

## JURNAL MODUL 13

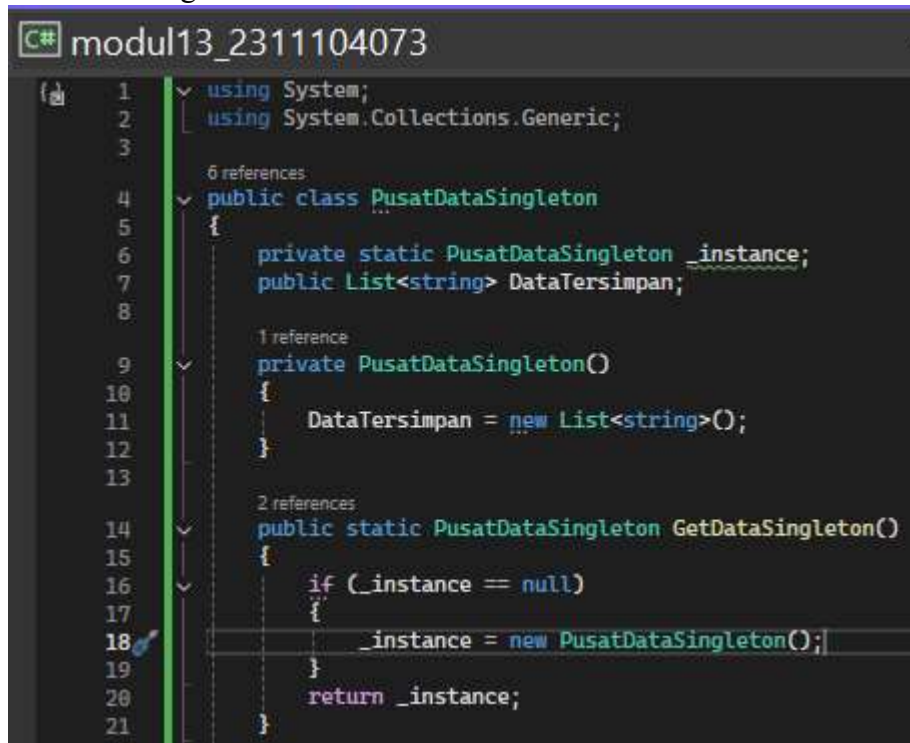
Nama : Farhan Kurniawan

NIM : 2311104073

Kelas : SE-07-02

Link Github : [https://github.com/farkurr/KPL\\_FARHANKURNIAWAN\\_2311104073\\_SE-07-02.git](https://github.com/farkurr/KPL_FARHANKURNIAWAN_2311104073_SE-07-02.git)

### 1. PusatDataSingleton.cs

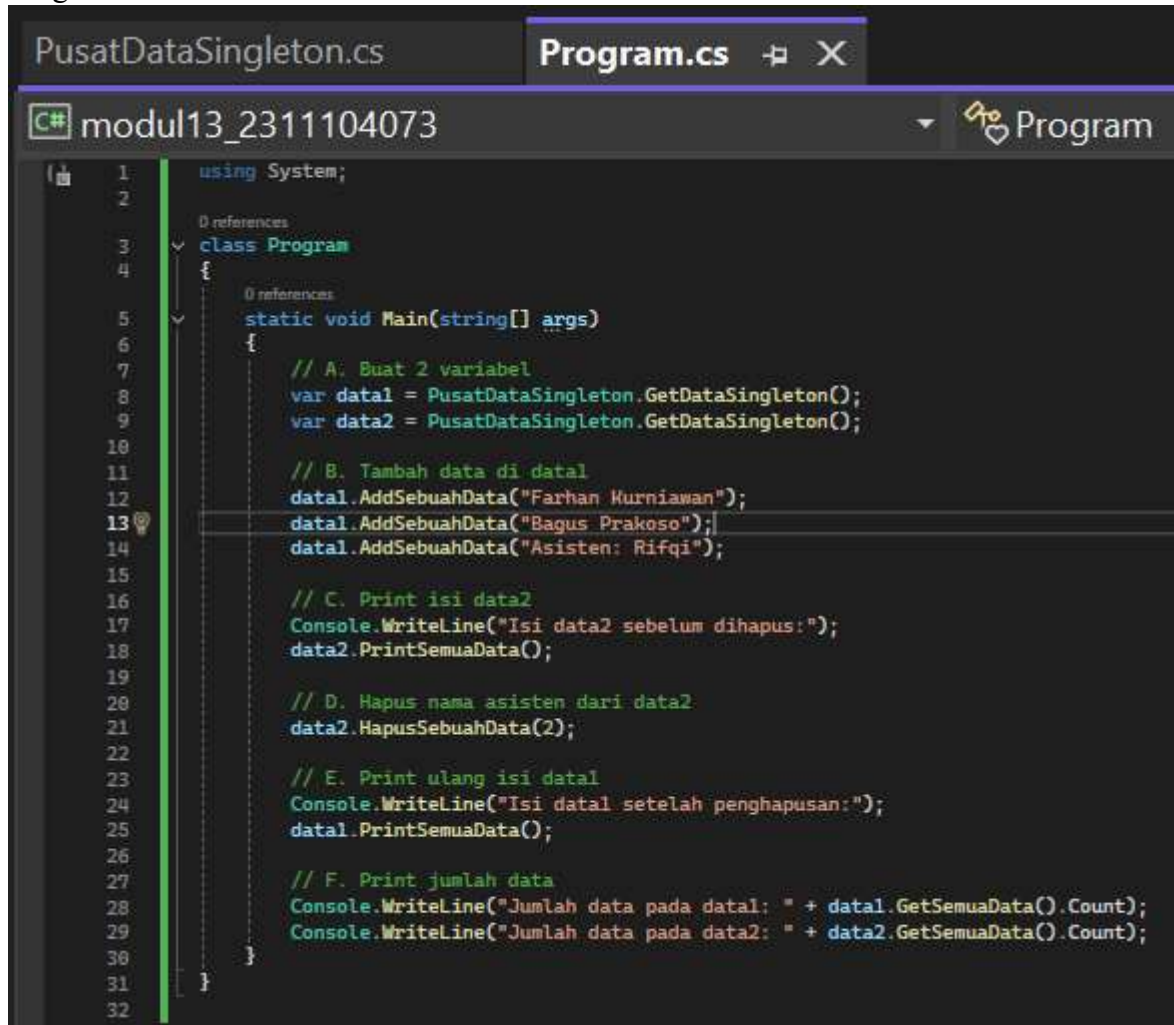


```
modul13_2311104073
1  using System;
2  using System.Collections.Generic;
3
4  6 references
5  public class PusatDataSingleton
6  {
7      private static PusatDataSingleton _instance;
8      public List<string> DataTersimpan;
9
10     1 reference
11     private PusatDataSingleton()
12     {
13         DataTersimpan = new List<string>();
14     }
15
16     2 references
17     public static PusatDataSingleton GetDataSingleton()
18     {
19         if (_instance == null)
20         {
21             _instance = new PusatDataSingleton();
22         }
23         return _instance;
24     }
25 }
```

```
C# modul13_2311104073
22
23 2 references
24 public List<string> GetSemuaData()
25 {
26     return DataTersimpan;
27 }
28
29 2 references
30 public void PrintSemuaData()
31 {
32     foreach (var data in DataTersimpan)
33     {
34         Console.WriteLine(data);
35     }
36 }
37
38 3 references
39 public void AddSebuahData(string input)
40 {
41     DataTersimpan.Add(input);
42 }
43
44 1 reference
45 public void HapusSebuahData(int index)
46 {
47     if (index >= 0 && index < DataTersimpan.Count)
48     {
49         DataTersimpan.RemoveAt(index);
50     }
51 }
```

File ini berisi implementasi class PusatDataSingleton yang menggunakan **design pattern Singleton**. Tujuannya adalah memastikan bahwa hanya satu objek PusatDataSingleton yang dibuat selama runtime aplikasi. Class ini memiliki atribut DataTersimpan bertipe List<string> yang digunakan untuk menyimpan data berupa string. Atribut \_instance bertipe PusatDataSingleton bersifat static dan digunakan sebagai penampung instance tunggal dari class ini. Konstruktor dibuat private agar tidak bisa diakses dari luar class, sesuai prinsip Singleton. Method GetDataSingleton() digunakan untuk mengembalikan instance yang sama setiap kali dipanggil. Selain itu, terdapat method AddSebuahData(), HapusSebuahData(), GetSemuaData(), dan PrintSemuaData() untuk memanipulasi dan menampilkan data dalam list.

## 2. Program.cs



```
1  using System;
2
3  class Program
4  {
5      static void Main(string[] args)
6      {
7          // A. Buat 2 variabel
8          var data1 = PusatDataSingleton.GetDataSingleton();
9          var data2 = PusatDataSingleton.GetDataSingleton();
10
11         // B. Tambah data di data1
12         data1.AddSebuahData("Farhan Kurniawan");
13         data1.AddSebuahData("Bagus Prakoso");
14         data1.AddSebuahData("Asisten: Rifqi");
15
16         // C. Print isi data2
17         Console.WriteLine("Isi data2 sebelum dihapus:");
18         data2.PrintSemuaData();
19
20         // D. Hapus nama asisten dari data2
21         data2.HapusSebuahData(2);
22
23         // E. Print ulang isi data1
24         Console.WriteLine("Isi data1 setelah penghapusan:");
25         data1.PrintSemuaData();
26
27         // F. Print jumlah data
28         Console.WriteLine("Jumlah data pada data1: " + data1.GetSemuaData().Count);
29         Console.WriteLine("Jumlah data pada data2: " + data2.GetSemuaData().Count);
30     }
31 }
32
```

File Program.cs berisi method Main() sebagai titik awal eksekusi program. Di dalamnya dibuat dua variabel data1 dan data2 yang keduanya diisi melalui pemanggilan PusatDataSingleton.GetDataSingleton(). Karena menggunakan pattern Singleton, data1 dan data2 sebenarnya merujuk ke objek yang sama. Program kemudian menambahkan beberapa data string ke dalam list melalui data1, termasuk nama anggota kelompok dan nama asisten praktikum. Setelah itu, data dalam list ditampilkan melalui data2, dan salah satu data (nama asisten praktikum) dihapus menggunakan method HapusSebuahData(). Program kemudian mencetak ulang isi list menggunakan data1 untuk menunjukkan bahwa data telah terhapus, membuktikan bahwa data1 dan data2 mengakses list yang sama. Terakhir, jumlah elemen data ditampilkan dari kedua variabel sebagai konfirmasi lebih lanjut atas keberhasilan implementasi Singleton.

### 3. Output:

```
Microsoft Visual Studio Debu: X + -
Isi data2 sebelum dihapus:
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\13_Design_Pattern_Implementation 14\JM\modul13_2311104073\modul13_2311104073
\bin\Debug\net8.0\modul13_2311104073.exe (process 15616) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console
when debugging stops.
Press any key to close this window . . .
```