

JURNAL MODUL 14

Nama : Farhan Kurniawan

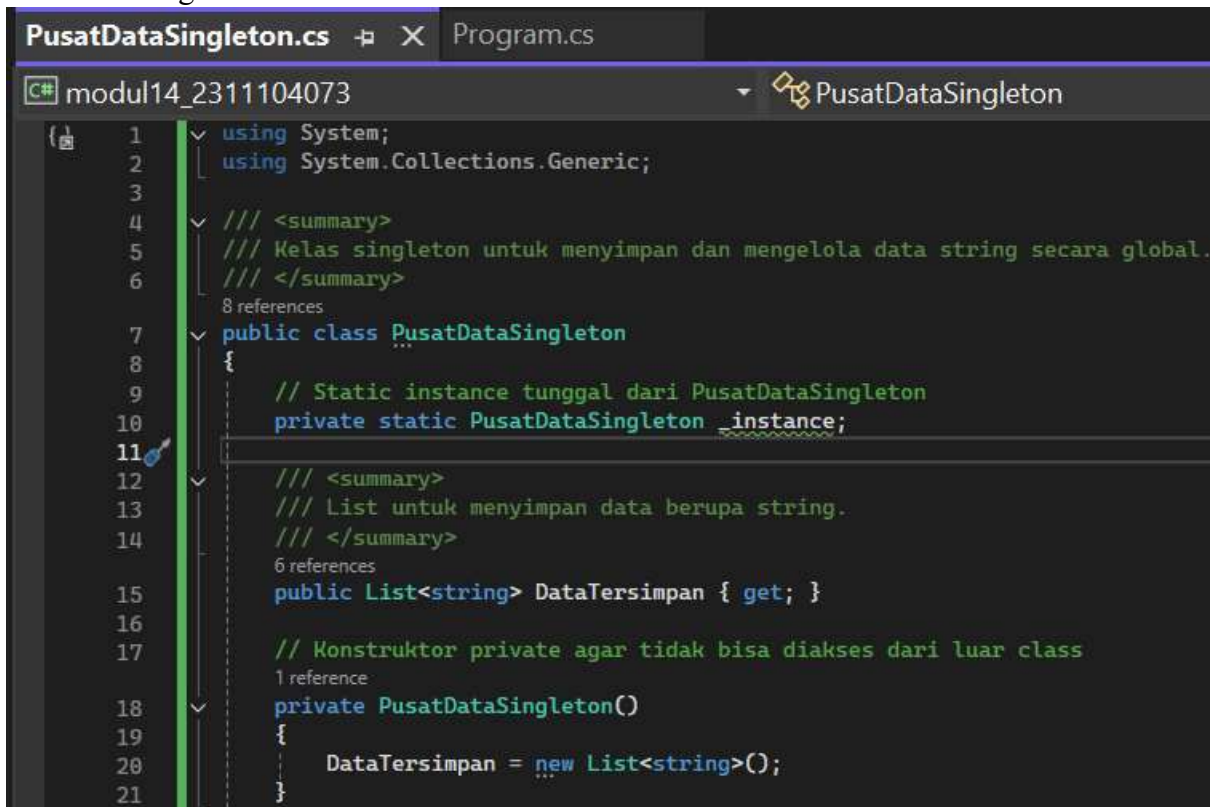
NIM : 2311104073

Kelas : SE-07-02

Link Github : https://github.com/farkurr/KPL_FARHANKURNIAWAN_2311104073_SE-07-02.git

Diambil dari : Jurnal Modul 13

1. PusatDataSingleton.cs

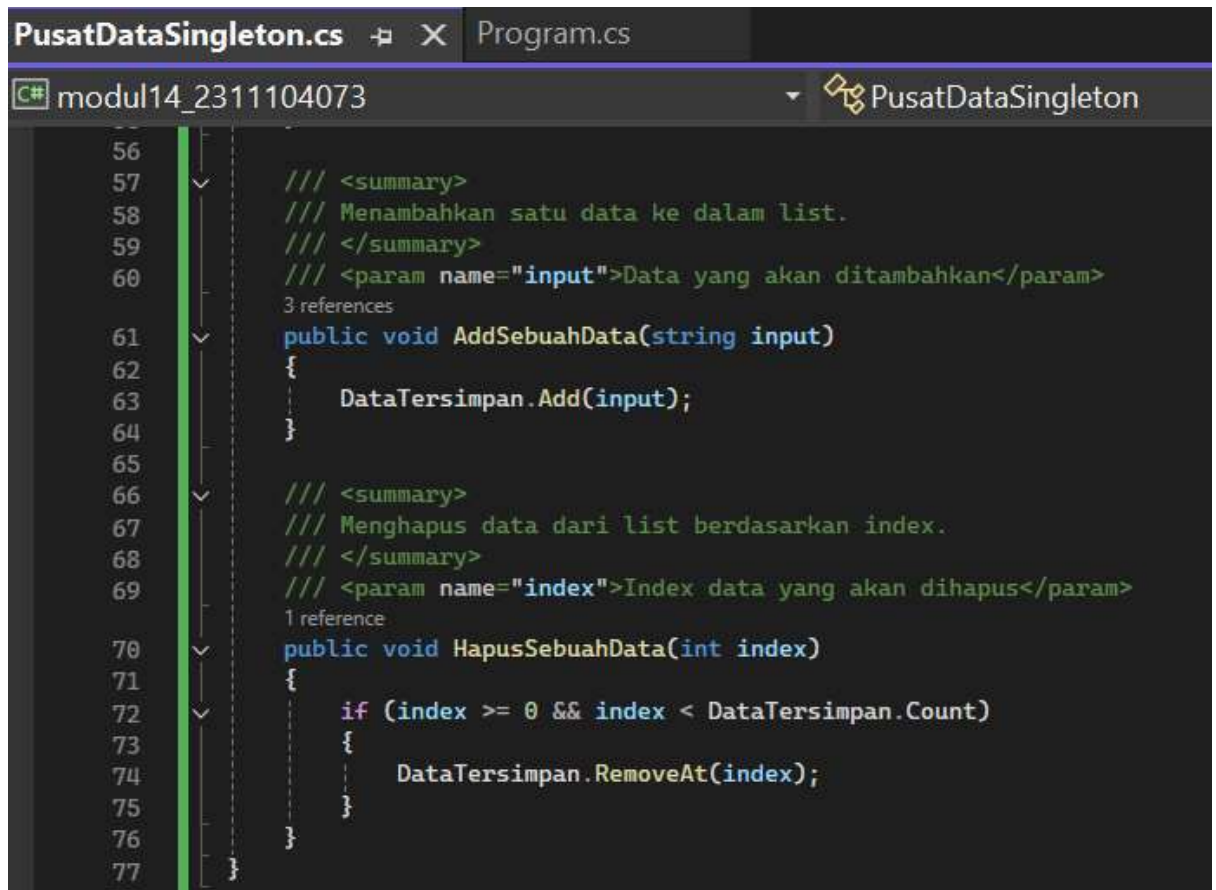


```
PusatDataSingleton.cs Program.cs
modul14_2311104073 PusatDataSingleton

1 using System;
2 using System.Collections.Generic;
3
4 /// <summary>
5 /// Kelas singleton untuk menyimpan dan mengelola data string secara global.
6 /// </summary>
7 public class PusatDataSingleton
8 {
9     // Static instance tunggal dari PusatDataSingleton
10    private static PusatDataSingleton _instance;
11
12    /// <summary>
13    /// List untuk menyimpan data berupa string.
14    /// </summary>
15    public List<string> DataTersimpan { get; }
16
17    // Konstruktor private agar tidak bisa diakses dari luar class
18    private PusatDataSingleton()
19    {
20        DataTersimpan = new List<string>();
21    }
22 }
```

```
PusatDataSingleton.cs X Program.cs
[C#] modul14_2311104073 PusatDataSingleton
22
23 // <summary>
24 // Mengembalikan instance tunggal dari PusatDataSingleton.
25 // </summary>
26 // <returns>Instance dari PusatDataSingleton</returns>
27 // 2 references
28 public static PusatDataSingleton GetDataSingleton()
29 {
30     if (_instance == null)
31     {
32         _instance = new PusatDataSingleton();
33     }
34     return _instance;
35 }
```

```
PusatDataSingleton.cs X Program.cs
[C#] modul14_2311104073 Pusat
36
37 // <summary>
38 // Mengembalikan seluruh data dalam list.
39 // </summary>
40 // <returns>List string</returns>
41 // 2 references
42 public List<string> GetSemuaData()
43 {
44     return DataTersimpan;
45 }
46
47 // <summary>
48 // Menampilkan seluruh data satu per satu.
49 // </summary>
50 // 2 references
51 public void PrintSemuaData()
52 {
53     foreach (var data in DataTersimpan)
54     {
55         Console.WriteLine(data);
56     }
57 }
```



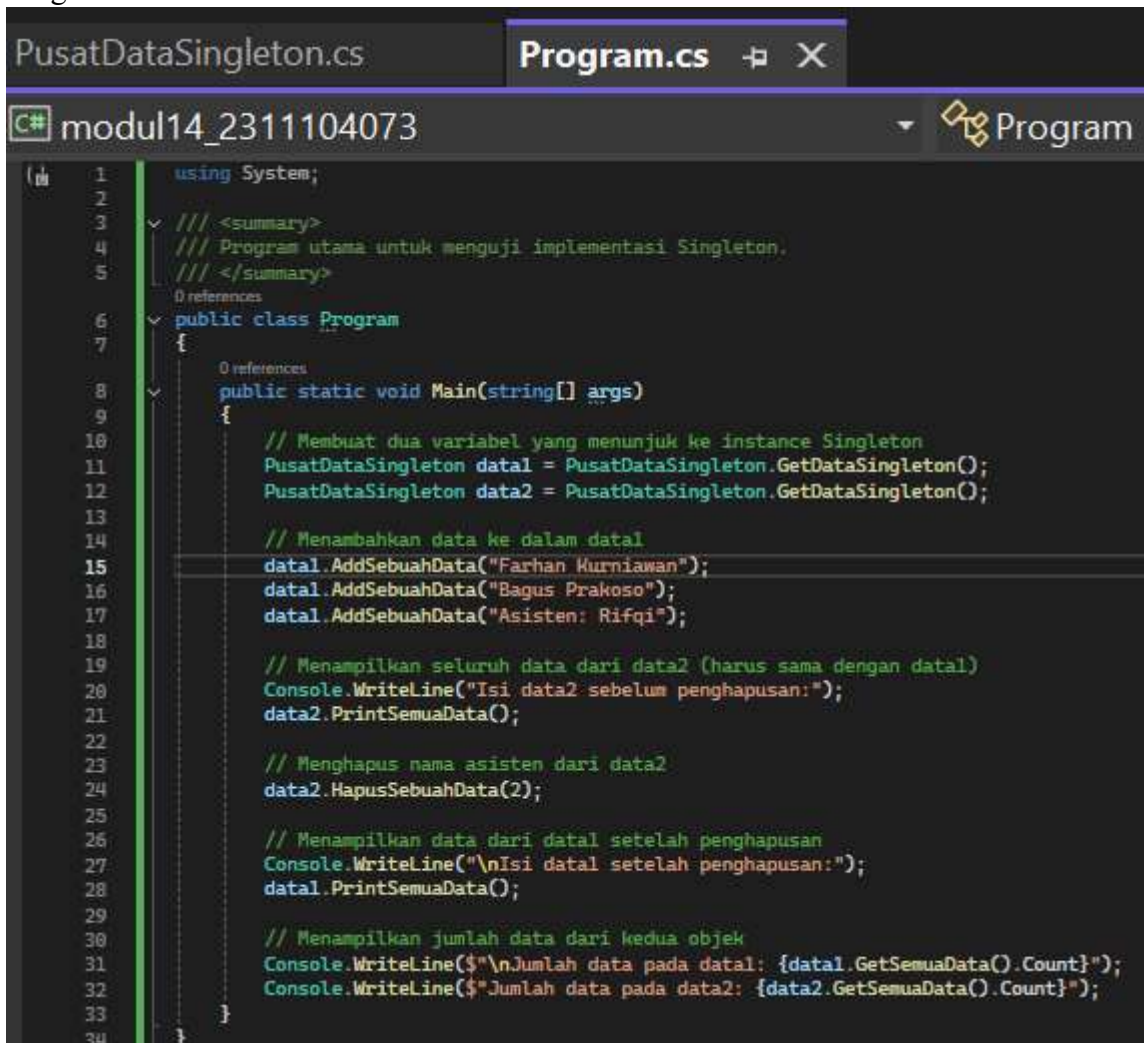
```
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78

/// <summary>
/// Menambahkan satu data ke dalam list.
/// </summary>
/// <param name="input">Data yang akan ditambahkan</param>
3 references
public void AddSebuahData(string input)
{
    DataTersimpan.Add(input);
}

/// <summary>
/// Menghapus data dari list berdasarkan index.
/// </summary>
/// <param name="index">Index data yang akan dihapus</param>
1 reference
public void HapusSebuahData(int index)
{
    if (index >= 0 && index < DataTersimpan.Count)
    {
        DataTersimpan.RemoveAt(index);
    }
}
```

File `PusatDataSingleton.cs` berisi class `PusatDataSingleton` yang telah difaktorkan mengikuti standar penulisan kode .NET. Penamaan class, property, method, dan atribut menggunakan konvensi PascalCase. Konstruktor class dibuat private untuk menerapkan prinsip Singleton, sementara instance-nya disimpan dalam field static `_instance`. Properti `DataTersimpan` digunakan untuk menyimpan list string, dan method-method seperti `AddSebuahData`, `HapusSebuahData`, `PrintSemuaData`, dan `GetSemuaData` disediakan untuk memanipulasi data tersebut. Komentar XML ditambahkan pada setiap bagian penting untuk mendokumentasikan fungsionalitas secara jelas dan mendukung praktik clean code. Whitespace dan indentasi juga telah diperbaiki agar lebih rapi dan mudah dibaca.

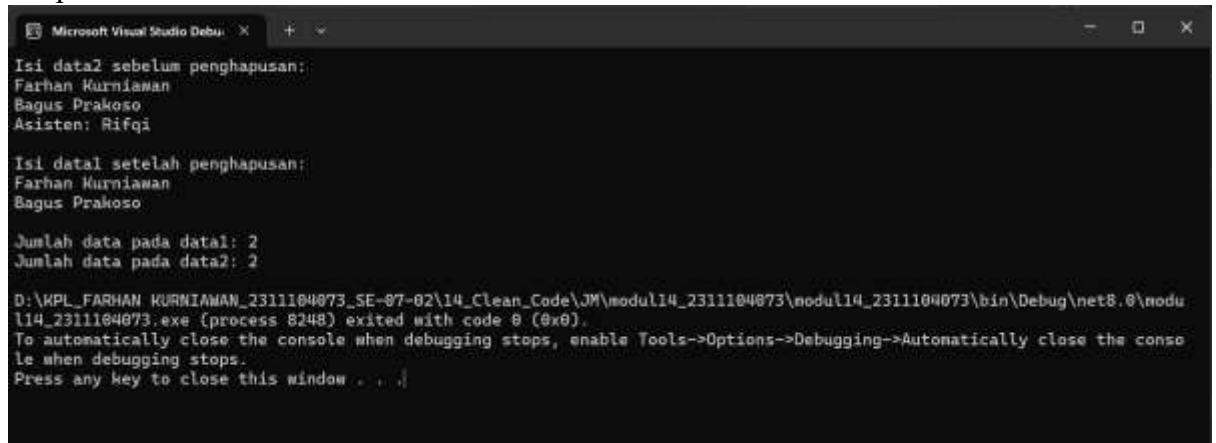
2. Program.cs



```
1  using System;
2
3  /// <summary>
4  /// Program utama untuk menguji implementasi Singleton.
5  /// </summary>
6  public class Program
7  {
8      public static void Main(string[] args)
9      {
10         // Membuat dua variabel yang menunjuk ke instance Singleton
11         PusatDataSingleton data1 = PusatDataSingleton.GetDataSingleton();
12         PusatDataSingleton data2 = PusatDataSingleton.GetDataSingleton();
13
14         // Menambahkan data ke dalam data1
15         data1.AddSebuahData("Farhan Kurniawan");
16         data1.AddSebuahData("Bagus Prakoso");
17         data1.AddSebuahData("Asisten: Rifqi");
18
19         // Menampilkan seluruh data dari data2 (harus sama dengan data1)
20         Console.WriteLine("Isi data2 sebelum penghapusan:");
21         data2.PrintSemuaData();
22
23         // Menghapus nama asisten dari data2
24         data2.HapusSebuahData(2);
25
26         // Menampilkan data dari data1 setelah penghapusan
27         Console.WriteLine("\nIsi data1 setelah penghapusan:");
28         data1.PrintSemuaData();
29
30         // Menampilkan jumlah data dari kedua objek
31         Console.WriteLine($"Jumlah data pada data1: {data1.GetSemuaData().Count}");
32         Console.WriteLine($"Jumlah data pada data2: {data2.GetSemuaData().Count}");
33     }
34 }
```

File Program.cs berfungsi sebagai titik eksekusi utama dan telah direfaktorkan untuk mengikuti standar struktur dan penamaan dalam bahasa C#. Di dalam method Main(), dibuat dua variabel data1 dan data2 yang berisi instance dari class Singleton. Program kemudian menambahkan data ke list melalui data1, dan mencetak serta memodifikasi data tersebut melalui data2, menunjukkan bahwa keduanya merujuk pada instance yang sama. Penamaan variabel menggunakan camelCase sesuai konvensi, dan komentar XML ditambahkan di awal class untuk menjelaskan fungsinya. Setiap blok kode diberi pemisahan whitespace untuk meningkatkan keterbacaan, serta struktur pemanggilan method dan penggunaan interpolasi string diformat agar lebih jelas dan konsisten.

3. Output:

A screenshot of the Microsoft Visual Studio Debug Console window. The window has a title bar with 'Microsoft Visual Studio Debug' and standard window controls. The output text is as follows:

```
Isi data2 sebelum penghapusan:  
Farhan Kurniawan  
Bagus Prakoso  
Asisten: Rifqi  
  
Isi data1 setelah penghapusan:  
Farhan Kurniawan  
Bagus Prakoso  
  
Jumlah data pada data1: 2  
Jumlah data pada data2: 2  
  
D:\KPL_FARHAN_KURNIAWAN_2311104073_SE-07-02\14_Clean_Code\JM\modul14_2311104073\modul14_2311104073\bin\Debug\net8.0\modu  
l14_2311104073.exe (process 8248) exited with code 0 (0x0).  
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso  
le when debugging stops.  
Press any key to close this window . . .
```