

Nama : Dimastian Aji Wibowo

NIM : 2311104058

JURNAL MODUL 8

Hasil Run

```
Microsoft Visual Studio Debu x + v
Masukkan jumlah uang yang akan di-transfer:
2000000
Biaya transfer = 5000
Total biaya = 2005000
Pilih metode transfer:
1. RTO (real-time)
2. SKN
3. RTGS
4. BI FAST
4
Ketik "ya" untuk mengkonfirmasi transaksi:
ya
Proses transfer berhasil

D:\College\Semester 4\PrakKPL\KPL_Dimastian_Aji_Wibowo_2311104058_SE07-02\08_Runtime_Configuration_dan_Internationalizat
ion\JURNAL\modul8_2311104058\modul8_2311104058\bin\Debug\net8.0\modul8_2311104058.exe (process 3200) 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 . . .|
```

```
Microsoft Visual Studio Debu x + v
Please insert the amount of money to transfer:
5000000
Transfer fee = 5000
Total amount = 5005000
Select transfer method:
1. RTO (real-time)
2. SKN
3. RTGS
4. BI FAST
4
Please type "yes" to confirm the transaction:
yes
The transfer is completed

D:\College\Semester 4\PrakKPL\KPL_Dimastian_Aji_Wibowo_2311104058_SE07-02\08_Runtime_Configuration_dan_Internationalizat
ion\JURNAL\modul8_2311104058\modul8_2311104058\bin\Debug\net8.0\modul8_2311104058.exe (process 7464) 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 . . .|
```

```
Microsoft Visual Studio Debu x + v
Masukkan jumlah uang yang akan di-transfer:
500000
Biaya transfer = 2500
Total biaya = 502500
Pilih metode transfer:
1. RTO (real-time)
2. SKN
3. RTGS
4. BI FAST
1
Ketik "ya" untuk mengkonfirmasi transaksi:
ya
Proses transfer berhasil

D:\College\Semester 4\PrakKPL\KPL_Dimastian_Aji_Wibowo_2311104058_SE07-02\08_Runtime_Configuration_dan_Internationalizat
ion\JURNAL\modul8_2311104058\modul8_2311104058\bin\Debug\net8.0\modul8_2311104058.exe (process 9904) 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 . . .
```

Source Code

BankTransferConfig.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Xml;
using System.IO;
using Newtonsoft.Json;

namespace modul8_2311104058
{
    public class BankTransferConfig
    {
        public string lang { get; set; }
        public Transfer transfer { get; set; }
        public List<string> methods { get; set; }
        public Confirmation confirmation { get; set; }

        private const string configFile = "bank_transfer_config.json";

        public BankTransferConfig()
        {
            if (File.Exists(configFile))
            {
                string json = File.ReadAllText(configFile);
                var configData =
                    JsonConvert.DeserializeObject<BankTransferConfigData>(json);

                this.lang = configData.lang;
                this.transfer = configData.transfer;
                this.methods = configData.methods;
                this.confirmation = configData.confirmation;
            }
            else
            {
                lang = "en";
                transfer = new Transfer
                {
                    threshold = 1000000,
                    low_fee = 2500,
                    high_fee = 5000
                };
                methods = new List<string> { "RTO (real-time)", "SKN", "RTGS", "BI FAST" };
                confirmation = new Confirmation { en = "yes", id = "ya" };

                var defaultData = new BankTransferConfigData
                {
                    lang = this.lang,
```

```

        transfer = this.transfer,
        methods = this.methods,
        confirmation = this.confirmation
    };

    string json = JsonConvert.SerializeObject(defaultData,
Newtonsoft.Json.Formatting.Indented);
    File.WriteAllText(configFile, json);
}
}

public class BankTransferConfigData
{
    public string lang { get; set; }
    public Transfer transfer { get; set; }
    public List<string> methods { get; set; }
    public Confirmation confirmation { get; set; }
}

public class Transfer
{
    public int threshold { get; set; }
    public int low_fee { get; set; }
    public int high_fee { get; set; }
}

public class Confirmation
{
    public string en { get; set; }
    public string id { get; set; }
}
}

```

Kelas utama BankTransferConfig bertugas untuk membaca konfigurasi dari file bank_transfer_config.json jika file tersebut sudah ada, atau membuat file dengan nilai default jika belum tersedia. Konfigurasi ini mencakup bahasa yang digunakan (lang), detail biaya transfer (Transfer), metode transfer (methods), serta pilihan konfirmasi dalam dua bahasa (Confirmation). Struktur data tambahan seperti BankTransferConfigData, Transfer, dan Confirmation dibuat sebagai representasi model data JSON untuk memudahkan proses serialisasi dan deserialisasi menggunakan pustaka Newtonsoft.Json. Pendekatan ini memungkinkan sistem memiliki fleksibilitas dalam mengelola pengaturan tanpa harus mengubah kode program secara langsung.

Program.cs

```

using modul8_2311104058;
using System;

class Program
{
    static void Main(string[] args)
    {
        BankTransferConfig config = new BankTransferConfig();
        string lang = config.lang;

        Console.WriteLine(lang == "en"
            ? "Please insert the amount of money to transfer:"
            : "Masukkan jumlah uang yang akan di-transfer:");
        int amount = int.Parse(Console.ReadLine());

        int fee = (amount <= config.transfer.threshold) ? config.transfer.low_fee :
config.transfer.high_fee;
        int total = amount + fee;

        if (lang == "en")
        {
            Console.WriteLine($"Transfer fee = {fee}");
            Console.WriteLine($"Total amount = {total}");
        }
        else
        {
            Console.WriteLine($"Biaya transfer = {fee}");
            Console.WriteLine($"Total biaya = {total}");
        }

        Console.WriteLine(lang == "en" ? "Select transfer method:" : "Pilih metode
transfer:");
        for (int i = 0; i < config.methods.Count; i++)
        {

```

```

        Console.WriteLine($"{i + 1}. {config.methods[i]}");
    }

    int methodChoice = int.Parse(Console.ReadLine());

    string confirmWord = lang == "en" ? config.confirmation.en :
config.confirmation.id;
    Console.WriteLine(lang == "en"
        ? $"Please type \"{confirmWord}\" to confirm the transaction:"
        : $"Ketik \"{confirmWord}\" untuk mengkonfirmasi transaksi:");
    string userInput = Console.ReadLine();

    if (userInput.Equals(confirmWord, StringComparison.OrdinalIgnoreCase))
    {
        Console.WriteLine(lang == "en" ? "The transfer is completed" : "Proses transfer
berhasil");
    }
    else
    {
        Console.WriteLine(lang == "en" ? "Transfer is cancelled" : "Transfer
dibatalakan");
    }
}
}

```

Program utama yang memanfaatkan kelas BankTransferConfig untuk menjalankan simulasi proses transfer uang dengan pengaturan dinamis berbasis file konfigurasi. Program ini meminta input jumlah uang dari pengguna, kemudian menghitung biaya transfer berdasarkan ambang batas (threshold) dan menentukan total biaya yang harus dibayarkan. Teks yang ditampilkan akan menyesuaikan bahasa yang dipilih (en untuk Inggris atau id untuk Indonesia). Pengguna juga diminta memilih metode transfer dari daftar yang disediakan, serta mengonfirmasi transaksi dengan mengetikkan kata tertentu sesuai bahasa yang digunakan. Validasi konfirmasi dilakukan secara case-insensitive, dan program akan menampilkan pesan sukses atau pembatalan transaksi berdasarkan input pengguna. Pendekatan ini menunjukkan bagaimana konfigurasi eksternal dapat digunakan untuk membuat aplikasi lebih fleksibel dan mudah disesuaikan tanpa perubahan langsung pada kode program.