

**TUGAS JURNAL
MODUL 7**



Disusun Oleh :

Izzaty Zahara Br Barus – 23111040452

Kelas :

SE-07-02

Dosen :

Yudha Islami Sulistya

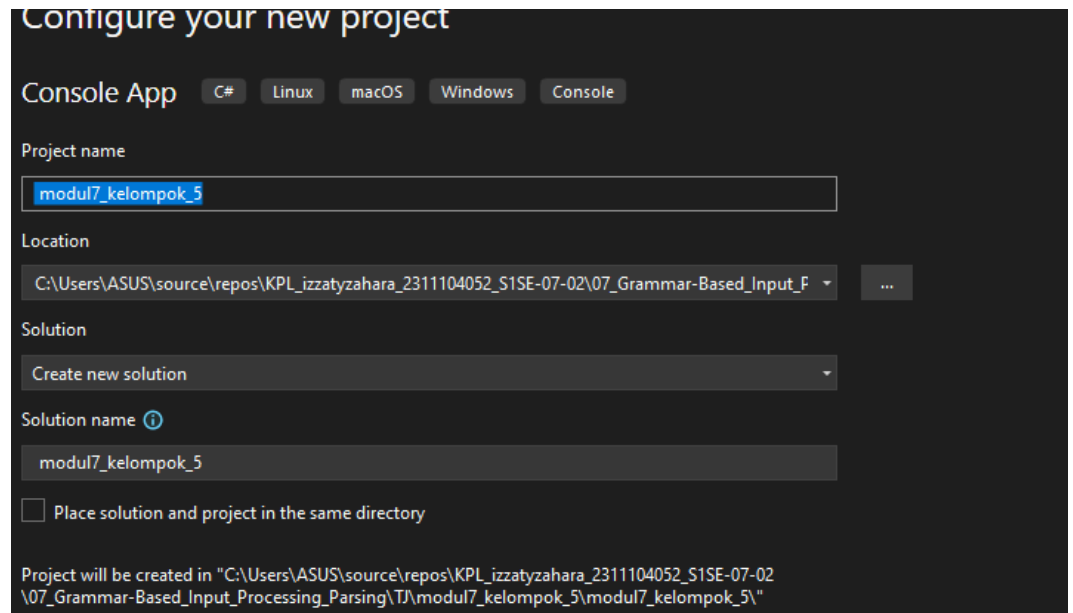
**PROGRAM STUDI SOFTWARE ENGINEERING
DIREKTORAT KAMPUS PURWOKERTO
TELKOM UNIVERSITY
PURWOKERTO
2025**

I. Link Github

- https://github.com/Izzaaaaaaaaa/KPL_izzatyzahara_2311104052_S1SE-07-02.git

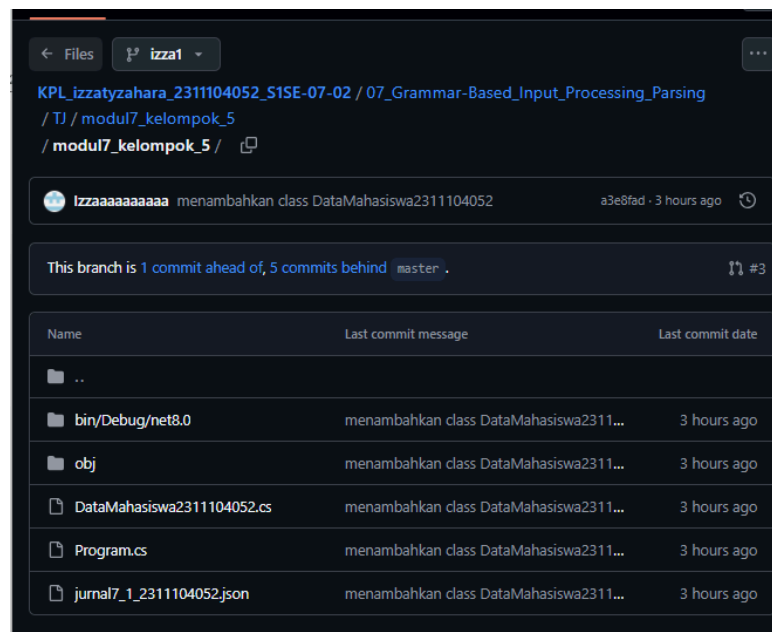
II. Penjelasan

1. Membuat New Project

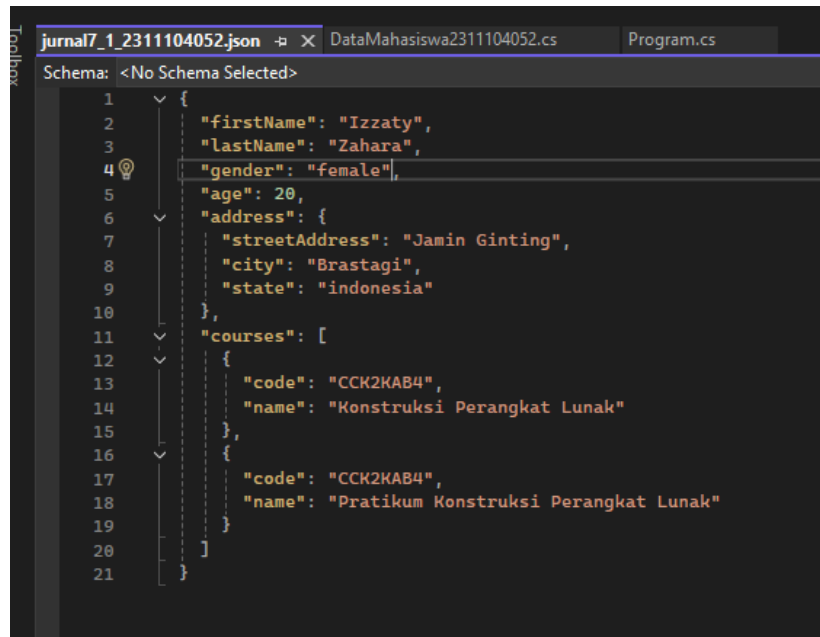


2. Menambah JSON Deserialization 1

a. Membuat Branch

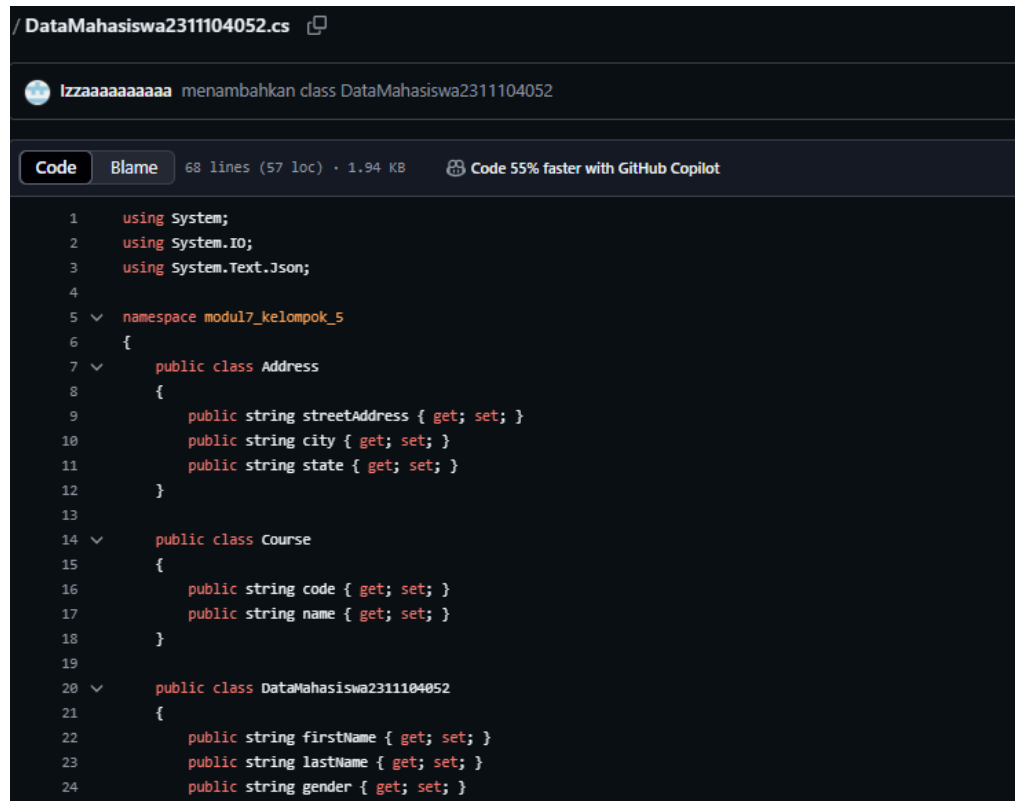


b. Menambahkan file JSON 1



```
jurnal7_1_2311104052.json  DataMahasiswa2311104052.cs  Program.cs
Schema: <No Schema Selected>
1  {
2    "firstName": "Izzaty",
3    "lastName": "Zahara",
4    "gender": "female",
5    "age": 20,
6    "address": {
7      "streetAddress": "Jamin Ginting",
8      "city": "Brastagi",
9      "state": "indonesia"
10   },
11   "courses": [
12     {
13       "code": "CCK2KAB4",
14       "name": "Konstruksi Perangkat Lunak"
15     },
16     {
17       "code": "CCK2KAB4",
18       "name": "Pratikum Konstruksi Perangkat Lunak"
19     }
20   ]
21 }
```

- c. Membuat Class baru dengan nama
“DataMahasiswa2311104052”



```
/DataMahasiswa2311104052.cs
Izzaaaaaaa menambahkan class DataMahasiswa2311104052

Code Blame 68 lines (57 loc) • 1.94 KB Code 55% faster with GitHub Copilot

1  using System;
2  using System.IO;
3  using System.Text.Json;
4
5  namespace modul7_kelompok_5
6  {
7      public class Address
8      {
9          public string streetAddress { get; set; }
10         public string city { get; set; }
11         public string state { get; set; }
12     }
13
14     public class Course
15     {
16         public string code { get; set; }
17         public string name { get; set; }
18     }
19
20     public class DataMahasiswa2311104052
21     {
22         public string firstName { get; set; }
23         public string lastName { get; set; }
24         public string gender { get; set; }
```

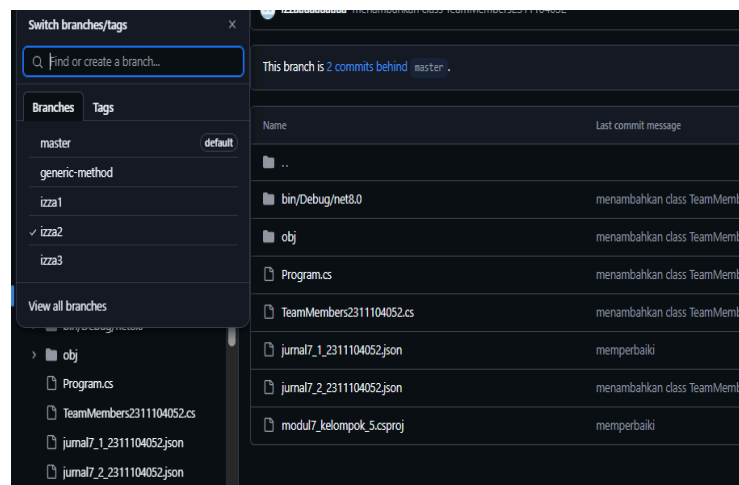
- d. Membuat method “ReadJSON() yang melakukan parsing untuk
file tersebut menjadi object sesuai.



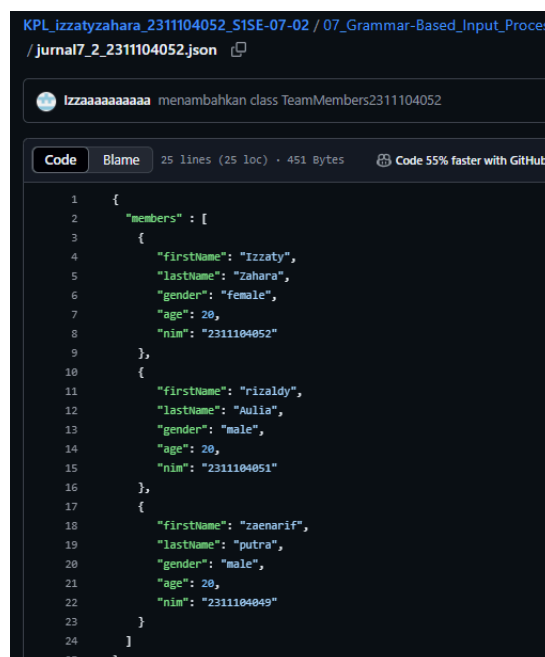
```
1 using System;
2 using modul7_kelompok_5;
3
4 class Program
5 {
6     static void Main(string[] args)
7     {
8         Console.WriteLine("=== [1] Menampilkan Parsing Data Mahasiswa ===");
9         var dataMahasiswa = new DataMahasiswa2311104052();
10        dataMahasiswa.ReadJSON();
11        Console.ReadLine();
12    }
13 }
```

3. Menambah JSON Deserialization 2

a. Membuat Branch

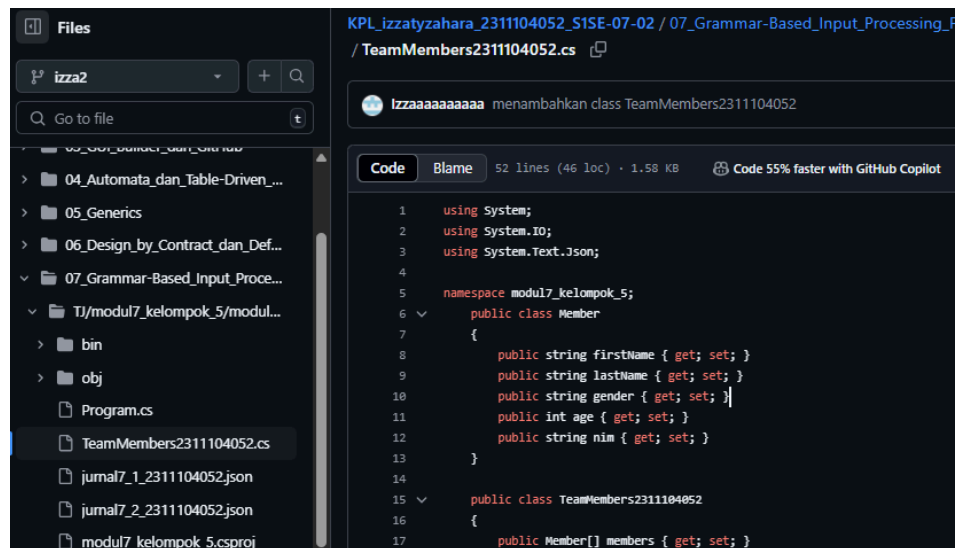


b. Menambahkan file JSON 2

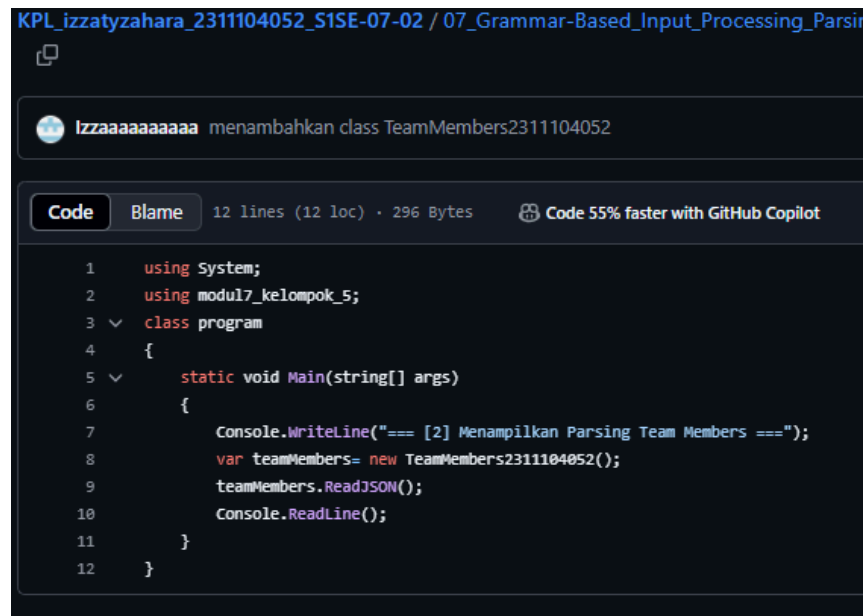


```
1 {
2   "members" : [
3     {
4       "firstName": "Izzaty",
5       "lastName": "Zahara",
6       "gender": "female",
7       "age": 20,
8       "nim": "2311104052"
9     },
10    {
11      "firstName": "rizaldy",
12      "lastName": "Aulia",
13      "gender": "male",
14      "age": 20,
15      "nim": "2311104051"
16    },
17    {
18      "firstName": "zaenarif",
19      "lastName": "putra",
20      "gender": "male",
21      "age": 20,
22      "nim": "2311104049"
23    }
24  ]
25 }
```

c. Membuat Class baru dengan nama “menambahkan class TeamMembers2311104052”

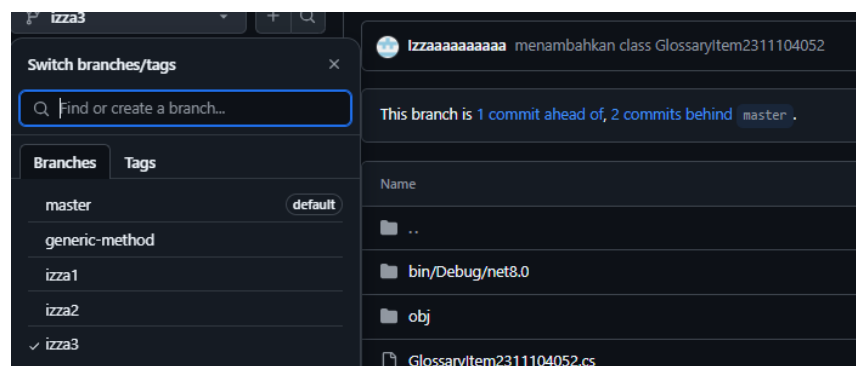


- d. Membuat method “ReadJSON()” yang melakukan parsing untuk file tersebut menjadi object sesuai.

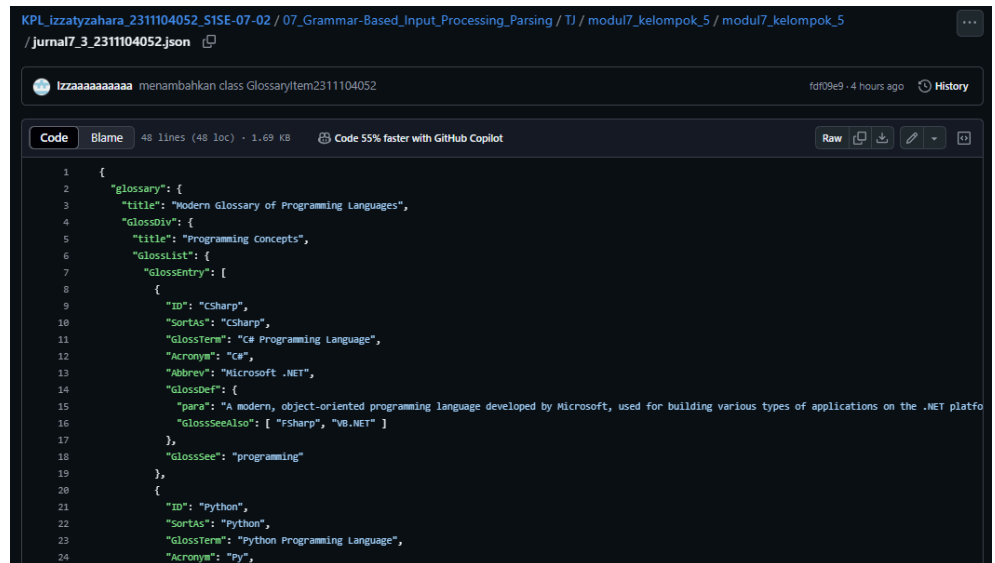


4. Menambah JSON Deserialization 3

- a. Membuat Branch



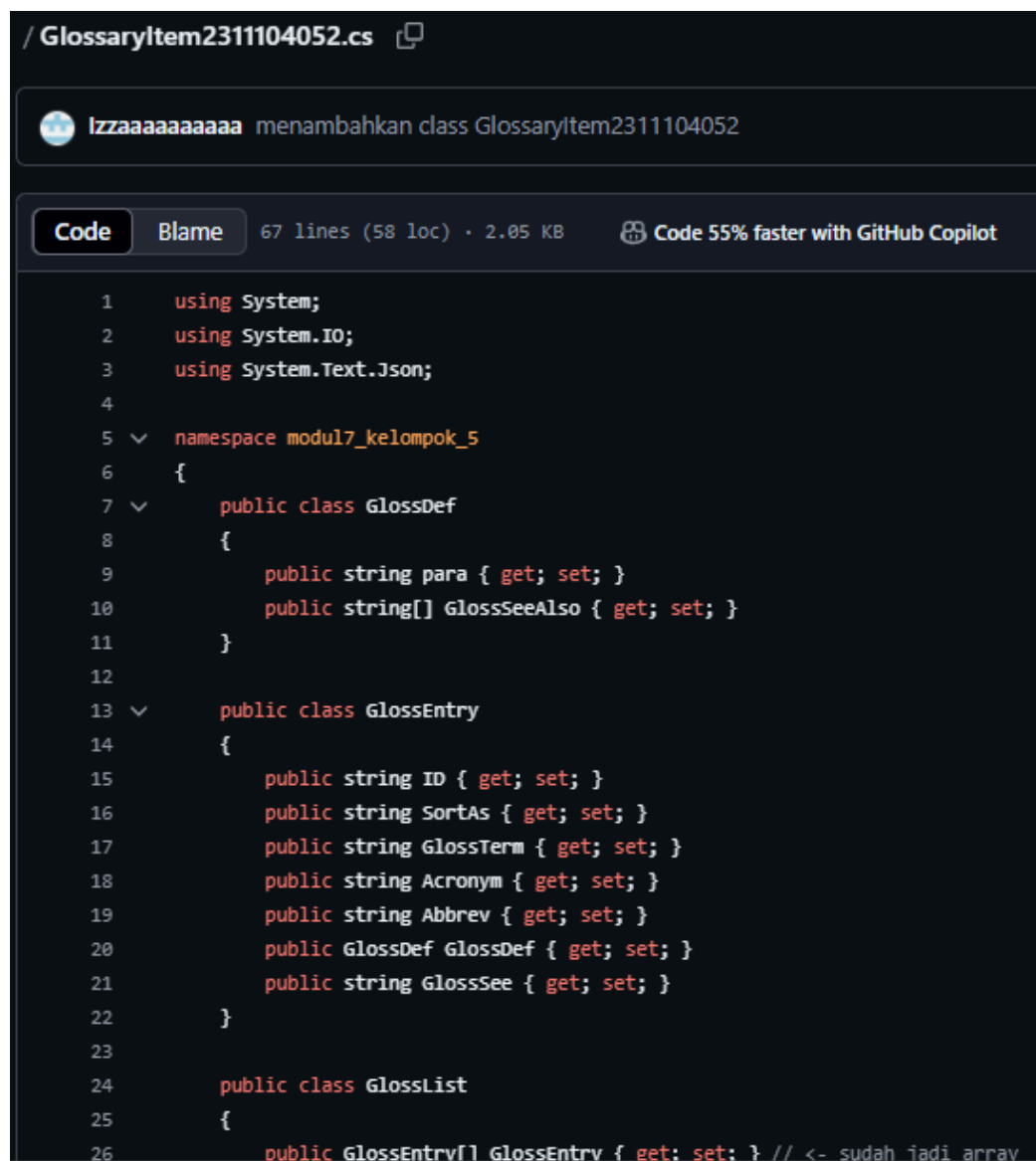
b. Menambahkan file JSON 3



The screenshot shows a code editor with a file named `jurnal7_3_2311104052.json`. The file contains a JSON object with a `"glossary"` property. This property is an object with `"title"` and `"glossList"` properties. The `"glossList"` property is an array of `"GlossEntry"` objects. The first entry is for C#, and the second is for Python. The `"GlossEntry"` objects have properties like `"ID"`, `"SortAs"`, `"GlossTerm"`, `"Acronym"`, `"Abbrev"`, `"GlossDef"` (which includes `"para"` and `"GlossSeeAlso"`), and `"GlossSee"`.

```
1 {
2   "glossary": {
3     "title": "Modern Glossary of Programming Languages",
4     "glossList": [
5       {
6         "title": "Programming Concepts",
7         "GlossEntry": {
8           {
9             "ID": "CSharp",
10            "SortAs": "Csharp",
11            "GlossTerm": "C# Programming Language",
12            "Acronym": "Ca",
13            "Abbrev": "Microsoft .NET",
14            "GlossDef": {
15              "para": "A modern, object-oriented programming language developed by Microsoft, used for building various types of applications on the .NET platfo
16              "GlossSeeAlso": [ "FSharp", "VB.NET" ]
17            },
18            "GlossSee": "programming"
19          },
20          {
21            "ID": "Python",
22            "SortAs": "Python",
23            "GlossTerm": "Python Programming Language",
24            "Acronym": "Py",
```

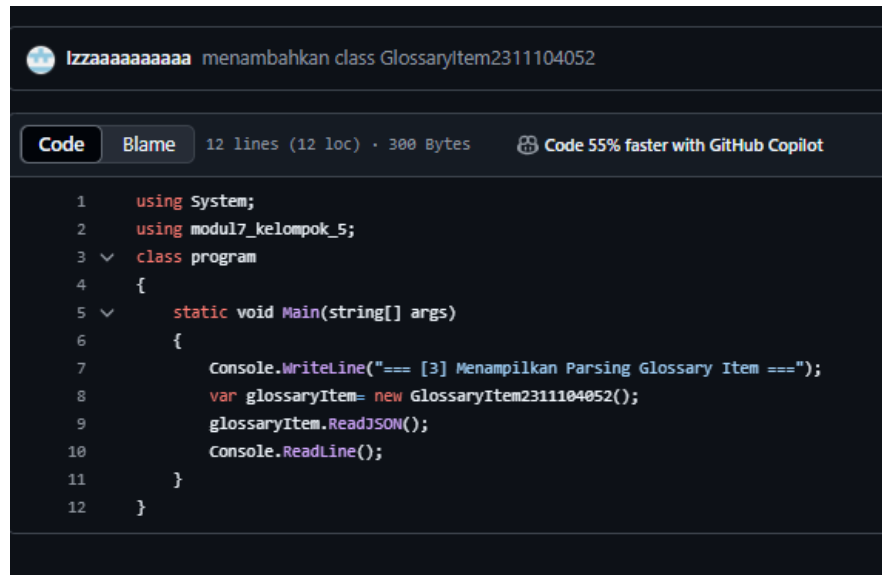
c. Membuat Class baru dengan nama “menambahkan class GlossaryItem2311104052”



The screenshot shows a code editor with a file named `GlossaryItem2311104052.cs`. The file contains a C# class definition. It starts with `using` statements for `System`, `System.IO`, and `System.Text.Json`. The `namespace` is `modul7_kelompok_5`. Inside the namespace, there are three public classes: `GlossDef`, `GlossEntry`, and `GlossList`. `GlossDef` has `para` and `GlossSeeAlso` properties. `GlossEntry` has `ID`, `SortAs`, `GlossTerm`, `Acronym`, `Abbrev`, `GlossDef`, and `GlossSee` properties. `GlossList` has a `GlossEntry[]` property.

```
1 using System;
2 using System.IO;
3 using System.Text.Json;
4
5 namespace modul7_kelompok_5
6 {
7     public class GlossDef
8     {
9         public string para { get; set; }
10        public string[] GlossSeeAlso { get; set; }
11    }
12
13    public class GlossEntry
14    {
15        public string ID { get; set; }
16        public string SortAs { get; set; }
17        public string GlossTerm { get; set; }
18        public string Acronym { get; set; }
19        public string Abbrev { get; set; }
20        public GlossDef GlossDef { get; set; }
21        public string GlossSee { get; set; }
22    }
23
24    public class GlossList
25    {
26        public GlossEntry[] GlossEntry { get; set; } // <- sudah jadi array
```

- d. Membuat method “ReadJSON() yang melakukan parsing untuk file tersebut menjadi object sesuai.

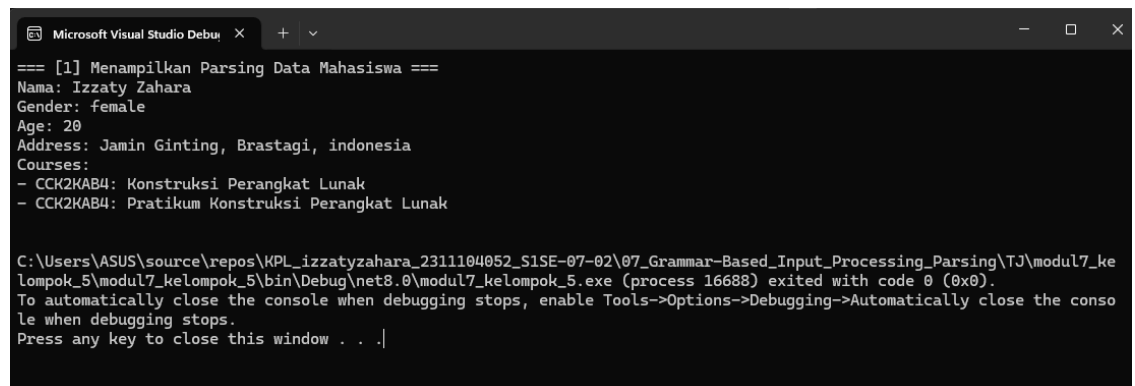


The screenshot shows a GitHub commit interface. At the top, it says "Izzaaaaaaaa menambahkan class GlossaryItem2311104052". Below this, there are tabs for "Code" and "Blame". The "Code" tab is selected, showing a C# code snippet. The code is as follows:

```
1 using System;
2 using modul7_kelompok_5;
3 class program
4 {
5     static void Main(string[] args)
6     {
7         Console.WriteLine("=== [3] Menampilkan Parsing Glossary Item ===");
8         var glossaryItem= new GlossaryItem2311104052();
9         glossaryItem.ReadJSON();
10        Console.ReadLine();
11    }
12 }
```

III. Hasil Running

1. Hasil Running JSON1

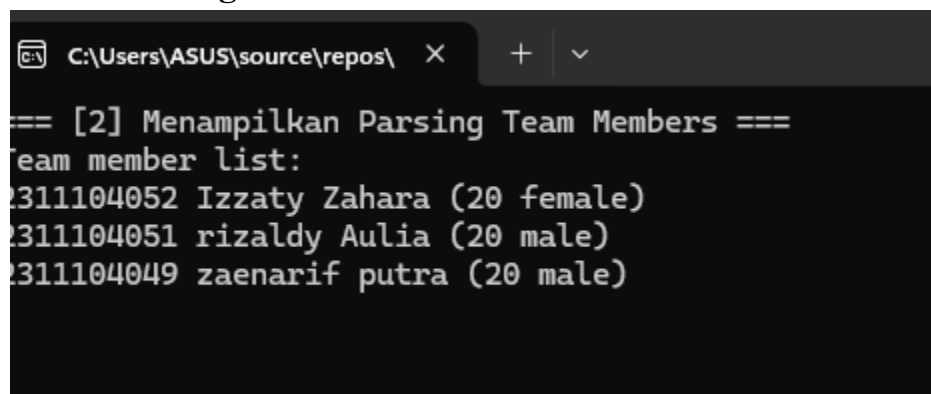


The screenshot shows the Microsoft Visual Studio Debug console window. The output is as follows:

```
=== [1] Menampilkan Parsing Data Mahasiswa ===
Nama: Izzaty Zahara
Gender: female
Age: 20
Address: Jamin Ginting, Brastagi, Indonesia
Courses:
- CCK2KAB4: Konstruksi Perangkat Lunak
- CCK2KAB4: Pratikum Konstruksi Perangkat Lunak

C:\Users\ASUS\source\repos\KPL_izzatyzahara_2311104052_SISE-07-02\07_Grammar-Based_Input_Processing_Parsing\TJ\modul7_ke
lompok_5\modul7_kelompok_5\bin\Debug\net8.0\modul7_kelompok_5.exe (process 16688) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .|
```

2. Hasil Running JSON 2



The screenshot shows a command prompt window. The output is as follows:

```
=== [2] Menampilkan Parsing Team Members ===
Team member list:
2311104052 Izzaty Zahara (20 female)
2311104051 rizaldy Aulia (20 male)
2311104049 zaenarif putra (20 male)
```

3. Hasil Running JSON 3

```
C:\Users\ASUS\source\repos\ x + v
=== [3] Menampilkan Parsing Glossary Item ===
GlossEntry List:
-----
ID: CSharp
SortAs: CSharp
GlossTerm: C# Programming Language
Acronym: C#
Abbrev: Microsoft .NET
Definition: A modern, object-oriented programming language developed by Microsoft, used for building various types of applications on the .NET platform.
See Also: FSharp, VB.NET
See: programming
-----
ID: Python
SortAs: Python
GlossTerm: Python Programming Language
Acronym: Py
Abbrev: Python 3.x
Definition: A high-level, interpreted programming language known for its readability and versatility, widely used in AI, data science, web development, and more.
See Also: NumPy, Django
See: programming
-----
ID: Java
SortAs: Java
GlossTerm: Java Programming Language
Acronym: Java
Abbrev: Java SE
Definition: A class-based, object-oriented language that is designed to have as few implementation dependencies as possible, commonly used for enterprise applications.
See Also: JVM, Kotlin
See: programming
-----
```

IV. Kesimpulan

Membuat program menggunakan bahasa pemrograman C# yang mampu melakukan deserialisasi data dari file JSON ke dalam objek adalah tujuan utama dari kegiatan ini. Kegiatan ini dirancang agar praktikan dapat memahami konsep dasar pemrosesan data berformat JSON, terutama dalam konteks pemrograman berorientasi objek serta penggunaan struktur data kompleks seperti objek bersarang dan daftar. Proyek dimulai dengan pembuatan proyek baru yang dinamakan modul7_kelompok_<nomor_kelompok> di Visual Studio, yang kemudian dihubungkan dengan Git dan GitHub untuk mendukung kolaborasi tim melalui sistem pengendalian versi.

Setelah repository berhasil dibuat, praktikan diminta untuk membuat cabang baru berdasarkan nama panggilan masing-masing. File jurnal7_3_<NIM>.json diunduh, diubah namanya sesuai dengan NIM praktikan, dan disimpan di dalam folder proyek. Praktikan juga diminta untuk memperbarui data JSON tersebut dengan informasi baru sesuai instruksi (misalnya daftar istilah pemrograman), lalu membuat kelas model bernama GlossaryItem<NIM> yang strukturnya disesuaikan dengan isi file JSON. Dengan memanfaatkan System.Text.Json, praktikan membuat metode bernama ReadJSON() yang berfungsi untuk membaca isi file JSON, melakukan deserialisasi, dan menampilkannya di konsol dengan format yang rapi dan mudah dipahami.

Selanjutnya, praktikan melakukan commit terhadap perubahan kode di cabang masing-masing, kemudian melakukan push ke GitHub dan membuat pull request ke cabang utama. Proses ini mencerminkan praktik profesional dalam pengembangan perangkat lunak, di mana setiap perubahan dikembangkan dalam cabang terpisah dan ditinjau sebelum digabungkan ke repository utama.

Melalui tugas ini, praktikan tidak hanya belajar memahami struktur JSON dan cara mengubahnya menjadi objek di C#, tetapi juga terbiasa menggunakan alat profesional seperti Git dan GitHub untuk mendukung kerja kolaboratif dan pengelolaan kode yang efektif.