\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Create a webpage with multiline textbox and two buttons, viz. saveContents and loadContents. On click of saveContents button, contents from the textbox should be retained and on click of loadContents button, the previously saved contents should be displayed back on the textbox.

CODE: using System;

namespace list {

public partial class listselect : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void btnAdd\_Click(object sender, EventArgs e)

{ int i; for (i = 0; i < lstEmployee.Items.Count; i++)

{

if (lstEmployee.Items[i].Selected == true)

txtEmployee.Text += lstEmployee.Items[i].Text + "\n";

}

} } }

* Store 3 objects of the furniture class having 3 data members (name, manufacturer, and cost) in 3 session objects. Display a panel to include a listbox displaying the names of all three furniture objects, and a button named "More Information". On click of the button retrieve the selected object (from listbox) information and display it in a label.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Display the no. of visitors on a given web page.
* Create a registration form having text fields for accepting, Name, Age, Email, Address and Mobile number. Perform the following validations for the same:

1. All fields need to be filled compulsorily

2. Name should contain only alphabets and should not be more

than 25 characters long 3. Validate email and mobile number appropriately

4. Age should be between 18 and 32 only. Include submit and cancel buttons. On click of submit button, open a new page and display all the information entered by the user and on click of cancel button, all text fields should be cleared.

CODE:

ValidateControlForm.aspx

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace ValidationControl

{

public partial class ValidationControlForm : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void CustomValidator1\_ServerValidate(object source, ServerValidateEventArgs args)

{

string str = args.Value;

args.IsValid = false;

if (str.Length < 7 || str.Length > 20)

{

return;

}

bool capital = false;

foreach (char ch in str)

{

if (ch >= 'A' && ch <= 'Z')

{

capital = true;

break;

}

}

if (!capital)

return;

bool digit = false;

foreach (char ch in str)

{

if (ch >= '0' && ch <= '9')

{

digit = true; break;

}

}

if (!digit)

return;

args.IsValid = true;

}

protected void btnSubmit\_Click(object sender, EventArgs e)

{

}

} }

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Create a simple web page to demonstrate use of built in DivideByZeroException and IndexOutOfRangeException exceptions using textbox and label control.
* Create a simple web page to show data in Tree view control and datalist using web. Sitemap file containing navigation information

Default.aspx:

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="YourNamespace.Default" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title>Web Page with Tree View and DataList</title>

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

</head>

<body>

<form id="form1" runat="server">

<header>

<h1>Web Page Example</h1>

</header>

<nav>

<asp:SiteMapDataSource ID="SiteMapDataSource1" runat="server" ShowStartingNode="false" />

<asp:Menu ID="Menu1" runat="server" DataSourceID="SiteMapDataSource1" />

</nav>

<main>

<section id="treeview">

<h2>Tree View Control</h2>

<asp:TreeView ID="TreeView1" runat="server">

<!-- Tree view nodes will be added in code-behind -->

</asp:TreeView>

</section>

<section id="datalist">

<h2>Data List</h2>

<asp:DataList ID="DataList1" runat="server">

<!-- Data list items will be added in code-behind -->

</asp:DataList>

</section>

</main>

</form>

</body>

</html>

Default.aspx.cs

using System;

namespace YourNamespace

{

public partial class Default : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

if (!IsPostBack)

{

// Populate TreeView

PopulateTreeView();

// Populate DataList

PopulateDataList();

}

}

private void PopulateTreeView()

{

// You can add nodes to the TreeView dynamically

// For simplicity, let's add a static node

TreeNode rootNode = new TreeNode("Root Node");

TreeView1.Nodes.Add(rootNode);

}

private void PopulateDataList()

{

// You can bind the DataList to a data source

// For simplicity, let's add static data

string[] data = { "Item 1", "Item 2", "Item 3" };

DataList1.DataSource = data;

DataList1.DataBind();

}

}

}

Styles.css

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

}

header {

background-color: #4CAF50;

color: white;

text-align: center;

padding: 1em;

}

nav {

background-color: #333;

color: white;

}

nav a {

display: block;

color: white;

text-align: center;

padding: 14px 16px;

text-decoration: none;

}

nav a:hover {

background-color: #ddd;

color: black;

}

main {

padding: 20px;

}

section {

margin-bottom: 30px;

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Write an Application to:

1. Generate Fibonacci series

2. Test for prime numbers

3. Test for vowels

4. Reverse a number

Generate Fibonacci numbers

CODE:

using System;

namespace ConsoleApplication3

{

class Program

{ static void Main(string[] args)

{ int num1=0,num2=1,num3,num4,num,counter;

Console.Write ("Upto how many number you want fibonacci series:"); num=int.Parse(Console.ReadLine());

counter=3;

Console.Write(num1+"\t"+num2);

while(counter<=num)

{

num3 = num1 + num2;

if (counter >= num)

break;

Console.Write("\t" + num3);

num1 = num2;

num2 = num3;

counter++;

}

}

}

}

OUTPUT:

Test for prime numbers

CODE:

using System;

namespace testprime

{

class Program

{

static void Main(string[] args)

{

int num, counter;

Console.Write("Enter number:");

num = int.Parse(Console.ReadLine());

for (counter = 2;

counter <= num / 2;

counter++)

{

if ((num % counter) == 0)

break;

}

if (num == 1)

Console.WriteLine(num + "is neither prime nor composite");

else if (counter<(num/2))

Console.Writeline(num+”is not a prime number”);

Else

Console.writeline(num+”is a prime number”);

}

}

}

Test for vowels

CODE: using System;

namespace vowels

{

class Program

{

static void Main(string[] args)

{

char ch;

Console.Write("Enter a character : ");

ch = (char)Console.Read();

switch (ch)

{

case 'a':

case 'A':

case 'e':

case 'E':

case 'i':

case 'I':

case 'o':

case 'O':

case 'u':

case 'U':

Console.WriteLine(ch + "is vowel");

break;

default: Console.Write(ch + "is not a vowel");

break;

}

Console.ReadKey();

}

}

}

Reverse a number

CODE:

using System;

namespace reverseNumber

{

class Program

{

static void Main(string[] args)

{

int num,actualnumber,revnum=0,digit,sumDigits=0;

Console.Write("Enter number:");

num = int.Parse(Console.ReadLine());

actualnumber = num;

while (num > 0)

{

digit = num % 10;

revnum = revnum \* 10 + digit;

sumDigits=sumDigits+digit;

num = num / 10;

}

Console.WriteLine("Reverse of " + actualnumber + "=" + revnum);

Console.WriteLine("Sum of its digits:" + sumDigits);

}

}

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Create a web application to bind data in a multiline textbox by querying in another textbox.
* Write a program to create a DLL to print a factorial of a number.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Create a web applications to demonstrate Form Security and Windows Security with proper Authentication and Authorization properties
* Create a web application to demonstrate GridView paging and Creating own table format using GridView.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Create a web application to demonstrate use of HtmlEditorExtender Ajax control.

Create a web application to demonstrate use of Master Page with applying Styles and Themes for page beautification.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Create a delegate dell. Create display1() and display2() static methods. Create a simple application to call these two methods by the through the delegate.

CODE:

TrafficSignal.cs

using System;

namespace TrafficDelegateExample

{

public delegate void TrafficDel();

class TrafficSignal

{

public static void Yellow()

{

Console.WriteLine("Yellow light signals to get ready");

}

public static void Green()

{

Console.WriteLine("Green light signals to go");

}

public static void Red()

{

Console.WriteLine("Red light signals to stop");

}

TrafficDel[] td = new TrafficDel[3];

public void IdentifySignal()

{

td[0] = new TrafficDel(Yellow);

td[1] = new TrafficDel(Green);

td[2] = new TrafficDel(Red);

}

public void display()

{

td[0]();

td[1]();

td[2]();

} } }

Program.cs

using System;

namespace TrafficDelegateExample

{

class Program

{

static void Main(string[] args)

{

TrafficSignal ts = new TrafficSignal();

ts.IdentifySignal();

ts.display();

} } }

* Create a simple web page to show how to write and read a cookie from a client's computer.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Create a simple web page to demonstrate all string operations.

StringOperation.cs

<%@ Page Language="C#" %>

<!DOCTYPE html>

<html>

<head>

<title>String Operations Demo</title>

</head>

<body>

<form id="form1" runat="server">

<div>

<h1>String Operations Demo</h1>

<asp:TextBox ID="txtInput" runat="server" placeholder="Enter a string"></asp:TextBox>

<asp:Button ID="btnSubmit" runat="server" Text="Submit" OnClick="btnSubmit\_Click" />

<hr />

<h2>Results:</h2>

<asp:Label ID="lblOutput" runat="server"></asp:Label>

</div>

</form>

</body>

</html>

Strings.aspx

using System;

using System.Web.UI;

public partial class StringOperationsDemo : Page

{

protected void btnSubmit\_Click(object sender, EventArgs e)

{

string inputString = txtInput.Text;

// String operations

string reversedString = ReverseString(inputString);

string upperCaseString = inputString.ToUpper();

string lowerCaseString = inputString.ToLower();

int stringLength = inputString.Length;

string[] words = inputString.Split(' '); // Splitting by space, can change delimiter

// Displaying results

lblOutput.Text = $"<strong>Original String:</strong> {inputString}<br/>" +

$"<strong>Reversed String:</strong> {reversedString}<br/>" +

$"<strong>Uppercase:</strong> {upperCaseString}<br/>" +

$"<strong>Lowercase:</strong> {lowerCaseString}<br/>" +

$"<strong>Length:</strong> {stringLength}<br/>" +

$"<strong>Words:</strong> {string.Join(", ", words)}";

}

// Function to reverse a string

private string ReverseString(string input)

{

char[] charArray = input.ToCharArray();

Array.Reverse(charArray);

return new string(charArray);

    }

}

* Design an asp.net webpage with 2 groups of Radio Buttons, DropDownList, label and TextBox to perform the following operations:-

1.On click of Radio Buttons each at the same time from two different groups, change the font size and font-face of the label's Text.

2.Also on the same webpage show that, on selecting a country name from the dropdown list, its respective country code gets displayed in a textbox.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Create a simple web page to display the Date properties (year, month, day, hour, minute, second, millisecond etc.) as well as to display the number of days of the year between two specified years.
* Create a web page containing the student details (RollNo, Name, Class, Phone, Email) and show result using Databinding and dropdownlist control.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Create XML file employees with nodes (eid, ename, edept, salary). Bind the eid and ename to a dropdownlist and the ename should be in the ascending order.
* Create simple web page that takes a number as input and display it four times in a row (separated by blank spaces), and then four times in the next row, with no separation.

Like-

Enter a digit: 22

Expected Output:

22222222

22222222

using System;

// This is the beginning of the Exercise12 class

public class Exercise12

{

// This is the main method where the program execution starts

public static void Main()

{

int num; // Variable to store the digit entered by the user

// Prompting the user to enter a digit

Console.WriteLine("Enter a digit: ");

// Reading the digit entered by the user and converting it to an integer

num = Convert.ToInt32(Console.ReadLine());

// Part A: "num num num num" using Write

Console.Write(num);

Console.Write(" ");

Console.Write(num);

Console.Write(" ");

Console.Write(num);

Console.Write(" ");

Console.Write(num);

Console.WriteLine();

// Part B: "numnumnumnum" using Write

Console.Write(num);

Console.Write(num);

Console.Write(num);

Console.WriteLine(num);

Console.WriteLine();

// Part C: "num num num num" using {0}

Console.WriteLine("{0} {0} {0} {0}", num);

// Part D: "numnumnumnum" using {0}

Console.WriteLine("{0}{0}{0}{0}", num);

}

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Create Web Form to demonstrate use of User Control. Create footer named user control having copyright reserved (ex. "©company name") and use it in a webpage.
* Create Web Form to demonstrate use of Ad rotator Control with five advertisements. Also demonstrate how keyword filter works.

Prac1.xml

<?xml-stylesheet type="text/css" href="Rule.css"?>

<books>

    <heading>Welcome To GeeksforGeeks </heading>

    <book>

        <title>Title -: Web Programming</title>

        <author>Author -: Chrisbates</author>

        <publisher>Publisher -: Wiley</publisher>

        <edition>Edition -: 3</edition>

        <price> Price -: 300</price>

    </book>

    <book>

        <title>Title -: Internet world-wide-web</title>

        <author>Author -: Ditel</author>

        <publisher>Publisher -: Pearson</publisher>

        <edition>Edition -: 3</edition>

        <price>Price -: 400</price>

    </book>

    <book>

        <title>Title -: Computer Networks</title>

        <author>Author -: Foruouzan</author>

        <publisher>Publisher -: Mc Graw Hill</publisher>

        <edition>Edition -: 5</edition>

        <price>Price -: 700</price>

    </book>

    <book>

        <title>Title -: DBMS Concepts</title>

        <author>Author -: Navath</author>

        <publisher>Publisher -: Oxford</publisher>

        <edition>Edition -: 5</edition>

        <price>Price -: 600</price>

    </book>

    <book>

        <title>Title -: Linux Programming</title>

        <author>Author -: Subhitab Das</author>

        <publisher>Publisher -: Oxford</publisher>

        <edition>Edition -: 8</edition>

        <price>Price -: 300</price>

    </book>

</books>

Rule.css

books {

    color: white;

    background-color : gray;

    width: 100%;

}

heading {

    color: green;

    font-size : 40px;

    background-color : powderblue;

}

heading, title, author, publisher, edition, price {

    display : block;

}

title {

    font-size : 25px;

    font-weight : bold;

}

Adrotator code

<Advertisements>

<Ad>

<ImageUrl></ImageUrl>

<NavigateUrl></NavigateUrl>

<AlternateText></AlternateText>

<Impressions></Impressions>

<Keyword></Keyword>

</Ad>

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Create simple application to perform following operations

i. Finding factorial Value

ii. Money Conversion

iii. Cube of given number

iv. Generate Fibonacci series

Code:

using System;

class Program

{

static void Main()

{

while (true)

{

Console.WriteLine("Choose an operation:");

Console.WriteLine("1. Finding factorial value");

Console.WriteLine("2. Money conversion");

Console.WriteLine("3. Cube of a given number");

Console.WriteLine("4. Generate Fibonacci series");

Console.WriteLine("5. Exit");

Console.Write("Enter your choice (1-5): ");

int choice = Convert.ToInt32(Console.ReadLine());

switch (choice)

{

case 1:

Console.Write("Enter a number to find its factorial: ");

int factorialNumber = Convert.ToInt32(Console.ReadLine());

Console.WriteLine($"Factorial of {factorialNumber} is: {Factorial(factorialNumber)}");

break;

case 2:

Console.Write("Enter the amount in dollars: $");

double dollars = Convert.ToDouble(Console.ReadLine());

Console.WriteLine($"Equivalent amount in INR: {MoneyConversion(dollars):C}");

break;

case 3:

Console.Write("Enter a number to find its cube: ");

double cubeNumber = Convert.ToDouble(Console.ReadLine());

Console.WriteLine($"Cube of {cubeNumber} is: {Cube(cubeNumber)}");

break;

case 4:

Console.Write("Enter the length of the Fibonacci series: ");

int fibonacciLength = Convert.ToInt32(Console.ReadLine());

Console.WriteLine($"Fibonacci series: {GenerateFibonacciSeries(fibonacciLength)}");

break;

case 5:

Console.WriteLine("Exiting the program. Goodbye!");

return;

default:

Console.WriteLine("Invalid choice. Please enter a valid option.");

break;

}

Console.WriteLine("\n--------------------------------------------\n");

}

}

static int Factorial(int number)

{

if (number == 0 || number == 1)

{

return 1;

}

return number \* Factorial(number - 1);

}

static double MoneyConversion(double dollars)

{

// Assuming 1 USD = 75 INR for simplicity

return dollars \* 75;

}

static double Cube(double number)

{

return Math.Pow(number, 3);

}

static string GenerateFibonacciSeries(int length)

{

int a = 0, b = 1;

string series = $"{a}, {b}";

for (int i = 2; i < length; i++)

{

int temp = a + b;

series += $", {temp}";

a = b;

b = temp;

}

return series;

}

}

* Demonstrate the use of Calendar control to perform following operations.
  + - Display messages in a calendar control
    - Display vacation in a calendar control c) Selected day in a calendar control using style
    - Difference between two calendar dates

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Create a simple web page with various sever controls to demonstrate setting and use of their properties. (Example: AutoPostBack)

20

1. On click of a button control display the selected items from the listbox in a textbox. Also in the same webpage display the name of the selected item from the DropDownList1 in a label. Also change the font size of the same label accor

ding to the font size selected from the Dropdownlist2.

2. Display Image control for photo.

3. Check Boxes provides special formatting (viz. underline, bold, italic) and Radio Buttons provides color for label.

4. Use of AutoPostBack property.

* Create a simple web page to count the number of times the current webpage is submitted to the server onclick event of a Button.

Default.aspx:

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="YourNamespace.Default" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title>Page Submission Counter</title>

</head>

<body>

<form id="form1" runat="server">

<div>

<h1>Page Submission Counter</h1>

<asp:Label ID="lblCounter" runat="server" Text="Number of submissions: 0"></asp:Label>

<br />

<asp:Button ID="btnSubmit" runat="server" Text="Submit" OnClientClick="incrementCounter();" OnClick="btnSubmit\_Click" />

</div>

</form>

<script type="text/javascript">

function incrementCounter() {

// Increment the counter using JavaScript on client side

var lblCounter = document.getElementById('<%= lblCounter.ClientID %>');

var currentCount = parseInt(lblCounter.innerText.split(":")[1].trim());

lblCounter.innerText = "Number of submissions: " + (currentCount + 1);

}

</script>

</body>

</html>

Default.aspx.cs:

using System;

namespace YourNamespace

{

public partial class Default : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

if (!IsPostBack)

{

// Initialize the counter on the first page load

Session["SubmissionCounter"] = 0;

}

}

protected void btnSubmit\_Click(object sender, EventArgs e)

{

// Increment the counter on server side

int currentCount = (int)Session["SubmissionCounter"];

currentCount++;

Session["SubmissionCounter"] = currentCount;

// Update the label with the new count

lblCounter.Text = $"Number of submissions: {currentCount}";

}

}

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Create a XML document for student database and apply the style sheet effects and display on webpage.

Prac1.xml:

<?xml-stylesheet type="text/css" href="Rule.css"?>

<books>

    <heading>Welcome To GeeksforGeeks </heading>

    <book>

        <title>Title -: Web Programming</title>

        <author>Author -: Chrisbates</author>

        <publisher>Publisher -: Wiley</publisher>

        <edition>Edition -: 3</edition>

        <price> Price -: 300</price>

    </book>

    <book>

        <title>Title -: Internet world-wide-web</title>

        <author>Author -: Ditel</author>

        <publisher>Publisher -: Pearson</publisher>

        <edition>Edition -: 3</edition>

        <price>Price -: 400</price>

    </book>

    <book>

        <title>Title -: Computer Networks</title>

        <author>Author -: Foruouzan</author>

        <publisher>Publisher -: Mc Graw Hill</publisher>

        <edition>Edition -: 5</edition>

        <price>Price -: 700</price>

    </book>

    <book>

        <title>Title -: DBMS Concepts</title>

        <author>Author -: Navath</author>

        <publisher>Publisher -: Oxford</publisher>

        <edition>Edition -: 5</edition>

        <price>Price -: 600</price>

    </book>

    <book>

        <title>Title -: Linux Programming</title>

        <author>Author -: Subhitab Das</author>

        <publisher>Publisher -: Oxford</publisher>

        <edition>Edition -: 8</edition>

        <price>Price -: 300</price>

    </book>

</books>

Rule.css:

books {

    color: white;

    background-color : gray;

    width: 100%;

}

heading {

    color: green;

    font-size : 40px;

    background-color : powderblue;

}

heading, title, author, publisher, edition, price {

    display : block;

}

title {

    font-size : 25px;

    font-weight : bold;

}

* Create a simple web page containing the student details (RollNo, Name, Class, Phone, Email). Write a program to store the data in the database and retrieve it using Data reader in tabular format.

4(1)

AIM: Set the label border color of rollno to red using css.

CODE: <%@ Page Language="C#" AutoEventWireup="true" CodeBehind="cssexample.aspx.cs"

Inherits="practical4css.cssexample" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title></title>

</head>

<body>

<form id="form1" runat="server">

<div>

<asp:Label ID="Label1" runat="server" Text="Enter Roll No.:"

BorderStyle="Dotted" BackColor="Coral"></asp:Label>

<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>

<br />

<asp:Label ID="Label2" runat="server" Text="Enter Name:"></asp:Label>

<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>

<br />

<asp:Label ID="Label3" runat="server" Text="Enter Marks:"></asp:Label>

<asp:TextBox ID="TextBox3" runat="server"></asp:TextBox>

<br />

<br />

<asp:Button ID="Button1" runat="server" Text="Submit" />

&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

<asp:Button ID="Button2" runat="server" Text="Clear" />

</div>

</form>

</body>

</html>

4(2)

AIM: Set the font-Arial , font style-bond , font size-18px of different controls(ie. Label, textbox, button) using css.

Myformat.css

.BtnStyle

{

font-family:Times New Roman;

font-size:large;

font-weight:bold;

}

.TxtStyle

{

font-family:Georgia;

font-size:larger;

font-weight:400;

background-color:Maroon;

border:2px solid goldenrod;

}

.Common

{

background-color:Aqua;

color:Red;

font-family:Courier New;

font-size:20px;

font-weight:bolder;

}

Myformatting.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="cssexample.aspx.cs"

Inherits="practical4css.cssexample" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title></title>

<body>

<form id="form1" runat="server">

<div>

<asp:Label ID="Label1" runat="server" Text="Enter Roll No.:" BorderStyle="Dotted"

BackColor="Coral"></asp:Label>

<asp:TextBox ID="TextBox1" runat="server" CssClass="TxtStyle"></asp:TextBox>

<br />

<asp:Label ID="Label2" runat="server" Text="Enter Name:"

CssClass="Common"></asp:Label>

<asp:TextBox ID="TextBox2" runat="server" CssClass="TxtStyle"></asp:TextBox>

<br />

<asp:Label ID="Label3" runat="server" Text="Enter Marks:"

CssClass="Common"></asp:Label>

<asp:TextBox ID="TextBox3" runat="server" CssClass="TxtStyle"></asp:TextBox>

<br />

<br />

<asp:Button ID="Button1" runat="server" Text="Submit" CssClass="BtnStyle" />

<asp:Button ID="Button2" runat="server" Text="Clear" CssClass="BtnStyle" />

</div>

</form>

</body>

</html>

4(3)

AIM: Design the same webpages for BMS, BAF, BscIT students and apply same background color for all the pages using css

CODE:

Myformat.css

.BtnStyle

{

font-family:Times New Roman;

font-size:large;

font-weight:bold;

}

.TxtStyle

{

font-family:Georgia;

font-size:larger;

font-weight:400;

background-color:Lime;

border:2px solid goldenrod;

}

.Common

{

background-color:Aqua;

color:Red;

font-family:Courier New;

font-size:20px;

font-weight:bolder;

}

.bk

{

background-color:Lime;

}

BScIT.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="BScIT.aspx.cs"

Inherits="cssExample.BScIT" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title></title>

<link rel="Stylesheet" type="text/css" href="MyFormat.css" />

</head>

<body text="Welcome to BScIT">

<form id="form1" runat="server">

<div class="bk">

<asp:Label ID="lblBScIT" runat="server" Text="Welcome to BscIT"></asp:Label>

</div>

</form>

</body>

</html>

BAF.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="BAF.aspx.cs"

Inherits="cssExample.BAF" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title></title>

<link rel="Stylesheet" type="text/css" href="MyFormat.css" />

</head>

<body>

<form id="form1" runat="server">

<div class="bk">

<asp:Label ID="lblBAF" runat="server" Text="Welcome to BAF"></asp:Label>

</div>

</form>

</body>

BMS.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="BMS.aspx.cs"

Inherits="cssExample.BMS" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title></title>

<link rel="Stylesheet" type="text/css" href="MyFormat.css" />

</head>

<body>

<form id="form1" runat="server" class="bk">

<asp:Label ID="lblBMS" runat="server" Text="Welcome to BMS"></asp:Label>

</form>

</body>

</html>

CSSExample1.aspx:

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="CSSExample1.aspx.cs"

Inherits="cssExample.CSSExample1" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title></title>

<link rel="Stylesheet" type="text/css" href="MyFormat.css" />

</head>

<body>

<form id="form1" runat="server">

<div>

<asp:Label ID="lblRollNo" runat="server" Text="Enter Roll No. :"

BorderStyle="Dotted" BackColor="Coral"></asp:Label>

&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbs

p;&nbsp;&nbsp;&nbsp;

<asp:TextBox ID="txtRoll" runat="server" CssClass="TxtStyle"></asp:TextBox>

<br />

<br />

<asp:Label ID="lblName" runat="server" Text="Enter Name :"

CssClass="Common"></asp:Label>

&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;

<asp:TextBox ID="txtName" runat="server" CssClass="TxtStyle"></asp:TextBox>

<br />

<br />

<asp:Label ID="lblMarks" runat="server" Text="Enter Marks :"

CssClass="Common"></asp:Label>

&nbsp;&nbsp;&nbsp;

<asp:TextBox ID="txtMarks" runat="server" CssClass="TxtStyle"></asp:TextBox>

<br />

<br />

<asp:Button ID="btnSubmit" runat="server" onclick="btnSubmit\_Click"

Text="Submit" CssClass="BtnStyle" />

&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbs

p;

<asp:Button ID="btnClear" runat="server" Text="Clear" CssClass="BtnStyle"/>

<br>

<br>

<br>

<h1><a href="BScIT.aspx"</a>Bsc IT</h1>

<h2><a href ="BAF.aspx"</a>BAF</h2>

<h3><a href ="BMS.aspx"</a>BMS</h3>

<a href="http://www.vsit.edu.in/">

Contact us</a>

</div>

</form>

</body>

</html

4(4)

AIM: Change the font family and color of all heading of above webpage using css.

CODE:

myformating.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="myformatting.aspx.cs"

Inherits="WebApplication1.myformatting" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title></title>

<link rel="Stylesheet" type="text/css" href="MyFormat.css" />

<style type="text/css">

h1,h2,h3{color:Blue; font-family:Agency FB;}

</style>

</head>

<body>

<form id="form1" runat="server">

<div>

<asp:Label ID="Label1" runat="server" Text="Enter Roll No.:" BorderStyle="Dotted"

BackColor="Coral"></asp:Label>

<asp:TextBox ID="TextBox1" runat="server" CssClass="TxtStyle"></asp:TextBox>

<br />

<asp:Label ID="Label2" runat="server" Text="Enter Name:"

CssClass="Common"></asp:Label>

<asp:TextBox ID="TextBox2" runat="server" CssClass="TxtStyle"></asp:TextBox>

<br />

<asp:Label ID="Label3" runat="server" Text="Enter Marks:"

CssClass="Common"></asp:Label>

<asp:TextBox ID="TextBox3" runat="server" CssClass="TxtStyle"></asp:TextBox>

<br />

<br />

<asp:Button ID="Button1" runat="server" Text="Submit" CssClass="BtnStyle" />

<asp:Button ID="Button2" runat="server" Text="Clear" CssClass="BtnStyle" />

<h1><a href="bscit.aspx"</a>Bsc IT</h1>

<h2><a href ="baf.aspx"</a>BAF</h2>

<h3><a href ="bms.aspx"</a>BMS</h3>

<a href="http://www.vsit.edu.in/">

Contact us</a>

<br />

<br />

<br />

<br />

</div>

</form>

</body>

</html>

4(5)

AIM: Use pseudo classes and display link, visited link and active link of contact us differently.

CODE:

myformatting.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="myformatting.aspx.cs"

Inherits="WebApplication1.myformatting" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<title></title>

<link rel="Stylesheet" type="text/css" href="MyFormat.css" />

<style type="text/css">

h1,h2,h3{color:Blue; font-family:Agency FB;}

A:link{color:Red;}

A:visited{color:Green;}

A:active{color:Orange;}

</style>

</head>

<body>

<form id="form1" runat="server">

<div>

<asp:Label ID="Label1" runat="server" Text="Enter Roll No.:" BorderStyle="Dotted"

BackColor="Coral"></asp:Label>

<asp:TextBox ID="TextBox1" runat="server" CssClass="TxtStyle"></asp:TextBox>

<br />

<asp:Label ID="Label2" runat="server" Text="Enter Name:"

CssClass="Common"></asp:Label>

<asp:TextBox ID="TextBox2" runat="server" CssClass="TxtStyle"></asp:TextBox>

<br />

<asp:Label ID="Label3" runat="server" Text="Enter Marks:"

CssClass="Common"></asp:Label>

<asp:TextBox ID="TextBox3" runat="server" CssClass="TxtStyle"></asp:TextBox>

<br /><br />

<asp:Button ID="Button1" runat="server" Text="Submit" CssClass="BtnStyle" />

<asp:Button ID="Button2" runat="server" Text="Clear" CssClass="BtnStyle" />

<h1><a href="bscit.aspx"</a>Bsc IT</h1>

<h2><a href ="baf.aspx"</a>BAF</h2>

<h3><a href ="bms.aspx"</a>BMS</h3>

<a href="http://www.vsit.edu.in/">

Contact us</a>

<br /><br /><br /><br />

</div>

</form>

</body>

</html>

Multiple inheritance using interfaces

CODE:

Gross.cs

using System;

namespace MultipleInheritance

{

interface Gross

{

int ta

{

get;

set;

}

int da

{

get;

set;

}

int GrossSal();

} }

Employee.cs

using System;

namespace MultipleInheritance

{

class Employee

{

string name;

public Employee(string name)

{ this.name = name; }

public int BasicSal(int basicSal)

{ return basicSal; }

public void ShowData()

{

Console.WriteLine("Name : " + name);

} } }

Salary.cs

using System;

namespace MultipleInheritance

{

class Salary:employee,Gross

{

int hra;

public Salary(string name, int hra):base(name)

{ this.hra = hra; }

public int ta

{

get {return S\_ta; }

set { S\_ta = value; }

}

private int S\_ta;

public int da

{

get { return S\_da; }

set { S\_da = value; }

}

private int S\_da;

public int GrossSal()

{

int gSal;

gSal = hra + ta + da + BasicSal(15000);

return gSal;

}

public void dispSal()

{ base.ShowData();

Console.WriteLine("Gross Sal : " + GrossSal());

} } }

Program.cs

using System;

namespace MultipleInheritance

{

class Program

{

static void Main(string[] args)

{

Salary s = new Salary("Prachit", 35000);

s.da = 20000;

s.ta = 30000;

s.dispSal();

} } }