

MODUL 14 KONTRUKSI PERANGKAT LUNAK

Nama: Dimas Abhi

NIM : 2311104069

Kelas : SE-07-02

Link Github:

Sebelum direfactoring

```
1 using System;
2 using System.Collections.Generic;
3
4 // Interface untuk observer yang menerima update berita
5 public interface IObserver
6 {
7     void Update(string message);
8 }
9
10 // Interface untuk subject yang mengelola observer
11 public interface ISubject
12 {
13     void Attach(IObserver observer);
14     void Detach(IObserver observer);
15     void Notify();
16 }
17
18 // Implementasi subject: NewsAgency
19 public class NewsAgency : ISubject
20 {
21     // Daftar observer yang terdaftar
22     private readonly List<IObserver> observers = new List<IObserver>();
23
24     // Berita terbaru yang akan dikirim ke observer
25     private string latestNews;
26
27     public void Attach(IObserver observer)
28     {
29         if (observer == null) throw new ArgumentNullException(nameof(observer));
30         if (!observers.Contains(observer))
31             observers.Add(observer);
32     }
33
34     public void Detach(IObserver observer)
35     {
36         if (observer == null) throw new ArgumentNullException(nameof(observer));
37         observers.Remove(observer);
38     }
39
40     public void Notify()
41     {
42         // Jika tidak ada berita, tidak perlu notify
43         if (string.IsNullOrEmpty(latestNews)) return;
44
45         foreach (var observer in observers)
46         {
47             observer.Update(latestNews);
48         }
49     }
50 }
```

```

49     }
50
51     // Menambahkan berita baru dan notify observer
52     2 references
53     public void AddNews(string news)
54     {
55         if (string.IsNullOrEmpty(news)) return;
56
57         latestNews = news;
58         Notify();
59     }
60
61     // Implementasi observer: Subscriber
62     4 references
63     public class Subscriber : IObserver
64     {
65         private readonly string name;
66
67         3 references
68         public Subscriber(string name)
69         {
70             if (string.IsNullOrEmpty(name)) throw new ArgumentException("Name cannot be empty", nameof(name));
71             this.name = name;
72         }
73
74         2 references
75         public void Update(string message)
76         {
77             Console.WriteLine($"{name} menerima berita terbaru: {message}");
78         }
79     }
80
81     // Program utama untuk menjalankan demo
82     0 references
83     class Program
84     {
85         0 references
86         static void Main()
87         {
88             var agency = new NewsAgency();
89
90             var sub1 = new Subscriber("Andi");
91             var sub2 = new Subscriber("Budi");
92             var sub3 = new Subscriber("Cici");
93
94             agency.Attach(sub1);
95             agency.Attach(sub2);
96             agency.Attach(sub3);
97
98             agency.AddNews("Berita Hari Ini: Cuaca Cerah!");
99             agency.Detach(sub2);
100            agency.AddNews("Update: Harga BBM Naik!");
101        }
102    }

```

Sesudah direfactoring

```
1  using System;
2  using System.Collections.Generic;
3
4  // Interface untuk observer yang menerima update berita
5
6  public interface IObserver
7  {
8      void Update(string message);
9  }
10
11 // Interface untuk subject yang mengelola observer
12
13 public interface ISubject
14 {
15     void Attach(IObserver observer);
16     void Detach(IObserver observer);
17     void Notify();
18 }
19
20 // Implementasi subject: NewsAgency
21
22 public class NewsAgency : ISubject
23 {
24     // Menyimpan daftar observer yang terdaftar
25     private readonly List<IObserver> _observers = new();
26
27     // Menyimpan berita terbaru yang akan dikirimkan ke observer
28     private string _latestNews;
29
30     4 references
31     public void Attach(IObserver observer)
32     {
33         if (observer == null)
34             throw new ArgumentNullException(nameof(observer));
35
36         if (!_observers.Contains(observer))
37             _observers.Add(observer);
38     }
39
40     public void Detach(IObserver observer)
41     {
42         if (observer == null)
43             throw new ArgumentNullException(nameof(observer));
44
45         _observers.Remove(observer);
46     }
47
48     2 references
49     public void Notify()
50     {
51         // Jika tidak ada berita, tidak perlu memberi notifikasi
52         if (string.IsNullOrEmpty(_latestNews)) return;
53
54         foreach (var observer in _observers)
55         {
56             observer.Update(_latestNews);
57         }
58     }
59 }
```

```
52     }
53 }
54
55 // Menambahkan berita baru dan memberi notifikasi ke semua observer
56 2 references
57 public void AddNews(string news)
58 {
59     if (string.IsNullOrEmpty(news)) return;
60     _latestNews = news;
61     Notify();
62 }
63
64 // Implementasi observer: Subscriber
65 4 references
66 public class Subscriber : IObserver
67 {
68     private readonly string _name;
69
70     1 reference
71     public Subscriber(string name)
72     {
73         if (string.IsNullOrEmpty(name))
74             throw new ArgumentException("Name cannot be empty", nameof(name));
75         _name = name;
76     }
77
78     2 references
79     public void Update(string message)
80     {
81         Console.WriteLine($"{_name} menerima berita terbaru: {message}");
82     }
83 }
84
85 // Program utama untuk menjalankan demo
86 0 references
87 public class Program
88 {
89     0 references
90     public static void Main()
91     {
92         var agency = new NewsAgency();
93         var andi = new Subscriber("Andi");
94         var budi = new Subscriber("Budi");
95         var cici = new Subscriber("Cici");
96
97         agency.Attach(andi);
98         agency.Attach(budi);
99         agency.Attach(cici);
100
101         agency.AddNews("Berita Hari Ini: Cuaca Cerah!");
102
103         agency.Detach(budi);
104
105         agency.AddNews("Update: Harga BBM Naik!");
106     }
107 }
```

1. Naming Convention

Pastikan semua nama menggunakan PascalCase untuk class, method, dan properti, serta camelCase untuk variabel lokal dan parameter.

Sudah benar:

NewsAgency, AddNews, latestNews

Attach(IObserver observer)

Update(string message)

Tidak perlu ubah apapun, naming convention sudah baik.

2. Whitespace dan Indentation

Pastikan:

Tidak ada spasi berlebih.

Setiap blok kode { } ter-indentasi dengan baik.

Contoh sudah baik:

csharp

CopyEdit

foreach (var observer in observers)

{

observer.Update(latestNews);

}

3. Variable/Attribute Declarations

Sudah menggunakan pendekatan deklarasi yang jelas:

csharp

CopyEdit

private readonly List<IObserver> observers = new List<IObserver>();

private string latestNews;

Jika ingin lebih ekspresif:

csharp

CopyEdit

private readonly List<IObserver> _observers = new();

private string _latestNews;

(menggunakan prefix _ adalah praktik umum di banyak tim .NET modern)

4. Comments

Tambahkan komentar untuk menjelaskan fungsi dari class, method, atau blok penting.

Contoh yang bisa ditambahkan:

```
csharp
```

```
CopyEdit
```

```
// Menyimpan daftar subscriber (observer) yang akan menerima notifikasi berita
```

```
private readonly List<IObserver> observers = new List<IObserver>();
```

```
// Method untuk menambahkan berita baru dan memberitahu semua subscriber
```

```
public void AddNews(string news)
```

```
{
```

```
    if (string.IsNullOrEmpty(news)) return;
```

```
    latestNews = news;
```

```
    Notify();
```

```
}
```

Output

```
● apple@Abiww-2 jmodul14_2311104069 % dotnet run
/Users/apple/Documents/KPL_DIMASABHIPRAYARAMANSYAH_SE0702/14_Clean_Code/modul14_2311104069/jmodul14_2311104069.csproj(1,1): warning CS8618: Non-nullable field 'latestNews' must contain a non-null value when exiting constructor. Consider declaring it as nullable.
Andi menerima berita terbaru: Berita Hari Ini: Cuaca Cerah!
Budi menerima berita terbaru: Berita Hari Ini: Cuaca Cerah!
Cici menerima berita terbaru: Berita Hari Ini: Cuaca Cerah!
Andi menerima berita terbaru: Update: Harga BBM Naik!
Cici menerima berita terbaru: Update: Harga BBM Naik!
○ apple@Abiww-2 jmodul14_2311104069 %
```