TP\_Modul 4

2311104059

Zhafir Zaidan Avail

Hasil Run:



Source Code:

* DoorMachine.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace TP\_Modul4

{

public class DoorMachine

{

private enum State { Terkunci, Terbuka }

private State currentState;

public DoorMachine()

{

currentState = State.Terkunci;

Console.WriteLine("Pintu terkunci");

}

public void KunciPintu()

{

if (currentState == State.Terbuka)

{

currentState = State.Terkunci;

Console.WriteLine("Pintu terkunci");

}

else

{

Console.WriteLine("Pintu sudah terkunci");

}

}

public void BukaPintu()

{

if (currentState == State.Terkunci)

{

currentState = State.Terbuka;

Console.WriteLine("Pintu tidak terkunci");

}

else

{

Console.WriteLine("Pintu sudah terbuka");

}

}

}

}

Penjelasan:

Class DoorMachine berfungsi untuk mensimulasikan kondisi pintu menggunakan teknik state-based construction. Class ini memiliki dua state utama yaitu "Terkunci" dan "Terbuka", yang direpresentasikan menggunakan enum State. Terdapat dua method yaitu KunciPintu() untuk mengubah pintu menjadi terkunci dan BukaPintu() untuk membuka pintu (tidak terkunci). Setiap perubahan state akan menampilkan pesan kondisi pintu ke console, seperti "Pintu terkunci" atau "Pintu tidak terkunci".

* KodePos.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace TP\_Modul4

{

public class KodePos

{

private Dictionary<string, string> kodePosTable = new Dictionary<string, string>(StringComparer.OrdinalIgnoreCase) // agar tidak case-sensitive

{

{"Batununggal", "40266"},

{"Kujangsari", "40287"},

{"Mengger", "40267"},

{"Wates", "40256"},

{"Cijaura", "40287"},

{"Jatisari", "40286"},

{"Margasari", "40286"},

{"Sekejati", "40286"},

{"Kebonwaru", "40272"},

{"Maleer", "40274"},

{"Samoja", "40273"}

};

// Method untuk menampilkan semua tabel kode pos

public void tampilkanTabel()

{

Console.WriteLine("=======================================");

Console.WriteLine("| Kelurahan | Kode Pos |");

Console.WriteLine("=======================================");

foreach (KeyValuePair<string, string> entry in kodePosTable)

{

Console.WriteLine($"| {entry.Key,-15} | {entry.Value,-16} |");

}

Console.WriteLine("=======================================");

}

// Method untuk mencari kode pos berdasarkan nama kelurahan

public void cariKodePos(string namaKelurahan)

{

if (kodePosTable.ContainsKey(namaKelurahan))

{

Console.WriteLine($"Kode pos untuk kelurahan {namaKelurahan} adalah: {kodePosTable[namaKelurahan]}");

}

else

{

Console.WriteLine($"Kelurahan {namaKelurahan} tidak ditemukan dalam tabel.");

}

}

}

}

Penjelasan:

Class KodePos berfungsi untuk menyimpan data tabel kelurahan beserta kode pos-nya menggunakan Dictionary. Class ini menerapkan teknik table-driven dengan menyediakan dua method, yaitu tampilkanTabel() yang berfungsi untuk menampilkan seluruh data kelurahan dan kode pos dalam bentuk tabel di console, dan cariKodePos(string namaKelurahan) yang digunakan untuk mencari kode pos berdasarkan nama kelurahan yang diinput oleh user. Jika kelurahan ditemukan, maka akan ditampilkan kode pos-nya, jika tidak ditemukan akan muncul pesan bahwa data tidak ada.

* Program.cs (Main Class)

using TP\_Modul4;

class Program

{

static void Main(string[] args)

{

// Membuat object dari KodePos dan DoorMachine

KodePos kodePos = new KodePos();

DoorMachine door = new DoorMachine();

Console.WriteLine("\n===== TABEL KODE POS =====");

kodePos.tampilkanTabel();

// Input user untuk mencari kode pos

Console.WriteLine("\nMasukkan nama kelurahan untuk mencari kode pos-nya:");

Console.Write("Nama Kelurahan: ");

string inputKelurahan = Console.ReadLine();

kodePos.cariKodePos(inputKelurahan);

// Simulasi Door Machine

Console.WriteLine("\n===== SIMULASI DOOR MACHINE =====");

door.BukaPintu(); // Harusnya pintu tidak terkunci

door.KunciPintu(); // Harusnya pintu terkunci

door.KunciPintu(); // Sudah terkunci

door.BukaPintu(); // Pintu tidak terkunci

door.BukaPintu(); // Sudah terbuka

}

}

Penjelasan:

Class Program merupakan class utama (main class) yang menjalankan keseluruhan program. Di dalam class ini dibuat object dari class KodePos dan DoorMachine. Program ini akan menampilkan tabel kode pos terlebih dahulu, kemudian meminta input dari user untuk mencari kode pos berdasarkan nama kelurahan yang dimasukkan. Setelah itu, program juga akan menjalankan simulasi pintu dengan memanggil method BukaPintu() dan KunciPintu() dari class DoorMachine, untuk menunjukkan perubahan state pintu (terkunci/terbuka). Semua proses ini berjalan di dalam method Main() sebagai titik awal aplikasi.