

**TUGAS JURNAL MODUL 14  
KONSTRUKSI PERANGKAT LUNAK**

**SECURE CODING PRACTICE**



**DISUSUN OLEH:**

**Alfian Mutakim**

**2211104024**

**SE 06 01**

**S1 REKAYASA PERANGKAT LUNAK**

**FAKULTAS INFORMATIKA**

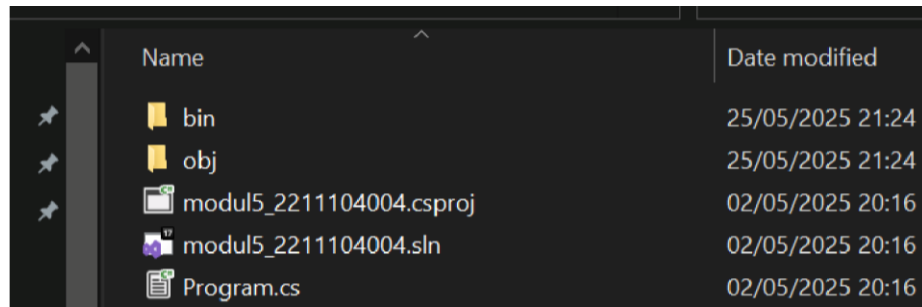
**TELKOM UNIVERSITY**

**2025**

**A. MEMBUAT PROJECT**

Buka IDE misalnya dengan Visual Studio

- a. Copy folder tugas modul 5, kemudian rename folder hasil copy-paste tersebut dengan modul14\_NIM. Misalnya menggunakan Visual Studio, bukanlah project/folder yang di-copy sebelumnya.



Name	Date modified
bin	25/05/2025 21:24
obj	25/05/2025 21:24
modul5_2211104004.csproj	02/05/2025 20:16
modul5_2211104004.sln	02/05/2025 20:16
Program.cs	02/05/2025 20:16

## B. REFACTORING DENGAN STANDAR CODE

Dengan mengikuti standard code yang digunakan (misal C# dengan standar dari .NET), pastikan kode yang dikumpulkan memenuhi faktor-faktor berikut:

1. Naming convention
  - i. Variable / Property / Attribute
    - angka1, angka2, angka3 → diubah menjadi number1, number2, number3 (mengikuti *camelCase*).
    - StoredData dan InputDates dipertahankan karena sudah sesuai konvensi properti (*PascalCase*).
    - penjumlahan → diubah menjadi calculator (mengikuti *camelCase* untuk variabel lokal/objek).
  - ii. Method / Function / Procedure
    - JumlahTigaAngka → diubah menjadi SumThreeNumbers (mengikuti *PascalCase* dan penggunaan nama deskriptif dalam Bahasa Inggris).
2. White space dan indentation: Penambahan spasi dan baris kosong
  - Menambahkan baris kosong antar definisi method dan class untuk meningkatkan keterbacaan.
  - Menyesuaikan indentasi menjadi konsisten 4 spasi (sesuai standar .NET).
3. Variable / attribute declarations: Deklarasi eksplisit dan jelas
  - Variabel seperti number1, number2, number3 dideklarasikan secara eksplisit dengan tipe double.
  - Penggunaan var hanya pada objek yang tipe-nya jelas dan mudah dikenali (calculator, database).
4. Comments: Komentar yang lebih deskriptif dan konsisten
  - a. Menambahkan komentar untuk menjelaskan maksud setiap class dan method.
  - b. Menghindari komentar yang tidak perlu atau membingungkan (seperti komentar yang bercampur konflik Git sebelumnya).

File Before Refactoring:

```

1  <<<<<< HEAD
2  using System;
3
4  class Penjumlahan
5  {
6      // Method generic untuk menjumlahkan tiga angka
7      public T JumlahTigaAngka<T>(T angka1, T angka2, T angka3)
8      {
9          dynamic a = angka1;
10         dynamic b = angka2;
11         dynamic c = angka3;
12         return a + b + c;
13     }
14 }
15
16 =====
17 //using System;
18
19 //class Penjumlahan
20 //{
21 //    // Method generic untuk menjumlahkan tiga angka
22 //    public T JumlahTigaAngka<T>(T angka1, T angka2, T angka3)
23 //    {
24 //        dynamic a = angka1;
25 //        dynamic b = angka2;
26 //        dynamic c = angka3;
27 //        return a + b + c;
28 //    }
29 //}
30
31 //class Program
32 //{
33 //    static void Main()
34 //    {
35 //        Penjumlahan penjumlahan = new Penjumlahan();
36
37 //        // Menggunakan tipe data double karena NIM berakhiran 4
38 //        double angka1 = 12.0, angka2 = 34.0, angka3 = 56.0;
39 //        double hasil = penjumlahan.JumlahTigaAngka(angka1, angka2, angka3);
40
41 //        Console.WriteLine($"Hasil penjumlahan (double): {hasil}");
42 //        Console.ReadLine();
43 //    }
44 //}
45
46
47
48 using System;
49 using System.Collections.Generic;
50
51 class SimpleDataBase<T>
52 {
53     // Property untuk menyimpan data generic
54     public List<T> StoredData { get; private set; }
55
56     // Property untuk menyimpan waktu input data
57     public List<DateTime> InputDates { get; private set; }
58

```

```

58
59 // Konstruktor untuk inisialisasi list kosong
60 public SimpleDataBase()
61 {
62     StoredData = new List<T>();
63     InputDates = new List<DateTime>();
64 }
65
66 // Method untuk menambahkan data baru
67 public void AddNewData(T data)
68 {
69     StoredData.Add(data);
70     InputDates.Add(DateTime.UtcNow); // Menyimpan waktu dalam format UTC
71 }
72
73 // Method untuk menampilkan semua data yang tersimpan
74 public void PrintAllData()
75 {
76     for (int i = 0; i < StoredData.Count; i++)
77     {
78         Console.WriteLine($"Data {i + 1} berisi: {StoredData[i]}, yang disimpan pada waktu UTC: {InputDates[i]}");
79     }
80 }
81
82
83 // Program Utama
84 >>>>>> implementasi-generic-class
85 class Program
86 {
87     static void Main()
88     {
89         <<<<<< HEAD
90         Penjumlahan penjumlahan = new Penjumlahan();
91
92         // Menggunakan tipe data double karena NIM berakhiran 4
93         double angka1 = 12.0, angka2 = 34.0, angka3 = 56.0;
94         double hasil = penjumlahan.JumlahTigaAngka(angka1, angka2, angka3);
95
96         Console.WriteLine($"Hasil penjumlahan (double): {hasil}");
97
98         // Mencegah console langsung tertutup
99         Console.WriteLine("\nTekan ENTER untuk keluar...");
100
101         =====
102         // NIM berakhiran 4 → gunakan tipe data double
103         SimpleDataBase<double> database = new SimpleDataBase<double>();
104
105         // Menambahkan tiga angka (misalnya 12, 34, 56 dari NIM)
106         database.AddNewData(12.0);
107         database.AddNewData(34.0);
108         database.AddNewData(56.0);
109
110         // Menampilkan semua data
111         Console.WriteLine("Data yang tersimpan:");
112         database.PrintAllData();
113     }
114     >>>>>> implementasi-generic-class
115     Console.ReadLine();
116 }
117

```

File After Refactoring:

```

1  using System;
2  using System.Collections.Generic;
3
4  // Kelas untuk operasi penjumlahan dengan method generic
5  class Calculator
6  {
7      // Method generic untuk menjumlahkan tiga angka
8      public T SumThreeNumbers<T>(T number1, T number2, T number3)
9      {
10         dynamic a = number1;
11         dynamic b = number2;
12         dynamic c = number3;
13         return a + b + c;
14     }
15 }
16
17 // Kelas untuk menyimpan data generic dengan waktu input
18 class SimpleDatabase<T>
19 {
20     // Menyimpan data generic
21     public List<T> StoredData { get; private set; }
22
23     // Menyimpan tanggal/waktu saat data ditambahkan
24     public List<DateTime> InputDates { get; private set; }
25
26     // Konstruktor untuk inisialisasi list
27     public SimpleDatabase()
28     {
29         StoredData = new List<T>();
30         InputDates = new List<DateTime>();
31     }
32
33     // Method untuk menambahkan data baru beserta waktu
34     public void AddNewData(T data)
35     {
36         StoredData.Add(data);
37         InputDates.Add(DateTime.UtcNow);
38     }
39
40     // Method untuk mencetak seluruh data beserta waktu input
41     public void PrintAllData()
42     {
43         for (int i = 0; i < StoredData.Count; i++)
44         {
45             Console.WriteLine($"Data {i + 1} berisi: {StoredData[i]}, yang disimpan pada waktu UTC: {InputDates[i]}");
46         }
47     }
48 }
49
50 // Program utama
51 class Program
52 {
53     static void Main()
54     {
55         // Inisialisasi kalkulator
56         var calculator = new Calculator();
57
58         // Contoh penjumlahan tiga angka (tipe data double karena NIM berakhiran 4)
59         double number1 = 12.0;
60         double number2 = 34.0;
61         double number3 = 56.0;
62
63         double result = calculator.SumThreeNumbers(number1, number2, number3);
64         Console.WriteLine($"Hasil penjumlahan (double): {result}");
65
66         // Inisialisasi database dan menambahkan data
67         var database = new SimpleDatabase<double>();
68         database.AddNewData(number1);
69         database.AddNewData(number2);
70         database.AddNewData(number3);
71
72         // Menampilkan seluruh data
73         Console.WriteLine("\nData yang tersimpan:");
74         database.PrintAllData();
75
76         // Pause aplikasi sebelum keluar
77         Console.WriteLine("\nTekan ENTER untuk keluar...");
78         Console.ReadLine();
79     }
80 }

```