

**PRACTICAL NO.03****Write the program for the following:**

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

Code:

**Aspx code:**

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs"
Inherits="autopostbackExampleNew.WebForm1" %>
```

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Demo Controls</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <h2>ASP.NET Server Controls Demo</h2>

            <asp:Label ID="lblInfo" runat="server" Text="Select a color:"></asp:Label>
            <asp:DropDownList ID="ddlColors" runat="server" AutoPostBack="true"
OnSelectedIndexChanged="ddlColors_SelectedIndexChanged">
                <asp:ListItem Text="Red" Value="Red"></asp:ListItem>
                <asp:ListItem Text="Green" Value="Green"></asp:ListItem>
                <asp:ListItem Text="Blue" Value="Blue"></asp:ListItem>
            </asp:DropDownList>

            <asp:Label ID="lblColor" runat="server" ForeColor="Black"></asp:Label>

            <br /><br />

            <asp:CheckBox ID="chkEnableButton" runat="server" Text="Enable Button"
AutoPostBack="true" OnCheckedChanged="chkEnableButton_CheckedChanged" />
            <asp:Button ID="btnSubmit" runat="server" Text="Submit" Enabled="false"
OnClick="btnSubmit_Click" />

            <br /><br />

            <asp:TextBox ID="txtInput" runat="server" AutoPostBack="true"
OnTextChanged="txtInput_TextChanged" />
            <asp:Label ID="lblTextBox" runat="server" Text="You entered: "></asp:Label>
        </div>
    </form>
</body>
</html>
```

**C# code:**

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace autopostbackExampleNew
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void ddlColors_SelectedIndexChanged(object sender, EventArgs e)
        {

        }
    }
}
```

```
        lblColor.Text = "You selected: " + ddlColors.SelectedItem.Text;
    }

    protected void chkEnableButton_CheckedChanged(object sender, EventArgs e)
    {
        btnSubmit.Enabled = chkEnableButton.Checked;
    }

    protected void btnSubmit_Click(object sender, EventArgs e)
    {
        lblColor.Text = "Button clicked!";
    }

    protected void txtInput_TextChanged(object sender, EventArgs e)
    {
        lblTextBox.Text = "You entered: " + txtInput.Text;
    }
}
}
```

Output:



b. Create a Registration form to demonstrate use of various Validation controls.

Code:

**Aspx code:**

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs"
Inherits="valadtion.WebForm1" %>

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Registration Form</title>

</head>
<body>
    <form id="form1" runat="server">
        <div>
            <h2>Registration Form</h2>

            <asp:Label ID="lblUsername" runat="server" Text="Username:"></asp:Label>
            <asp:TextBox ID="txtUsername" runat="server"></asp:TextBox>
            <asp:RequiredFieldValidator ID="rfvUsername" runat="server"
                ControlToValidate="txtUsername"
                ErrorMessage="Username is required">
```

```

        ForeColor="Red"></asp:RequiredFieldValidator>
    <br /><br />

```

```

<asp:Label ID="lblEmail" runat="server" Text="Email:"></asp:Label>
<asp:TextBox ID="txtEmail" runat="server"></asp:TextBox>
<asp:RequiredFieldValidator ID="rfvEmail" runat="server"
    ControlToValidate="txtEmail"
    ErrorMessage="Email is required"
    ForeColor="Red"></asp:RequiredFieldValidator>
<asp:RegularExpressionValidator ID="revEmail" runat="server"
    ControlToValidate="txtEmail"
    ValidationExpression="\b[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Z|a-z]{2,}\b"
    ErrorMessage="Invalid email format"
    ForeColor="Red"></asp:RegularExpressionValidator>
<br /><br />

```

```

<asp:Label ID="lblAge" runat="server" Text="Age:"></asp:Label>
<asp:TextBox ID="txtAge" runat="server"></asp:TextBox>
<asp:RequiredFieldValidator ID="rfvAge" runat="server"
    ControlToValidate="txtAge"
    ErrorMessage="Age is required"
    ForeColor="Red"></asp:RequiredFieldValidator>
<asp:RangeValidator ID="rvAge" runat="server"
    ControlToValidate="txtAge"
    MinimumValue="18"
    MaximumValue="120"
    Type="Integer"
    ErrorMessage="Age must be between 18 and 120"
    ForeColor="Red"></asp:RangeValidator>

```

```

<br />

```

```

<br />
<asp:Label ID="phnlbl" runat="server" Text="Phone No.:"></asp:Label>
<asp:TextBox ID="phntxt" runat="server"></asp:TextBox>
<asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
    ControlToValidate="phntxt" ErrorMessage="Phone Number is required"
    ForeColor="Red"></asp:RequiredFieldValidator>
<asp:CustomValidator ID="CustomValidator1" runat="server"
    ControlToValidate="phntxt" ErrorMessage="Enter 10 digits Phone Number."
    ForeColor="Red"
    OnServerValidate="CustomValidator1_ServerValidate"></asp:CustomValidator>
<br />
<br />

```

```

<asp:Label ID="lblPassword" runat="server" Text="Password:"></asp:Label>
<asp:TextBox ID="txtPassword" runat="server"
    TextMode="Password"></asp:TextBox>
<asp:RequiredFieldValidator ID="rfvPassword" runat="server"
    ControlToValidate="txtPassword"
    ErrorMessage="Password is required"
    ForeColor="Red"></asp:RequiredFieldValidator>
<br /><br />

```

```

<asp:Label ID="lblConfirmPassword" runat="server" Text="Confirm
Password:"></asp:Label>
<asp:TextBox ID="txtConfirmPassword" runat="server"
    TextMode="Password"></asp:TextBox>
<asp:RequiredFieldValidator ID="rfvConfirmPassword" runat="server"
    ControlToValidate="txtConfirmPassword"
    ErrorMessage="Confirm Password is required"
    ForeColor="Red"></asp:RequiredFieldValidator>
<asp:CompareValidator ID="cvPassword" runat="server"
    ControlToCompare="txtPassword"

```

```

        ControlToValidate="txtConfirmPassword"
        ErrorMessage="Passwords do not match"
        ForeColor="Red"></asp:CompareValidator>
<br /><br />

<asp:Button ID="btnSubmit" runat="server" Text="Register"
OnClick="btnSubmit_Click" />
<br />
</div>
<asp:Label ID="Label1" runat="server"></asp:Label>
<asp:ValidationSummary ID="ValidationSummary1" runat="server" />
</form>
</body>
</html>

```

**C# code:**

```

using System;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace valadtion
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
        }

        protected void btnSubmit_Click(object sender, EventArgs e)
        {
            if (Page.IsValid)
            {
                Label1.Text = "Registration successful!";
            }
        }

        protected void CustomValidator1_ServerValidate(object source, ServerValidateEventArgs
args)
        {
            string ph= args.Value;
            if (ph.Length==10)
            {

                args.IsValid = true;
            }
            else
            {
                args.IsValid = false;
            }
        }
    }
}

```

Output:

### Registration Form

Username:  Username is required

Email:  Email is required

Age:  Age is required

Phone No.:  Phone Number is required

Password:  Password is required

Confirm Password:  Confirm Password is required

Register

- Username is required
- Email is required
- Age is required
- Phone Number is required
- Password is required
- Confirm Password is required

### Registration Form

Username:

Email:

Age:

Phone No.:

Password:

Confirm Password:

Register

Registration successful!