TASK 12:

Using the System.NetWebClient to Retrieve or Upload Data:

Demonstrate how to create windows form that can use HTTP to download and save a resource from a specified URI, upload a resource to a specified URI.

Procedure :

1. **Create a Windows Forms Application:**
   * Open Visual Studio and create a new Windows Forms App (.NET Framework).
2. **Design** the Windows Form (MainForm.cs):
   * Add a TextBox for the uploading URL.
   * Add a TextBox for the down loading URL.
   * Add a Button for downloading data.
   * Add a Button for uploading data.
   * Add a Label to display messages.

3.double click the button to open .cs code

Code:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.IO;

using System.Linq;

using System.Net;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace WindowsFormsApp1

{

public partial class Form1 : Form

{

WebClient client = new WebClient();

//It is used to communicate with web servers using HTTP

public Form1()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

try

{

string url = textBox1.Text.Trim();

if (string.IsNullOrEmpty(url))

{

label2.Text = "Please enter a download URL.";

return;

}

string savePath = "C:\\Users\\cvr\\Desktop\\download.txt";

client.DownloadFile(url, savePath);

label2.Text = "File downloaded to " + savePath;

}

catch (Exception ex)

{

label2.Text = "Error: " + ex.Message;

}

}

private void button2\_Click(object sender, EventArgs e)

{

try

{

string url = textBox2.Text; // Example: https://httpbin.org/post

string filePath = "G:\\TEMP\\dummy.txt";

client.UploadFile(url, "POST", filePath);

MessageBox.Show("File uploaded successfully!");

label4.Text = "Upload completed";

}

catch (Exception ex)

{

label4.Text = "Error: " + ex.Message;

} } } }

Out put:

