

Nama : Gideon Toranawa Ladiyo
NIM : 2211104022
Kelas : SE0601

https://github.com/gideonladiyo/tpmodul7_2211104022

```
Microsoft Visual Studio Debu... x + v
Nama Gideon Toranawa Ladiyo dengan nim 2211104022 dari fakultas informatika
=====

Daftar mata kuliah yang diambil:
MK 1 CCK2KAB4 - KONSTRUKSI PERANGKAT LUNAK
MK 2 CCK3KAB3 - MANAJEMEN KONFIGURASI DAN EVOLUSI PERANGKAT LUNAK
MK 3 CCK3GAB4 - PENGUJIAN PERANGKAT LUNAK
MK 4 CCK2LAB3 - KECERDASAN BUATAN
MK 5 CCK2DAB3 - PROSES PERANGKAT LUNAK
MK 6 CCK1LAB3 - DESIGN THINKING
MK 7 CCK2JAC2 - PROYEK TINGKAT II
MK 8 CCK3IAB2 - TATA TULIS ILMIAH

D:\kuliah\semester 6\praktikum kpl\KPL_GideonToranawaLadiyo_2211104022_SE0601\
tp\tpmodul7_2211104022\tpmodul7_2211104022\bin\Debug\net8.0\tpmodul7_2211104022
0x0).
To automatically close the console when debugging stops, enable Tools->Options
le when debugging stops.
Press any key to close this window . . .|
```

```
1 namespace tpmodul7_2211104022
2 {
3     class Nama
4     {
5         [JsonPropertyName("depan")]
6         public string Depan { get; set; }
7
8         [JsonPropertyName("belakang")]
9         public string Belakang { get; set; }
10    }
11    class DataMahasiswa2211104022
12    {
13        [JsonPropertyName("nama")]
14        public Nama Nama { get; set; }
15
16        [JsonPropertyName("nim")]
17        public string Nim { get; set; }
18
19        [JsonPropertyName("fakultas")]
20        public string Fakultas { get; set; }
```

Buat class Nama untuk parsing nested object yang berisi nama depan dan nama belakang. Kemudian class DataMahasiswa2211104022 untuk parsing semua data yang ada di dalam file JSON.

```

public static void ReadJSON()
{
    string pathFile = Path.Combine(Directory.GetCurrentDirectory(), "solution", "tp7_1_2211104022.json");
    if (File.Exists(pathFile))
    {
        string jsonData = File.ReadAllText(pathFile);
        DataMahasiswa2211104022 mahasiswa = JsonSerializer.Deserialize<DataMahasiswa2211104022>(jsonData);

        Console.WriteLine($"Nama {mahasiswa>Nama.Depan} {mahasiswa>Nama.Belakang} dengan nim {mahasiswa.Nim} dari fakultas {mahasiswa.Fakultas}");
    }
    else
    {
        Console.WriteLine("File tidak ditemukan");
    }
}

```

Fungsi ReadJSON untuk mem-parsing data yang ada di file json. File berada pada direktori “tpmodul7_2211104022\bin\Debug\net8.0\solution”, dengan membuat string file path untuk file json-nya. Variable jsonData adalah variable hasil read yang akan diparsing dengan JsonSerializer. Setelah itu akan dikeluarkan hasil parsing dengan console.writeline

```

namespace tpmodul7_2211104022
{
    class Course
    {
        [JsonPropertyName("code")]
        public string Code { get; set; }

        [JsonPropertyName("name")]
        public string Name { get; set; }
    }

    class KuliahMahasiswa2211104022
    {
        [JsonPropertyName("courses")]
        public List<Course> courses { get; set; }
    }
}

```

Kemudian untuk parsing yang kedua kita pertama buat class Course karena teradapat nested object yang berisi list data course. Kemudian kita parsing list tersebut di dalam class KuliahMahasiswa 2211104022.

```

public static void ReadJSON()
{
    string pathFile = Path.Combine(Directory.GetCurrentDirectory(), "solution", "tp7_2_2211104022.json");
    if (File.Exists(pathFile))
    {
        string jsonData = File.ReadAllText(pathFile);
        KuliahMahasiswa2211104022 courses = JsonSerializer.Deserialize<KuliahMahasiswa2211104022>(jsonData);
        Console.WriteLine("Daftar mata kuliah yang diambil:");
        int index = 1;
        foreach (var course in courses.courses)
        {
            Console.WriteLine($"MK {index} {course.Code} - {course.Name}");
            index++;
        }
    }
    else
    {
        Console.WriteLine("File tidak ditemukan");
    }
}

```

Fungsi ini sama seperti tadi, bedanya ini hasil parsing adalah sebuah list, yang di mana kita mengeluarkan hasil parsing dengan foreach.