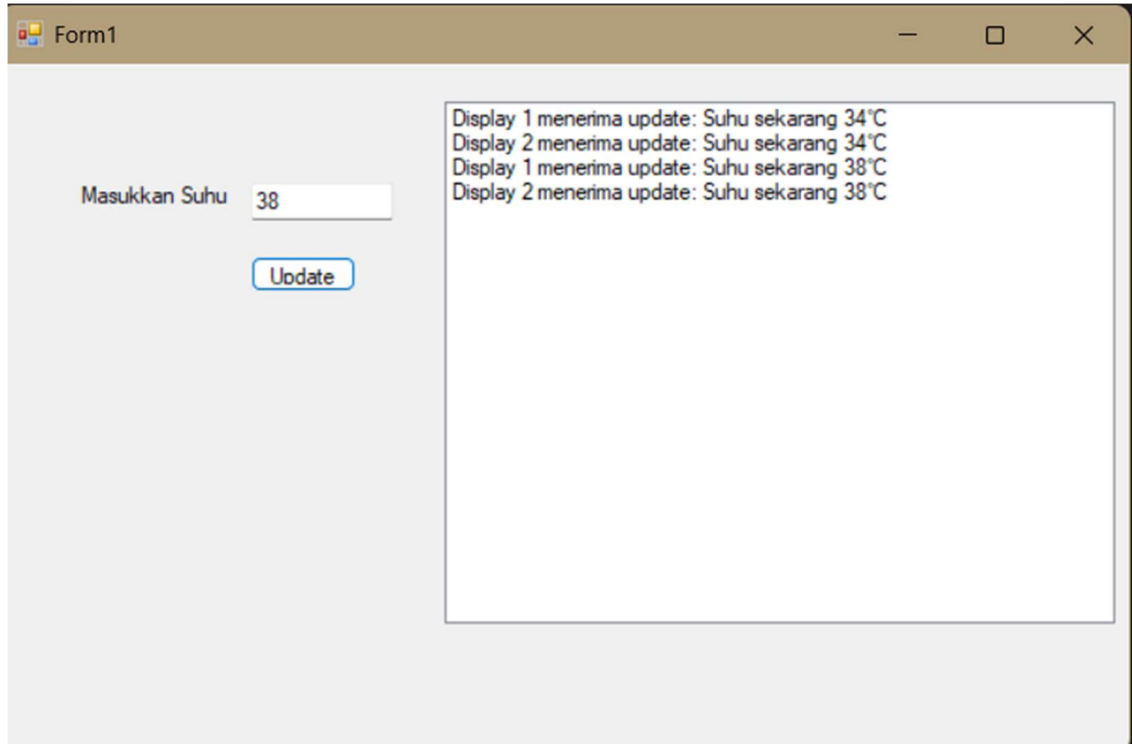


JURNAL MODUL 13

Nama: Zhaafir Zaidan Avail

NIM: 2311104059

Hasil Run:



Struktur:

TJ13_2311104059/

- IObserver.cs
- ISubject.cs
- WeatherStation.cs
- DisplayDevice.cs
- Form1.cs (GUI + logic)
- Program.cs

Buat Interface: IObserver dan ISubject

```
--IObserver.cs--
public interface IObserver
{
    void Update(float temperature);
}

--ISubject.cs--
public interface ISubject
{
    void RegisterObserver(IObserver observer);
    void RemoveObserver(IObserver observer);
    void NotifyObservers();
}
```

Buat Subject: WeatherStation.cs

```
using System.Collections.Generic;
```

```

public class WeatherStation : ISubject
{
    private List<IObserver> observers = new List<IObserver>();
    private float temperature;

    public void RegisterObserver(IObserver observer)
    {
        observers.Add(observer);
    }

    public void RemoveObserver(IObserver observer)
    {
        observers.Remove(observer);
    }

    public void NotifyObservers()
    {
        foreach (var observer in observers)
        {
            observer.Update(temperature);
        }
    }

    public void SetTemperature(float temp)
    {
        this.temperature = temp;
        NotifyObservers();
    }
}

```

Buat Observer: DisplayDevice.cs

```

using System.Windows.Forms;

public class DisplayDevice : IObserver
{
    private string deviceName;
    private ListBox outputBox;

    public DisplayDevice(string name, ListBox listBox)
    {
        deviceName = name;
        outputBox = listBox;
    }

    public void Update(float temperature)
    {
        outputBox.Items.Add($"{deviceName}    menerima    update:    Suhu    sekarang
{temperature}°C");
    }
}

```

Kode di Form1.cs

```

using System;
using System.Windows.Forms;

namespace ObserverWeatherApp
{
    public partial class Form1 : Form
    {
        private WeatherStation weatherStation;
        private DisplayDevice display1;
        private DisplayDevice display2;

        public Form1()
        {
            InitializeComponent();

            weatherStation = new WeatherStation();
            display1 = new DisplayDevice("Display 1", listBoxOutput);

```

```
        display2 = new DisplayDevice("Display 2", listBoxOutput);

        weatherStation.RegisterObserver(display1);
        weatherStation.RegisterObserver(display2);
    }

    private void btnUpdate_Click(object sender, EventArgs e)
    {
        if (float.TryParse(txtSuhu.Text, out float suhu))
        {
            weatherStation.SetTemperature(suhu);
        }
        else
        {
            MessageBox.Show("Masukkan suhu yang valid!", "Error",
                MessageBoxButtons.OK, MessageBoxIcon.Error);
        }
    }
}
```