

1. **Develop a simple ASP.NET web form that generates the front cover for a magazine. The form should provide the options for selecting the image, background color, changing font size, colour etc. Input messages must be taken from the user so as to display it on the front cover with legible font family and font size. The controls added must be proportionate to the web page size.**

Code:

CoverGenerator.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="CoverGenerator.aspx.cs" Inherits="q4.CoverGenerator" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Front Cover Generator</title>
</head>
<body>
    <form id="form1" runat="server">
        <div style="height: auto; width: 100%">
            <asp:Panel ID="ConfigureDiv" runat="server" style="padding: 15px; border-style: inset;
border-width: medium; height: 705px; width: 300px; font-family: Calibri; font-size: medium; font-
weight: bold; background-color: #5C87D8; position: absolute; z-index: auto; display: block; float:
left; cursor: auto; clip: rect(auto, auto, auto, auto); color: #FFFFFF; top: 15px; left: 10px;
overflow: scroll;">
                Choose the bacground image:<br />
                <asp:DropDownList ID="ImageDropDownList" runat="server" AutoPostBack="True"
OnSelectedIndexChanged="ControlChanged" Height="25px" Width="175px">
                    <asp:ListItem>None</asp:ListItem>
                    <asp:ListItem>Abstract Image 1</asp:ListItem>
                    <asp:ListItem>Abstract Image 2</asp:ListItem>
                    <asp:ListItem>Abstract Image 3</asp:ListItem>
                </asp:DropDownList><br /><br />
                Choose the background colour:<br />
                <asp:DropDownList ID="BackgoundDropDownList" runat="server" AutoPostBack="true"
OnSelectedIndexChanged="ControlChanged" Height="25px" Width="175px"> </asp:DropDownList><br /><br />
                Choose the text colour:<br />
                <asp:DropDownList ID="TextDropDownList" runat="server" AutoPostBack="True"
OnSelectedIndexChanged="ControlChanged" Height="25px" Width="175px"></asp:DropDownList><br /><br />
                Choose the font name:<br />
                <asp:DropDownList ID="FontDropDownList" runat="server" AutoPostBack="True"
OnSelectedIndexChanged="ControlChanged" Height="25px" Width="175px"></asp:DropDownList><br /><br />
                Enter the font size:<br />
                <asp:TextBox ID="SizeTextBox" runat="server" AutoPostBack="true"
OnTextChanged="ControlChanged" Height="16px" Width="175px"></asp:TextBox><br /><br />
                Choose the border style:<br />
                <asp:RadioButtonList ID="BorderRadioButtonList" runat="server" AutoPostBack="True"
OnSelectedIndexChanged="ControlChanged" Height="23px" RepeatColumns="2" Width="175px">
                    </asp:RadioButtonList><br />
                Enter the text to be displayed on the cover:<br />
                <asp:TextBox ID="GreetingsTextBox" runat="server" AutoPostBack="True"
OnTextChanged="ControlChanged" Height="75px" Rows="2" TextMode="MultiLine"
Width="175px"></asp:TextBox>
                <br /><br />
                <br />
                <asp:Button ID="PreviewButton" runat="server" Text="Preview"
OnClick="PreviewButton_Click" Height="31px" Width="76px" />
                &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~
                <asp:Button ID="ClearButton" runat="server" Text="Clear" OnClick="ClearButton_Click"
Height="31px" Width="76px" />
                <br />
                <br />
            </div>
        </form>
    </body>
</html>
```



```

        BorderRadiusButtonList.DataBind();
        BorderRadiusButtonList.SelectedIndex = 0;

        //Choose default image
        ImageDropDownList.SelectedIndex = -1;

        //Make cover panel visible
        CoverPanel.Visible = true;

        //Make edit button not visible
        EditButtonPanel.Visible = false;
    }
}

private void UpdateCard()
{
    //Update the colours
    CoverPanel.BackColor = Color.FromName(BackgroundDropDownList.SelectedItem.Text);
    GreetingsLabel.ForeColor = Color.FromName(TextDropDownList.SelectedItem.Text);

    //Update background image
    if (ImageDropDownList.SelectedIndex != 0)
    {
        string imageURL = "~/Images/" + ImageDropDownList.SelectedItem.Text + ".jpg";
        CoverPanel.BackImageUrl = imageURL;
    }
    else
        CoverPanel.BackImageUrl = "";

    //Update the font name and size
    GreetingsLabel.Font.Name = FontDropDownList.SelectedItem.Text;
    try
    {
        if (Int32.Parse(SizeTextBox.Text) > 0)
        {
            GreetingsLabel.Font.Size = Int32.Parse(SizeTextBox.Text);
        }
    }
    catch
    {
    }

    //Get the correct TypeConverter for BorderStyle enumeration
    TypeConverter converter = TypeDescriptor.GetConverter(typeof(BorderStyle));

    //Upfate the border style
    CoverPanel.BorderStyle =
(BorderStyle)converter.ConvertFromString(BorderRadioButtonList.SelectedItem.Text);

    //Set the text
    GreetingsLabel.Text = GreetingsTextBox.Text;
}

protected void ControlChanged(Object sender, EventArgs e)
{
    //Update and refresh the cover
    UpdateCard();
}

```

```

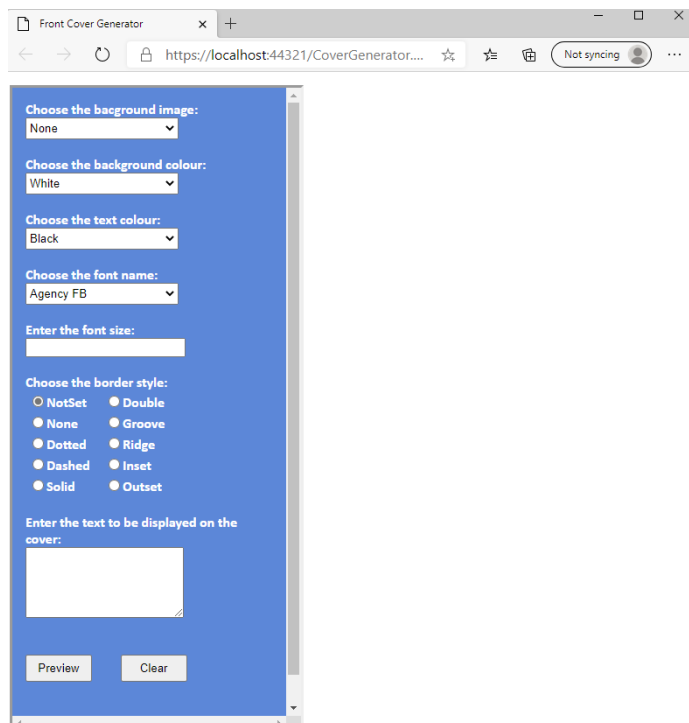
protected void PreviewButton_Click(object sender, EventArgs e)
{
    //Update and refresh the cover and hide the div
    UpdateCard();
    ConfigureDiv.Visible = false;
    EditButtonPanel.Visible = true;
}

protected void Button1_Click(object sender, EventArgs e)
{
    //Make the properties visible to configure and hide the edit button
    ConfigureDiv.Visible = true;
    EditButtonPanel.Visible = false;
}

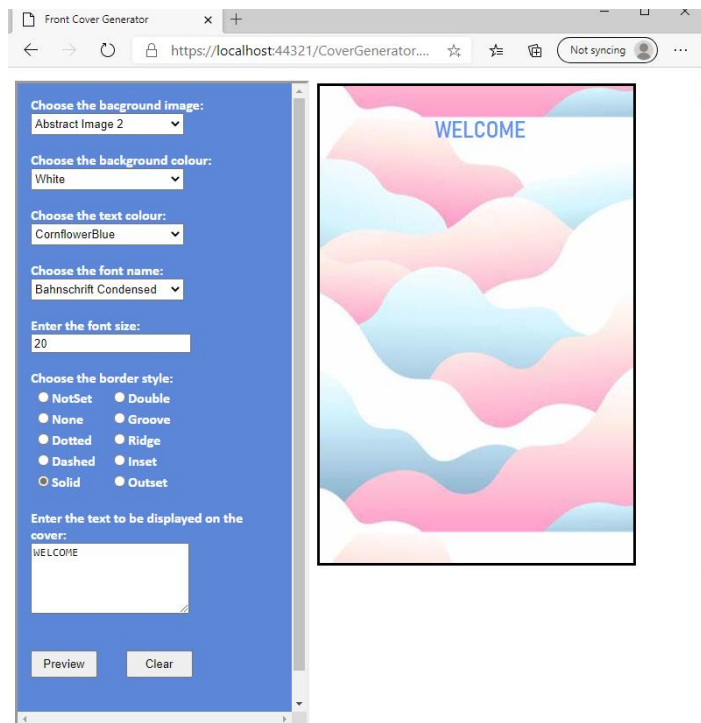
protected void ClearButton_Click(object sender, EventArgs e)
{
    //Clear entered texts and selections and set it back to default
    BackgroundDropDownList.SelectedIndex = 163;
    TextDropDownList.SelectedIndex = 34;
    BorderRadiusButtonList.SelectedIndex = 0;
    ImageDropDownList.SelectedIndex = -1;
    FontDropDownList.SelectedIndex = -1;
    SizeTextBox.Text = "";
    GreetingsTextBox.Text = "";
    UpdateCard();
}
}
}

```

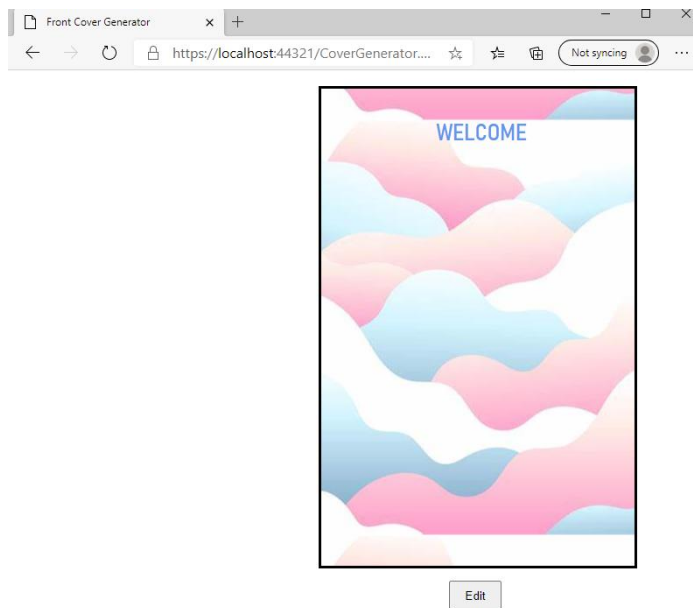
Output:



a) Default configurations. On clicking Clear it will default to this.



b) On selecting the properties



c) On clicking Preview

2. Create a web form using web controls, with employee ids in dropdown list. When the user selects the employee id, his image should be displayed in the image control. Let the next focus be on the TextBox named “name of the employee”. Use another TextBox for date of joining. Add a button as “Am I Eligible for Promotion”. On clicking the button, if he has more than 5 years of experience, then display “YES” else “NO” in the label control.

Employee.aspx:

<!DOCTYPE html>

```

        <br />
        <br />
        Date of Joining:<br />
        <asp:TextBox ID="DOJTextBox" runat="server" Height="30px" Width="145px"
TextMode="Date"></asp:TextBox>
        <br />
        <br />
        <asp:Button ID="EligibleButton" runat="server" Text="Am I Eligible?"
OnClick="PrintResult" Height="30px" Width="130px" />
        <br />
        <br />
        <asp:Label ID="ResultLabel" runat="server"></asp:Label>
        <br />
        <br />
    </div>
    <div style="left: 220px; position: absolute; top:44px; height: 150px; width: 150px;">

        <asp:Image ID="EmployeeImage" runat="server" Height="150px" Width="150px"
ImageAlign="Middle" />

    </div>
</form>
</body>
</html>

```

Employee.aspx.cs:

```

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

namespace q2
{
    public partial class Employee : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (this.IsPostBack == false)
            {
                //Add Employee IDs to the dropdown list
                EmployeeIDList.Items.Add("");
                EmployeeIDList.Items.Add("1001");
                EmployeeIDList.Items.Add("1002");
                EmployeeIDList.Items.Add("1003");
                EmployeeIDList.Items.Add("1004");
                EmployeeIDList.Items.Add("1005");

                //EmployeeIDList.Items.Insert(0, new ListItem(string.Empty, string.Empty));
                EmployeeIDList.SelectedIndex = -1;
            }
        }
        protected void UpdateInfo(Object sender, EventArgs e)
        {
            //Select Employee image
            if (EmployeeIDList.SelectedIndex != 0)
            {
                string imageUrl = "~/Images/" + EmployeeIDList.SelectedItem.Text + ".png";
                EmployeeImage.ImageUrl = imageUrl;
            }
        }
    }
}

```

```

else
    EmployeeImage.ImageUrl = "";

//Clear the other selections
EmployeeTextBox.Text = "";
DOJTextBox.Text = "";
ResultLabel.Text = "";
}

protected void PrintResult(object sender, EventArgs e)
{
    //Find years of experience
    DateTime date = DateTime.Today;
    int year = date.Year;
    string[] data = DOJTextBox.Text.Split('-');
    var JoinedYear = data[0];
    if (year - Int32.Parse(JoinedYear) > 5)
        ResultLabel.Text = "YES";
    else
        ResultLabel.Text = "NO";
}
}
}

```

Output:

The output shows two instances of the 'Employee Promotion' web application running on a local host. Both forms have the same layout: a dropdown for 'Select Employee ID', a text box for 'Name of Employee', a date picker for 'Date of Joining', and a button labeled 'Am I Eligible?'. Below the button, the result is displayed.

Employee ID	Name of Employee	Date of Joining	Result
1005	Employee5	12-06-2013	YES
1004	Employee4	23-10-2018	NO

3. Develop a simple dynamic e-card generator. E-card generator should provide options to select background colour, type of font, font size. An image can be included in the e-card with an option to provide input message.

Code:

e-card Generator.aspx:

```

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="CoverGenerator.aspx.cs"
Inherits="q4.CoverGenerator" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>e-Card Generator</title>
</head>
<body>
    <form id="form1" runat="server">
        <div style="height: auto; width: 100%">

```



```

using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Web.UI.HtmlControls;
using System.Drawing.Text;

namespace q4
{
    public partial class CoverGenerator : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (this.IsPostBack != true)
            {
                //Get list of available colours
                string[] colours = Enum.GetNames(typeof(KnownColor));

                BackgroundDropDownList.DataSource = colours;
                BackgroundDropDownList.DataBind();
                BackgroundDropDownList.SelectedIndex = 163;

                //Get list of available font families
                InstalledFontCollection fonts = new InstalledFontCollection();
                foreach (FontFamily family in fonts.Families)
                    FontDropDownList.Items.Add(family.Name);

                //Choose default image
                ImageDropDownList.SelectedIndex = -1;

                //Make cover panel visible
                CoverPanel.Visible = true;

                //Make edit button not visible
                EditButtonPanel.Visible = false;
            }
        }

        private void UpdateCard()
        {
            //Update the colours
            CoverPanel.BackColor = Color.FromName(BackgroundDropDownList.SelectedItem.Text);

            //Update image
            if (ImageDropDownList.SelectedIndex != 0)
            {
                Image1.ImageUrl = "~/Images/" + ImageDropDownList.SelectedItem.Text + ".jpg";
            }
            else
            {
                Image1.ImageUrl = "";
            }

            //Update the font name and size
            GreetingsLabel.Font.Name = FontDropDownList.SelectedItem.Text;
            try
            {
                if (Int32.Parse(SizeTextBox.Text) > 0)

```

```

        {
            GreetingsLabel.Font.Size = Int32.Parse(SizeTextBox.Text);
        }
    }
    catch
    {
    }

    //Set the text
    GreetingsLabel.Text = GreetingsTextBox.Text;
}

protected void ControlChanged(Object sender, EventArgs e)
{
    //Update and refresh the cover
    UpdateCard();
}

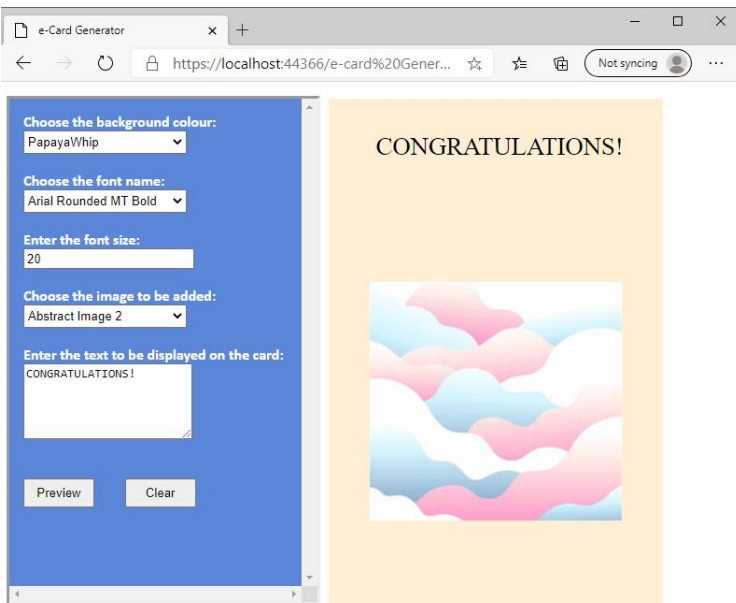
protected void PreviewButton_Click(object sender, EventArgs e)
{
    //Update and refresh the cover and hide the div
    UpdateCard();
    ConfigureDiv.Visible = false;
    EditButtonPanel.Visible = true;
}

protected void Button1_Click(object sender, EventArgs e)
{
    //Make the properties visible to configure and hide the edit button
    ConfigureDiv.Visible = true;
    EditButtonPanel.Visible = false;
}

protected void ClearButton_Click(object sender, EventArgs e)
{
    //Clear entered texts and selections and set it back to default
    BackgroundDropDownList.SelectedIndex = 163;
    ImageDropDownList.SelectedIndex = 0;
    FontDropDownList.SelectedIndex = -1;
    SizeTextBox.Text = "";
    GreetingsTextBox.Text = "";
    UpdateCard();
}
}
}

```

Output:



4. Develop simple application using web controls to reproduce the given Captcha. Upon match, suitable message has to be displayed. If there is a mismatch for more than 3 times, TextBox has to be disabled. (Hint: Use Hidden TextBox).

Code:

Captcha.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="captcha.aspx.cs" Inherits="q4.captcha" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Captcha</title>
</head>
<body>
    <form id="form1" runat="server">
        <div style="height: 213px; width: 268px">
            &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~
                <br />
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;& Enter the text<br />
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&
                <br />
                &nbsp;&nbsp;&
                <asp:Image ID="CaptchaImage" runat="server" Height="44px" ImageUrl="~/Images/botdetect- captcha-cut.jpg" Width="185px" />
                <br />
                <br />
&nbsp;&nbsp;&~
                <asp:TextBox ID="CaptchaTextBox" runat="server" style="margin-top: 0px" Width="157px" OnTextChanged="AttemptMade" AutoPostBack="true" Height="18px"></asp:TextBox>
                <br />
                <br />
&nbsp;&~
                <asp:Label ID="ResultLabel" runat="server"></asp:Label>
            </div>
        </form>
    </body>
</html>
```

Captcha.aspx.cs:

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

namespace q4
{
    public partial class captcha : System.Web.UI.Page
    {
        //To keep a count on number of tries
        static int count;

        //Initialise the text in each image
        string[] text = { "T4Y9", "D56Y", "3G6B", "P35B" };

        //To choose a random captcha from the set of images
        Random r = new Random();
        static int num;
```

```

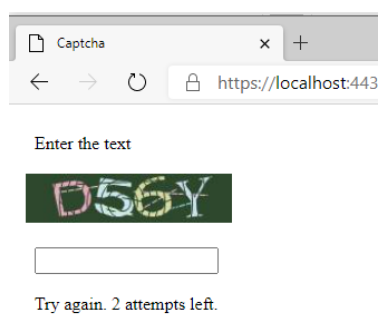
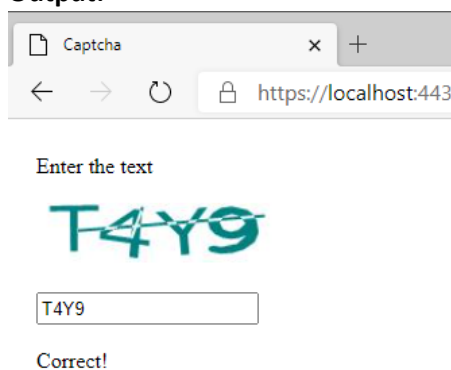
protected void Page_Load(object sender, EventArgs e)
{
    if(this.IsPostBack == false)
    {
        count = 0;
        num = r.Next(0, 3);

        string url = "~/Images/captcha" + num.ToString() + ".jpg";
        CaptchaImage.ImageUrl = url;
    }
}

protected void AttemptMade(object sender, EventArgs e)
{
    count++;
    if(count < 3)
    {
        string attempt = CaptchaTextBox.Text;
        if(attempt == text[num])
        {
            ResultLabel.Text = "Correct!";
            CaptchaTextBox.Text = attempt;
        }
        else
        {
            string res = "Try again. " + (3 - count).ToString() + " attempts left.";
            ResultLabel.Text = res;
            CaptchaTextBox.Text = "";
        }
    }
    else
    {
        CaptchaTextBox.Visible = false;
        ResultLabel.Text = "Too many wrong attempts.";
    }
}
}
}

```

Output:



a) When entered correctly

b) When entered incorrectly

c) On 3 incorrect attempts