

TUGAS JURNAL
KONSTRUKSI PERANGKAT LUNAK
MODUL 14



Disusun oleh :
Althafia Defiyandrea Laskanadya Wibowo

S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO

1. MEMBUAT PROJECT MODUL

Modul 5

2. REFACTORING DENGAN STANDAR CODE

a. Tujuan

Menyesuaikan struktur, penamaan, dan gaya penulisan kode agar mengikuti standar penulisan yang rapi dan profesional sesuai .NET C# Coding Convention.

b. Naming Convention

Variable / Property / Attribute

- Sebelumnya (kurang tepat):

```
int angka1;  
List<T> storedData;
```

- Setelah refactor (sesuai .NET):

```
int firstNumber;  
private List<T> _storedData;
```

- Penjelasan:
 - Gunakan camelCase untuk parameter dan variabel lokal.
 - Gunakan PascalCase untuk public property.
 - Gunakan prefix _ untuk private field (misalnya _storedData).

Method / Function / Procedure

- Sebelumnya:

```
public T JumlahTigaAngka<T>(T a, T b, T c) { ... }
```

- Setelah refactor:

```
public T SumThreeNumbers<T>(T a, T b, T c) { ... }
```

- Penjelasan:
 - Gunakan PascalCase untuk nama method (awali huruf kapital).
 - Gunakan nama method deskriptif dan dalam Bahasa Inggris.

c. White Space dan Indentation

- Sebelumnya (rapat, susah dibaca):

```
public void AddNewData(T data){  
    storedData.Add(data);  
    inputDates.Add(DateTime.UtcNow);}
```

- Setelah refactor:

```
public void AddNewData(T data)
{
    _storedData.Add(data);
    _inputDates.Add(DateTime.UtcNow);
}
```

- Penjelasan:
 - Gunakan 4 spasi untuk indentasi.
 - Tambahkan jarak antar blok kode untuk meningkatkan keterbacaan.
 - Blok {} selalu diletakkan di baris baru.

d. Variable / Attribute Declarations

- Sebelumnya:
- dynamic tempA = a;

dynamic tempB = b;

- Setelah refactor:

dynamic x = a, y = b, z = c;

- Penjelasan:
 - Gunakan deklarasi ringkas dan efisien.
 - Gunakan var jika tipe dapat diketahui, tapi hindari berlebihan.

e. Comments

- Sebelumnya: Kode tanpa penjelasan sama sekali.
- Setelah refactor (ditambahkan komentar):

```
// Melakukan penjumlahan terhadap tiga angka bertipe generic
public T SumThreeNumbers<T>(T a, T b, T c)

// Menyimpan data beserta waktu inputnya

public void AddNewData(T data)
```

- Penjelasan:
 - Tambahkan komentar pendek sebelum method atau logika penting.
 - Gunakan komentar untuk menjelaskan tujuan dari bagian kode, bukan isi teknis per baris.

3. Hasil Refactoring Code

```
1  using System;
2  using System.Collections.Generic;
3
4  namespace TpModul14_2211104011
5  {
6      public class Calculator
7      {
8          public T SumThreeNumbers<T>(T a, T b, T c)
9          {
10             dynamic x = a, y = b, z = c;
11             return (T)(x + y + z);
12          }
13      }
14
15      public class SimpleDatabase<T>
16      {
17          private readonly List<T> _storedData;
18          private readonly List<DateTime> _inputDates;
19
20          public SimpleDatabase()
21          {
22              _storedData = new List<T>();
23              _inputDates = new List<DateTime>();
24          }
25
26          public void AddNewData(T data)
27          {
28              _storedData.Add(data);
29              _inputDates.Add(DateTime.UtcNow);
30          }
31
32          public void PrintAllData()
33          {
34              for (int i = 0; i < _storedData.Count; i++)
35              {
36                  Console.WriteLine($"Data {i + 1}: {_storedData[i]}, saved at UTC: {_inputDates[i]}");
37              }
38          }
39      }
40
41      public class Program
42      {
43          public static void Main()
44          {
45              // Implementasi Generic Method
46              var calculator = new Calculator();
47              int a = 22, b = 40, c = 27;
48              int result = calculator.SumThreeNumbers(a, b, c);
49              Console.WriteLine($"Sum Result: {result}");
50
51              // Implementasi Generic Class
52              var database = new SimpleDatabase<int>();
53              database.AddNewData(a);
54              database.AddNewData(b);
55              database.AddNewData(c);
56              database.PrintAllData();
57          }
58      }
59  }
```