

TUGAS JURNAL MODUL 13
KONSTRUKSI PERANGKAT LUNAK
DESIGN PATTERN IMPLEMENTATION



DISUSUN OLEH:
KHOLIFAH DINA
2211104004

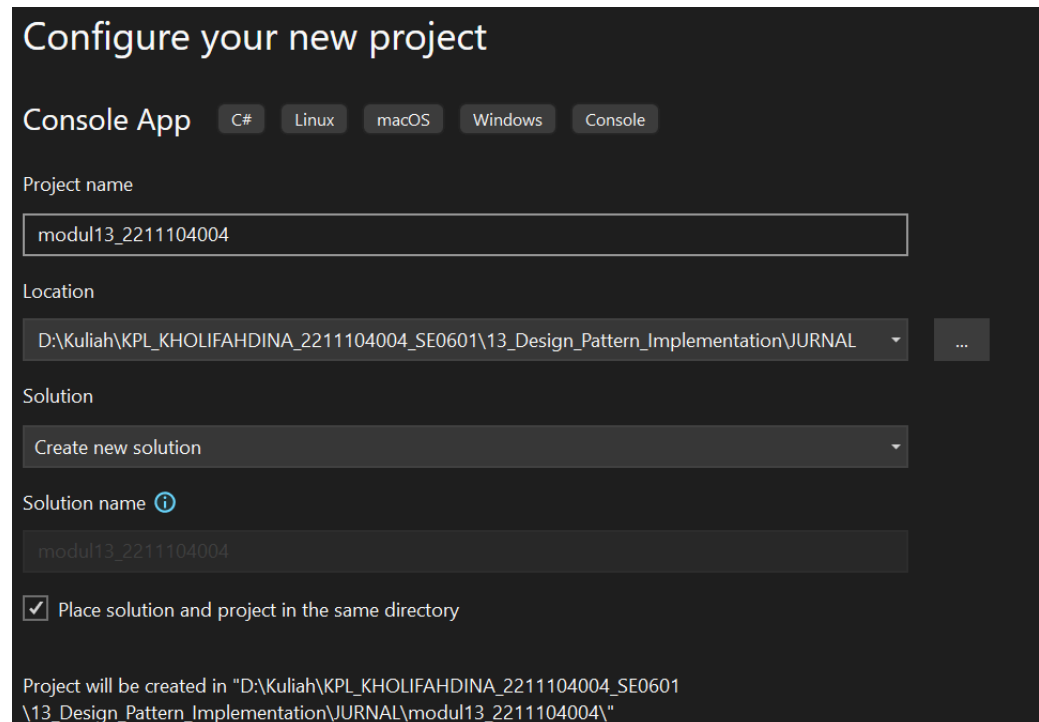
SE 06 01

S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY
2025

A. MEMBUAT PROJECT WEB API

Buka IDE misalnya dengan Visual Studio

- a. Misalnya menggunakan Visual Studio, buatlah project baru dengan nama modul13_NIM.



B. MENJELASKAN SALAH SATU DESIGN PATTERN

Buka halaman web <https://refactoring.guru/design-patterns/catalog> kemudian baca design pattern dengan nama “Observer”, dan jawab pertanyaan berikut ini (dalam Bahasa Indonesia):

- a. Berikan salah dua contoh kondisi dimana design pattern “Singleton” dapat digunakan.
 1. **Logger System**
Dalam aplikasi besar, biasanya hanya ada satu objek Logger untuk mencatat log ke file, database, atau konsol agar konsistensi dan efisiensi terjaga.
 2. **Koneksi Database**
Untuk menjaga efisiensi dan mencegah konflik, aplikasi hanya boleh memiliki satu instance koneksi database aktif (misalnya `DatabaseConnection.Instance`).
- b. Berikan penjelasan singkat mengenai langkah-langkah dalam mengimplementasikan design pattern “Singleton”.
 1. Buat konstruktor private, agar tidak bisa diinstansiasi dari luar.
 2. Buat static field untuk menyimpan satu-satunya instance dari kelas tersebut.

3. Buat method publik static (biasanya bernama `GetInstance()`) untuk mengakses instance-nya. Jika instance-nya null, buat instance baru, jika tidak, kembalikan yang lama.

c. Berikan kelebihan dan kekurangan dari design pattern “Singleton”.

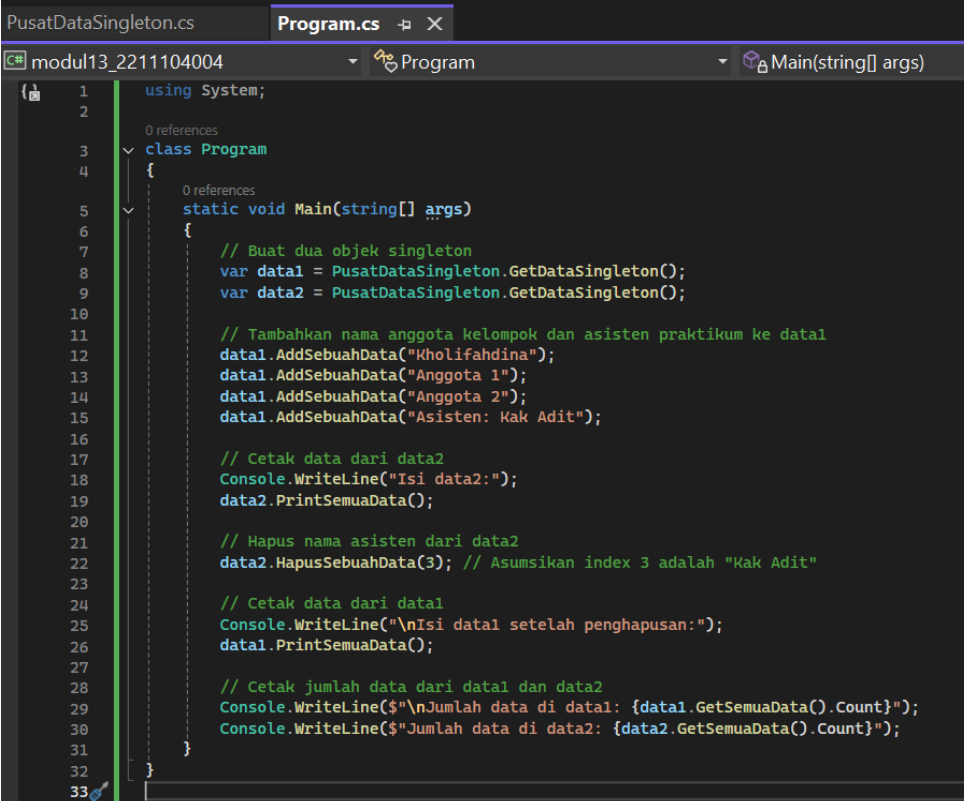
Kelebihan	Kekurangan
Kontrol global atas objek tertentu	Sulit di-test karena ketergantungan global
Hanya satu instance yang digunakan (hemat resource)	Bisa menyulitkan ketika ingin me-reset atau mengubah state
Mudah diakses dari mana saja	Bisa disalahgunakan dan membuat code menjadi tightly-coupled

C. IMPLEMENTASI DAN PEMAHAMAN DESIGN PATTERN SINGLETON

Buka halaman web berikut <https://refactoring.guru/design-patterns/observer> dan scroll ke bagian “Code Examples”, pilih kode yang akan dilihat misalnya C# dan ikuti langkah-langkah berikut:

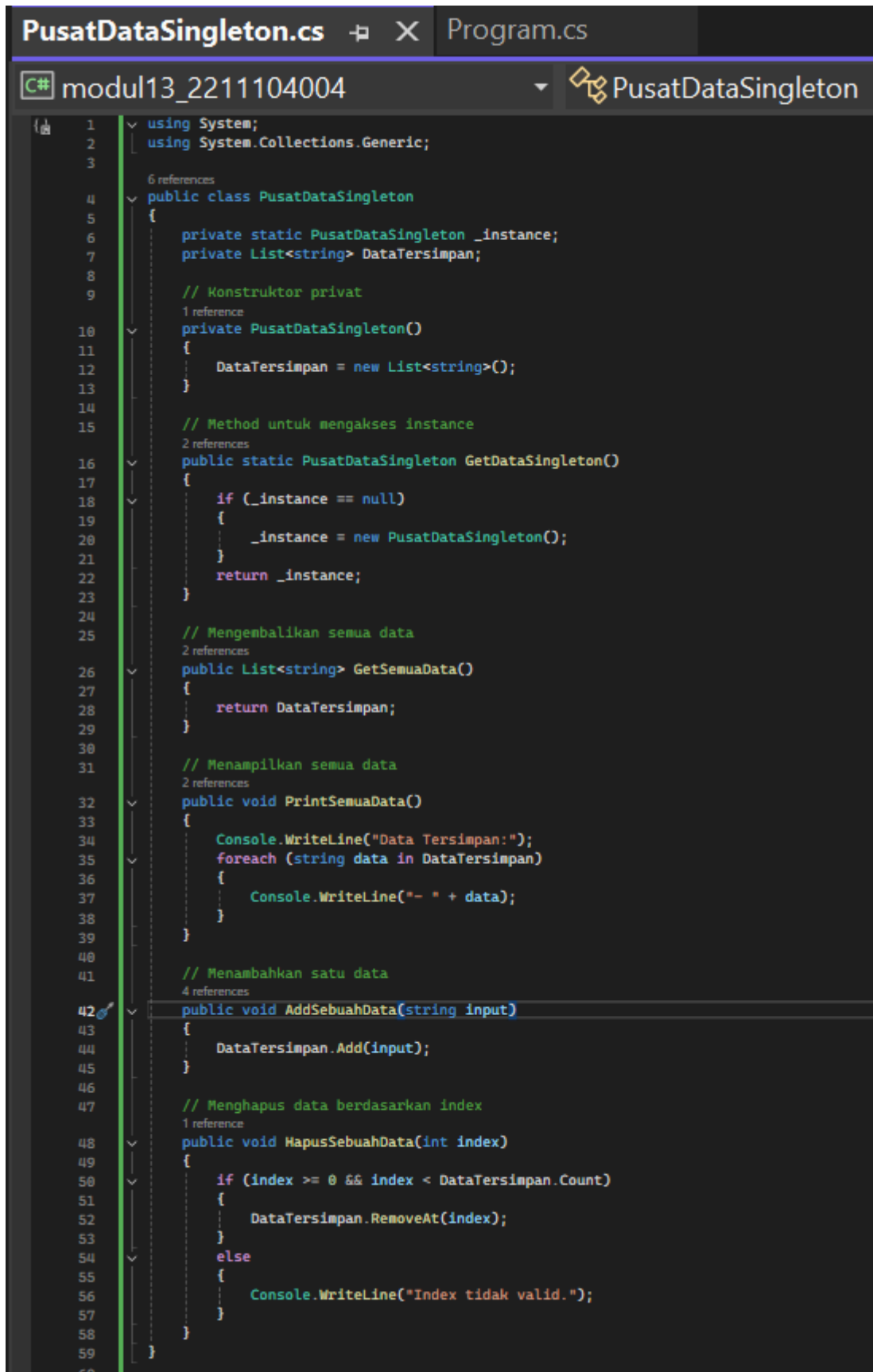
- a. Dengan contoh yang sudah diberikan, buatlah sebuah class dengan design pattern singleton dengan nama “PusatDataSingleton”.
- b. Class “PusatDataSingleton” mempunyai dua atribut yaitu “DataTersimpan” yang mempunyai tipe berupa List<string> dan property singleton dengan nama “_instance” dengan tipe data “PusatDataSingleton” itu sendiri.
- c. Class tersebut juga memiliki beberapa method yaitu:
 - i. Konstruktor dari kelas tersebut yang mengisi atribut “DataTersimpan” dengan list kosong.
 - ii. GetDataSingleton() yang mengembalikan “_instance” jika tidak null dan memanggil konstruktor terlebih dahulu apabila nilainya masih null.
 - iii. GetSemuaData() yang mengembalikan list dari property “DataTersimpan”
 - iv. PrintSemuaData() yang melakukan print satu per satu dari string yang ada di list “DataTersimpan”.
 - v. AddSebuahData(string input) yang menambahkan satu data baru “input” ke dalam list “DataTersimpan”.
 - vi. HapusSebuahData(int index) yang menghapus sebuah data berdasarkan index tertentu.

File program.cs



```
1 using System;
2
3 class Program
4 {
5     static void Main(string[] args)
6     {
7         // Buat dua objek singleton
8         var data1 = PusatDataSingleton.GetDataSingleton();
9         var data2 = PusatDataSingleton.GetDataSingleton();
10
11         // Tambahkan nama anggota kelompok dan asisten praktikum ke data1
12         data1.AddSebuahData("Kholifahdina");
13         data1.AddSebuahData("Anggota 1");
14         data1.AddSebuahData("Anggota 2");
15         data1.AddSebuahData("Asisten: Kak Adit");
16
17         // Cetak data dari data2
18         Console.WriteLine("Isi data2:");
19         data2.PrintSemuaData();
20
21         // Hapus nama asisten dari data2
22         data2.HapusSebuahData(3); // Asumsikan index 3 adalah "Kak Adit"
23
24         // Cetak data dari data1
25         Console.WriteLine("\nIsi data1 setelah penghapusan:");
26         data1.PrintSemuaData();
27
28         // Cetak jumlah data dari data1 dan data2
29         Console.WriteLine($"Jumlah data di data1: {data1.GetSemuaData().Count}");
30         Console.WriteLine($"Jumlah data di data2: {data2.GetSemuaData().Count}");
31     }
32 }
33
```

File PusatDataSingleton.cs



```
1 using System;
2 using System.Collections.Generic;
3
4 public class PusatDataSingleton
5 {
6     private static PusatDataSingleton _instance;
7     private List<string> DataTersimpan;
8
9     // Konstruktor privat
10    private PusatDataSingleton()
11    {
12        DataTersimpan = new List<string>();
13    }
14
15    // Method untuk mengakses instance
16    public static PusatDataSingleton GetDataSingleton()
17    {
18        if (_instance == null)
19        {
20            _instance = new PusatDataSingleton();
21        }
22        return _instance;
23    }
24
25    // Mengembalikan semua data
26    public List<string> GetSemuaData()
27    {
28        return DataTersimpan;
29    }
30
31    // Menampilkan semua data
32    public void PrintSemuaData()
33    {
34        Console.WriteLine("Data Tersimpan:");
35        foreach (string data in DataTersimpan)
36        {
37            Console.WriteLine("- * + data);
38        }
39    }
40
41    // Menambahkan satu data
42    public void AddSebuahData(string input)
43    {
44        DataTersimpan.Add(input);
45    }
46
47    // Menghapus data berdasarkan index
48    public void HapusSebuahData(int index)
49    {
50        if (index >= 0 && index < DataTersimpan.Count)
51        {
52            DataTersimpan.RemoveAt(index);
53        }
54        else
55        {
56            Console.WriteLine("Index tidak valid.");
57        }
58    }
59 }
```

D. IMPLEMENTASI PROGRAM UTAMA

Tambahkan beberapa implementasi di program/method utama atau “main”:

- Buatlah dua variable dengan tipe “PusatDataSingleton” bernama data1 dan data2.
- Isi kedua variable tersebut dengan hasil keluaran dari GetDataSingleton().
- Pada data1 lakukan pemanggilan method AddSebuahData() beberapa kali dengan input nama anggota kelompok dan asisten praktikum.
- Pada data2 panggil method PrintSemuaData(), pastikan keluaran dari hasil print data2 menampilkan nama-nama anggota kelompok dan asisten praktikum.
- Pada data2 panggil HapusSebuahData() untuk menghapus nama asisten praktikum anda sekarang.
- Pada data1 panggil PrintSemuaData(), dan seharusnya nama asisten praktikum anda tidak muncul di hasil print tersebut.
- Langkah terakhir, pada data1 dan data2 panggil GetSemuaData() dan lakukan print dari jumlah “Count” atau elemen yang ada di list pada data1 dan data2.

```
C:\> Microsoft Visual Studio Debug Console

Isi data2:
Data Tersimpan:
- Kholifahdina
- Anggota 1
- Anggota 2
- Asisten: Kak Adit

Isi data1 setelah penghapusan:
Data Tersimpan:
- Kholifahdina
- Anggota 1
- Anggota 2

Jumlah data di data1: 3
Jumlah data di data2: 3

D:\Kuliah\KPL_KHOLIFAHDINA_2211104004_
```