

**LAPORAN PRAKTIKUM
MODUL 4
AUTOMATA DAN TABLE-DRIVEN CONSTRUCTION**



Disusun Oleh :

Zaenarif Putra 'Ainurdin – 2311104049

Kelas :

SE-07-02

Dosen :

Yudha Islami Sulistya

**PROGRAM STUDI SOFTWARE ENGINEERING
DIREKTORAT KAMPUS PURWOKERTO
TELKOM UNIVERSITY
PURWOKERTO
2024**

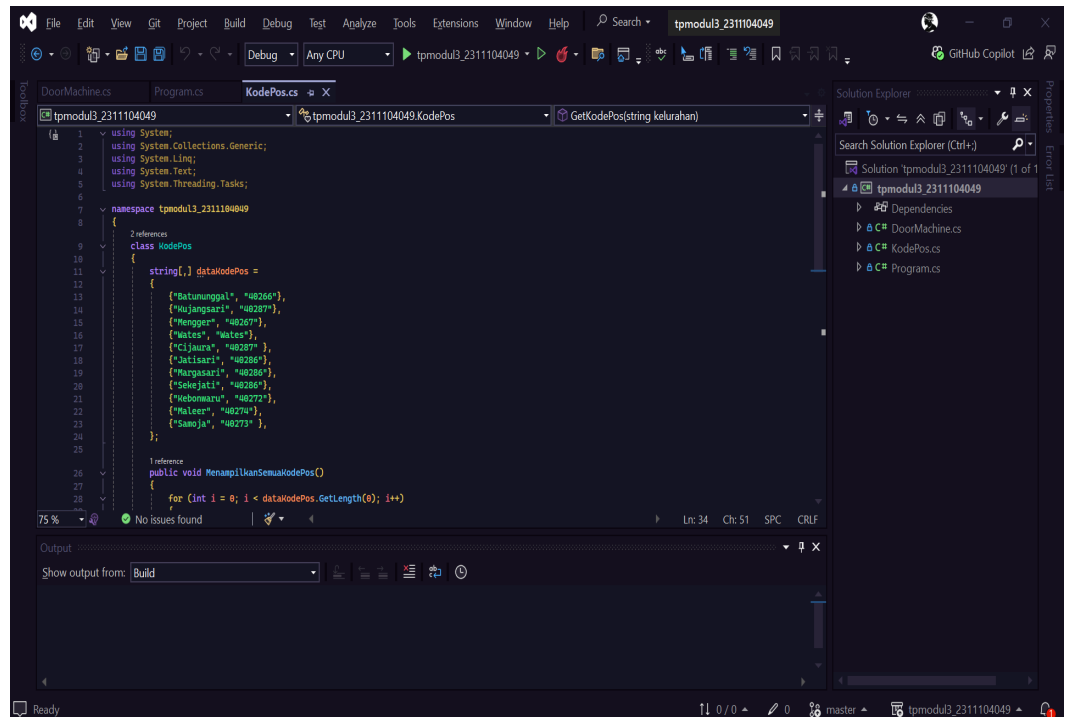
I. Link Github

- https://github.com/zaenarifputra/04_Automata_dan_Table-Driven_Construction.git

II. Penjelasan Syntax

a. Class KodePos.cs

i. Syntax



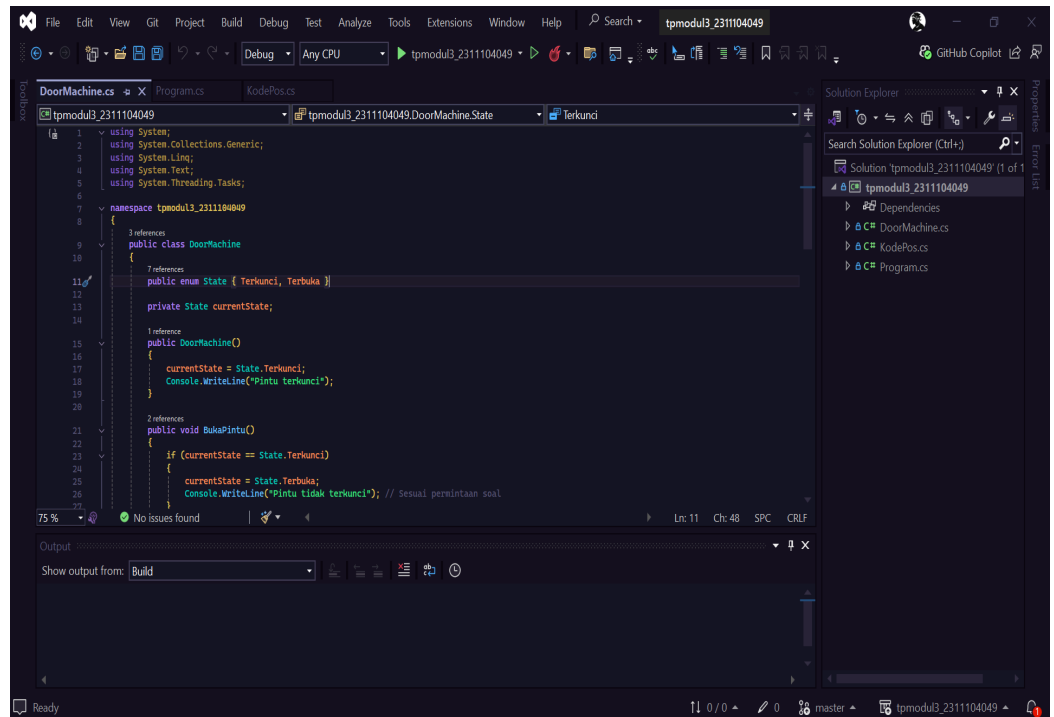
```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace tpmodul3_2311104049
8 {
9     internal class KodePos
10     {
11         string[,] dataKodePos =
12         {
13             {"Batununggal", "40266"},
14             {"Mujaningsari", "40287"},
15             {"Menger", "40267"},
16             {"Wates", "40267"},
17             {"Teljaur", "40287"},
18             {"Jatisari", "40286"},
19             {"Margasari", "40286"},
20             {"Sekejati", "40286"},
21             {"Sebonmaru", "40272"},
22             {"Naler", "40274"},
23             {"Samoja", "40273"}
24         };
25
26         internal void MenampilkanSemuaKodePos()
27         {
28             for (int i = 0; i < dataKodePos.GetLength(0); i++)
```

The screenshot shows the Visual Studio IDE with a C# project named 'tpmodul3_2311104049'. The file 'KodePos.cs' is open, displaying a class definition for 'KodePos'. The class contains a 2D array 'dataKodePos' with 11 rows and 2 columns, representing location names and their corresponding codes. A method 'MenampilkanSemuaKodePos()' is also defined, which iterates through the array. The Solution Explorer on the right shows the project structure, including 'DoorMachine.cs', 'KodePos.cs', and 'Program.cs'. The Output window at the bottom shows the build process.

- Menampilkan Semua Kode Pos() : Digunakan untuk melakukan pengulangan yang bertujuan membaca seluruh data dalam array dataKodePos.
- GetKodePos(string kelurahan) : Mencari kode pos dengan menggunakan nama kelurahan yang dimasukkan. Proses ini melibatkan perulangan untuk mencocokkan kelurahan yang dicari dengan data yang ada dalam array. Apabila kelurahan tersebut ditemukan, kode pos akan dikembalikan. Namun, jika tidak ditemukan, akan muncul pesan yang menyatakan bahwa data tidak tersedia.

b. Class DoorMachine.cs

i. Syntax



```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace tpmodul3_2311104049
8  {
9      3 references
10     public class DoorMachine
11     {
12         7 references
13         public enum State { Ter Kunci, Terbuka }
14         private State currentState;
15
16         1 reference
17         public DoorMachine()
18         {
19             currentState = State.Ter Kunci;
20             Console.WriteLine("Pintu terkunci");
21         }
22
23         2 references
24         public void BukaPintu()
25         {
26             if (currentState == State.Ter Kunci)
27             {
28                 currentState = State.Terbuka;
29                 Console.WriteLine("Pintu tidak terkunci"); // Sesuai permintaan soal
30             }
31         }
32     }
33 }

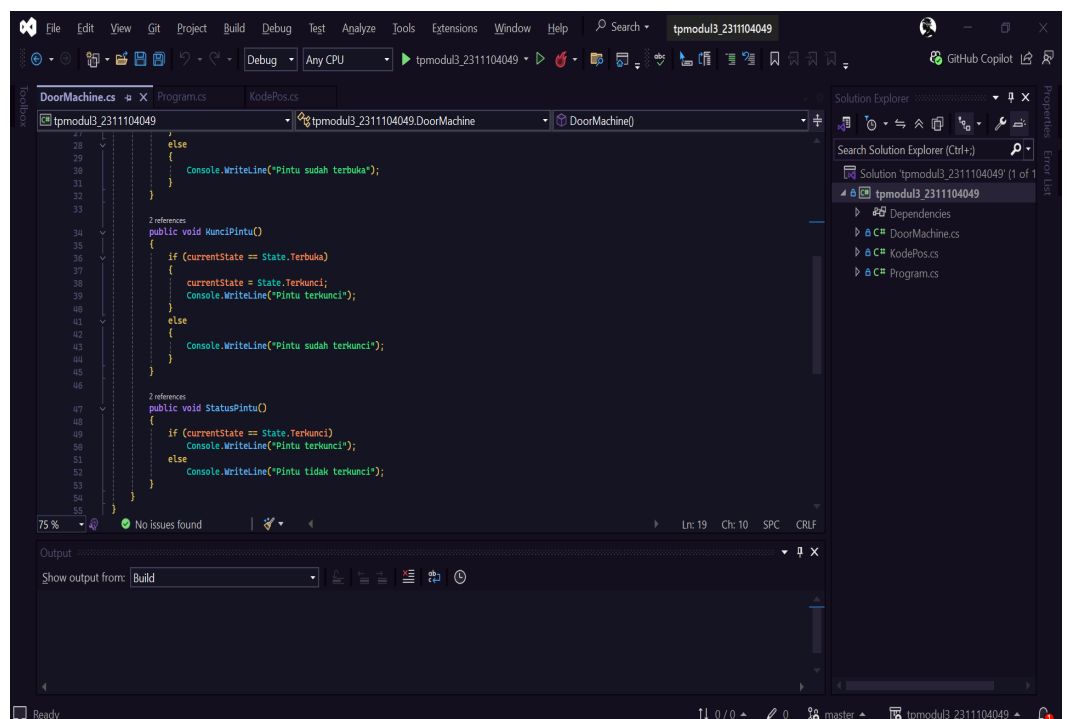
```

Output

```

Show output from: Build
Pintu terkunci
Pintu tidak terkunci

```



```

27     else
28     {
29         Console.WriteLine("Pintu sudah terbuka");
30     }
31 }
32
33 2 references
34 public void kunciPintu()
35 {
36     if (currentState == State.Terbuka)
37     {
38         currentState = State.Ter Kunci;
39         Console.WriteLine("Pintu terkunci");
40     }
41     else
42     {
43         Console.WriteLine("Pintu sudah terkunci");
44     }
45 }
46
47 2 references
48 public void StatusPintu()
49 {
50     if (currentState == State.Ter Kunci)
51     {
52         Console.WriteLine("Pintu terkunci");
53     }
54     else
55     {
56         Console.WriteLine("Pintu tidak terkunci");
57     }
58 }
59 }
60 }

```

Output

```

Show output from: Build
Pintu terkunci
Pintu tidak terkunci
Pintu terkunci
Pintu tidak terkunci

```

ii . Penjelasan syntax

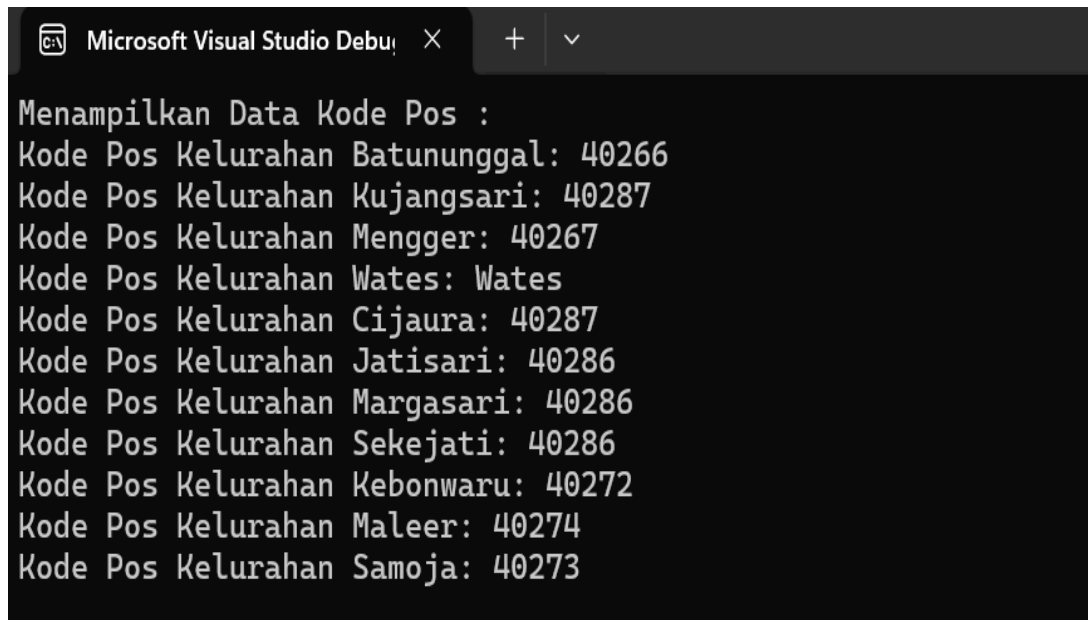
- State (Enum) → Menyimpan dua kondisi pintu yaitu :

Terkunci dan Terbuka.

- `currentState` (Variabel Privat) → Menyimpan status pintu saat ini.
- `DoorMachine()` (Constructor) → Menginisialisasi pintu dalam keadaan terkunci saat objek dibuat.
- `BukaPintu()` → Mengubah status pintu menjadi terbuka jika sebelumnya terkunci.
- `KunciPintu()` → Mengubah status pintu menjadi terkunci jika sebelumnya terbuka.
- `StatusPintu()` → Menampilkan status pintu saat ini (terkunci atau tidak).

III. Hasil Running

1. Class KodePos.cs



```
Microsoft Visual Studio Debug Console
Menampilkan Data Kode Pos :
Kode Pos Kelurahan Batununggal: 40266
Kode Pos Kelurahan Kujangsari: 40287
Kode Pos Kelurahan Mengger: 40267
Kode Pos Kelurahan Wates: Wates
Kode Pos Kelurahan Cijaura: 40287
Kode Pos Kelurahan Jatisari: 40286
Kode Pos Kelurahan Margasari: 40286
Kode Pos Kelurahan Sekejati: 40286
Kode Pos Kelurahan Kebonwaru: 40272
Kode Pos Kelurahan Maleer: 40274
Kode Pos Kelurahan Samoja: 40273
```

2. Class DoorMachine.cs

```
Pintu terkunci  
  
Pintu tidak terkunci  
Pintu tidak terkunci  
  
Pintu terkunci  
Pintu terkunci  
  
Pintu sudah terkunci  
Pintu tidak terkunci  
  
C:\Users\LENOVO\source\repos\tpmodul3_2311104049\tpmodul3_2311104049\bin\Debug\net8.0\tpmodul3_2311104049.exe (process 18856) exited  
with code 0 (0x0).  
Press any key to close this window . . .|
```