

**TUGAS JURNAL
MODUL 8**



Disusun Oleh :

Pradana Argo Pangestu – 2311104079

Kelas :

SE-07-02

Dosen :

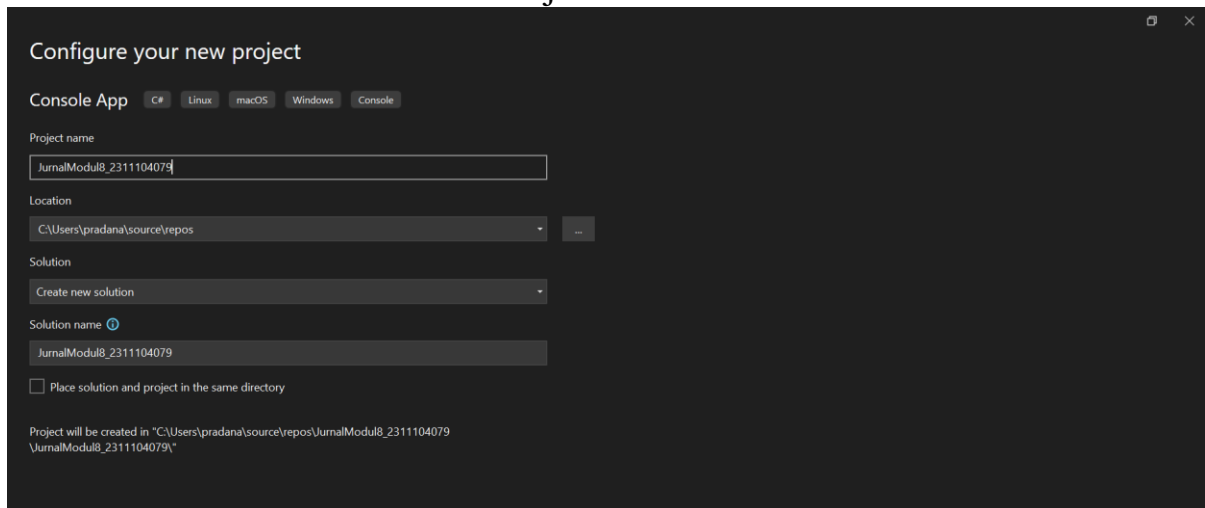
Yudha Islami Sulistya

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

I. Link Github

II. Penjelasan

1. Membuat New Project



2. Menambah JSON Deserialization 1

a. Membuat File JSON “bank_transfer_config.json”.

```
1. {  
2.   "lang": "en", "transfer": {  
3.     "threshold": 25000000,  
4.     "low_fee": 6500,  
5.     "high_fee": 15000  
6.   },  
7.   "methods": [ "RTO (real-time)", "SKN", "RTGS", "BI FAST" ],  
8.   "confirmation": { "en": "yes",  
9.     "id": "ya"  
10.  }  
11. }
```

Penjelasan Singkat:

File `bank_transfer_config.json` adalah file konfigurasi utama untuk aplikasi. File ini menyimpan informasi penting yang digunakan secara dinamis oleh program saat dijalankan, seperti

pilihan bahasa (lang), batas dan biaya transfer (transfer), metode transfer yang tersedia (methods), serta kata konfirmasi dalam dua bahasa (confirmation). Dengan adanya file ini, program dapat beradaptasi tanpa perlu mengubah kode, cukup dengan mengedit isi file konfigurasi.

- b. Membuat Class baru dengan nama
“BankTransferConfig.cs”

```
1. using System.Text.Json;
2.
3. public class BankTransferConfig
4. {
5.     public string lang { get; set; } public Transfer transfer { get; set; }
6.     public List<string> methods { get; set; } public Confirmation confirmation
       { get; set; }
7. }
8.
9. public static BankTransferConfig Load(string path)
10. {
11.     if (!File.Exists(path))
12.     {
13.         var defaultConfig = new BankTransferConfig
14.         {
15.             lang = "en",
16.             transfer = new Transfer { threshold = 25000000, low_fee = 6500,
               high_fee = 15000
17.         },
18.             methods = new List<string> { "RTO (real-time)", "SKN", "RTGS", "BI
               FAST" },
19.             confirmation = new Confirmation { en = "yes", id = "ya" }
20.         };
21.
22.         var json = JsonSerializer.Serialize(defaultConfig, new
               JsonSerializerOptions { WriteIndented = true
23.         });
24.         File.WriteAllText(path, json); return defaultConfig;
25.     }
26.     else
27.     {
28.         string json = File.ReadAllText(path); return
               JsonSerializer.Deserialize<banktransferconfi g="">(json);
29.     }
30. }
31. }
32. }
33. public class Transfer
34. {
```

```

35.     public int threshold { get; set; } public int low_fee { get; set; }
        public int high_fee { get; set; }
36.     }
37.     public class Confirmation
38.     {
39.         public string en { get; set; } public string id { get; set; }
40.     }
41.     </banktransferconfi></string></string>

```

Penjelasan Singkat:

Kelas BankTransferConfig berfungsi sebagai representasi dari struktur data konfigurasi JSON yang digunakan dalam program. Kelas ini memiliki properti seperti lang (bahasa), transfer (objek biaya transfer), methods (daftar metode transfer), dan confirmation (objek konfirmasi transaksi). Metode Load() di dalam kelas ini berfungsi untuk memuat konfigurasi dari file bank_transfer_config.json, dan jika file tersebut tidak ada, program akan otomatis membuat file konfigurasi baru dengan nilai default. Dengan cara ini, program dapat menggunakan konfigurasi runtime tanpa perlu mengubah kode sumber setiap kali ada perubahan konfigurasi.

c. Mengisi code class program “program .cs”.

```

1. class Program
2. {
3.     static void Main()
4.     {
5.         var config = BankTransferConfig.Load("bank_transfer_config.j son");
6.
7.         string promptAmount = config.lang == "en"
8.         ? "Please insert the amount of money to transfer:"
9.         : "Masukkan jumlah uang yang akan di-transfer:";
10.
11.         Console.WriteLine(promptAmount);
12.         int amount = int.Parse(Console.ReadLine());

```

```

13.
14.     int fee = amount <= config.transfer.threshold
15.     ? config.transfer.low_fee
16.     : config.transfer.high_fee; int total = amount + fee;
17.     if (config.lang == "en")
18.     {
19.
20.
21.         fee); total);
22.
23.         Console.WriteLine("Transfer fee = " + Console.WriteLine("Total
amount = " +
24.         Console.WriteLine("Select transfer
25.
26.         method:");
27.     }
28.     else
29.     {
30.         Console.WriteLine("Biaya transfer = " +
31.         fee);
32.         Console.WriteLine("Total biaya = " +
33.         total);
34.         Console.WriteLine("Pilih metode transfer:");
35.     }
36.     for (int i = 0; i < config.methods.Count; i++)
37.     {
38.         Console.WriteLine($"{i + 1}.
39.         {config.methods[i]}");
40.     }
41.     Console.ReadLine(); // pilihan metode string confirmMsg =
config.lang == "en"
42.     ? $"Please type
43.     \"{config.confirmation.en}\" to confirm the transaction:"
44.     : $"Ketik \"{config.confirmation.id}\" untuk mengkonfirmasi
transaksi:";
45.     Console.WriteLine(confirmMsg); string input = Console.ReadLine();
46.     bool confirmed = (config.lang == "en" && input ==
config.confirmation.en)
47.     || (config.lang == "id" && input == config.confirmation.id);
48.     string result = config.lang == "en"
49.     ? confirmed ? "The transfer is completed" : "Transfer is cancelled"
50.     : confirmed ? "Proses transfer berhasil" : "Transfer dibatalkan";
51.
52.     Console.WriteLine(result);
53.     }
54.     }

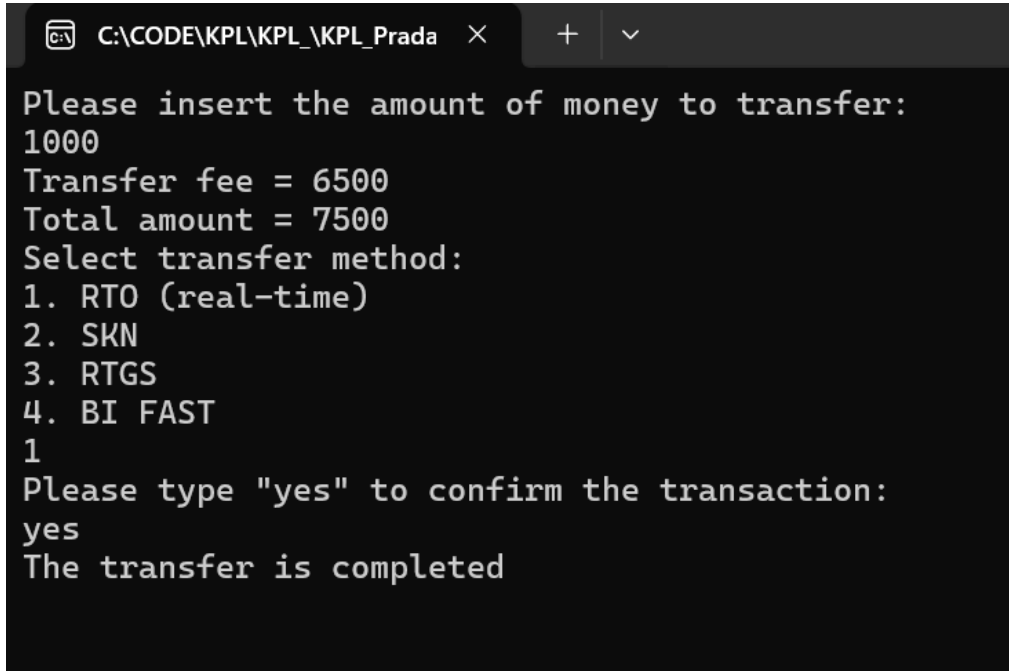
```

Penjelasan Singkat :

Kelas Program merupakan kelas utama dalam aplikasi konsol ini yang mengelola seluruh alur logika program. Di dalam metode Main(), aplikasi pertama-tama memuat file konfigurasi JSON Melalui BankTransferConfig.Load(). Berdasarkan bahasa (lang) yang ada dalam konfigurasi, program akan menampilkan prompt input kepada pengguna dalam Bahasa Indonesia atau Inggris. Selanjutnya, pengguna diminta untuk memasukkan jumlah uang yang ingin ditransfer, dan program akan menghitung biaya transfer berdasarkan nilai ambang batas (threshold) serta menampilkan total biaya. Program kemudian akan menunjukkan daftar metode transfer, meminta konfirmasi dari pengguna, dan memberikan hasil apakah proses transfer berhasil atau dibatalkan sesuai dengan input konfirmasi yang diberikan.

Hasil Running

1. Hasil Running



```
C:\CODE\KPL\KPL_KPL_Prada x + v

Please insert the amount of money to transfer:
1000
Transfer fee = 6500
Total amount = 7500
Select transfer method:
1. RTO (real-time)
2. SKN
3. RTGS
4. BI FAST
1
Please type "yes" to confirm the transaction:
yes
The transfer is completed
```

III. Kesimpulan

Melalui pelaksanaan tugas ini, kita dapat menyimpulkan bahwa teknik konfigurasi runtime memberikan fleksibilitas yang tinggi dalam pengembangan aplikasi. Dengan memisahkan konfigurasi dari kode program melalui file `bank_transfer_config.json`, aplikasi menjadi lebih dinamis dan dapat menyesuaikan alur serta output sesuai kebutuhan pengguna tanpa harus mengubah logika utama. Selain itu, pembagian program ke dalam kelas-kelas terstruktur seperti `BankTransferConfig`, `Transfer`, dan `Confirmation` juga berkontribusi pada keterbacaan, pemeliharaan kode, serta mendukung praktik pemrograman modular. Secara keseluruhan, tugas ini tidak hanya mengajarkan penerapan konfigurasi runtime dalam C#, tetapi juga memperdalam pemahaman tentang konsep OOP, struktur JSON, serta penggunaan Git dan GitHub dalam proyek nyata.