

CSWP for Boxing and Unboxing

Front code:

```
<form id="form1" runat="server">
<div>
    <asp:Button ID="ButtonBoxUnbox" runat="server" Text="Box and Unbox"
OnClick="ButtonBoxUnbox_Click" />
    <br /><br />
    <asp:Label ID="LabelBoxed" runat="server" Text=""></asp:Label>
    <br /><br />
    <asp:Label ID="LabelUnboxed" runat="server" Text=""></asp:Label>
</div>
</form>
```

Backend Code:

```
protected void ButtonBoxUnbox_Click(object sender, EventArgs e)
{
    int valueType = 123;
    object boxedValue = valueType;

    LabelBoxed.Text = "Boxed value (stored as object): " + boxedValue.ToString();

    int unboxedValue = (int)boxedValue;

    LabelUnboxed.Text = "Unboxed value (cast back to int): " + unboxedValue;
}
}
```

CSWP for count no of visitor visting an application

Front end:

```
<form id="form1" runat="server">
    <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
    <br />
    <br />
    <br />
    <br />
    <asp:Button ID="Button1" runat="server" Height="21px" OnClick="Button1_Click" Text="Button"
/>
    <br />
    <div>
    </div>
</form>
```

Backend :

```
protected void Button1_Click(object sender, EventArgs e)
{
    ViewState["x"] = Convert.ToInt32(ViewState["x"])+1;
    Label1.Text = ViewState["x"].ToString();// Using when to get different users

    Application["x"] = Convert.ToInt32(Application["x"]) + 1;
    Label1.Text = Application["x"].ToString();// to use when to count all users as one
}
}
```

CSWP for Country Code And Two Group Of RadioDDI

Front End Code:

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

    <div>

        <asp:RadioButton ID="RadioButton1" runat="server" AutoPostBack="True" Font-Names="Curlz MT" GroupName="Fonts" OnCheckedChanged="RadioButton1_CheckedChanged" Text="Curlz MT" />

        <asp:RadioButton ID="RadioButton2" runat="server" AutoPostBack="True" Font-Names="Colonna MT" GroupName="Fonts" OnCheckedChanged="RadioButton1_CheckedChanged" Text="Colonna MT" />

        <asp:RadioButton ID="RadioButton3" runat="server" AutoPostBack="True" Font-Names="Gill Sans MT Ext Condensed Bold" GroupName="Fonts" OnCheckedChanged="RadioButton1_CheckedChanged" Text="Gill Sans MT Ext Condensed Bold" />

        <br />

        <br />

        <br />

        <asp:RadioButton ID="RadioButton4" runat="server" AutoPostBack="True" GroupName="Size" OnCheckedChanged="RadioButton4_CheckedChanged" Text="10" />

        <asp:RadioButton ID="RadioButton5" runat="server" AutoPostBack="True" GroupName="Size" OnCheckedChanged="RadioButton4_CheckedChanged" Text="30" />

        <asp:RadioButton ID="RadioButton6" runat="server" AutoPostBack="True" GroupName="Size" OnCheckedChanged="RadioButton4_CheckedChanged" Text="50" />

        <br />

        <br />

        <br />

        <asp:Label ID="Label1" runat="server" Font-Names="Curlz MT" Text="Hello Friends"></asp:Label>

        <br />

        <br />

        <asp:DropDownList ID="DropDownList1" runat="server" AutoPostBack="True" OnSelectedIndexChanged="DropDownList1_SelectedIndexChanged">

            <asp:ListItem Value="91">IND</asp:ListItem>

            <asp:ListItem Value="98">CHI</asp:ListItem>
```

```
        <asp:ListItem Value="100">BAN</asp:ListItem>
    </asp:DropDownList>
    <br />
    <br />
    <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
    <br />
</div>
```

Backend Code:

```
protected void RadioButton1_CheckedChanged(object sender, EventArgs e)
{

    Label1.Font.Name = ((RadioButton)sender).Text;
}

protected void RadioButton4_CheckedChanged(object sender, EventArgs e)
{
    Label1.Font.Size = Convert.ToInt32( ((RadioButton)sender).Text );
}

protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
{
    TextBox1.Text = DropDownList1.SelectedValue.ToString();
}
}
```

CSWP For disconnented data access in grid view

Front End:

```
<form id="form1" runat="server">
    <div>
        <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="True">
        </asp:GridView>
        <br />
        <br />
        <br />
    </div>
</form>
```

Back End:

```
public partial class WebForm1 : System.Web.UI.Page
{
    string con = ConfigurationManager.ConnectionStrings["AspConnectionString"].ConnectionString;
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            BindGridView();
        }
    }

    private void BindGridView()
    {

```

```

SqlConnection conn = new SqlConnection(con);
try
{
    string query = "SELECT * FROM tbl_student";

    SqlDataAdapter da = new SqlDataAdapter(query, conn);

    DataSet ds = new DataSet();

    da.Fill(ds);

    GridView1.DataSource = ds;
    GridView1.DataBind();
}
catch (Exception ex)
{
    Response.Write("Error: " + ex.Message);
}
}
}

```

CSWP to Create Master Page with Indian Culture

Front End:

```

<html>
<head runat="server">
    <title>Indian Culture Web Application</title>

```

```
<asp:ContentPlaceHolder ID="head" runat="server">
</asp:ContentPlaceHolder>
</head>
<body>
<header>
<h1 style="text-align: center;">Welcome to Indian Culture</h1>
<nav>
<ul style="list-style-type:none;">
<li style="display:inline; margin-right: 20px;"><a href="Home.aspx">Home</a></li>
<li style="display:inline; margin-right: 20px;"><a href="Festivals.aspx">Festivals</a></li>
<li style="display:inline; margin-right: 20px;"><a href="Cuisine.aspx">Cuisine</a></li>
<li style="display:inline; margin-right: 20px;"><a href="Art.aspx">Art</a></li>
</ul>
</nav>
</header>

<main>
<asp:ContentPlaceHolder id="MainContent" runat="server"></asp:ContentPlaceHolder>
</main>

<footer>
<p style="text-align: center;">&copy; 2024 Indian Culture. All rights reserved.</p>
</footer>

</body>
</html>
```

Home.aspx

```
<asp:Content ID="Content1" ContentPlaceHolderID="MainContent" runat="server">
<h2>Welcome to the Incredible Culture of India</h2>
<p>India is a country with a rich heritage, diverse traditions, and vibrant culture.</p>
<p>Explore the various facets of Indian culture, including its festivals, cuisine, and art forms.</p>
```

</asp:Content>

Festivals.aspx

<asp:Content ID="Content1" ContentPlaceHolderID="MainContent" runat="server">

<h2>Festivals of India</h2>

<p>India is known for its colorful and vibrant festivals celebrated throughout the year. Some of the most famous festivals are:</p>

Diwali - The Festival of Lights

Holi - The Festival of Colors

Durga Puja - Celebrating Goddess Durga

Eid - A day of celebration and prayer

Christmas - Celebrating the birth of Jesus Christ

</asp:Content>

Cuisine.aspx:

<asp:Content ID="Content1" ContentPlaceHolderID="MainContent" runat="server">

<h2>Indian Cuisine</h2>

<p>Indian cuisine is a delightful fusion of flavors, spices, and cooking techniques. Some popular Indian dishes include:</p>

Biryani

Butter Chicken

Paneer Tikka

Samosa

Masala Dosa

</asp:Content>

Art.aspx:

<asp:Content ID="Content1" ContentPlaceHolderID="MainContent" runat="server">

<h2>Indian Art</h2>

<p>India has a rich tradition of art and craft. Some renowned Indian art forms are:</p>


```
<ul>
    <li>Madhubani Painting</li>
    <li>Tanjore Painting</li>
    <li>Warli Art</li>
    <li>Pattachitra</li>
</ul>
</asp:Content>
```

CSWP for Server Controls all controls such as checkbox radio button and many more:

Frontend code:

```
<body>
    <form id="form1" runat="server">
        <div>
            <asp:ListBox ID="ListBox1" runat="server">
                <asp:ListItem>item1</asp:ListItem>
                <asp:ListItem>item2</asp:ListItem>
                <asp:ListItem>item3</asp:ListItem>
                <asp:ListItem>item4</asp:ListItem>
                <asp:ListItem>item5</asp:ListItem>
                <asp:ListItem>item6</asp:ListItem>
                <asp:ListItem>item7</asp:ListItem>
            </asp:ListBox>
            <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Button" />
            <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
            <br />
            <br />
            <asp:DropDownList ID="DropDownList1" runat="server" AutoPostBack="True"
                OnSelectedIndexChanged="DropDownList1_SelectedIndexChanged">
                <asp:ListItem>item1</asp:ListItem>
```

```
<asp:ListItem>item2</asp:ListItem>
<asp:ListItem>item3</asp:ListItem>
<asp:ListItem>item4</asp:ListItem>
<asp:ListItem>item5</asp:ListItem>
<asp:ListItem>item6</asp:ListItem>
<asp:ListItem>item7</asp:ListItem>
</asp:DropDownList>
<asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
<br />
<br />
<asp:DropDownList ID="DropDownList2" runat="server" AutoPostBack="True"
OnSelectedIndexChanged="DropDownList2_SelectedIndexChanged">
    <asp:ListItem>10</asp:ListItem>
    <asp:ListItem>20</asp:ListItem>
    <asp:ListItem>30</asp:ListItem>
    <asp:ListItem>40</asp:ListItem>
    <asp:ListItem>50</asp:ListItem>
    <asp:ListItem>60</asp:ListItem>
</asp:DropDownList>
<br />
<br />
<br />
<br />
<asp:Image ID="Image1" runat="server" ImageUrl="~/p1.jpg" Width="100px" />
<br />
<br />
<br />
    <asp:Label ID="Label2" runat="server" Text="Hello Friends"></asp:Label><br />
<br />
<br />
    <asp:CheckBox ID="CheckBox2" runat="server" Text="Bold"
OnCheckedChanged="CheckBox2_CheckedChanged" AutoPostBack="True" />
<br />
```

```

        <asp:CheckBox ID="CheckBox1" runat="server"
OnCheckedChanged="CheckBox1_CheckedChanged" Text="UnderLine" AutoPostBack="True" />

        &nbsp;

        <asp:CheckBox ID="CheckBox3" runat="server" Text="Italic"
OnCheckedChanged="CheckBox3_CheckedChanged" AutoPostBack="True" />

        <br />

        <asp:RadioButton ID="RadioButton1" runat="server" Text="RED" AutoPostBack="True"
GroupName="Color" OnCheckedChanged="RadioButton1_CheckedChanged" />

        <asp:RadioButton ID="RadioButton2" runat="server" Text="Green" AutoPostBack="True"
GroupName="Color" OnCheckedChanged="RadioButton2_CheckedChanged" />

        <br />

        <br />

        <br />

    </div>

</form>

```

Backend code:

```

public partial class WebForm1 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            UpdateLabelStyle();
        }
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        TextBox1.Text = ListBox1.SelectedValue;
    }
}

```

```
protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
{
    Label1.Text = DropDownList1.SelectedValue.ToString();
}
```

```
protected void DropDownList2_SelectedIndexChanged(object sender, EventArgs e)
{
    Label1.Font.Size = Convert.ToInt32(DropDownList2.SelectedValue);
}
```

```
protected void CheckBox1_CheckedChanged(object sender, EventArgs e)
{
    if (CheckBox1.Checked)
    {
        Label2.Font.Underline = true;
    }
    else
    {
        Label2.Font.Underline = false;
    }
}
```

```
protected void CheckBox2_CheckedChanged(object sender, EventArgs e)
{
    if (CheckBox2.Checked)
    {
        Label2.Font.Bold = true;
    }
    else
    {
        Label2.Font.Bold = false;
    }
}
```

```
}
```

```
protected void CheckBox3_CheckedChanged(object sender, EventArgs e)
```

```
{
```

```
    if (CheckBox3.Checked)
```

```
    {
```

```
        Label2.Font.Italic = true;
```

```
    }
```

```
    else
```

```
    {
```

```
        Label2.Font.Italic = false;
```

```
    }
```

```
}
```

```
protected void RadioButton1_CheckedChanged(object sender, EventArgs e)
```

```
{
```

```
    Label2.ForeColor = System.Drawing.Color.Red;
```

```
}
```

```
protected void RadioButton2_CheckedChanged(object sender, EventArgs e)
```

```
{
```

```
    Label2.ForeColor = System.Drawing.Color.Green;
```

```
}
```

```
}
```

Create a simple application for Furniture class with 3 data members:

Front end code:

```
<body>
```

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

```
        <div>
```

```

        <asp:ListBox ID="ListBox1" runat="server" Height="238px" Width="93px"></asp:ListBox>
    &nbsp;
    <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Show Info" />
    <br />
    <br />
    <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
    <br />
    <asp:Label ID="Label2" runat="server" Text="Label"></asp:Label>
    <br />
</div>
</form>
</body>

```

Backend Code:

```

protected void Page_Load(object sender, EventArgs e)
{
    if (!IsPostBack)
    {
        Session["s1"] = new Furniture("chair1", "xyz1", "1001");
        Session["s2"] = new Furniture("chair2", "xyz2", "1002");
        Session["s3"] = new Furniture("chair3", "xyz3", "1003");

        ListBox1.Items.Add((((Furniture)Session["s1"]).name));
        ListBox1.Items.Add((((Furniture)Session["s2"]).name));
        ListBox1.Items.Add((((Furniture)Session["s3"]).name));

    }

}

protected void Button1_Click(object sender, EventArgs e)
{

```

```

if (ListBox1.SelectedIndex != -1)
{
    if(ListBox1.SelectedIndex == 0)
    {
        Label1.Text = ((Furniture)Session["s1"]).manu;
        Label2.Text = ((Furniture)Session["s1"]).cost;
    }

    else if (ListBox1.SelectedIndex == 1)
    {
        Label1.Text = ((Furniture)Session["s2"]).manu;
        Label2.Text = ((Furniture)Session["s2"]).cost;
    }

    else if (ListBox1.SelectedIndex == 2)
    {
        Label1.Text = ((Furniture)Session["s3"]).manu;
        Label2.Text = ((Furniture)Session["s3"]).cost;
    }
}
}

```

```

class Furniture
{
    public string name;
    public string manu;
    public string cost;

    public Furniture(string name, string manu, string cost)
    {
        this.name = name;
        this.manu = manu;
    }
}

```

```
        this.cost = cost;

    }

}

}
```

CSWP For Fibonacci series and test of prime number

Front End:

```
<body>

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

        <div>

            Enter Number:

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

            <br />

            <br />

            <asp:Button ID="Button1" runat="server" Text="Fibonacci"

                onclick="Button1_Click" />

            <br />

            <br />

            <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>

            <br />

            <br />

            <asp:Button ID="Button2" runat="server" Text="Prime" onclick="Button2_Click" />

            <br />
```



```
<br />

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

</div>

</form>

</body>
```

Backend:

```
protected void Button1_Click(object sender, EventArgs e)
{
    int a, b, c, i, n;

    a = 0;
    b = 1;

    Label1.Text = a.ToString() + b.ToString();
    n = Convert.ToInt32(TextBox1.Text);
    for (i = 1; i <= n; ++i)
    {
        c = a + b;
        Label1.Text = Label1.Text + c.ToString();
        a = b;
        b = c;
    }
}

protected void Button2_Click(object sender, EventArgs e)
{
    int n, i, s = 0; // s is used to track if the number is prime
    n = Convert.ToInt32(TextBox1.Text);

    if (n <= 1) // Check for numbers less than or equal to 1
    {
        s = 1; // Not prime
    }
}
```

```

else
{
    for (i = 2; i <= n / 2; ++i) // Check for factors from 2 to n/2
    {
        if (n % i == 0) // If n is divisible by i, it's not prime
        {
            s = 1; // Not prime
            break; // Exit the loop if a factor is found
        }
    }
}

if (s == 0)
    Label2.Text = "The given number is prime";
else
    Label2.Text = "The given number is not prime";
}
}
}

```

CSWP IndexOutOfRangeException And DivideByZeroException

Front End Code:

```

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

        <br />
        <asp:Label ID="Label1" runat="server" Text="Enter any Index:"></asp:Label>
        <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
        <br />
        <br />
        <asp:Button ID="Button1" runat="server" Text="Button" OnClick="Button1_Click" />
    </div>
</form>

```

[illegible]

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



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

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


`<asp:Label ID="Label4" runat="server" Text="Enter Denominator:"></asp:Label>`

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


[illegible]

```
<asp:Button ID="Button2" runat="server" Text="Button" OnClick="Button2_Click" />
```

[illegible]

```
<asp:Label ID="Label5" runat="server" Text=" "></asp:Label>
```


</form>

BackEnd Code:

```
protected void Button1_Click(object sender, EventArgs e)
```

 $\{$

try

 $\{$

```
int[] numbers = { 10, 20, 30 };
```

```
int index = int.Parse(TextBox1.Text);
```

```
int value = numbers[index];
```

```
    Label2.Text = "Value at index " + index + " is: " + value;
```

```
}
```

```
catch (IndexOutOfRangeException ex)
```

```
{
```

```
    Label2.Text = "Error: Index out of range! Please enter a valid index between 0 and 2.";
```

```
}
```

```
catch (FormatException ex)
```

```
{
```

```
    Label2.Text = "Error: Please enter a valid number. ";
```

```
}
```

```
}
```

```
protected void Button2_Click(object sender, EventArgs e)
```

```
{
```

```
    try
```

```
{
```

```
        int numerator = int.Parse(TextBox2.Text);
```

```
        int denominator = int.Parse(TextBox3.Text);
```

```
        int result = numerator / denominator;
```

```

        Label5.Text = "Result: " + result;
    }
    catch (DivideByZeroException)
    {

        Label5.Text = "Error: Cannot divide by zero!";
    }
    catch (FormatException)
    {

        Label5.Text = "Error: Please enter valid numbers for both numerator and denominator.";
    }
}

```

CSWP For Student details in tabular format in connected data base using datareader to read:

Front end code:

```

<body>
    <form id="form1" runat="server">
        <table>
            <tr><td>Roll No</td><td>
                <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
            </td></tr>
            <tr><td>name</td><td>
                <asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>
            </td></tr>
            <tr><td>class</td><td>
                <asp:TextBox ID="TextBox3" runat="server"></asp:TextBox>
            </td></tr>
            <tr><td>phone no</td><td>

```

```

        <asp:TextBox ID="TextBox5" runat="server"></asp:TextBox>
    </td></tr>
    <tr><td>email</td><td>
        <asp:TextBox ID="TextBox6" runat="server"></asp:TextBox>
    </td></tr>
    <tr><td></td><td>
        <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Add Data" />
        <br />
        <br />
        <asp:Button ID="Button2" runat="server" OnClick="Button2_Click" Text="Button" />
        <br />
    </td></tr>
</table>
</form>

```

Backend Code:

```

public partial class WebForm1 : System.Web.UI.Page
{
    SqlConnection con = new SqlConnection("data source=LAPTOP-LTCDQN0N\\SQLEXPRESS01;
initial catalog=asp; integrated security=true");

    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        try
        {

```

```
string query = "insert into tbl_student values('"+ TextBox1.Text + "','"+ TextBox2.Text + "','"+  
TextBox3.Text + "','"+ TextBox5.Text + "','"+ TextBox6.Text + "')";
```

```
SqlCommand cmd = new SqlCommand(query, con);
```

```
con.Open();
```

```
cmd.ExecuteNonQuery();
```

```
}
```

```
catch (Exception ex)
```

```
{
```

```
    Response.Write(ex.ToString());
```

```
}
```

```
finally
```

```
{
```

```
    con.Close();
```

```
}
```

```
}
```

```
protected void Button2_Click(object sender, EventArgs e)
```

```
{
```

```
    try
```

```
    {
```

```
        string squery = "select * from tbl_student";
```

```
        SqlCommand cmd = new SqlCommand(squery, con);
```

```
        con.Open();
```

```
        SqlDataReader dr = cmd.ExecuteReader();
```

```
        string optop = "<table border>";
```

```
        if(dr.HasRows)
```

```
        {
```

```
            while(dr.Read())
```

```
            {
```

```

        optop += "<tr><td>" + dr[0] + "</td><td>" + dr[1] + "</td><td>" + dr[2] + " </td><td>" +
dr[3] + " </td><td>" + dr[4] + " </td><td>";

    }

}

optop += "</table>";

Response.Write(optop);

}

catch(Exception ex)

{

    Response.Write(ex.Message);

}

finally

{

    con.Close() ;

}

}

}

```

CSWP TreeVIEW Control

Front End:

```

<body>

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

        <div>

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

                <Nodes>

                    <asp:TreeNode Text="Home" Value="Home">

                        <asp:TreeNode NavigateUrl="~/Employee.aspx" Text="Employee"
Value="Employee"></asp:TreeNode>

                        <asp:TreeNode NavigateUrl="~/Upload Resume.aspx" Text="Upload Resume"
Value="Resume"></asp:TreeNode>

                    </asp:TreeNode>

                    <asp:TreeNode Text="Contact" Value="Contact">

```



```
        <asp:TreeNode NavigateUrl="~/AboutUs.aspx" Text="AboutUs"
Value="Address"></asp:TreeNode>
    </asp:TreeNode>
</Nodes>
</asp:TreeView>
<br />
<br />
<br />
<br />
<br />
</div>
</form>
</body>
```

Employee.aspx:

```
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Label ID="Label1" runat="server" Text="Employee"></asp:Label>
            <br />
        </div>
    </form>
```

Resume.aspx:

```
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Label ID="Label1" runat="server" Text="Upload your Resume"></asp:Label>
        </div>
    </form>
</body>
```

About US:

```
<body>
  <form id="form1" runat="server">
    <asp:Label ID="Label1" runat="server" Text="About Us"></asp:Label>
    <div>
      </div>
    </div>
  </form>
</body>
```

CSWP for User Defined Exception :

FrontEnd:

```
<form id="form1" runat="server">
  <div>
    <h2>User-Defined Exception Demo</h2>

    <label for="inputNumber">Enter a number greater than 0:</label>
    <asp:TextBox ID="inputNumber" runat="server"></asp:TextBox>
    <br /><br />

    <asp:Button ID="btnCheckNumber" runat="server" Text="Check Number"
    OnClick="btnCheckNumber_Click" />
    <br /><br />

    <asp:Label ID="lblResult" runat="server" ForeColor="Red"></asp:Label>
  </div>
</form>
```

Backend:

```
protected void btnCheckNumber_Click(object sender, EventArgs e)
{
    try
    {
```

```

int number = int.Parse(inputNumber.Text);

if (number <= 0)
{

    throw new Exception("The number must be greater than zero.");
}

lblResult.ForeColor = System.Drawing.Color.Green;
lblResult.Text = "Success! You entered a valid number.";
}
catch (FormatException)
{

    lblResult.Text = "Error: Please enter a valid number.";
}
catch (Exception ex)
{

    lblResult.Text = $"Error: {ex.Message}";
}
}

```

CSWP For Multiple inheritance:

Code:

```

public interface IPrintable
{

```

```
    void Print();  
}
```

```
public interface IScannable  
{  
    void Scan();  
}
```

```
public class MultifunctionPrinter : IPrintable, IScannable  
{  
    public void Print()  
    {  
        Console.WriteLine("Printing...");  
    }  
  
    public void Scan()  
    {  
        Console.WriteLine("Scanning...");  
    }  
}
```

```
internal class Program  
{  
    static void Main(string[] args)  
    {  
        MultifunctionPrinter mfp = new MultifunctionPrinter();  
        mfp.Print();  
        mfp.Scan();  
  
    }  
}
```

CSWP For CalenDar Control

Frontend:

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

    <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>

    <br />

    <asp:Button ID="Button2" runat="server" Text="Button" OnClick="Button2_Click" />

    <asp:Calendar ID="Calendar1" runat="server" BackColor="White" BorderColor="Black"
    DayNameFormat="Shortest" Font-Names="Times New Roman" Font-Size="10pt" ForeColor="Black"
    Height="220px" NextPrevFormat="FullMonth" OnDayRender="Calendar1_DayRender"
    OnSelectionChanged="Calendar1_SelectionChanged" TitleFormat="Month" Width="400px">

        <DayHeaderStyle BackColor="#CCCCCC" Font-Bold="True" Font-Size="7pt"
        ForeColor="#333333" Height="10pt" />

        <DayStyle Width="14%" />

        <NextPrevStyle Font-Size="8pt" ForeColor="White" />

        <OtherMonthDayStyle ForeColor="#999999" />

        <SelectedDayStyle BackColor="#CC3333" ForeColor="White" />

        <SelectorStyle BackColor="#CCCCCC" Font-Bold="True" Font-Names="Verdana" Font-Size="8pt"
        ForeColor="#333333" Width="1%" />

        <TitleStyle BackColor="Black" Font-Bold="True" Font-Size="13pt" ForeColor="White"
        Height="14pt" />

        <TodayDayStyle BackColor="#CCCC99" />

    </asp:Calendar>

    <br />

    <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Button" />

    <asp:Calendar ID="Calendar2" runat="server"></asp:Calendar>

    <asp:Label ID="lblVacationDates" runat="server" Text="Vacation Dates:"></asp:Label>

    <br />

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

    <br />

    <br />

    <br />

    <br />

</div>
```

```
</div>  
</form>
```

Backend:

```
protected void Page_Load(object sender, EventArgs e)  
{  
    if (!IsPostBack)  
    {  
        Calendar1.TodaysDate = DateTime.Now;  
    }  
}  
  
protected void Calendar1_DayRender(object sender, DayRenderEventArgs e)  
{  
    if (e.Day.Date.Year == 2024 && e.Day.Date.Month == 10 && e.Day.Date.Day == 28)  
    {  
        Label l1 = new Label();  
        l1.Text = "Diwali";  
        e.Cell.Controls.Add(l1);  
    }  
}  
  
protected void Calendar1_SelectionChanged(object sender, EventArgs e)  
{  
    Calendar1.SelectedDayStyle.BackColor = System.Drawing.Color.Red;  
    Calendar1.SelectedDayStyle.ForeColor = System.Drawing.Color.Yellow;  
  
    lblVacationDates.Text = "Selected Date: " + Calendar1.SelectedDate.ToString("yyyy-MM-dd  
HH:mm:ss");  
}
```

```
protected void Button1_Click(object sender, EventArgs e)
{
    TimeSpan ts = Calendar1.SelectedDate - Calendar2.SelectedDate;
    Label1.Text = ts.TotalDays.ToString();

}

protected void Button2_Click(object sender, EventArgs e)
{
    DateTime selectedDate = Calendar1.SelectedDate;

    DateTime currentTime = DateTime.Now;

    DateTime finalDateTime = new DateTime(selectedDate.Year, selectedDate.Month,
selectedDate.Day,
currentTime.Hour, currentTime.Minute, currentTime.Second);

    Label2.Text = "Formatted Date and Time: " + finalDateTime.ToString();
}
}
```