

**TUGAS PENDAHULUAN  
KONSTRUKSI PERANGKAT LUNAK**

**MODUL VII  
GRAMMAR-BASED INPUT PROCESSING  
PARSING**



**Disusun Oleh:**

Nadia Putri Rahmانيar / 2211104012

S1 SE-06-01

**Dosen Pengampu:**

Yudha Islami Sulistya, S.Kom., M.Cs

**PROGRAM STUDI S1 SOFTWARE ENGINEERING FAKULTAS  
INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO  
2025**

## TUGAS JURNAL

### 1. MENAMBAHKAN JSON DESERIALIZATON 1

Buatlah branch baru dengan nama branch “nama\_panggilan\_praktikan” dan checkout kesana

- 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.
- Ganti isi dari file json tersebut dengan detail yang benar dari praktikan.
- Buatlah sebuah file class baru dengan nama “DataMahasiswa”.
- 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 asalkan semua nilai ditampilkan di console/output.”

**Jawab:**

**File jurnal7\_1\_2211104012.json**

```
1  {
2      "firstName": "Nadia",
3      "lastName": "Putri Rahmaniar",
4      "gender": "female",
5      "age": 22,
6      "address": {
7          "streetAddress": "JL krapyak",
8          "city": "Purwokerto",
9          "state": "Central Java"
10     },
11     "courses": [
12         {
13             "code": "CRI2C4",
14             "name": "Konstruksi Perangkat Lunak"
15         },
16         {
17             "code": "CRI2XX",
18             "name": "Kecerdasan Buatan"
19         }
20     ]
21 }
```

## File DataMahasiswa2211104012.cs

```
1  using System;
2  using System.IO;
3  using System.Text.Json;
4
5  0 references
6  public class DataMahasiswa2211104012
7  {
8      // Class untuk merepresentasikan struktur JSON
9      1 reference
10     public class Address
11     {
12         1 reference
13         public string streetAddress { get; set; }
14         1 reference
15         public string city { get; set; }
16         1 reference
17         public string state { get; set; }
18     }
19
20     1 reference
21     public class Course
22     {
23         1 reference
24         public string code { get; set; }
25         1 reference
26         public string name { get; set; }
27     }
28
29     2 references
30     public class Mahasiswa
31     {
32         1 reference
33         public string firstName { get; set; }
34         1 reference
35         public string lastName { get; set; }
36         1 reference
37         public string gender { get; set; }
38         1 reference
39         public int age { get; set; }
40
41         3 references
42         public Address address { get; set; }
43         1 reference
44         public Course[] courses { get; set; }
45     }
46
47     0 references
48     public static void ReadJSON()
49     {
50         string filePath = "jurnal7_1_2211104012.json";
51
52         if (File.Exists(filePath))
53         {
54             string jsonString = File.ReadAllText(filePath);
55
56             Mahasiswa mahasiswa = JsonSerializer.Deserialize<Mahasiswa>(jsonString);
57
58             Console.WriteLine("==== Data Mahasiswa =====");
59             Console.WriteLine($"Nama : {mahasiswa.firstName} {mahasiswa.lastName}");
60             Console.WriteLine($"Gender: {mahasiswa.gender}");
61             Console.WriteLine($"Usia : {mahasiswa.age}");
62             Console.WriteLine($"Alamat: {mahasiswa.address.streetAddress}, " +
63                 $"{mahasiswa.address.city}, {mahasiswa.address.state}");
64             Console.WriteLine("\n===== Mata Kuliah =====");
65
66             foreach (var course in mahasiswa.courses)
67             {
68                 Console.WriteLine($"Kode: {course.code} - Nama: {course.name}");
69             }
70         }
71         else
72         {
73             Console.WriteLine("File JSON tidak ditemukan.");
74         }
75     }
76 }
```

## File Program.cs

```
0 references
class Program
{
    static void Main(string[] args)
    {
        DataMahasiswa2211104012.ReadJSON();
    }
}
```

## Output

```
===== Data Mahasiswa =====
Nama   : Nadia Putri Rahmaniar
Gender : female
Usia   : 21
Alamat : JL krapyak, Purwokerto, Central Java

===== Mata Kuliah =====
Kode: CRI2C4 - Nama: Konstruksi Perangkat Lunak
Kode: CRI2XX - Nama: Kecerdasan Buatan
```

## Penjelasan

Program ini dirancang untuk membaca serta menampilkan informasi mahasiswa yang disimpan dalam file JSON. File bernama jurnal7\_1\_2211104012.json berisi data mahasiswa seperti nama, usia, jenis kelamin, alamat, dan daftar mata kuliah. File JSON tersebut diletakkan dalam folder proyek dan akan dibaca oleh program untuk kemudian diubah menjadi objek di dalam bahasa pemrograman C# melalui proses yang disebut parsing. Parsing sendiri merupakan proses mengonversi data dalam format teks JSON ke bentuk objek yang dapat digunakan dalam kode program.

Dalam file DataMahasiswa2211104012.cs, terdapat kelas Mahasiswa yang mencerminkan struktur data JSON, serta kelas-kelas turunan seperti Address dan Course untuk merepresentasikan alamat dan mata kuliah. Method ReadJSON() digunakan untuk membaca isi file JSON dengan bantuan File.ReadAllText(), lalu mengubahnya menjadi objek Mahasiswa menggunakan JsonSerializer.Deserialize<Mahasiswa>(). Setelah data berhasil diparsing, program akan menampilkan informasi tersebut ke layar dalam format yang lebih mudah dibaca.

File Program.cs akan menjalankan method ReadJSON() saat program dieksekusi untuk memulai proses pembacaan dan konversi data dari JSON ke objek.

Dengan pendekatan ini, pengelolaan dan pemrosesan data JSON menjadi lebih terstruktur dan efisien.

## 2. 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

Jawab :

```
ASUS@DESKTOP-I50G73S MINGW64 /d/School/KPL_NadiaPutriRahmaniar_2211104012 (master)
$ git commit -m "menambahkan class DataMahasiswa2211104012"
[master 8615950] menambahkan class DataMahasiswa2211104012
3 files changed, 89 insertions(+)
 create mode 100644 Praktik/07_Grammar-Based_Input_Processing_Parsing/Jurnal/DataMahasiswa2211104012.cs
 create mode 100644 Praktik/07_Grammar-Based_Input_Processing_Parsing/Jurnal/Program.cs
 create mode 100644 Praktik/07_Grammar-Based_Input_Processing_Parsing/Jurnal/jurnal7_1_2211104012.json

ASUS@DESKTOP-I50G73S MINGW64 /d/School/KPL_NadiaPutriRahmaniar_2211104012 (master)
$ git push origin master
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 8 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (8/8), 1.42 KiB | 1.42 MiB/s, done.
```

## 3. MENAMBAHKAN JSON DESERIALIZATON 2

Buatlah branch baru dengan nama branch “nama\_panggilan\_praktikan” dan checkout kesana.

- 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.
- Ubah isi dari file json tersebut dengan daftar anggota kelompok (untuk tubes). C. Buatlah sebuah file class baru dengan nama “TeamMembers”.
- 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:

“Team member list:”

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

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

dst.

**Jawab :**

File jurnal7\_2211104012.json

```
{
  "members": [
    {
      "firstName": "Nita",
      "lastName": "Fitrotul Mar'ah",
      "gender": "Female",
      "age": 21,
      "nim": "2211104005"
    },
    {
      "firstName": "Muhammad",
      "lastName": "Edgar Nadhif",
      "gender": "Male",
      "age": 21,
      "nim": "2211104019"
    },
    {
      "firstName": "Ahmad",
      "lastName": "Junaidi",
      "gender": "Male",
      "age": 21,
      "nim": "2211104002"
    },
    {
      "firstName": "Rafla",
      "lastName": "Dafin kamil",
      "gender": "Male",
      "age": 22,
      "nim": "2211104018"
    },
    {
      "firstName": "Muhammad",
      "lastName": "Dhimas Afrizal",
      "gender": "Male",
      "age": 21,
      "nim": "2211104023"
    },
    {
      "firstName": "Alfian",
```

```
      "lastName": "Mutakim",
      "gender": "Male",
      "age": 21,
      "nim": "2211104000"
    }
  ]
}
```

File TeamMembers2211104012.cs

```
1  using System;
2  using System.IO;
3  using System.Text.Json;
4
5  0 references
6  class TeamMembers2211104012
7  {
8      1 reference
9      class Member
10     {
11         1 reference
12         public string firstName { get; set; }
13         1 reference
14         public string lastName { get; set; }
15         1 reference
16         public string gender { get; set; }
17         1 reference
18         public int age { get; set; }
19         1 reference
20         public string nim { get; set; }
21     }
22
23     2 references
24     class Team
25     {
26         1 reference
27         public Member[] members { get; set; }
28     }
29
30     0 references
31     public static void ReadJSON()
32     {
33         string filePath = "jurnal7_2_2211104012.json";
34
35         if (!File.Exists(filePath))
36         {
37             Console.WriteLine("File JSON tidak ditemukan.");
38             return;
39         }
40
41         // Membaca dan melakukan parsing JSON
42         Team team = JsonSerializer.Deserialize<Team>(File.ReadAllText(filePath));
43
44         Console.WriteLine("Team member list:");
45         foreach (var member in team.members)
46         {
47             Console.WriteLine($"{member.nim} {member.firstName} {member.lastName} " +
48                               $"{member.age} {member.gender}");
49         }
50     }
51 }
```

File Program.cs

```
class Program
{
    0 references
    static void Main(string[] args)
    {
        //DataMahasiswa2211104012.ReadJSON();
        TeamMembers2211104012.ReadJSON();
    }
}
```

Hasil run :

```
Team member list:
2211104005 Nita Fitrotul Mar'ah      (21 Female)
2211104019 Muhammad Edgar Nadhif   (21 Male)
2211104002 Ahmad Junaidi            (21 Male)
2211104018 Rafli Dafin kamil        (22 Male)
2211104023 Muhammad Dhimas Afrizal  (21 Male)
2211104000 Alfian Mutakim           (21 Male)
```

#### Penjelasan :

Program ini dibuat untuk membaca serta menampilkan data anggota kelompok dari sebuah file JSON menggunakan metode parsing JSON. File yang digunakan bernama jurnal7\_2\_2211104012.json, berisi informasi anggota tim seperti NIM, nama, jenis kelamin, dan usia dalam format JSON. File tersebut disimpan di dalam folder proyek agar dapat diakses oleh program.

Parsing merupakan proses untuk mengubah data berbentuk teks JSON menjadi objek dalam bahasa C# agar data tersebut dapat diproses lebih lanjut dalam program. Di dalam file TeamMembers2211104012.cs, program membaca isi file JSON menggunakan File.ReadAllText(). Kemudian, data tersebut diubah menjadi objek Team melalui JsonSerializer.Deserialize<Team>(). Objek Team ini memiliki array members yang menyimpan seluruh informasi anggota kelompok.

Setelah proses parsing selesai, data anggota tim ditampilkan ke console menggunakan format yang telah ditentukan. File Program.cs bertugas memanggil method ReadJSON() agar proses pembacaan dan penampilan data dapat berjalan saat program dijalankan.

#### 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



```

ASUS@DESKTOP-I50G73S MINGW64 /d/School/KPL_NadiaPutriRahmaniar_2211104012 (master)
$ git commit -m "menambahkan class TeamMembers2211104012"
[master ebcf6f9] menambahkan class TeamMembers2211104012
3 files changed, 95 insertions(+), 2 deletions(-)
create mode 100644 Praktik/07_Grammar-Based_Input_Processing_Parsing/Jurnal/TeamMembers2211104012.cs
create mode 100644 Praktik/07_Grammar-Based_Input_Processing_Parsing/Jurnal/jurnal7_2_2211104012.json

ASUS@DESKTOP-I50G73S MINGW64 /d/School/KPL_NadiaPutriRahmaniar_2211104012 (master)
$ git push origin master
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 8 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (8/8), 1.29 KiB | 1.29 MiB/s, done.
Total 8 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/nadiaputrirah/KPL_NadiaPutriRahmaniar_2211104012.git
8615950..ebcf6f9 master -> master

```

## 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.
- Buatlah sebuah file class baru dengan nama “GlossaryItem”.
- 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.

**Jawab :**

File jurnal7\_3\_2211104012.json

```

{
  "glossary": {
    "title": "example glossary",
    "GlossDiv": {
      "title": "S",
      "GlossList": {
        "GlossEntry": {
          "ID": "SGML",
          "SortAs": "SGML",
          "GlossTerm": "Standard Generalized Markup Language",
          "Acronym": "SGML",
          "Abbrev": "ISO 8879:1986",
          "GlossDef": {
            "para": "A meta-markup language, used to create markup languages such as DocBook.",
            "GlossSeeAlso": ["GML", "XML"]
          },
          "GlossSee": "Markup"
        }
      }
    }
  }
}

```

File GlossaryItem2211104012.cs

```
1  using System;
2  using System.IO;
3  using System.Text.Json;
4
5  0 references
6  class GlossaryItem2211104012
7  {
8      1 reference
9      class GlossDef
10     {
11         1 reference
12         public string para { get; set; }
13         1 reference
14         public string[] GlossSeeAlso { get; set; }
15     }
16
17     2 references
18     class GlossEntry
19     {
20         1 reference
21         public string ID { get; set; }
22         1 reference
23         public string SortAs { get; set; }
24         1 reference
25         public string GlossTerm { get; set; }
26         1 reference
27         public string Acronym { get; set; }
28         1 reference
29         public string Abbrev { get; set; }
30         2 references
31         public GlossDef GlossDef { get; set; }
32         1 reference
33         public string GlossSee { get; set; }
34     }
35
36     1 reference
37     class GlossList
38     {
39         1 reference
40         public GlossEntry GlossEntry { get; set; }
41     }
42
43     }
44
45     1 reference
46     class GlossDiv
47     {
48         0 references
49         public string title { get; set; }
50         1 reference
51         public GlossList GlossList { get; set; }
52     }
53
54     1 reference
55     class Glossary
56     {
57         0 references
58         public string title { get; set; }
59         1 reference
60         public GlossDiv GlossDiv { get; set; }
61     }
62
63     2 references
64     class Root
65     {
66         1 reference
67         public Glossary glossary { get; set; }
68     }
69
70     0 references
71     public static void ReadJSON()
72     {
73         string filePath = "jurnal7_3_2211104012.json";
74
75         if (!File.Exists(filePath))
76         {
77             Console.WriteLine("File JSON tidak ditemukan.");
78             return;
79         }
80
81         // Membaca dan parsing JSON
82         Root data = JsonSerializer.Deserialize<Root>(File.ReadAllText(filePath));
83     }
```

```

58
59 // Mengambil bagian GlossEntry
60 GlossEntry entry = data.glossary.GlossDiv.GlossList.GlossEntry;
61
62 Console.WriteLine("==== Glossary Entry =====");
63 Console.WriteLine($"ID : {entry.ID}");
64 Console.WriteLine($"SortAs : {entry.SortAs}");
65 Console.WriteLine($"Term : {entry.GlossTerm}");
66 Console.WriteLine($"Acronym : {entry.Acronym}");
67 Console.WriteLine($"Abbrev : {entry.Abbrev}");
68 Console.WriteLine($"Definition: {entry.GlossDef.para}");
69 Console.WriteLine("See Also : ");
70 Console.WriteLine(string.Join(", ", entry.GlossDef.GlossSeeAlso));
71 Console.WriteLine($"GlossSee : {entry.GlossSee}");
72
73

```

File Program.cs

```

0 references
class Program
{
    0 references
    static void Main(string[] args)
    {
        //DataMahasiswa2211104012.ReadJSON();
        //TeamMembers2211104012.ReadJSON();
        GlossaryItem2211104012.ReadJSON();
    }
}

```

Hasil run :

```

===== Glossary Entry =====
ID      : SGML
SortAs   : SGML
Term     : Standard Generalized Markup Language
Acronym  : SGML
Abbrev   : ISO 8879:1986
Definition: A meta-markup language, used to create markup languages such as DocBook.
See Also : GML, XML

```

### Penjelasan :

Tujuan dari program ini adalah untuk membaca dan menampilkan informasi dari file JSON bernama jurnal7\_3\_2211104012.json, yang berisi data tentang glossary atau daftar istilah. Struktur utama dalam file JSON ini terdiri dari elemen "glossary", yang di dalamnya terdapat "GlossDiv", lalu dilanjutkan dengan "GlossList", dan akhirnya "GlossEntry" yang memuat informasi penting seperti ID, GlossTerm, Acronym, Abbreviation, Definition, serta referensi terkait lainnya.

Program ini membaca file JSON tersebut menggunakan metode parsing JSON, yaitu proses mengonversi data dalam bentuk teks JSON menjadi objek dalam bahasa C# yang bisa digunakan langsung di dalam kode program. Di dalam file GlossaryItem2211104012.cs, terdapat sejumlah kelas yang dibuat untuk mencerminkan struktur JSON tersebut.

Melalui method ReadJSON(), program membaca isi file dengan File.ReadAllText() dan melakukan parsing menggunakan JsonSerializer.Deserialize<Root>(). Setelah

proses parsing selesai, program mengambil bagian GlossEntry dari objek hasil deserialisasi, lalu mencetak isinya ke console dengan tampilan yang lebih terstruktur dan mudah dipahami.

Penggunaan teknik parsing ini mempermudah pengambilan informasi dari elemen-elemen JSON seperti ID, GlossTerm, Acronym, serta daftar referensi yang tercantum dalam GlossSeeAlso, tanpa perlu membaca isi file secara manual.

## 6. 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

```
ASUS@DESKTOP-I50G73S MINGW64 /d/School/KPL_NadiaPutriRahmaniar_2211104012 (master)
$ git commit -m "menambahkan class GlossaryItem2211104012"
[master 93ac75e] menambahkan class GlossaryItem2211104012
 2 files changed, 95 insertions(+)
 create mode 100644 Praktik/07_Grammar-Based_Input_Processing_Parsing/Jurnal/GlossaryItem2211104012.cs
 create mode 100644 Praktik/07_Grammar-Based_Input_Processing_Parsing/Jurnal/jurnal7_3_2211104012.json

ASUS@DESKTOP-I50G73S MINGW64 /d/School/KPL_NadiaPutriRahmaniar_2211104012 (master)
$ git push origin master
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 1.35 KiB | 1.35 MiB/s, done.
Total 7 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/nadiaputrirah/KPL_NadiaPutriRahmaniar_2211104012.git
 029c368..93ac75e  master -> master
```