

JURNAL MODUL 14

Nama : Kafka Putra Riyadi

Kelas : SE 07-02

NIM : 2311104041

1. Source code class program.cs

```
1 using JurnalModul13_2311104041;
2
3 0 references
4 public class Program
5 {
6     0 references
7     public static void Main(string[] args)
8     {
9         var pusatData1 = PusatDataSingleton.GetInstance();
10        var pusatData2 = PusatDataSingleton.GetInstance();
11
12        pusatData1.TambahData("Kafka");
13        pusatData1.TambahData("Adit");
14        pusatData1.TambahData("Bu Nia");
15
16        Console.WriteLine("Data di pusatData2:");
17        pusatData2.TampilkanSemuaData();
18
19        pusatData2.HapusData(2);
20
21        Console.WriteLine("\nSetelah penghapusan, data di pusatData1:");
22        pusatData1.TampilkanSemuaData();
23
24        Console.WriteLine($"Jumlah data di pusatData1: {pusatData1.AmbilSemuaData().Count}");
25        Console.WriteLine($"Jumlah data di pusatData2: {pusatData2.AmbilSemuaData().Count}");
26    }
27 }
```

Penjelasan :

Class Program.cs di atas berfungsi sebagai titik masuk program utama yang mendemonstrasikan penggunaan pola desain Singleton melalui class PusatDataSingleton. Objek pusatData1 dan pusatData2 mengacu pada instance yang sama, sehingga setiap perubahan data yang dilakukan oleh salah satunya akan tercermin pada yang lain. Program ini menambahkan tiga data ke dalam koleksi, menampilkannya melalui pusatData2, lalu menghapus salah satu data, dan menampilkan hasil akhirnya melalui pusatData1. Di akhir program, jumlah data pada kedua objek ditampilkan untuk menunjukkan bahwa keduanya mengakses data yang sama secara global melalui instance tunggal.

2. Source code class PusatDataSingleton.cs

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

namespace JurnalModul13_2311104041
{
    6 references
    public class PusatDataSingleton
    {
        private static PusatDataSingleton _instance;

        6 references
        public List<string> DataTersimpan { get; private set; }

        1 reference
        private PusatDataSingleton()
        {
            DataTersimpan = new List<string>();
        }

        2 references
        public static PusatDataSingleton GetInstance()
        {
            if (_instance == null)
            {
                _instance = new PusatDataSingleton();
            }

            return _instance;
        }

        2 references
        public List<string> AmbilSemuaData()
        {
            return DataTersimpan;
        }

        2 references
        public void TampilkanSemuaData()
        {
            foreach (var data in DataTersimpan)
            {
                Console.WriteLine(data);
            }
        }

        3 references
        public void TambahData(string input)
        {
            DataTersimpan.Add(input);
        }

        1 reference
        public void HapusData(int index)
        {
            if (index >= 0 && index < DataTersimpan.Count)
            {
                DataTersimpan.RemoveAt(index);
            }
        }
    }
}
```

Penjelasan :

Class PusatDataSingleton menerapkan pola desain Singleton yang memastikan hanya ada satu instance dari class ini yang digunakan selama program berjalan. Instance tunggal tersebut disimpan dalam variabel statis `_instance`, dan hanya dapat diakses melalui method `GetInstance()`. Class ini menyimpan data dalam bentuk list string `DataTersimpan` yang dapat dimanipulasi melalui method `TambahData()` untuk menambahkan data, `HapusData()` untuk menghapus data berdasarkan indeks, serta `TampilkanSemuaData()` untuk mencetak seluruh data ke konsol. Selain itu, method `AmbilSemuaData()` memungkinkan akses langsung ke seluruh isi data. Dengan desain ini, semua bagian program yang memanggil `GetInstance()` akan mengakses dan memodifikasi data yang sama secara konsisten.

Outputannya :

```
Data di data2:  
Kafka  
Adit  
Bu Nia  
  
Setelah penghapusan, data di data1:  
Kafka  
Adit  
  
Jumlah data di data1: 2  
Jumlah data di data2: 2
```