五專資工三甲

視窗程式設計

5B0G0024

莊鈞凱

ZHUANG JUN KAI

題目

請依照上課內容，設計一個擷取 AQI 網路資料並呈現於圖表的系統，包含如下功能：

- 透過非同步擷取方式取得網址內之資料：https://data.moenv.gov.tw/api/v2/aqx_p_432?api_key=e8dd42e6-9b8b-43f8-991e-b3dee723a52d&limit=1000&sort=ImportDate desc&format=JSON。
- 使用一個 DataGrid 來呈現此 JSON 資料。
- 動態產生空氣品質指標 CheckBox。
- 使用 LiveChart 元件來呈現資料。

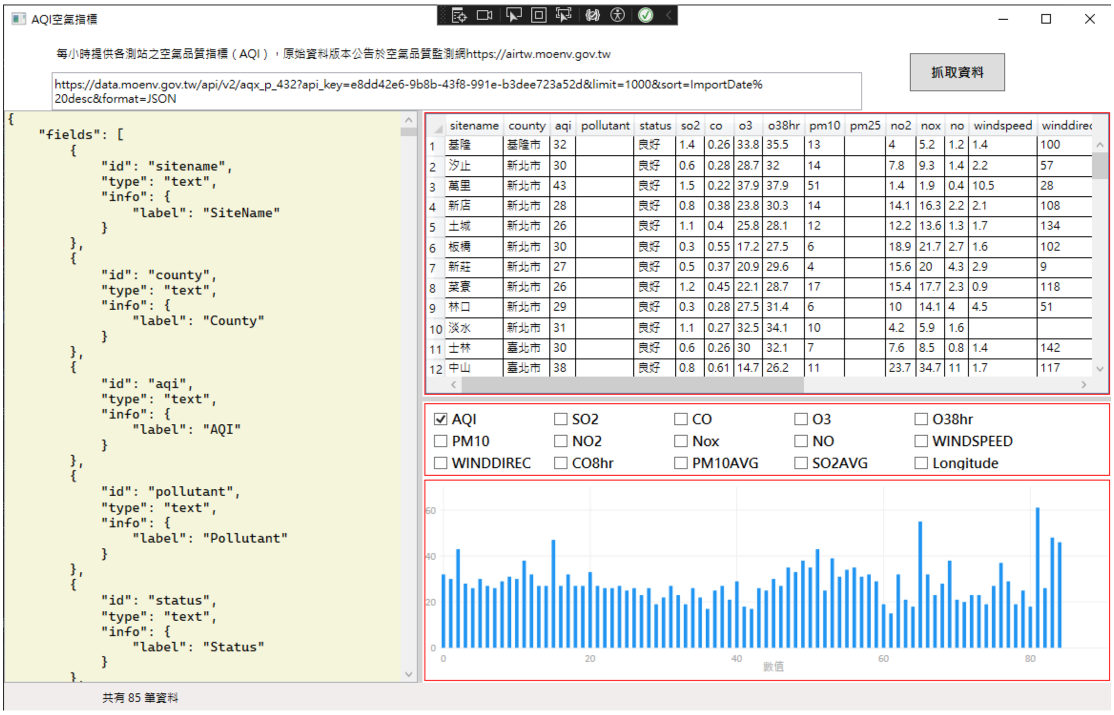
請將你的程式碼與程式執行畫面，輸出並彙整成一個 pdf 檔案來繳交，pdf 檔名請以學號+作業編號方式命名，列如第一個作業檔名為 A123456789ex1.pdf。請注意作業繳交要求、日期與格式。

```xml
<Window x:Class="WebBug.MainWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
        xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
        xmlns:lvc="clr-namespace:LiveCharts.Wpf;assembly=LiveCharts.Wpf"
        xmlns:local="clr-namespace:WebBug"
        mc:Ignorable="d"
        Title="AQI空氣指標" Height="800" Width="1200">
    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="80"/>
            <RowDefinition Height="500*"/>
            <RowDefinition Height="30"/>
        </Grid.RowDefinitions>
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="300*"/>
            <ColumnDefinition Width="5"/>
            <ColumnDefinition Width="500*"/>
        </Grid.ColumnDefinitions>

        <StackPanel Grid.Row="0" Grid.Column="0" Grid.ColumnSpan="3" Height="80" Width="Auto" Orientation="Horizontal">
            <StackPanel Orientation="Vertical" Width="900">
                <Label Content="每小時提供各測站之空氣品質指標（AQI），原始資料版本公告於空氣品質監測網https://airtw.moenv.gov.tw" Height="40" VerticalAlignment="Center" VerticalContentAlignment="Center" Margin="50, 0, 0, 0"/>
                <TextBox x:Name="UrlTextBox" VerticalAlignment="Center" Height="40" Margin="50, 0, 0, 0" TextWrapping="Wrap" VerticalContentAlignment="Center"/>
            </StackPanel>
            <Button x:Name="GetWebDataButton" Content="抓取資料" VerticalAlignment="Center" Height="40" Width="100" Margin="50, 0, 0, 0" Click="GetWebDataButton_Click" FontSize="14" FontWeight="Bold"/>
        </StackPanel>

        <TextBox x:Name="ContentTextBox" Grid.Row="1" Grid.Column="0" Background=□"Beige" FontSize="14" FontFamily="Cascadia Code" ScrollViewer.VerticalScrollBarVisibility="Auto"/>

        <GridSplitter Grid.Row="1" Grid.Column="1" Width="5" VerticalAlignment="Stretch" HorizontalAlignment="Center" Background=□"LightGray"/>

        <Grid Grid.Row="1" Grid.Column="2">
            <Grid.RowDefinitions>
                <RowDefinition Height="300"/>
                <RowDefinition Height="5"/>
                <RowDefinition Height="80"/>
                <RowDefinition/>
            </Grid.RowDefinitions>
            <Border Grid.Row="0" Margin="2, 2, 2, 2" BorderBrush=🟥"Red"
               BorderThickness="1">
                <DataGrid x:Name="RecordDataGrid"
                  AutoGenerateColumns="True"
                  SelectionChanged="RecordDataGrid_SelectionChanged"
                  LoadingRow="RecordDataGrid_LoadingRow"
                  IsReadOnly="True"/>
            </Border>
            <GridSplitter Grid.Row="1" Height="5"
              HorizontalAlignment="Stretch" VerticalAlignment="Center"
              Background=⬜"LightGray"/>
            <Border Grid.Row="2" Margin="2, 2, 2, 2" BorderBrush=🟥"Red"
               BorderThickness="1">
                <WrapPanel Name="DataWrapPanel" Orientation="Horizontal"
                   Margin="5, 5, 5, 5">
                </WrapPanel>
            </Border>
            <Border Grid.Row="3" Margin="2, 2, 2, 2" BorderBrush=🟥"Red"
               BorderThickness="1">
                <lvc:CartesianChart x:Name="AQIChart" MinWidth="500">
                    <lvc:CartesianChart.AxisX>
                        <lvc:Axis Title="數值"/>
                    </lvc:CartesianChart.AxisX>
                </lvc:CartesianChart>
            </Border>
        </Grid>

        <StatusBar Grid.Row="2" Grid.Column="0" Grid.ColumnSpan="3">
            <StatusBarItem Margin="100, 0, 0, 0">
                <TextBlock x:Name="StatusTextBlock" Text="Status" />
            </StatusBarItem>
        </StatusBar>
    </Grid>
</Window>
```

```csharp
using LiveCharts;
using LiveCharts.Wpf;
using System.Collections.Generic;
using System;
using System.Net.Http;
using System.Text.Json;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using WebBug;
using System.Linq;

namespace WebBug
{
    /// <summary>
    /// Interaction logic for MainWindow.xaml
    /// </summary>
    public partial class MainWindow : Window
    {
        string url = "https://data.moenv.gov.tw/api/v2/aqx_p_432?api_key=e8dd42e6-9b8b-43f8-991e-b3dee723a52d&limit=1000&sort=ImportDate%20desc&format=JSON";
        AQIdata aqidata = new AQIdata();
        List<Field> fields = new List<Field>();
        List<Record> records = new List<Record>();
        List<Record> selectedRecords = new List<Record>();
        SeriesCollection seriesCollection = new SeriesCollection();
        public MainWindow()
        {
            InitializeComponent();
            UrlTextBox.Text = url;
            selectedRecords.Clear();
        }

        private async void GetWebDataButton_Click(object sender, RoutedEventArgs e)
        {
            ContentTextBox.Text = "正在抓取網路資料 ...";

            string jsontext = await FetchContentAsync(url);
            ContentTextBox.Text = jsontext;
            aqidata = JsonSerializer.Deserialize<AQIdata>(jsontext);
            fields = aqidata.fields.ToList();
            records = aqidata.records.ToList();
            selectedRecords = records;
            StatusTextBlock.Text = $"共有 {records.Count} 筆資料";
            DisplayAQIData();
        }

        private void DisplayAQIData()
        {
            RecordDataGrid.ItemsSource = records;
            DataWrapPanel.Children.Clear();

            foreach (var field in fields)
            {
                var propertyInfo = typeof(Record).GetProperty(field.id);
                if (propertyInfo != null)
                {
                    string value = propertyInfo.GetValue(records[0]) as string;
                    if (double.TryParse(value, out double v))
                    {
                        CheckBox cb = new CheckBox
                        {
                            Content = field.info.label,
                            Tag = field.id,
                            Margin = new Thickness(3),
                            Width = 120,
                            FontSize = 14,
                            FontWeight = FontWeights.Bold,
                        };

                        cb.Checked += UpdateChart;
                        cb.Unchecked += UpdateChart;
                        DataWrapPanel.Children.Add(cb);
                    }
                }
            }
        }

        private void UpdateChart(object sender, RoutedEventArgs e)
        {
            seriesCollection.Clear();

            foreach (CheckBox cb in DataWrapPanel.Children)
            {
                if (cb.IsChecked == true)
                {
                    var tag = cb.Tag as String;
                    ColumnSeries columnSeries = new ColumnSeries();
                    ChartValues<double> values = new ChartValues<double>();
                    List<String> labels = new List<String>();

                    foreach (var record in selectedRecords)
                    {
                        var propertyInfo = typeof(Record).GetProperty(tag);
                        if (propertyInfo != null)
                        {
                            string value = propertyInfo.GetValue(record) as string;
                            if (double.TryParse(value, out double v))
                            {
                                values.Add(v);
                                labels.Add(record.sitename);
                            }
                        }
                    }
                    columnSeries.Values = values;
                    columnSeries.Title = tag;
                    columnSeries.LabelPoint = point => $"{labels[(int)point.X]}: {point.Y.ToString()}";
                    seriesCollection.Add(columnSeries);
                }
            }
            AQIChart.Series = seriesCollection;
        }

        private async Task<string> FetchContentAsync(string url)
        {
            try
            {
                using (HttpClient client = new HttpClient())
                {
                    return await client.GetStringAsync(url);
                }
            }
            catch (Exception ex)
            {
                return $"發生錯誤: {ex.Message}";
            }
        }

        private void RecordDataGrid_SelectionChanged(object sender, SelectionChangedEventArgs e)
        {
            selectedRecords = RecordDataGrid.SelectedItems.Cast<Record>().ToList();
            StatusTextBlock.Text = $"共選取 {selectedRecords.Count} 筆資料";
        }

        private void RecordDataGrid_LoadingRow(object sender, DataGridRowEventArgs e)
        {
            e.Row.Header = e.Row.GetIndex() + 1;
        }
    }
}
```

```csharp
namespace WebBug
{

    public class AQIdata
    {
        public Field[] fields { get; set; }
        public string resource_id { get; set; }
        public __Extras __extras { get; set; }
        public bool include_total { get; set; }
        public string total { get; set; }
        public string resource_format { get; set; }
        public string limit { get; set; }
        public string offset { get; set; }
        public _Links _links { get; set; }
        public Record[] records { get; set; }
    }

    public class __Extras
    {
        public string api_key { get; set; }
    }

    public class _Links
    {
        public string start { get; set; }
        public string next { get; set; }
    }

    public class Field
    {
        public string id { get; set; }
        public string type { get; set; }
        public Info info { get; set; }
    }

    public class Info
    {
        public string label { get; set; }
    }

    public class Record
    {
        public string sitename { get; set; }
        public string county { get; set; }
        public string aqi { get; set; }
        public string pollutant { get; set; }
        public string status { get; set; }
        public string so2 { get; set; }
        public string co { get; set; }
        public string o3 { get; set; }
        public string o3_8hr { get; set; }
        public string pm10 { get; set; }
        public string pm25 { get; set; }
        public string no2 { get; set; }
        public string nox { get; set; }
        public string no { get; set; }
        public string wind_speed { get; set; }
        public string wind_direc { get; set; }
        public string publishtime { get; set; }
        public string co_8hr { get; set; }
        public string pm25_avg { get; set; }
        public string pm10_avg { get; set; }
        public string so2_avg { get; set; }
        public string longitude { get; set; }
        public string latitude { get; set; }
        public string siteid { get; set; }
    }

}

```

輸出結果：



每小時提供各測站之空氣品質指標（AQI），原始資料版本公告於空氣品質監測網https://airtw.moenv.gov.tw

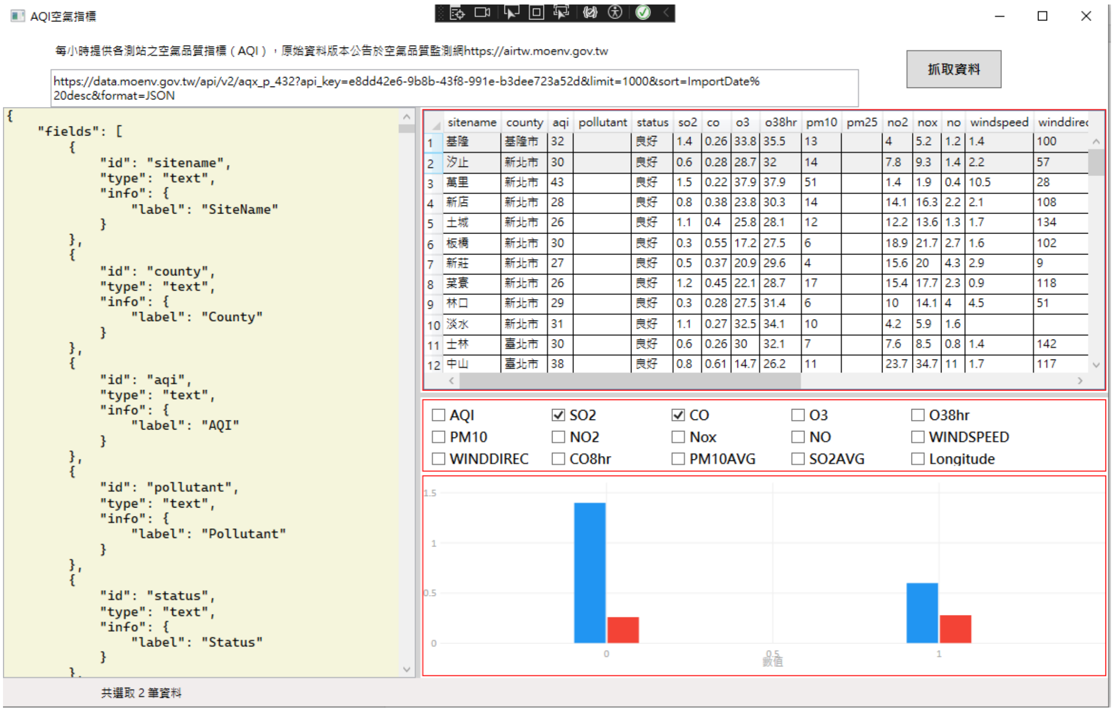
https://data.moenv.gov.tw/api/v2/aqx_p_432?api_key=e8dd42e6-9b8b-43f8-991e-b3dee723a52d&limit=1000&sort=ImportDate%20desc&format=JSON

```
{
    "fields": [
        {
            "id": "sitename",
            "type": "text",
            "info": {
                "label": "SiteName"
            }
        },
        {
            "id": "county",
            "type": "text",
            "info": {
                "label": "County"
            }
        },
        {
            "id": "aqi",
            "type": "text",
            "info": {
                "label": "AQI"
            }
        },
        {
            "id": "pollutant",
            "type": "text",
            "info": {
                "label": "Pollutant"
            }
        },
        {
            "id": "status",
            "type": "text",
            "info": {
                "label": "Status"
            }
        },
```

	sitename	county	aqi	pollutant	status	so2	co	o3	o38hr	pm10	pm25	no2	nox	no	windspeed	winddirec
1	基隆	基隆市	32		良好	1.4	0.26	33.8	35.5	13		4	5.2	1.2	1.4	100
2	汐止	新北市	30		良好	0.6	0.28	28.7	32	14		7.8	9.3	1.4	2.2	57
3	萬里	新北市	43		良好	1.5	0.22	37.9	37.9	51		1.4	1.9	0.4	10.5	28
4	新店	新北市	28		良好	0.8	0.38	23.8	30.3	14		14.1	16.3	2.2	2.1	108
5	土城	新北市	26		良好	1.1	0.4	25.8	28.1	12		12.2	13.6	1.3	1.7	134
6	板橋	新北市	30		良好	0.3	0.55	17.2	27.5	6		18.9	21.7	2.7	1.6	102
7	新莊	新北市	27		良好	0.5	0.37	20.9	29.6	4		15.6	20	4.3	2.9	9
8	菜寮	新北市	26		良好	1.2	0.45	22.1	28.7	17		15.4	17.7	2.3	0.9	118
9	林口	新北市	29		良好	0.3	0.28	27.5	31.4	6		10	14.1	4	4.5	51
10	淡水	新北市	31		良好	1.1	0.27	32.5	34.1	10		4.2	5.9	1.6		
11	士林	臺北市	30		良好	0.6	0.26	30	32.1	7		7.6	8.5	0.8	1.4	142
12	中山	臺北市	38		良好	0.8	0.61	14.7	26.2	11		23.7	34.7	11	1.7	117

☑ AQI ☐ SO2 ☐ CO ☐ O3 ☐ O38hr
☐ PM10 ☐ NO2 ☐ Nox ☐ NO ☐ WINDSPEED
☐ WINDDIREC ☐ CO8hr ☐ PM10AVG ☐ SO2AVG ☐ Longitude

共有 85 筆資料



每小時提供各測站之空氣品質指標（AQI），原始資料版本公告於空氣品質監測網https://airtw.moenv.gov.tw

https://data.moenv.gov.tw/api/v2/aqx_p_432?api_key=e8dd42e6-9b8b-43f8-991e-b3dee723a52d&limit=1000&sort=ImportDate%20desc&format=JSON

```
{
    "fields": [
        {
            "id": "sitename",
            "type": "text",
            "info": {
                "label": "SiteName"
            }
        },
        {
            "id": "county",
            "type": "text",
            "info": {
                "label": "County"
            }
        },
        {
            "id": "aqi",
            "type": "text",
            "info": {
                "label": "AQI"
            }
        },
        {
            "id": "pollutant",
            "type": "text",
            "info": {
                "label": "Pollutant"
            }
        },
        {
            "id": "status",
            "type": "text",
            "info": {
                "label": "Status"
            }
        },
```

	sitename	county	aqi	pollutant	status	so2	co	o3	o38hr	pm10	pm25	no2	nox	no	windspeed	winddirec
1	基隆	基隆市	32		良好	1.4	0.26	33.8	35.5	13		4	5.2	1.2	1.4	100
2	汐止	新北市	30		良好	0.6	0.28	28.7	32	14		7.8	9.3	1.4	2.2	57
3	萬里	新北市	43		良好	1.5	0.22	37.9	37.9	51		1.4	1.9	0.4	10.5	28
4	新店	新北市	28		良好	0.8	0.38	23.8	30.3	14		14.1	16.3	2.2	2.1	108
5	土城	新北市	26		良好	1.1	0.4	25.8	28.1	12		12.2	13.6	1.3	1.7	134
6	板橋	新北市	30		良好	0.3	0.55	17.2	27.5	6		18.9	21.7	2.7	1.6	102
7	新莊	新北市	27		良好	0.5	0.37	20.9	29.6	4		15.6	20	4.3	2.9	9
8	菜寮	新北市	26		良好	1.2	0.45	22.1	28.7	17		15.4	17.7	2.3	0.9	118
9	林口	新北市	29		良好	0.3	0.28	27.5	31.4	6		10	14.1	4	4.5	51
10	淡水	新北市	31		良好	1.1	0.27	32.5	34.1	10		4.2	5.9	1.6		
11	士林	臺北市	30		良好	0.6	0.26	30	32.1	7		7.6	8.5	0.8	1.4	142
12	中山	臺北市	38		良好	0.8	0.61	14.7	26.2	11		23.7	34.7	11	1.7	117

☐ AQI ☑ SO2 ☑ CO ☐ O3 ☐ O38hr
☐ PM10 ☐ NO2 ☐ Nox ☐ NO ☐ WINDSPEED
☐ WINDDIREC ☐ CO8hr ☐ PM10AVG ☐ SO2AVG ☐ Longitude

共選取 2 筆資料