Bahria University,

Karachi Campus



LAB EXPERIMENT NO.

**8**

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| **1** | Develop a web service in ASP.NET to retrieve current weather data based on location. Then build an ASP.NET web application to consume the weather service and display current weather information for a given location. |
| 2 | Design and implement a web service that solves quadratic equations using different method (at least 3), then build an ASP.NET web application to consume this service. |

Submitted On:

6 June 2024

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

(Date: DD/MM/YY)

**Task 1**

Develop a web service in ASP.NET to retrieve current weather data based on location. Then build an ASP.NET web application to consume the weather service and display current weather information for a given location.

**Web Service**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Web;

using System.Web.Services;

namespace Lab8

{

[WebService(Namespace = "http://tempuri.org/")]

[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1\_1)]

[System.ComponentModel.ToolboxItem(false)]

public class WeatherService : System.Web.Services.WebService

{

[WebMethod]

public string GetWeatherData(string location)

{

//Normally you would use a API to get realtime weather info but API keys are paid :,)

location = location.ToLower();

switch (location)

{

case "hyderabad":

return "Weather is Sunny and Temprature is 41C";

break;

case "karachi":

return "Weather is Rainy and Temprature is 34C";

break;

default:

return "Invalid location";

break;

}

}

}

}

**Web Service Consumer Class**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace WeatherServiceConsumer

{

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

{

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

{

}

protected void btnWeatherClick(object sender, EventArgs e)

{

WeatherServiceRef.WeatherService service = new WeatherServiceRef.WeatherService();

string location = txtLocation.Text;

lblResult.Text = service.GetWeatherData(location);

}

}

}

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

<!DOCTYPE html>

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

<head runat="server">

<title></title>

</head>

<body>

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

<div>

<label for="txtLocation">Location:</label>

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

</div>

<div>

<asp:Button ID="btnGetWeather" runat="server" Text="Get Weather" OnClick="btnWeatherClick" />

</div>

<div>

<label>Result:</label>

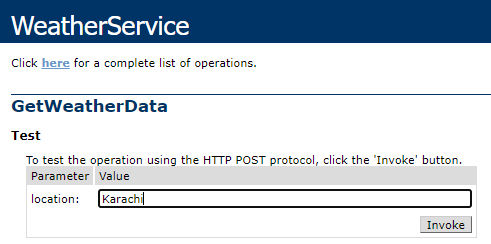
<asp:Label ID="lblResult" runat="server"></asp:Label>

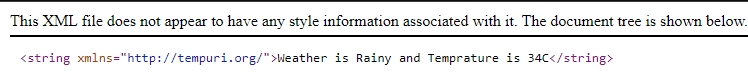
</div>

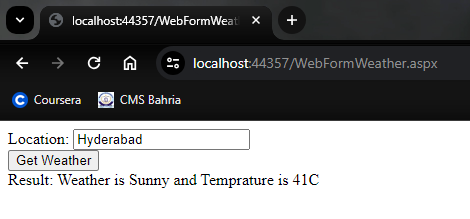
</form>

</body>

</html>







**Task 2**

Design and implement a web service that solves quadratic equations using different method (atleast 3), then build an ASP.NET web application to consume this service.

**Web Service**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Services;

namespace Lab8

{

[WebService(Namespace = "http://tempuri.org/")]

[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1\_1)]

[System.ComponentModel.ToolboxItem(false)]

public class QuadCalculator : System.Web.Services.WebService

{

[WebMethod]

public string method1(double a, double b, double c)

{

string ans= "Solve Quadratic Using Formula:\n";

double discriminant = b \* b - 4 \* a \* c;

if (discriminant > 0)

{

double root1 = (-b + Math.Sqrt(discriminant)) / (2 \* a);

double root2 = (-b - Math.Sqrt(discriminant)) / (2 \* a);

ans += "\nRoots are real and distinct.";

ans += "\nRoot 1: " + root1;

ans += "\nRoot 2: " + root2;

}

else if (discriminant == 0)

{

double root = -b / (2 \* a);

ans += "\nRoots are real and equal.";

ans += "\nRoot: " + root;

}

else

{

double realPart = -b / (2 \* a);

double imaginaryPart = Math.Sqrt(-discriminant) / (2 \* a);

ans += "\nRoots are complex and distinct.";

ans += "\nRoot 1: " + realPart + " + " + imaginaryPart + "i";

ans += "\nRoot 2: " + realPart + " - " + imaginaryPart + "i";

}

return ans;

}

[WebMethod]

public string method2(double a, double b, double c)

{

double m = b / (2 \* a);

double n = c-((b\*b)/(4\*a));

if (n < 0)

{

n\*=-1;

}

m\*=-1;

double root1 = m + Math.Sqrt(n);

double root2 = m - Math.Sqrt(n);

return "Solve Quadratic Using Complete Square:\nRoot 1: " + root1 +"\nRoot 2: "+ root2;

}

[WebMethod]

public string method3(double a, double b, double c)

{

string ans = "Solve Quadratic Using Factoring:\n";

if (a == 0)

{

ans += "\nThis is not a quadratic equation.";

return ans;

}

// Finding factors of 'a \* c' that sum up to 'b'

double product = a \* c;

double factor1 = 0, factor2 = 0;

for (double i = 1; i <= Math.Abs(product); i++)

{

if (product % i == 0)

{

if (product / i + i == b / a)

{

factor1 = i;

factor2 = product / i;

break;

}

}

}

if (factor1 == 0 && factor2 == 0)

{

ans += "\nThe equation cannot be factored over the integers.";

return ans;

}

// Roots are (-b/a) + factor1 and (-b/a) + factor2

double root1 = (-b / a) + factor1;

double root2 = (-b / a) + factor2;

ans += "\nRoot 1: " + root1;

ans += "\nRoot 2: " + root2;

return ans;

}

}

}

**Web Service Consumer**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace QuadCalServiceConsumer

{

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

{

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

{

}

protected void btnMethod1Click(object sender, EventArgs e)

{

QuadCalculatorServiceRef.QuadCalculator qc = new QuadCalculatorServiceRef.QuadCalculator();

double a = double.Parse(txta.Text);

double b = double.Parse(txtb.Text);

double c = double.Parse(txtc.Text);

lblResult.Text = qc.method1(a, b, c);

}

protected void btnMethod2Click(object sender, EventArgs e)

{

QuadCalculatorServiceRef.QuadCalculator qc = new QuadCalculatorServiceRef.QuadCalculator();

double a = double.Parse(txta.Text);

double b = double.Parse(txtb.Text);

double c = double.Parse(txtc.Text);

lblResult.Text = qc.method2(a, b, c);

}

protected void btnMethod3Click(object sender, EventArgs e)

{

QuadCalculatorServiceRef.QuadCalculator qc = new QuadCalculatorServiceRef.QuadCalculator();

double a = double.Parse(txta.Text);

double b = double.Parse(txtb.Text);

double c = double.Parse(txtc.Text);

lblResult.Text = qc.method3(a, b, c);

}

}

}

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

<!DOCTYPE html>

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

<head runat="server">

<title></title>

</head>

<body>

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

<div>

<label for="txtFormulla">Location:ax^2 + bx + c</label>

</div>

<div>

<label for="txta">a:</label>

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

</div>

<div>

<label for="txtb">b:</label>

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

</div>

<div>

<label for="txtc">c:</label>

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

</div>

<div>

<asp:Button ID="btnMethod1" runat="server" Text="Quadratic Formula" OnClick="btnMethod1Click" />

</div>

<div>

<asp:Button ID="btnMethod2" runat="server" Text="Complete Square" OnClick="btnMethod2Click" />

</div>

<div>

<asp:Button ID="btnMethod3" runat="server" Text="Factorization" OnClick="btnMethod3Click" />

</div>

<div>

<label>Result:</label>

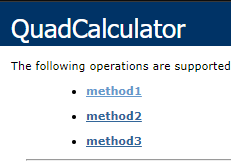
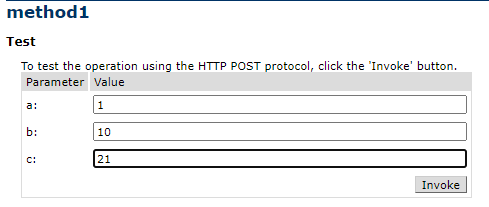
<asp:Label ID="lblResult" runat="server"></asp:Label>

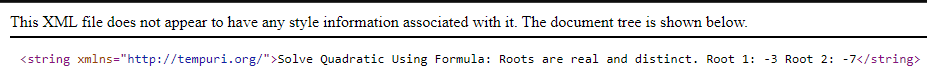
</div>

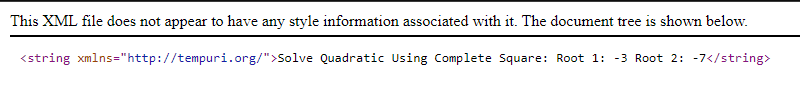
</form>

</body>

</html>



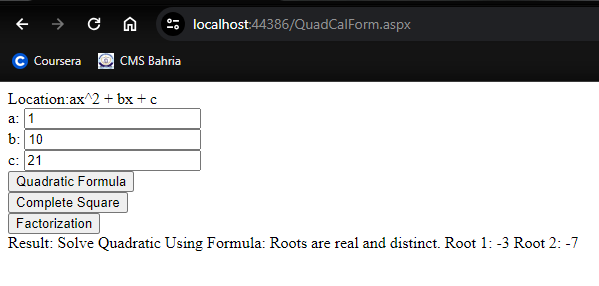


**Using Quad Formula**

A screenshot of a computer

Description automatically generated

**Using Complete Square**



**Using Factorization**

