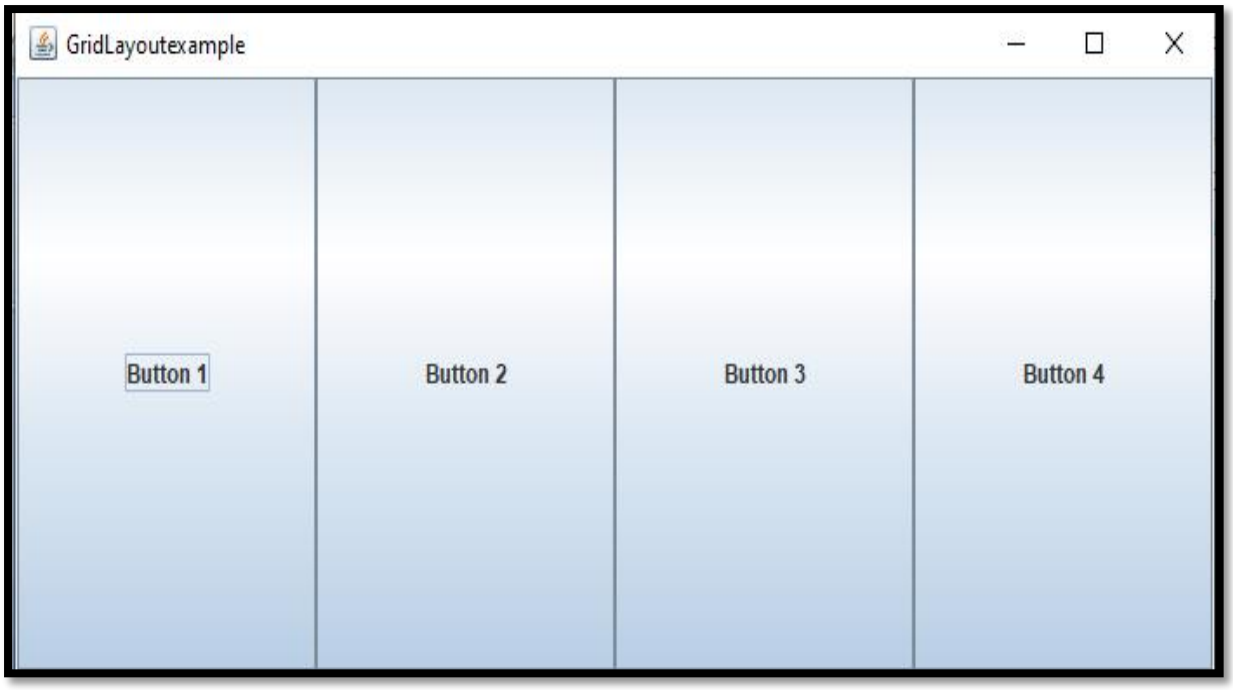
4. Card Layout

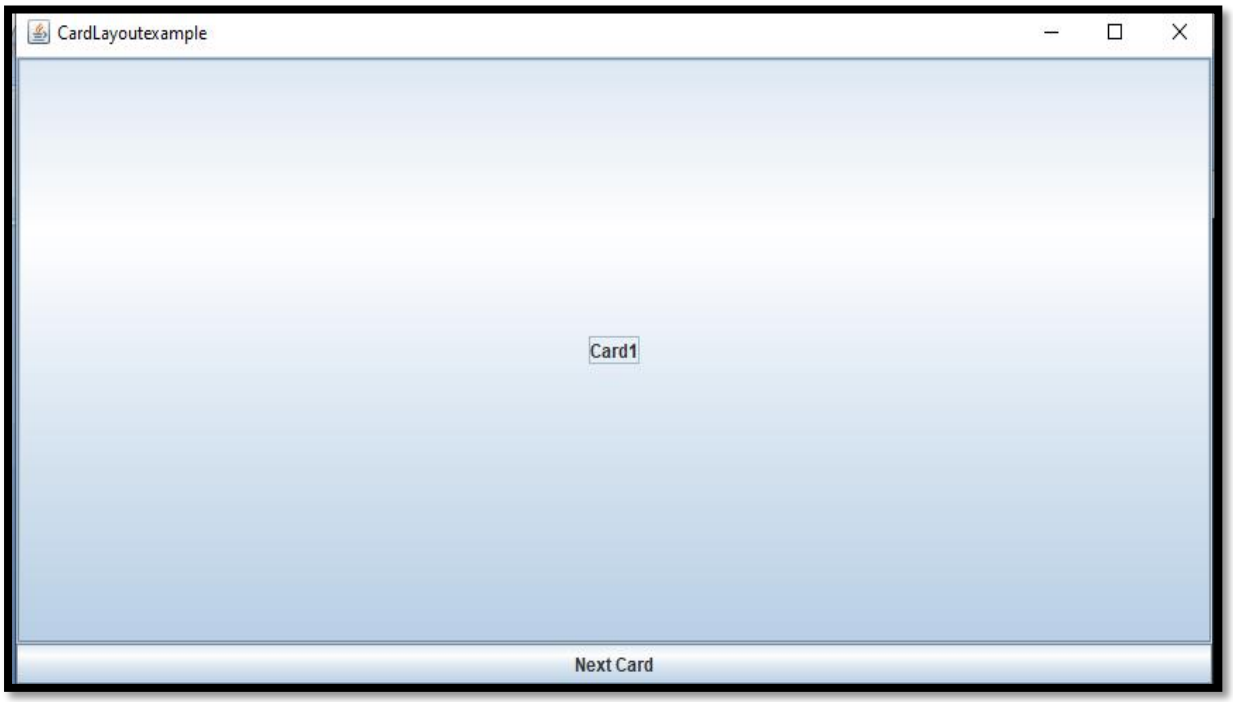
```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
public class CardLayoutexample extends Frame
{
public static void main(String[] args)
{
JFrame frame=new JFrame("CardLayoutexample");
CardLayout cardLayout=new CardLayout();
JPanel cardPanel=new JPanel(cardLayout);
//Add components
cardPanel.add(new JButton("Card1"),"Card1");
cardPanel.add(new JButton("Card2"),"Card2");
cardPanel.add(new JButton("Card3"),"Card3");
JButton nextButton=new JButton("Next Card");
nextButton.addActionListener(new ActionListener()
{
public void actionPerformed(ActionEvent e)
{
cardLayout.next(cardPanel);
}
});
frame.add(cardPanel,BorderLayout.CENTER);
```

```
frame.add(nextButton,BorderLayout.SOUTH);

frame.setSize(300,150);

frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.setVisible(true);

}

}
```

Output:







Result:

Thus the program to demonstrate the use of various layouts was designed and displayed successfully

Ex No : 03	Color Palette
Date :	

Aim:

To write programs in Java to create applets incorporating the following features:

I. Create a color palette with matrix of buttons

II. Set background and foreground of the control text area by selecting a color from color palette.

III. In order to select Foreground or background use check box control as radio buttons

Procedure:

Step 1 : Start the program.

Step 2 : Define the main class Platte.

Step 3 : Declare an array of buttons to set colors, two checkboxes for Foreground and background colors

Step 4 : Declare three panels, buttonpanel, palettepanel and checckpanel

Step 5 : Create the buttonpanel and set the layout to gridlayout of 3 X 3

Step 6 : Create a text area and change its font to desired one

Step 7 : write an empty itemStateChange() method

Step 8 : In the actionPerformed () Method get action command

Step 9 : To run the source code.

Step 10 : To stop the program.

Source code:

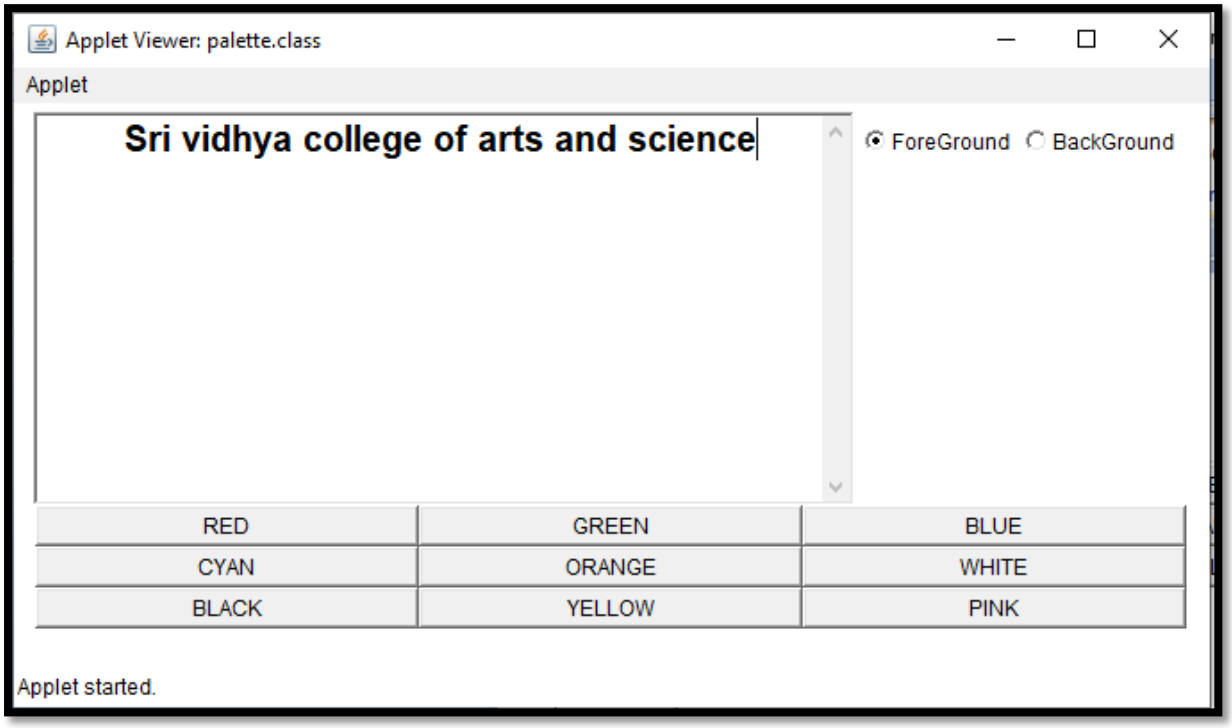
```
/* <applet code=palette.class , height=600 , width=600>

</applet> */

import java.awt.*;

import java.awt.event.*;
```

```
import java.applet.*;
public class palette extends Applet implements ActionListener,ItemListener
{
Button[] colors;
Checkbox foreground,background;
TextArea workarea;
CheckboxGroup cbg;
Panel buttonpanel,checkpanel,palettepanel;
String colour;
public void init()
{
buttonpanel=new Panel();
buttonpanel.setLayout(new GridLayout(3,3));
colors=new Button[9];
colors[0]=new Button("RED");
colors[1]=new Button("GREEN");
colors[2]=new Button("BLUE");
colors[3]=new Button("CYAN");
colors[4]=new Button("ORANGE");
colors[5]=new Button("WHITE");
colors[6]=new Button("BLACK");
colors[7]=new Button("YELLOW");
colors[8]=new Button("PINK");
for(int i=0;i<9;i++)
{
colors[i].addActionListener(this);
buttonpanel.add(colors[i]);
```

```
}
checkpanel=new Panel();
checkpanel.setLayout(new FlowLayout());
cbg=new CheckboxGroup();
foreground=new Checkbox("ForeGround",cbg,true);
background=new Checkbox("BackGround",cbg,false);
foreground.addItemListener(this);
background.addItemListener(this);
checkpanel.add(foreground);
checkpanel.add(background);
workarea=new TextArea(8,40);
workarea.setFont(new Font("Garamond",Font.BOLD,20));
palettepanel=new Panel();
palettepanel.setLayout(new BorderLayout());
palettepanel.add(workarea,BorderLayout.CENTER);
palettepanel.add(checkpanel,BorderLayout.EAST);
palettepanel.add(buttonpanel,BorderLayout.SOUTH);
add(palettepanel);
}
public void itemStateChanged(ItemEventie)
{
}
public void actionPerformed(ActionEventae)
{
colour=ae.getActionCommand();
if(foreground.getState()==true)
workarea.setForeground(getColour());
```

```
if(background.getState()==true)
workarea.setBackground(getColour());
}
public Color getColour()
{
Color mycolor=null;
if(colour.equals("RED"))
mycolor=Color.red;
if(colour.equals("GREEN"))
mycolor=Color.green;
if(colour.equals("BLUE"))
mycolor=Color.blue;
if(colour.equals("CYAN"))
mycolor=Color.cyan;
if(colour.equals("ORANGE"))
mycolor=Color.orange;
if(colour.equals("WHITE"))
mycolor=Color.white;
if(colour.equals("BLACK"))
mycolor=Color.black;
if(colour.equals("YELLOW"))
mycolor=Color.yellow;
if(colour.equals("PINK"))
mycolor=Color.pink;
return mycolor;
}
}
```

Output:



Result:

Thus the palette with various color styles using applet was entered and executed successfully.

Ex No : 04	Server Info Downloader
Date :	

Aim:

To write programs in Java to do the following.

I. Set the URL of another server.

II. Download the homepage of the server.

III. Display the contents of homepage with date, content type, and Expiration date. Last modified and length of the home page.

Procedure:

Step 1	:	Start the program.
Step 2	:	Define the main class ServerInfoDownloader.
Step 3	:	Using the URL class locate the server with the help of URL.
Step 4	:	Using the appropriate stream and open the home page of the Server.
Step 5	:	Display the contents of the homepage in the console.
Step 6	:	Using the method getDate(), getContentType(), getExpriationDtae(), getLastmodified() and getLength().
Step 7	:	To run the source code.
Step 8	:	To stop the program.

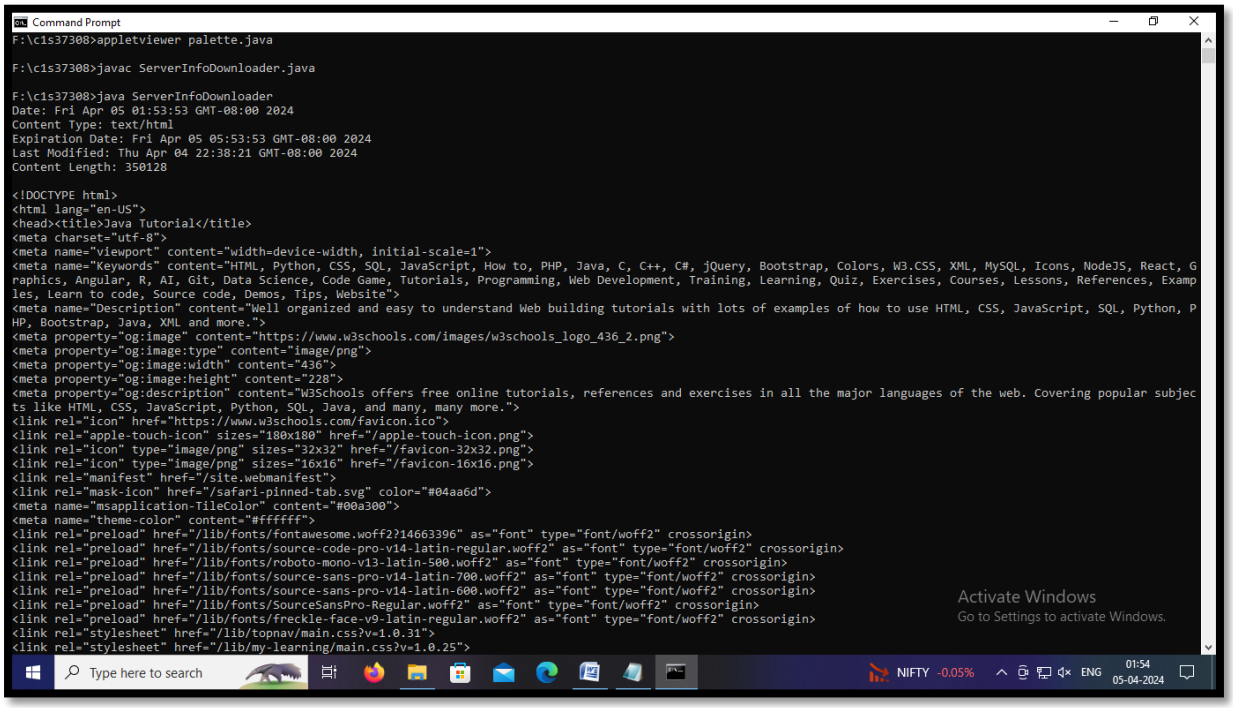
Source code:

```
import java.io.*;
import java.net.*;
import java.util.*;
```

```
public class ServerInfoDownloader
{
public static void main(String[] args)
{
try
{
        // Set the URL of another server
URL url = new URL("https://www.w3schools.com/java/");
        // Download the homepage of the server
HttpURLConnection connection = (HttpURLConnection) url.openConnection();
connection.setRequestMethod("GET");

        // Display the contents of homepage with date, content type, and expiration date
System.out.println("Date: " + new Date(connection.getDate()));
System.out.println("Content Type: " + connection.getContentType());
System.out.println("Expiration Date: " + new Date(connection.getExpiration()));
System.out.println("Last Modified: " + new Date(connection.getLastModified()));
System.out.println("Content Length: " + connection.getContentLength());
        // Read and display the content of the homepage
BufferedReader reader = new BufferedReader(new
InputStreamReader(connection.getInputStream()));
        String line;
while ((line = reader.readLine()) != null)
        {
System.out.println(line);
        }
reader.close();
connection.disconnect();
}
catch (MalformedURLException e)
```

```
{
e.printStackTrace();
}
catch (IOException e)
{
e.printStackTrace();
}
}
}
```

Output:



Result:

Thus the program to show the information about server was entered and executed successfully.

Ex No : 05	Simple Chat Application
Date :	

Aim:

To write a program in Java for creating simple chat application with datagram sockets and datagram packets.

Procedure:

Step 1 : Start the program.

Step 2 : Import the package java.net.

Step 3 : Declare the datagram sockets, datagram packets, buffer reader.

Step 4 : In the main function using while loop it performed

Step 5 : Terminate the UDP Client Program.

Step 6 : Terminate the UDP Server Program.

Step 7 : To run the source code.

Step 8 : To stop the program.

Source Code:

Server.Java

```
import java.io.*;
import java.net.*;
public class Server
{
  public static void main(String[] args) throws Exception
```

```
 {
    ServerSocket sersock = new ServerSocket(3000);
    System.out.println("Server  ready for chatting");
    Socket sock = sersock.accept( );
  BufferedReader keyRead = new BufferedReader(new InputStreamReader(System.in));
OutputStream ostream = sock.getOutputStream();
    PrintWriter pwrite = new PrintWriter(ostream, true);
    InputStream istream = sock.getInputStream();
    BufferedReader receiveRead = new BufferedReader(new InputStreamReader(istream));
    String receiveMessage, sendMessage;
    while(true)
    {
     if((receiveMessage = receiveRead.readLine()) != null)
     {
       System.out.println(receiveMessage);
     }
     sendMessage = keyRead.readLine();
     pwrite.println(sendMessage);
     pwrite.flush();
    }
  }
}
```

Client.Java

```
import java.io.*;
import java.net.*;
public class Client
{
```

```
 public static void main(String[] args) throws Exception
 {
   Socket sock = new Socket("127.0.0.1", 3000);
   BufferedReader keyRead = new BufferedReader(new InputStreamReader(System.in));
      OutputStream ostream = sock.getOutputStream();
   PrintWriter pwrite = new PrintWriter(ostream, true);
   InputStream istream = sock.getInputStream();
   BufferedReader receiveRead = new BufferedReader(new InputStreamReader(istream));
   System.out.println("Start the chat, type and press Enter key");
   String receiveMessage, sendMessage;
   while(true)
   {
     sendMessage = keyRead.readLine();
     pwrite.println(sendMessage);
       pwrite.flush();
     if((receiveMessage = receiveRead.readLine()) != null)
     {
        System.out.println(receiveMessage);
     }
    }
  }
}
```

Output:-

Result:

Thus the program for creating simple chat application with datagram sockets and datagram packets Entered and executed successfully.

Ex No : 06	Java Using Servlets
Date :	

Aim:

To write programs in Java using Servlets:

I. To invoke servlets from HTML forms

II. To invoke servlets from Applets

Procedure:

Step 1	:	Start the program.
Step 2	:	Create the PostParam.html.
Step 3	:	Save the PostParam.Html
Step 4	:	Create java servlet for invoke the html file.
Step 5	:	Open the web browser and type http://localhost:8080.
Step 6	:	Select tomcat manager, Deploy the ware file and run.
Step 7	:	To run the source code.
Step 8	:	To stop the program.

Source Code :

PostParam.Html

```
<html>
<body>
<center>
<FORM name = "postparam" method = "post"
action="http://localhost:8080/PostParam/PostParam">
```

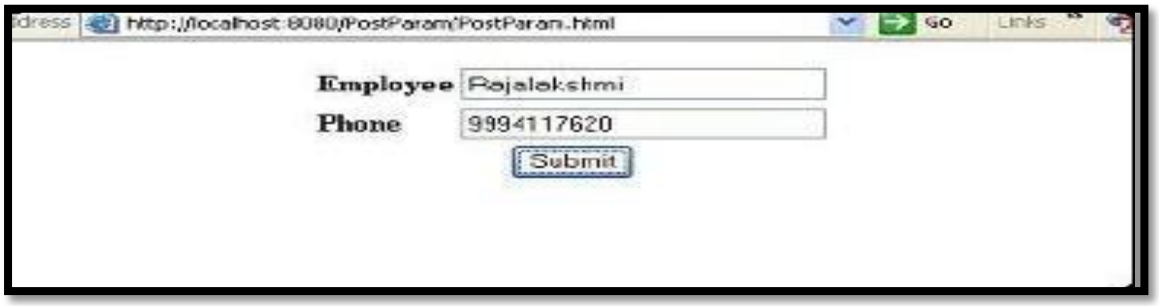
```
<TABLE>
<tr>
<td><B>Employee </B></td>
<td><input type = "textbox" name="ename" size="25"
value=""></td>
</tr>
<tr>
<td><B>Phone </B></td>
<td><input type = "textbox" name="phoneno" size="25"
value=""></td>
</tr>
</TABLE>
<INPUT type = "submit" value="Submit">
</body>
</html>
```

PostParam.java

```
import java.io.*;
import java.util.*;
import javax.servlet.*;
public class PostParam extends GenericServlet
{
public void service(ServletRequest request,ServletResponse response) throws
ServletException, IOException
{
PrintWriter pw = response.getWriter();
Enumeration e = request.getParameterNames();
while(e.hasMoreElements())
{
String pname = (String)e.nextElement();
pw.print(pname + " = ");
String pvalue = request.getParameter(pname);
pw.println(pvalue);
}
```

```
pw.close();
}
}
```

Output:



Result:

Thus the above program in java using servlets Entered and executed successfully.

Ex No : 07	Three Tier Applications Using Servlet
Date :	

Aim:

To write programs in Java to create three-tier applications using servlets for conducting on-line examination for displaying student mark list. Assume that student information is available in a database which has been stored in a database server.

Procedure:

Step 1 : Start the program.

Step 2 : Create the Indext.Jsp.

Step 3 : Save the Indext.Jsp.

Step 4 : The form tag action=”http://localhost:8080/example/servlet/exam”..

Step 5 : Import the necessary packages and declare class, class name in exam.

Step 6 : Declare the connection, statement and result set object.

Step 7 : To run the source code.

Step 8 : To stop the program.

Source Code:

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(); Stringname=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 Systemisa ..... </p>");
out.println("<input type=radio name=q1value=0>Hardware"); out.println("<br>");
out.println("<input type=radio name=q1value=1>Software"); out.println("<hr>");
out.println("<p>2.Developer of CLanguageis ..... </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,multiuser,multiprocessing);
out.println("OS. ..... </p>");
out.println("<br>");
out.println("<input type=radio name=q3 value=0>MSDOS"); out.println("<br>");
```

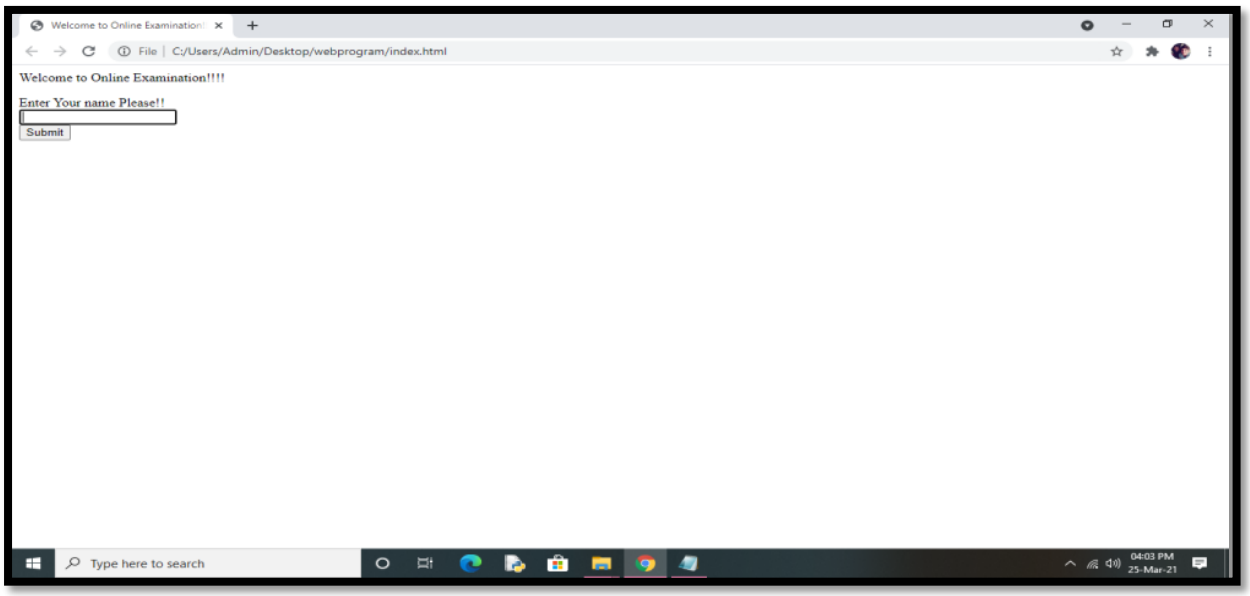
```
out.println("<input type=radio name=q3 value=1>Windows NT"); out.println("<hr>");
out.println("<p>4.Father ofComputersis ..... </p>");
out.println("<br>");
out.println("<input type=radio name=q4 value=1>Charles babbage"); out.println("<br>");
out.println("<input type=radio name=q4 value=0>Charles 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=q5value=0>Fifth"); out.println("<br>");
out.println("<input type=radio name=q5value=1>Sixth"); out.println("<hr>");
out.println("<input type=submitvalue=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>ExaminationResults</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 above program in java to create three-tier applications using servlets for conducting on-line examination Entered and executed successfully.

Ex No : 08	Web Page creation using map fix the hot Spots
Date :	

Aim:

To create a web page with the following using HTML

I. To embed a map in a web page

II. To fix the hot spots in that map

III. Show all the related information when the hot spots are clicked.

Procedure:

Step 1 : Start the program.

Step 2 : Create the html file with maptag.

Step 3 : Set the source attributes of the image tag to the location of the image.

Step 4 : Specify an area with name, shape and href set of the appropriate Value.

Step 5 : Repeat step3 as many hot spots you want to put in the map.

Step 6 : Create html file for each and every hot spots the user will select the Particular location it shows information about it.

Step 7 : To run the source code.

Step 8 : To stop the program.

Source Code:

Indiamap.HTML

```
<!DOCTYPE html>
<html>
<body>
<h1>The map and area elements</h1>
<p>Click on the Hotspots to go to a new page and read more about the Places:</p>
<img src="india.jpg" alt="Workplace" usemap="#workmap" width="209" height="242">
<map name="workmap">
<area shape="rect" coords="83,41,57,63" alt="delhi" href="delhi.html">
<area shape="rect" coords="149,129,124,83" alt="calcutta" href="calcutta.html">
<area shape="rect" coords="89,184,56,226" alt="tamilnadu" href="tamilnadu.html">
</map>
</body>
</html>
```

Tamilnadu.HTML

```
<html>
<body bgcolor="cyan">
<font face="Times New Roman" size="10" color="orange">
<center>Chennai is the capital of Tamil Nadu<br> and <br>More IT companies are camped at
Chennai</center>
<a href="Indiamap.html">Home Page</a>
</font>
</body>
</html>
```

Delhi.HTML

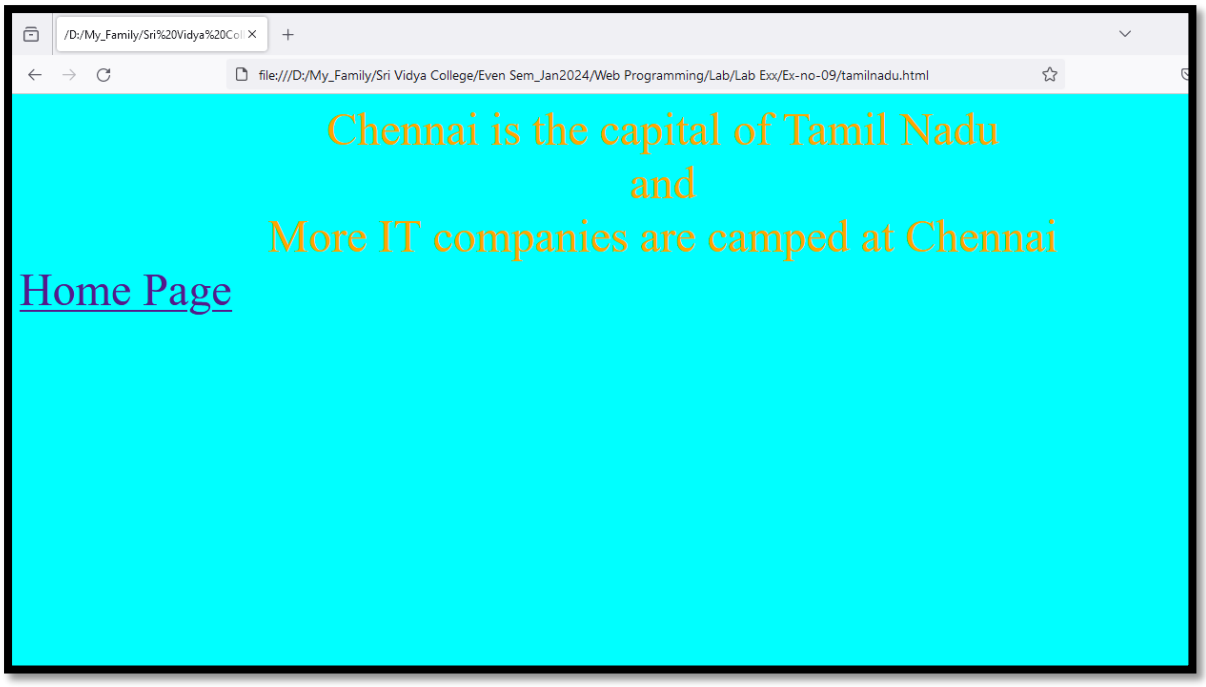
```
<html>
<body bgcolor="cyan">
<font face="Times New Roman" size="10" color="orange">
<center>Chennai is the capital of Tamil Nadu<br> and <br>More IT companies are camped at Chennai</center>
<a href="Indiamap.html">Home Page</a>
</font>
</body>
</html>
```

Calcutta.HTML

```
<html>
<body bgcolor="SKYBLUE">
<font face="Times New Roman" size="10" 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="Indiamap.html">Home Page</a>
</font>
</body>
</html>
```

Output:


/D:/My_Family/Sri%20Vidya%20Col
file:///D:/My_Family/Sri Vidya College/Even Sem_Jan2024/Web Programming/Lab/Lab Exx/Ex-no-09/Indiamap.html
The map and area elements
Click on the Hotspots to go to a new page and read more about the Places:


/D:/My_Family/Sri%20Vidya%20Col
file:///D:/My_Family/Sri Vidya College/Even Sem_Jan2024/Web Programming/Lab/Lab Exx/Ex-no-09/tamilnadu.html
Chennai is the capital of Tamil Nadu
and
More IT companies are camped at Chennai
Home Page

Result:

Thus the above web page are created and fix hotspots designed and executed successfully.

Ex No : 09	Web Page creation Using Cascading Style Sheets
Date :	

Aim:

To create a web page with the following.

I. Cascading style sheets.

II. Embedded style sheets.

III. Inline style sheets. Use our college information for the web pages.

Procedure:

Step 1 : Start the program.

Step 2 : Create the html document using Style Sheets information

Step 3 : Create a CSS file which contains the course details and link it to the Document using the <link> tag by applying external style concept.

Step 4 : Describe the current sem subjects in our own way of following by Applying internal style concept

Step 5 : To run the source code.

Step 6 : To stop the program.

Source Code :

Index.html

```
<html>
<frameset rows="40%,*">
<frame src="title.html" name="f1">
<frameset cols="35%,*">
<frame src="link.html" name="f2">
```

```
<frame name="f3">
</frameset>
</frameset>
</html>
```

Title.Html

```
<html>
<head><title>index</title>
</head>
<body bgcolor="magenta">
<center><h1>SRI VIDHYA COLLEGE OF ARTS & SCIENCE</h1>
<h2>Affiliated to Madurai Kamaraj University</h2>
<h3>Virudhunagar-626005</h3>
</center>
</body>
</html>
```

Link.HTML

```
<html>
<head><title>linking...</title>
</head>
<body bgcolor="Yellow">
<center>
<h2>DO U WANT TO KNOW...</h2>
<br>
<br>
<a href="dep.html" target="f3">DEPARTMENT</a><br>
```

```
<a href="faci.html" target="f3">FACILITIES</a><br>
<a href="cele.html" target="f3">CELEBRATION</a><br>
<a href="image.html" target="f3">GALLERY</a><br>
<a href="fees.html" target="f3">FEES</a><br>
<a href="add.html" target="f3">ADDRESS</a><br>
<br>
<br>
</body>
</html>
```

Dep.HTML

```
<html>
<head>
<title>Working with Style Sheets</title>
<style type="text/css">
h1
{
font-family:monotypecorsiva
}
h2
{
background-color:gray;background-repeat:repeat-x
}
p {font-size:12pt;font-weight:bold;color:#23238e;border-style:groove}
u1 {list-style-type:lower-roman}
</style>
</head>
<body bgcolor="red">
```

```
<center>
<h1>WELCOME TO OUR COLLEGE DEPARTMENTS</h1>

<h1><u>UG COURSES</u></h1>
<ul>
<li>B.Sc(Computer Science)</li>
<li>B.Sc(Physics)</li>
<li>B.Sc(Chemistry)</li>
<li>B.Sc(Maths)</li>
<li>BCA</li>
<li>BA(Tamil)</li>
<li>BA(English)</li>
<li>BA(History)</li>
<li>B.Com CA</li>
<li>B.Com</li>
<li>B.Com (CS)</li>
</ul>
<h1><u>PG COURSES</u></h1>
<ul>
<li>M.Sc(Computer Science)</li>
<li>M.Com</li>
</ul>
<br><br><br>
<a href ="fees.html">FEES</a>
<h1><marquee behaviour="right">Thanks for visiting.....</marquee></h1>
</body>
</html>
```

FACI.HTML

```
<html>
<head>
<title>external style sheets</title>
<link rel=stylesheet href="external.css">
</head>
<body bgcolor="cyan">
<center><h2>OUR COLLEGE FACILITIES</h2></center>Facilities are.....<br>
<ul type="1" color="blue">
<li>Hostel for Girls & Boys</li>
<li>Canteen</li>
<li>Library</li>
<li>lab Facility</li>
</ul>
</body>
</html>
```

Cele.HTML

```
<html>
<head>
<title>Working with style sheets</title>
<style type="text/css">
h2 {font-family:Playbill;background-color:red}
ul {list-style-type:lower-roman}
</style>
</head>
<body bgcolor="orange">
<center><h2>Our College Celebrations.....</h2></center>
```

```
<br> Celebrations are....<br>
<ul>
<li>Associations</li>
<li>Festivals</li>
<li>Cerificate Course</li>
<li>Talent Show</li>
</ul>
<br><br><br>
<h2><marquee behaviour="right">Thank You....</marquee></h2>
</body>
</html>
```

Image.HTML

```
<html>
<head>
<title>external style sheets</title>
<link rel=stylesheet href="external.css">
</head>
<body>
<center><h2>....IMAGES....</h2><center>
<br>
<center>Our College Lab....</center>
<br><br><br>
<img src="lab.jpeg">
<center>Our Library....</center>
<br><br><br>
<img src="Class.jpeg">
<center>Our ClassRoom....</center>
```

```
<br><br><br>
<img src="Exter.jpeg">
</body>
</html>
```

Fees.HTML

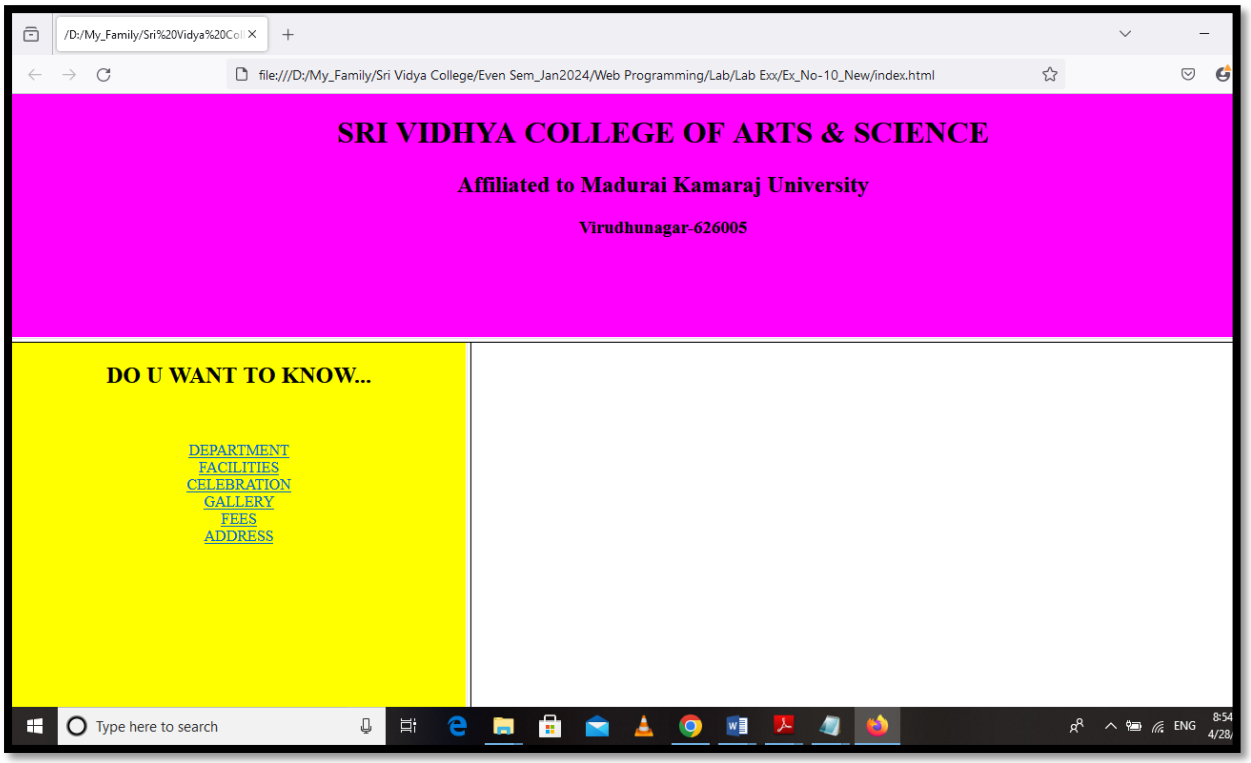
```
<html>
<head>
<title>fees</title>
<h1 style-color="cyan" font size=10% font family="Small Fonts"></h1>
</head>
<body bgcolor="brown">
<center><h1>COLLEGE FEES</h1></center>
<table border=5 align="center" BORDERCOLOR="#ac6d07" bgcolor="gray"
width="50%" height="50%">
<h2><tr><th align="center" bgcolor="red">DEGREE</h2>
<h2><th align="center" bgcolor="blue">FEES</h2>

<center><tr><td>B.Sc CS</td><td>13500</td></tr></center>
<center><tr><td>B.Sc Physics</td><td>12500</td></tr></center>
<center><tr><td>B.Sc Chemistry</td><td>14500</td></tr></center>
<center><tr><td>B.ScMaths</td><td>11500</td></tr></center>
<center><tr><td>BCA</td><td>14500</td></tr></center>
<center><tr><td>BA(Tamil)</td><td>11000</td></tr></center>
<center><tr><td>BA(English)</td><td>11000</td></tr></center>
<center><tr><td>BA(History)</td><td>11000</td></tr></center>
<center><tr><td>B.Com CA</td><td>14500</td></tr></center>
<center><tr><td>B.Com</td><td>10500</td></tr></center>
```

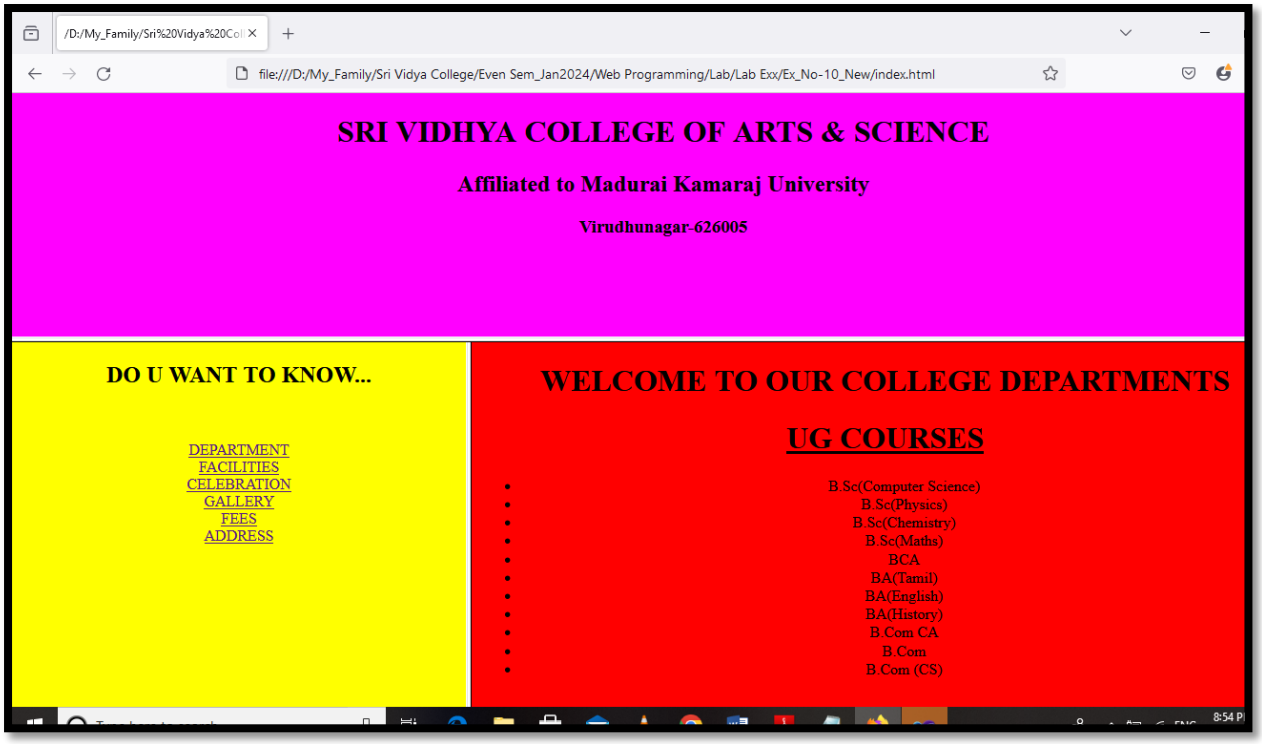
```
<center><tr><td>B.Com (CS)</td><td>11500</td></tr></center>
<center><tr><td>M.Sc CS</td><td>16000</td></tr></center>
<center><tr><td>M.Com</td><td>13500</td></tr></center>
</table>
</body>
</html>
```

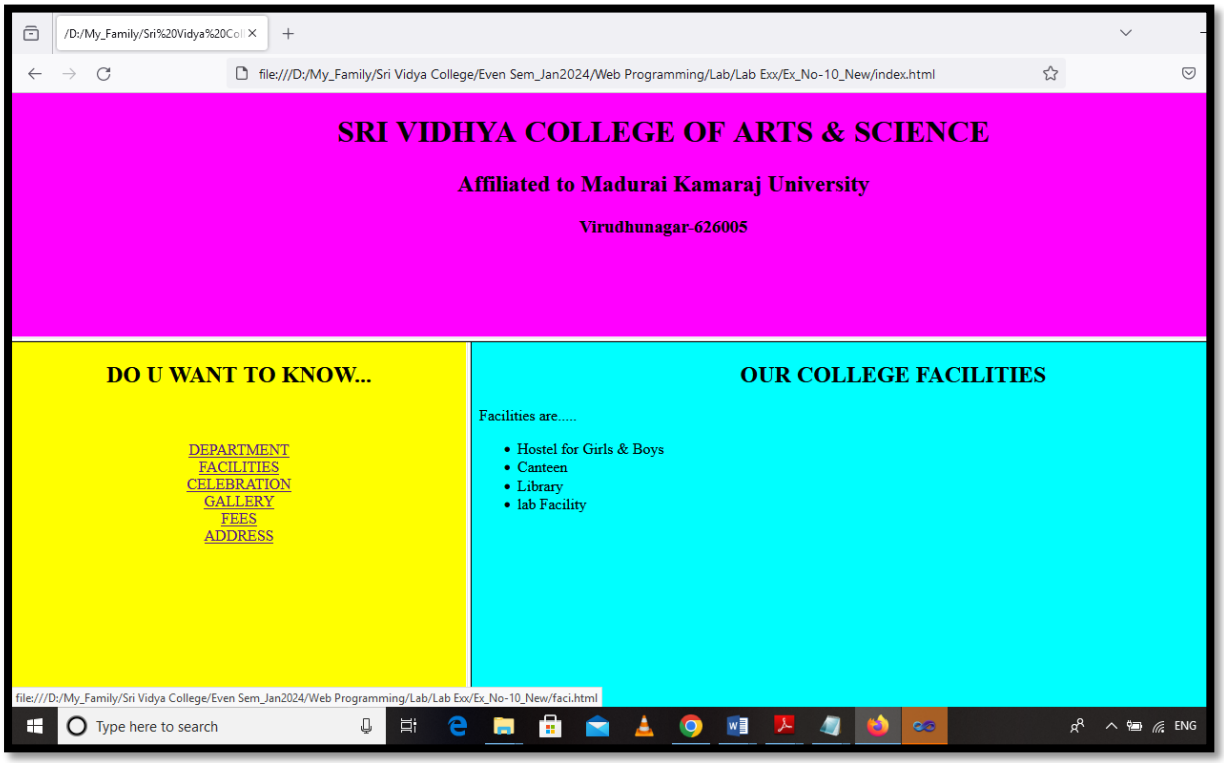
Add.HTML

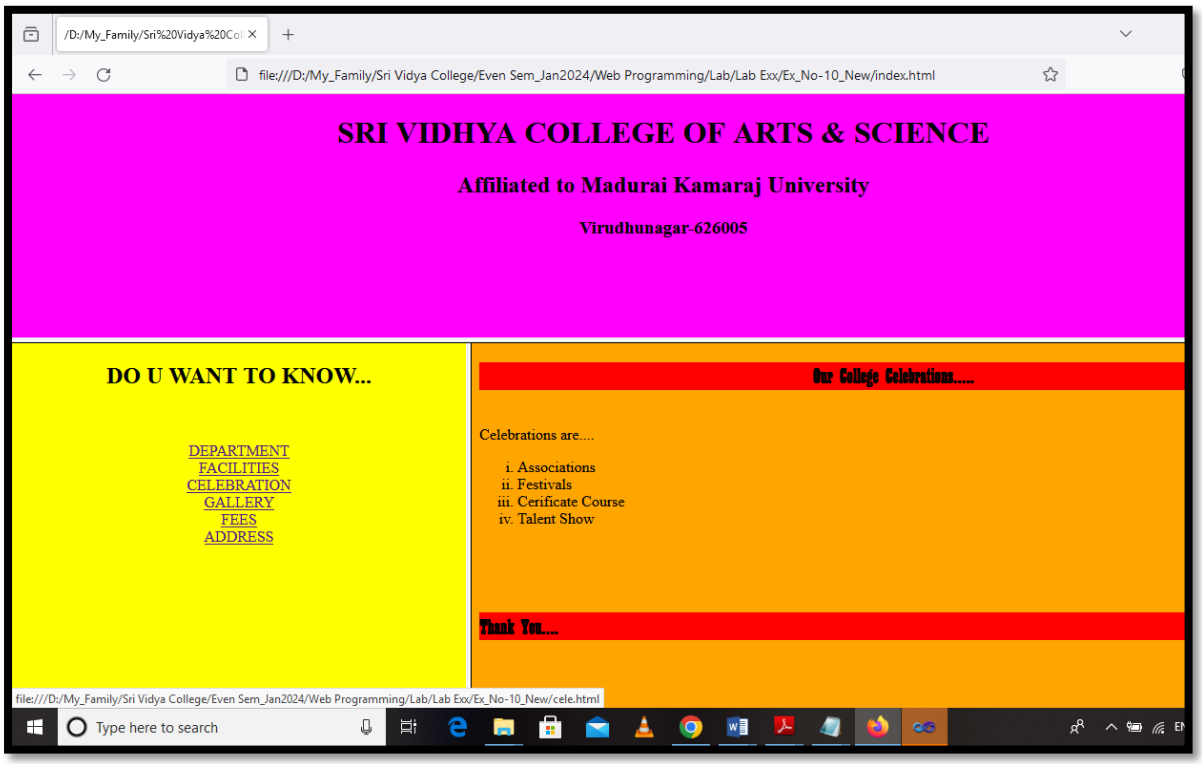
```
<html>
<head>
<title>working wid style sheet</title>
<style type="text/css">
h1 {font-family:RockWell;background-color:red}
</style>
</head>
<body bgcolor="pink">
<center><h1>Our Colege Address.....</h1></center>
<br>
<h1><u>Address.....</u></h1><br>
Sri Vidhya College of Arts & Science,<br>
Kumarlingapuram,<br>
Virudhunagar 626 005<br>
Ph:7094465623, 8870882201<br>
email id:srividhyaartssci@gmail.com
<h1><marquee behaviour="right">Thanking You.....</marquee></h1>
</body>
</html>
```

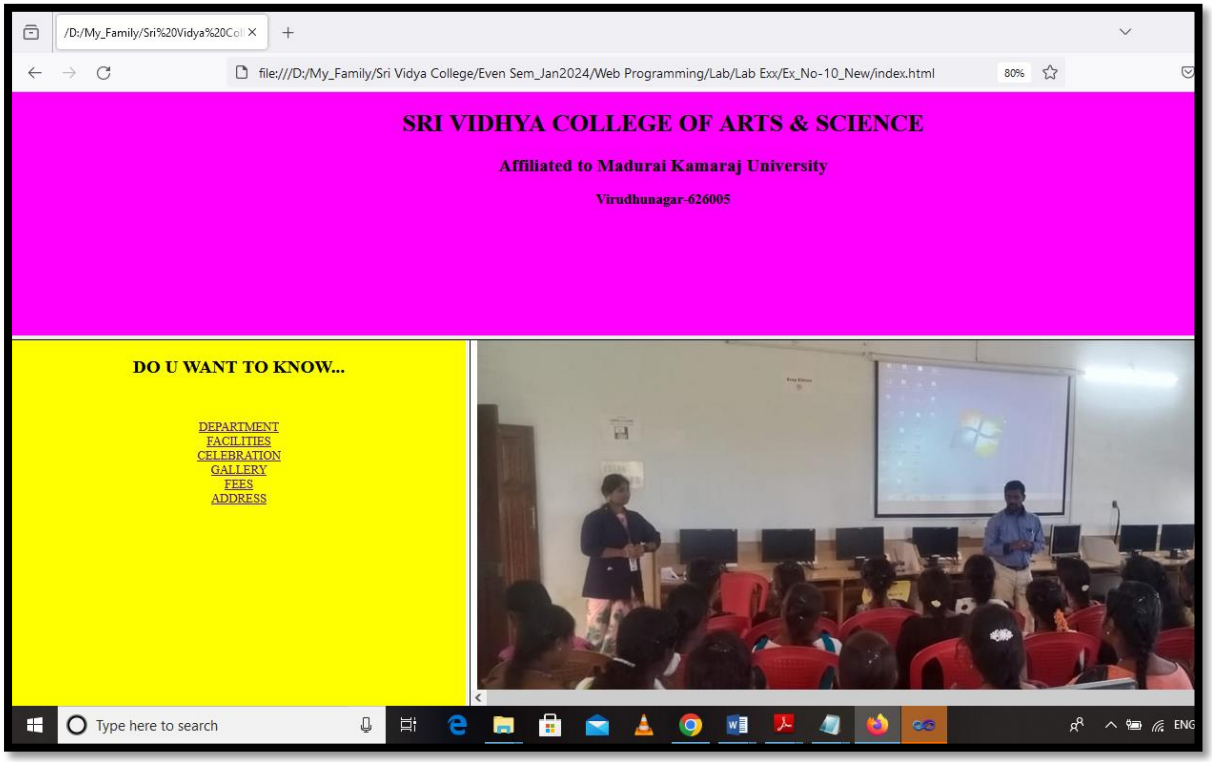
Output:

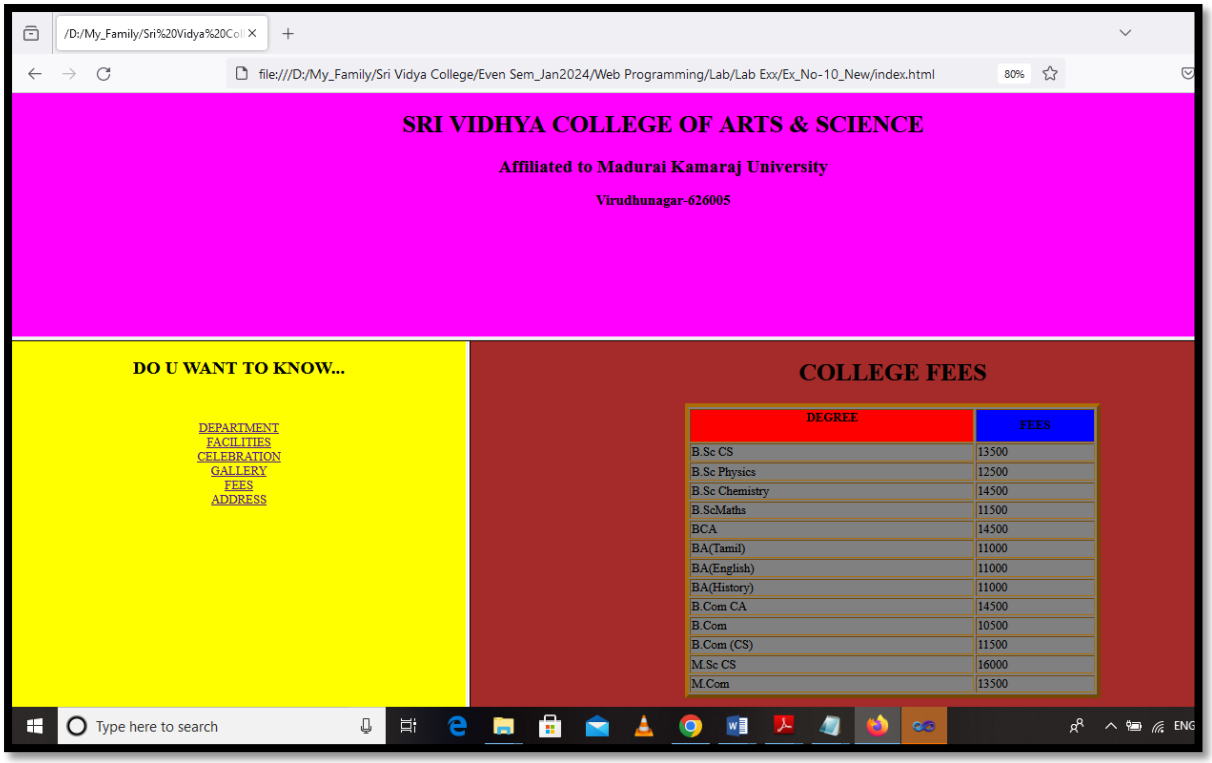

/D:/My_Family/Sri%20Vidya%20Col
file:///D:/My_Family/Sri Vidya College/Even Sem_Jan2024/Web Programming/Lab/Lab Exx/Ex_No-10_New/index.html
SRI VIDHYA COLLEGE OF ARTS & SCIENCE
Affiliated to Madurai Kamaraj University
Virudhunagar-626005
DO U WANT TO KNOW...
DEPARTMENT
FACILITIES
CELEBRATION
GALLERY
FEES
ADDRESS
Type here to search
ENG
8:54
4/28/

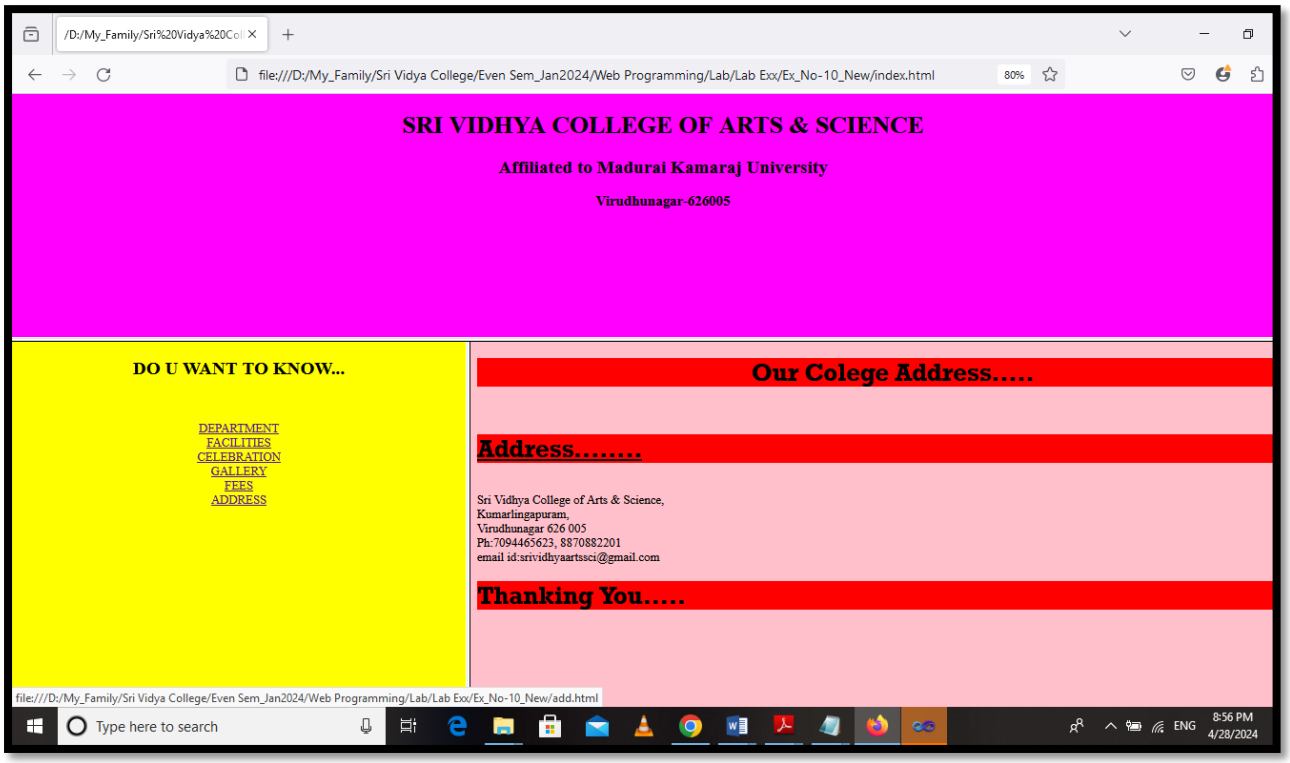

/D:/My_Family/Sri%20Vidya%20Col
file:///D:/My_Family/Sri Vidya College/Even Sem_Jan2024/Web Programming/Lab/Lab Exx/Ex_No-10_New/index.html
SRI VIDHYA COLLEGE OF ARTS & SCIENCE
Affiliated to Madurai Kamaraj University
Virudhunagar-626005
DO U WANT TO KNOW...
DEPARTMENT
FACILITIES
CELEBRATION
GALLERY
FEES
ADDRESS
WELCOME TO OUR COLLEGE DEPARTMENTS
UG COURSES
B.Sc(Computer Science)
B.Sc(Physics)
B.Sc(Chemistry)
B.Sc(Maths)
BCA
BA(Tamil)
BA(English)
BA(History)
B.Com CA
B.Com
B.Com (CS)


/D:/My_Family/Sri%20Vidya%20Col
file:///D:/My_Family/Sri Vidya College/Even Sem_Jan2024/Web Programming/Lab/Lab Exx/Ex_No-10_New/index.html
SRI VIDHYA COLLEGE OF ARTS & SCIENCE
Affiliated to Madurai Kamaraj University
Virudhunagar-626005
DO U WANT TO KNOW...
DEPARTMENT
FACILITIES
CELEBRATION
GALLERY
FEES
ADDRESS
OUR COLLEGE FACILITIES
Facilities are.....
Hostel for Girls & Boys
Canteen
Library
lab Facility
file:///D:/My_Family/Sri Vidya College/Even Sem_Jan2024/Web Programming/Lab/Lab Exx/Ex_No-10_New/faci.html
Type here to search
ENG


/D:/My_Family/Sri%20Vidya%20Col
file:///D:/My_Family/Sri Vidya College/Even Sem_Jan2024/Web Programming/Lab/Lab Exx/Ex_No-10_New/index.html
SRI VIDHYA COLLEGE OF ARTS & SCIENCE
Affiliated to Madurai Kamaraj University
Virudhunagar-626005
DO U WANT TO KNOW...
DEPARTMENT
FACILITIES
CELEBRATION
GALLERY
FEES
ADDRESS
Our College Celebrations.....
Celebrations are....
i. Associations
ii. Festivals
iii. Cerificate Course
iv. Talent Show
Thank You....
file:///D:/My_Family/Sri Vidya College/Even Sem_Jan2024/Web Programming/Lab/Lab Exx/Ex_No-10_New/cele.html
Type here to search


/D:/My_Family/Sri%20Vidya%20Col
file:///D:/My_Family/Sri Vidya College/Even Sem_Jan2024/Web Programming/Lab/Lab Exx/Ex_No-10_New/index.html
80%
SRI VIDHYA COLLEGE OF ARTS & SCIENCE
Affiliated to Madurai Kamaraj University
Virudhunagar-626005
DO U WANT TO KNOW...
DEPARTMENT
FACILITIES
CELEBRATION
GALLERY
FEES
ADDRESS
Type here to search
ENG


/D:/My_Family/Sri%20Vidya%20Col
file:///D:/My_Family/Sri Vidya College/Even Sem_Jan2024/Web Programming/Lab/Lab Exx/Ex_No-10_New/index.html
80%
SRI VIDHYA COLLEGE OF ARTS & SCIENCE
Affiliated to Madurai Kamaraj University
Virudhunagar-626005
DO U WANT TO KNOW...
DEPARTMENT
FACILITIES
CELEBRATION
GALLERY
FEES
ADDRESS
COLLEGE FEES
DEGREE
FEES
B.Sc CS 13500
B.Sc Physics 12500
B.Sc Chemistry 14500
B.ScMaths 11500
BCA 14500
BA(Tamil) 11000
BA(English) 11000
BA(History) 11000
B.Com CA 14500
B.Com 10500
B.Com (CS) 11500
M.Sc CS 16000
M.Com 13500
Type here to search
ENG



Result:

Thus the above web page are created and Inline, Embedded CSS File designed and executed successfully.