

```
1 using System;
2 using System.Net.Http;
3 using System.Text.Json;
4 using System.Threading.Tasks;
5 using System.Windows.Forms;
6
7 namespace WinFormsApp1
8 {
9     public partial class Form1 : Form
10    {
11        private readonly HttpClient _httpClient;
12        private readonly string _baseUrl;
13
14        public Form1()
15        {
16            InitializeComponent();
17
18            _httpClient = new HttpClient();
19
20            _baseUrl = "https://localhost:7227/";
21
22        }
23
24        private async void MainForm_Load(object sender, EventArgs e)
25        {
26            try
27            {
28                var responseDateTime = await _httpClient.GetAsync  ↗
29                    (_baseUrl + "DateTime");
30                responseDateTime.EnsureSuccessStatusCode();
31                var dateTime = await  ↗
32                    responseDateTime.Content.ReadAsStringAsync();
33
34                textBox_Date.Text = dateTime;
35            }
36            catch (HttpRequestException ex)
37            {
38                MessageBox.Show($"Request failed: {ex.Message}");
39            }
40        }
41
42        private async void buttonList_Click(object sender, EventArgs e)
43        {
44            try
45            {
46                var responseList = await _httpClient.GetAsync  ↗
47                    (_baseUrl + "List");
48                responseList.EnsureSuccessStatusCode();
49                var elementsJson = await  ↗
50                    responseList.Content.ReadAsStringAsync();
51                var elementsList =  ↗
52                    JsonSerializer.Deserialize<List<string>>
```

```
        (elementsJson);
49
50        listBox_List.Items.Clear();
51        foreach (var element in elementsList)
52        {
53            listBox_List.Items.Add(element);
54        }
55    }
56    catch (HttpRequestException ex)
57    {
58        MessageBox.Show($"Request failed: {ex.Message}");
59    }
60 }
61
62 private async void buttonDate_Click(object sender, EventArgs e)
63 {
64     try
65     {
66         var responseDateTime = await _httpClient.GetAsync      ↗
67             (_baseUrl + "DateTime");
68         responseDateTime.EnsureSuccessStatusCode();
69         var dateTime = await
70             responseDateTime.Content.ReadAsStringAsync();      ↗
71
72         textBox_Date.Text = dateTime;
73     }
74     catch (HttpRequestException ex)
75     {
76         MessageBox.Show($"Request failed: {ex.Message}");
77     }
78 }
79
80 private async void buttonFC_Click(object sender, EventArgs e)
81 {
82     try
83     {
84         int celsius = int.Parse(textBoxC.Text);
85         var responseCtoF = await _httpClient.GetAsync(_baseUrl  ↗
86             + $"CtoF?c={celsius}");
87         responseCtoF.EnsureSuccessStatusCode();
88         var fahrenheitString = await
89             responseCtoF.Content.ReadAsStringAsync();          ↗
90         textBoxR.Text = fahrenheitString;
91     }
92     catch (HttpRequestException ex)
93     {
94         MessageBox.Show($"Request failed: {ex.Message}");
95     }
96 }
97
98 private async void buttonCF_Click(object sender, EventArgs e)
99 {
100     try
```

```
97         {
98             int fahrenheit = int.Parse(textBoxF.Text);
99             var responseFtoC = await _httpClient.GetAsync(_baseUrl
100                 + $"FtoC?f={fahrenheit}");
101             responseFtoC.EnsureSuccessStatusCode();
102             var celsiusString = await
103                 responseFtoC.Content.ReadAsStringAsync();
104             textBoxR.Text = celsiusString;
105         }
106     catch (HttpRequestException ex)
107     {
108         MessageBox.Show($"Request failed: {ex.Message}");
109     }
110 }
111 private async void buttonConv_Click(object sender, EventArgs e)
112 {
113     try
114     {
115         int ron = int.Parse(textBoxRon.Text);
116         var responseCurCon = await _httpClient.GetAsync
117             (_baseUrl + $"RonEur?r={ron}");
118         responseCurCon.EnsureSuccessStatusCode();
119         var eurString = await
120             responseCurCon.Content.ReadAsStringAsync();
121         textBoxEur.Text = eurString;
122     }
123     catch (HttpRequestException ex)
124     {
125         MessageBox.Show($"Request failed: {ex.Message}");
126     }
127 }
```