

TUGAS JURNAL 07
KONSTRUKSI PERANGKAT LUNAK

GRAMMAR-BASED INPUT PROCESSING PARSING



DISUSUN OLEH:
KHOLIFAH DINA
2211104004


SE 06 01

S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY
2025

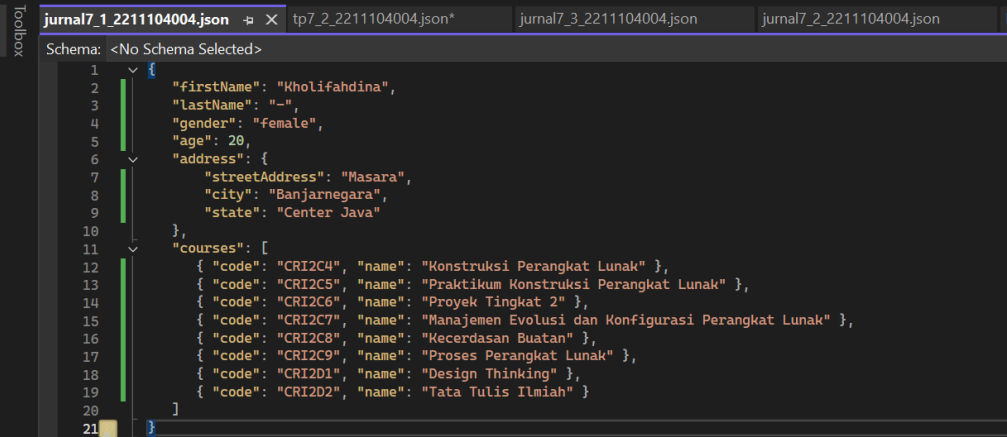
1. MENAMBAHKAN JSON DESERIALIZATION 1

Buatlah branch baru dengan nama branch “nama_panggilan_praktikan” dan checkout kesana.

- a. Download file “jurnal7_1_nim.json” dan rename file tersebut dengan mengganti “nim” dengan NIM praktikan kemudian pindahkan file json tersebut di folder solution projectnya.


 jurnal7_1_2211104004.json

- b. Ganti isi dari file json tersebut dengan detail yang benar dari praktikan.



```
1 {
2   "firstName": "Kholifahdina",
3   "lastName": "-",
4   "gender": "female",
5   "age": 20,
6   "address": {
7     "streetAddress": "Masara",
8     "city": "Banjarnegara",
9     "state": "Center Java"
10  },
11  "courses": [
12    { "code": "CRI2C4", "name": "Konstruksi Perangkat Lunak" },
13    { "code": "CRI2C5", "name": "Praktikum Konstruksi Perangkat Lunak" },
14    { "code": "CRI2C6", "name": "Proyek Tingkat 2" },
15    { "code": "CRI2C7", "name": "Manajemen Evolusi dan Konfigurasi Perangkat Lunak" },
16    { "code": "CRI2C8", "name": "Kecerdasan Buatan" },
17    { "code": "CRI2C9", "name": "Proses Perangkat Lunak" },
18    { "code": "CRI2D1", "name": "Design Thinking" },
19    { "code": "CRI2D2", "name": "Tata Tulis Ilmiah" }
20  ]
21 }
```

- c. Buatlah sebuah file class baru dengan nama “DataMahasiswa<NIM_PRAKTIKAN>”.

 DataMahasiswa_2211104004.cs

- d. Buat method “ReadJSON() yang melakukan parsing untuk file tersebut menjadi object sesuai.
- e. Pada method tersebut, lakukan print hasil deserialisasi dari object yang dibuat dengan format bebas asalkan semua nilai ditampilkan di console/output.

Source Code:

```
using System;
using System.IO;
using System.Text.Json;

public class Program
{
    public class Address
    {
        public string streetAddress { get; set; }
        public string city { get; set; }
        public string state { get; set; }
    }
}
```

```

public class Course
{
    public string code { get; set; }
    public string name { get; set; }
}

public class Student
{
    public string firstName { get; set; }
    public string lastName { get; set; }
    public string gender { get; set; }
    public int age { get; set; }
    public Address address { get; set; }
    public Course[] courses { get; set; }
}

public static void ReadJSON(string filePath)
{
    try
    {
        // Show where the program is looking for the
file
        Console.WriteLine($"Looking for file at:
{Path.GetFullPath(filePath)}");

        if (!File.Exists(filePath))
        {
            Console.WriteLine("Error: File not found.
Please ensure:");
            Console.WriteLine($"1. The file
'{Path.GetFileName(filePath)}' exists");
            Console.WriteLine($"2. It's in this
folder:
{Path.GetDirectoryPath(Path.GetFullPath(filePath))}");
            return;
        }

        string jsonString =
File.ReadAllText(filePath);
        Student student =
JsonSerializer.Deserialize<Student>(jsonString);

        Console.WriteLine("\n=== Student Data ===");
        Console.WriteLine($"Name: {student.firstName}
{student.lastName}");
        Console.WriteLine($"Gender:
{student.gender}");
        Console.WriteLine($"Age: {student.age}");

        Console.WriteLine("\n=== Address ===");
        Console.WriteLine($"Street:
{student.address.streetAddress}");
        Console.WriteLine($"City:
{student.address.city}");
    }
}

```

```

        Console.WriteLine($"State:
{student.address.state}");

        Console.WriteLine("\n=== Courses ===");
        foreach (var course in student.courses)
        {
            Console.WriteLine($"{{course.code}}:
{course.name}");
        }
    }
    catch (JsonException)
    {
        Console.WriteLine("Error: Invalid JSON
format");
    }
    catch (Exception ex)
    {
        Console.WriteLine($"Error: {ex.Message}");
    }
}

public static void Main()
{
    // Using the exact filename you specified
    string jsonFilePath = "jurnal7_1_2211104004.json";
    ReadJSON(jsonFilePath);

    Console.WriteLine("\nPress enter to exit...");
    Console.ReadLine();
}
}

```

Output:

```

=== Student Data ===
Name: Kholifahdina -
Gender: female
Age: 20

=== Address ===
Street: Masara
City: Banjarnegara
State: Center Java

=== Courses ===
CRI2C4: Konstruksi Perangkat Lunak
CRI2C5: Praktikum Konstruksi Perangkat Lunak
CRI2C6: Proyek Tingkat 2
CRI2C7: Manajemen Evolusi dan Konfigurasi Perangkat Lunak
CRI2C8: Kecerdasan Buatan
CRI2C9: Proses Perangkat Lunak
CRI2D1: Design Thinking
CRI2D2: Tata Tulis Ilmiah

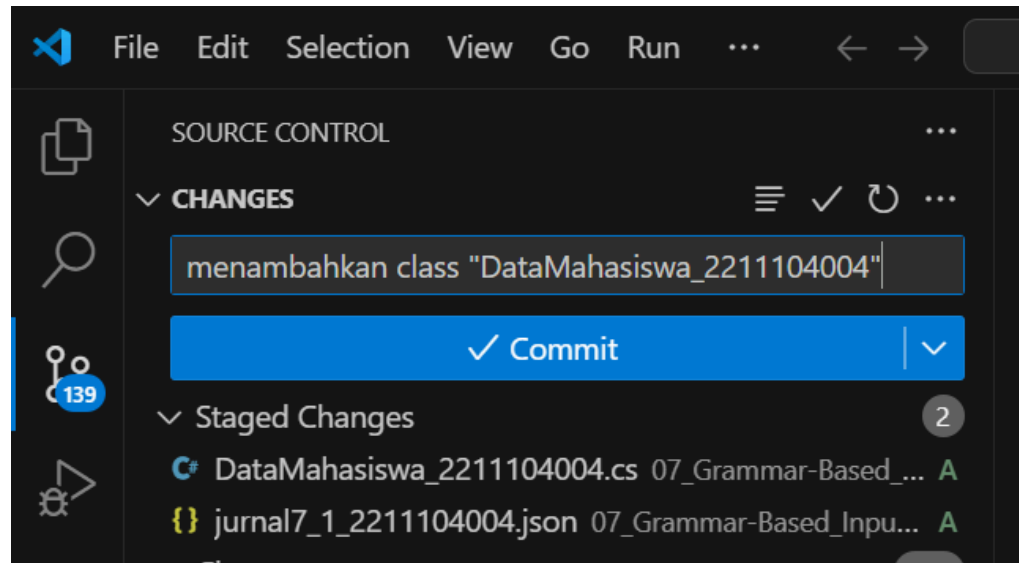
Press enter to exit...

```

2. MELAKUKAN COMMIT

Pada branch yang dibuat sebelumnya:

- a. Lakukan commit dengan pesan “menambahkan class <NAMA_CLASS>”.
- b. Lakukan push ke github ke branch yang dibuat di bagian sebelumnya.



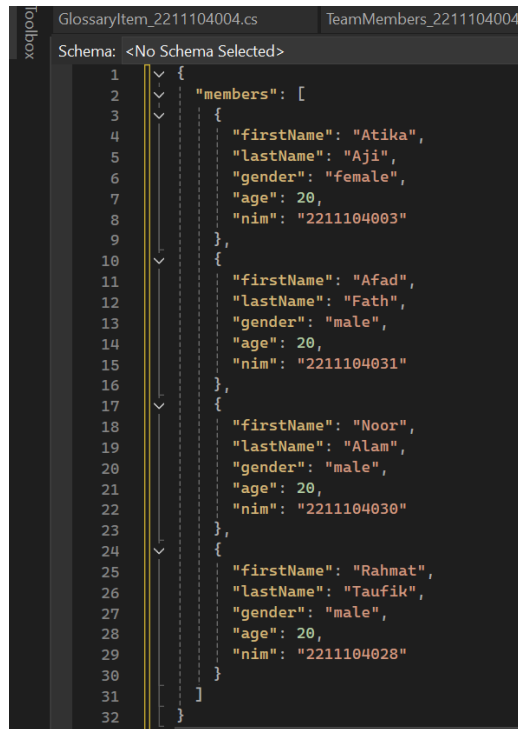
3. MENAMBAHKAN JSON DESERIALIZATION 2

Buatlah branch baru dengan nama branch “nama_panggilan_praktikan” dan checkout kesana.

- a. Download file “jurnal7_2_nim.json” dan rename file tersebut dengan mengganti “nim” dengan NIM praktikan kemudian pindahkan file json tersebut di folder solution projectnya.



- b. Ubah isi dari file json tersebut dengan daftar anggota kelompok (untuk tubes).



```

1  {
2    "members": [
3      {
4        "firstName": "Atika",
5        "lastName": "Aji",
6        "gender": "female",
7        "age": 20,
8        "nim": "2211104003"
9      },
10     {
11       "firstName": "Afad",
12       "lastName": "Fath",
13       "gender": "male",
14       "age": 20,
15       "nim": "2211104031"
16     },
17     {
18       "firstName": "Noor",
19       "lastName": "Alam",
20       "gender": "male",
21       "age": 20,
22       "nim": "2211104030"
23     },
24     {
25       "firstName": "Rahmat",
26       "lastName": "Taufik",
27       "gender": "male",
28       "age": 20,
29       "nim": "2211104028"
30     }
31   ]
32 }

```

- c. Buatlah sebuah file class baru dengan nama “TeamMembers<NIM_PRAKTIKAN>”.

C# TeamMembers_2211104004.cs

- d. Buat method “ReadJSON() yang melakukan parsing untuk file tersebut menjadi object sesuai.
- e. Pada method tersebut, lakukan print hasil deserialisasi dari object yang dibuat dengan format:

“Team member list:”

“<nim1> <firstname1 + lastname1> (<age1> <gender1>) ”

“<nim2> <firstname2 + lastname2> (<age2> <gender2>) ”

dst.

source Code

```

using System;
using System.IO;
using System.Collections.Generic;
using Newtonsoft.Json;

class TeamMembers
{
    class Member
    {

```

```

        public string NIM { get; set; }
        public string FirstName { get; set; }
        public string LastName { get; set; }
        public int Age { get; set; }
        public string Gender { get; set; }
    }

    class Team
    {
        public List<Member> Members { get; set; }
    }

    static void Main()
    {
        string filePath = "jurnal7_2_2211104004.json";
        // Pastikan file ini ada di direktori proyek

        if (!File.Exists(filePath))
        {
            Console.WriteLine("❌ File JSON tidak ditemukan!");
            return;
        }

        var jsonData = File.ReadAllText(filePath);
        var team =
        JsonConvert.DeserializeObject<Team>(jsonData);

        if (team?.Members == null ||
        team.Members.Count == 0)
        {
            Console.WriteLine("⚠️ Data kosong atau tidak valid.");
            return;
        }

        Console.WriteLine("\n✅ Team Member List:");
        foreach (var member in team.Members)
        {
            Console.WriteLine($"{member.NIM} -
            {member.FirstName} {member.LastName} ({member.Age}
            {member.Gender})");
        }

        Console.WriteLine("\nPress any key to
        exit...");
        Console.ReadLine();
    }
}

```

Output:

```

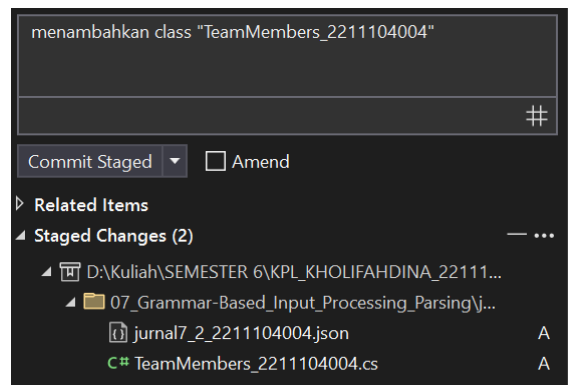
✓ Team Member List:
2211104003 - Atika Aji (20 female)
2211104031 - Afad Fath (20 male)
2211104030 - Noor Alam (20 male)
2211104028 - Rahmat Taufik (20 male)

```

4. MELAKUKAN COMMIT

Pada branch yang dibuat sebelumnya:

- Lakukan commit dengan pesan “menambahkan class <NAMA_CLASS>”.
- Lakukan push ke github ke branch yang dibuat di bagian sebelumnya.



5. MENAMBAHKAN JSON DESERIALIZATION 3

Buatlah branch baru dengan nama branch “nama_panggilan_praktikan” dan checkout kesana.

- Download file “jurnal7_3_nim.json” dan rename file tersebut dengan mengganti “nim” dengan NIM praktikan kemudian pindahkan file json tersebut di folder solution projectnya.

jurnal7_3_2211104004.json

- Buatlah sebuah file class baru dengan nama “GlossaryItem<NIM_PRAKTIKAN>”.

C# GlossaryItem_2211104004.cs

- Buat method “ReadJSON() yang melakukan parsing untuk file tersebut menjadi object sesuai.
- Pada method tersebut, lakukan print hasil deserialisasi dari object yang dibuat dengan format bebas untuk bagian “GlossEntry” saja.\

Source Code:

```
using System;
```



```

using System.IO;
using System.Text.Json;

class Program
{
    public class GlossDef
    {
        public string para { get; set; }
        public string[] GlossSeeAlso { get; set; }
    }

    public class GlossEntry
    {
        public string ID { get; set; }
        public string SortAs { get; set; }
        public string GlossTerm { get; set; }
        public string Acronym { get; set; }
        public string Abbrev { get; set; }
        public GlossDef GlossDef { get; set; }
        public string GlossSee { get; set; }
    }

    public class GlossList
    {
        public GlossEntry GlossEntry { get; set; }
    }

    public class GlossDiv
    {
        public string Title { get; set; }
        public GlossList GlossList { get; set; }
    }

    public class Glossary
    {
        public string Title { get; set; }
        public GlossDiv GlossDiv { get; set; }
    }

    public class Root
    {
        public Glossary glossary { get; set; }
    }

    static void ReadJSON(string filePath)
    {
        if (!File.Exists(filePath))
        {
            Console.WriteLine("File not found: " +
filePath);
            return;
        }

        string json = File.ReadAllText(filePath);

```

```

        Root data =
JsonSerializer.Deserialize<Root>(json);
        GlossEntry entry =
data.glossary.GlossDiv.GlossList.GlossEntry;

        Console.WriteLine("GlossEntry Details:");
        Console.WriteLine($"ID: {entry.ID}");
        Console.WriteLine($"SortAs: {entry.SortAs}");
        Console.WriteLine($"GlossTerm:
{entry.GlossTerm}");
        Console.WriteLine($"Acronym:
{entry.Acronym}");
        Console.WriteLine($"Abbrev: {entry.Abbrev}");
        Console.WriteLine($"GlossDef:
{entry.GlossDef.para}");
        Console.WriteLine($"GlossSeeAlso:
{string.Join(", ", entry.GlossDef.GlossSeeAlso)}");
        Console.WriteLine($"GlossSee:
{entry.GlossSee}");
    }

    static void Main()
    {
        string filePath = "jurnal7_3_2211104004.json";
        ReadJSON(filePath);
    }
}

```

Output:

```

GlossEntry Details:
ID: SGML
SortAs: SGML
GlossTerm: Standard Generalized Markup Language
Acronym: SGML
Abbrev: ISO 8879:1986
GlossDef: A meta-markup language, used to create markup languages such as DocBook.
GlossSeeAlso: GML, XML
GlossSee: markup

```

6. MELAKUKAN COMMIT TERAKHIR

Pada branch yang dibuat sebelumnya:

- Lakukan commit dengan pesan “menambahkan class <NAMA_CLASS>”.
- Lakukan push ke github ke branch yang dibuat di bagian sebelumnya.

Git Changes - KPL_KHOLIFAHINA_2211104004_SE0601

main

↓ ↓ ↑ ↺ ...

↑↓ 1 / 0 [View all commits](#)

menambahkan class "GlossaryItem_2211104004"

#

Commit Staged

☐ Amend

▶ Related Items

▲ Staged Changes (2) — ...

📁 D:\Kuliah\SEMESTER 6\KPL_KHOLIFAHINA_2211104004_SE0601

📁 07_Grammar-Based_Input_Processing_Parsing\jurnal7_2211104004\GlossaryItem

C# GlossaryItem_2211104004.cs A

📄 jurnal7_3_2211104004.json A