



TUGAS JURNAL
KONSTRUKSI PERANGKAT LUNAK
MODUL 13

D
I
S
U
S
U
N

OLEH :
TIURMA GRACE ANGELINA – 2311104042
S1-SE07-02

Dosen :
YIS

Prodi S1 Rekaya Perangkat Lunak
Direktorat Kampus Purwokerto

1. MENJELASKAN DESIGN PATTERN SINGLETON

a. Dua contoh kondisi penggunaan Singleton

1. Logger Aplikasi

Dalam aplikasi besar, hanya diperlukan satu objek logger untuk mencatat aktivitas/error log ke file agar konsisten.

2. Koneksi Database

Biasanya hanya satu koneksi aktif yang dipakai untuk efisiensi resource dan sinkronisasi akses database.

b. Langkah-langkah implementasi Singleton

1. Buat class dengan konstruktor private agar tidak bisa diinstansiasi dari luar.

2. Buat atribut static private bertipe class itu sendiri untuk menyimpan instance-nya.

3. Buat method static (misalnya `GetInstance()`) yang mengembalikan instance. Kalau belum ada, buat dulu.

4. Tambahkan mekanisme thread-safety jika dibutuhkan.

c. Tiga kelebihan dan kekurangan Singleton

Kelebihan:

1. Menghemat memori karena hanya ada satu instance.

2. Mudah diakses dari mana saja lewat `GetInstance()`.

3. Memastikan pengelolaan resource yang konsisten (misalnya file, DB, cache).

Kekurangan:

1. Sulit untuk diuji unit test (karena global state).

2. Bisa disalahgunakan seperti global variable.

3. Sulit dikembangkan dalam sistem multithread jika tidak hati-hati (butuh thread-safety).

2. IMPLEMENTASI DAN PEMAHAMAN DESIGN PATTERN SINGLETON

Implementasi Singleton - Class `PusatDataSingleton`

Struktur class:

- Property:
 - `List<string> DataTersimpan`
 - `static PusatDataSingleton _instance`
- Method:
 - Konstruktor: menginisialisasi `DataTersimpan = new List<string>()`
 - `GetDataSingleton()`: return `_instance` (buat jika null)
 - `GetSemuaData()`: return list
 - `PrintSemuaData()`: foreach print isi `DataTersimpan`

- AddSebuahData(string input)
- HapusSebuahData(int index)

Implementasi di Main()

1. Buat dua variabel data1, data2 dari PusatDataSingleton.GetDataSingleton()
2. Tambah data anggota kelompok dan asisten ke data1
3. Cetak isi data2 → harus sama dengan data1
4. Hapus asisten dari data2
5. Cetak ulang dari data1 → nama asisten harus hilang juga (karena Singleton)
6. Tampilkan .Count dari GetSemuaData() untuk data1 dan data2

Code :

1. PusatDataSingleton.cs

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

namespace tj13_2311104042
{
    public class PusatDataSingleton
    {
        private static PusatDataSingleton _instance;

        public List<string> DataTersimpan { get; set; }

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

        public static PusatDataSingleton GetDataSingleton()
        {
            if (_instance == null)
            {
                _instance = new PusatDataSingleton();
            }
            return _instance;
        }
    }
}
```

```
public List<string> GetSemuaData()
{
    return DataTersimpan;
}

public void PrintSemuaData()
{
    Console.WriteLine("Isi Data:");
    foreach (var data in DataTersimpan)
    {
        Console.WriteLine("- " + data);
    }
}

public void AddSebuahData(string input)
{
    DataTersimpan.Add(input);
}

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

2. Program.cs

```
using System;

namespace tj13_2311104042
{
    class Program
    {
        static void Main(string[] args)
        {
            var data1 = PusatDataSingleton.GetDataSingleton();
            var data2 = PusatDataSingleton.GetDataSingleton();

            data1.AddSebuahData("Tiurma Grace Angelina");
            data1.AddSebuahData("caca");
            data1.AddSebuahData("cici");
            data1.AddSebuahData("Asisten: cece");

            Console.WriteLine("Data2 sebelum penghapusan:");
            data2.PrintSemuaData();

            data2.HapusSebuahData(3);

            Console.WriteLine("\nData1 setelah penghapusan dari data2:");
            data1.PrintSemuaData();

            Console.WriteLine($"Jumlah data (data1): {data1.GetSemuaData().Count}");
            Console.WriteLine($"Jumlah data (data2): {data2.GetSemuaData().Count}");

            Console.WriteLine("\nTekan tombol apa saja untuk keluar...");
            Console.ReadKey();
        }
    }
}
```

Output :

```
Microsoft Visual Studio Debug Console
Data2 sebelum penghapusan:
Isi Data:
- Tiurma Grace Angelina
- caca
- cici
- Asisten: cece

Data1 setelah penghapusan dari data2:
Isi Data:
- Tiurma Grace Angelina
- caca
- cici

Jumlah data (data1): 3
Jumlah data (data2): 3

Tekan tombol apa saja untuk keluar...

C:\Users\USER\source\repos\KPL_TiurmaGraceA_2311104042_S1SE-07-02\13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\bin\Debug\net8.0\tjmod13_2311104042.exe (process 26088) 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 . . .
```

Penjelasan Singkat Kode Implementasi

Class PusatDataSingleton

- Merupakan implementasi dari design pattern **Singleton**.
- Atribut private static `_instance` memastikan hanya ada satu instance yang digunakan selama runtime.
- Konstruktor dibuat private agar tidak bisa diakses dari luar.
- Method `GetDataSingleton()` akan membuat objek saat pertama kali dipanggil, dan akan mengembalikan instance yang sama untuk semua pemanggilan selanjutnya.
- `DataTersimpan` menyimpan data dalam bentuk `List<string>`.
- Method:
 - `AddSebuahData(string input)`: menambahkan data ke list.
 - `HapusSebuahData(int index)`: menghapus data dari list berdasarkan index.
 - `PrintSemuaData()`: mencetak semua data ke console.
 - `GetSemuaData()`: mengembalikan list data.

Method Main() (dalam Program.cs)

- Membuat dua variabel data1 dan data2 dari Singleton (GetDataSingleton()), yang sebenarnya mengacu ke instance yang sama.
- Menambahkan beberapa data ke data1, lalu mencetak isi list lewat data2 (menunjukkan bahwa keduanya berbagi data).
- Data dihapus lewat data2, lalu data1 dicetak kembali → hasilnya berubah, menunjukkan bahwa Singleton bekerja.
- Terakhir, jumlah data di kedua variabel ditampilkan untuk memastikan mereka memang mengakses instance yang sama.

Commit

```
C:\WINDOWS\system32\cmd.exe
*****
** Visual Studio 2022 Developer Command Prompt v17.13.0
** Copyright (c) 2022 Microsoft Corporation
*****

C:\Users\USER\source\repos\KPL_TiurmaGraceA_2311104042_S1SE-07-02\13_Design_Pattern_Implementation\tjmod13_2311104042>cd ..

C:\Users\USER\source\repos\KPL_TiurmaGraceA_2311104042_S1SE-07-02\13_Design_Pattern_Implementation>git add tjmod13_2311104042

C:\Users\USER\source\repos\KPL_TiurmaGraceA_2311104042_S1SE-07-02\13_Design_Pattern_Implementation>
C:\Users\USER\source\repos\KPL_TiurmaGraceA_2311104042_S1SE-07-02\13_Design_Pattern_Implementation>
C:\Users\USER\source\repos\KPL_TiurmaGraceA_2311104042_S1SE-07-02\13_Design_Pattern_Implementation>git commit -m "IMPLEMENTASI DAN PEMAHAMAN DESIGN PATTERN SINGLETON"
[main 25334f1] IMPLEMENTASI DAN PEMAHAMAN DESIGN PATTERN SINGLETON
40 files changed, 518 insertions(+)
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042/.vs/ProjectEvaluation/tjmod13_2311104042.metadata.v9.bin
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042/.vs/ProjectEvaluation/tjmod13_2311104042.projects.v9.bin
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042/.vs/ProjectEvaluation/tjmod13_2311104042.strings.v9.bin
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042/.vs/tjmod13_2311104042/DesignTimeBuild/.dtbcache.v2
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042/.vs/tjmod13_2311104042/FileContentIndex/254493bb-adcd-4f5a-a5a5-ed0f6ea0c2ce.vsidx
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042/.vs/tjmod13_2311104042/FileContentIndex/b5f361a8-6ee6-4839-b671-289bcc0735bc.vsidx
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042/.vs/tjmod13_2311104042/v17/.futcachecache.v2
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042/.vs/tjmod13_2311104042/v17/.suo
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042/.vs/tjmod13_2311104042/v17/DocumentLayout.backup.json
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042/.vs/tjmod13_2311104042/v17/DocumentLayout.json
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042.sln
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\Program.cs
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\PusatDataSingleton.cs
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\bin\Debug\net8.0\tjmod13_2311104042.deps.json
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\bin\Debug\net8.0\tjmod13_2311104042.dll
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\bin\Debug\net8.0\tjmod13_2311104042.exe
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\bin\Debug\net8.0\tjmod13_2311104042.runtimeconfig.json
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\obj\Debug\net8.0\NETCoreApp,Version=v8.0.AssemblyAttributes.cs
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\obj\Debug\net8.0\apphost.exe
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\obj\Debug\net8.0\ref\tjmod13_2311104042.dll
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\obj\Debug\net8.0\refint\tjmod13_2311104042.dll
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\obj\Debug\net8.0\tjmod13_2311104042.AssemblyInfo.cs
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\obj\Debug\net8.0\tjmod13_2311104042.AssemblyInfoInputs.cache
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\obj\Debug\net8.0\tjmod13_2311104042.GeneratedMSBuildEditorConfig.editorconfig
create mode 100644 13_Design_Pattern_Implementation\tjmod13_2311104042\tjmod13_2311104042\obj\Debug\net8.0\tjmod13_2311104042.GlobalUsings.g.cs
```