

Nama:Faishal Arif Setiawan

Nim:2311104066

Kelas:SE-07-02

JURNAL MODUL 14

I.Link Github

https://github.com/faishalstwn/KPL_FaishalArifSetiawan_2311104066_SE_07_02

II.Code

Kode dari Jm2 setelah di refactor sesuai standar kode

Kodepos.cs

```
anModul14_2311104066
using System;
using System.Collections.Generic;

namespace JurnalModul14_2311104066
{
    1 reference
    internal class KodePos
    {
        private static readonly Dictionary<string, string> _kodePosMap = new Dictionary<string, string>
        {
            {"Batununggal", "40266"},
            {"Kujangsari", "40287"},
            {"Mengger", "40267"},
            {"Mates", "40256"},
            {"Cijaur", "40287"},
            {"Jatisari", "40286"},
            {"Margasari", "40286"},
            {"Sekarjati", "40286"},
            {"Kebonwaru", "40272"},
            {"Maleer", "40274"},
            {"Samoja", "40273"}
        };

        // Menampilkan semua data kode pos ke konsol
        1 reference
        public static void TampilkanSemua()
        {
            Console.WriteLine("==== DAFTAR KODE POS =====");
            foreach (var entry in _kodePosMap)
            {
                Console.WriteLine($"{entry.Key}: {entry.Value}");
            }
        }
    }
}
```

Kelas KodePos berfungsi sebagai representasi data wilayah dan kode pos yang tersimpan dalam struktur Dictionary, di mana setiap nama wilayah dipetakan ke kode pos-nya. Kelas ini menyediakan method statis TampilkanSemua() yang digunakan untuk menampilkan seluruh isi daftar kode pos ke konsol. Karena method-nya bersifat statis, maka bisa dipanggil langsung tanpa membuat objek dari kelas tersebut. Kelas ini berperan sebagai penyaji data statis dalam program, tanpa adanya logika pemrosesan yang kompleks.

Internal class doormachine

```

2 references
internal class DoorMachine
{
    private bool _isLocked = true;
    private bool _isOpen = false;

    1 reference
    public void CurrentState()
    {
        Console.WriteLine($"Status: {(_isLocked ? "Terkunci" : "Terbuka/Terkunci: Tidak")}, Pintu: {(_isOpen ? "Terbuka" : "Tertutup")}");
    }

    2 references
    public void ToggleLock()
    {
        if (_isOpen)
        {
            Console.WriteLine("Pintu terbuka, tidak bisa mengunci!");
        }
        else
        {
            _isLocked = !_isLocked;
            Console.WriteLine(_isLocked ? "Pintu dikunci." : "Pintu dibuka kuncinya.");
        }
    }

    1 reference
    public void OpenDoor()
    {
        if (_isLocked)
        {
            Console.WriteLine("Tidak bisa membuka pintu, masih terkunci!");
        }
        else
        {
            _isOpen = true;
            Console.WriteLine("Pintu berhasil dibuka.");
        }
    }

    1 reference
    public void CloseDoor()
    {
        if (_isOpen)
        {
            _isOpen = false;
            Console.WriteLine("Pintu ditutup kembali.");
        }
        else
        {
            Console.WriteLine("Pintu sudah dalam keadaan tertutup.");
        }
    }
}

```

Kelas DoorMachine adalah representasi sederhana dari sebuah sistem pintu otomatis yang memiliki dua status utama, yaitu apakah pintu terkunci (`_isLocked`) dan apakah pintu terbuka (`_isOpen`). Kelas ini menyediakan beberapa metode untuk mengelola dan menampilkan status pintu, seperti `CurrentState()` yang menampilkan kondisi terkini pintu, `ToggleLock()` untuk mengunci atau membuka kunci pintu dengan pengecekan jika pintu sedang terbuka agar tidak bisa dikunci, `OpenDoor()` yang hanya mengizinkan pintu dibuka jika tidak terkunci, serta `CloseDoor()` untuk menutup pintu jika sedang terbuka. Dengan mekanisme ini, kelas menjaga logika bahwa pintu tidak bisa dibuka saat terkunci dan tidak bisa dikunci saat sedang terbuka, sehingga mensimulasikan perilaku pintu otomatis yang realistis.

Internal class program.cs

```

0 references
internal class Program
{
    0 references
    static void Main()
    {
        // Menampilkan semua kode pos
        KodePos.TampilkanSemua();

        // Simulasi sistem pintu otomatis
        Console.WriteLine("\n===== PROGRAM DOOR MACHINE =====");
        DoorMachine pintu = new DoorMachine();

        pintu.CurrentState();
        pintu.ToggleLock();
        pintu.OpenDoor();
        pintu.CloseDoor();
        pintu.ToggleLock();

        Console.WriteLine("\nProgram selesai. Tekan Enter untuk keluar.");
        Console.ReadLine();
    }
}

```

Kelas Program dalam file Program.cs berfungsi sebagai titik masuk (entry point) utama dari aplikasi konsol ini. Di dalam kelas tersebut terdapat metode Main(), yang secara otomatis dijalankan saat program dimulai. Pada metode Main(), pertama-tama program memanggil KodePos.TampilkanSemua() untuk menampilkan daftar semua kode pos yang sudah didefinisikan sebelumnya. Setelah itu, program melanjutkan ke simulasi sistem pintu otomatis dengan membuat objek DoorMachine bernama pintu, lalu menjalankan serangkaian operasi seperti menampilkan status pintu, mengubah status kunci pintu, membuka dan menutup pintu, serta kembali mengunci atau membuka kunci pintu. Program ini juga menampilkan informasi status di setiap langkah agar pengguna dapat melihat perubahan kondisi pintu secara interaktif. Di akhir, program menunggu pengguna menekan tombol Enter untuk keluar, sehingga jendela konsol tidak langsung tertutup. Jadi, kelas Program mengatur alur utama eksekusi aplikasi sekaligus menghubungkan fitur daftar kode pos dan simulasi pintu otomatis dalam satu program yang terstruktur.

III. Output

===== DAFTAR KODE POS =====

Batununggal: 40266

Kujangsari: 40287

Mengger: 40267

Wates: 40256

Cijaura: 40287

Jatisari: 40286

Margasari: 40286

Sekejati: 40286

Kebonwaru: 40272

Maleer: 40274

Samoja: 40273

===== PROGRAM DOOR MACHINE =====

Status: Terkunci, Pintu: Tertutup

Pintu dibuka kuncinya.

Pintu berhasil dibuka.

Pintu ditutup kembali.

Pintu dikunci.

Program selesai. Tekan Enter untuk keluar.