

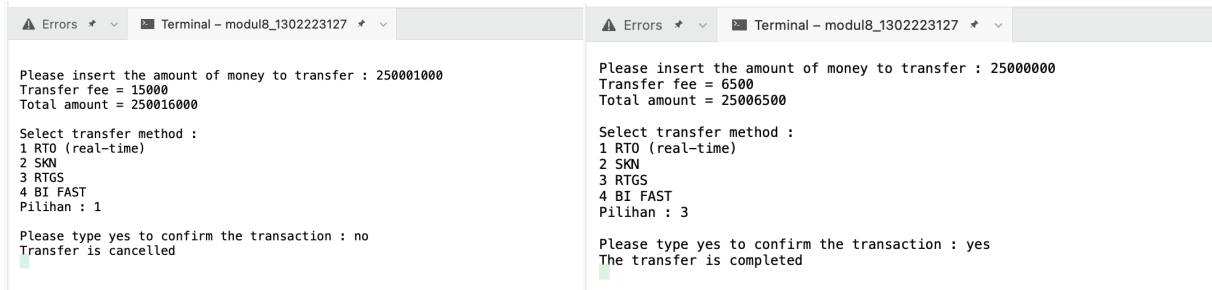
## Jurnal Modul 8 : Runtime Configuration

### Praktikum Konstruksi Perangkat Lunak

#### 1. Link Repository Github

<https://github.com/fauzeinmw2/jurnal-modul8-kpl-1302223127>

#### 2. Screenshot Hasil Program

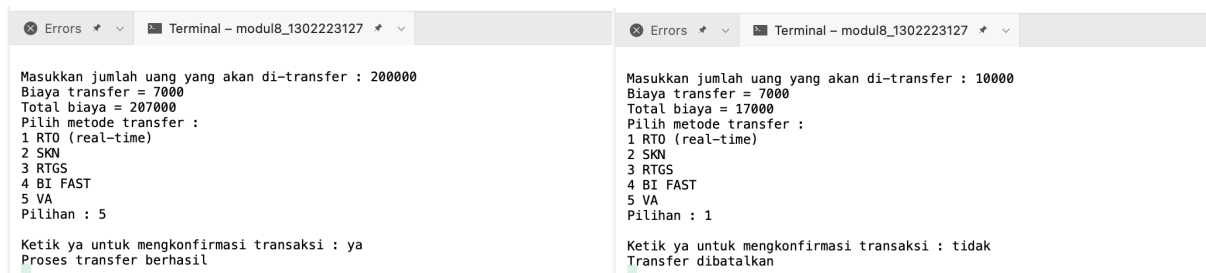


Jsonnya diubah

```
<No Schema Selected>

1  {
2    "lang": "id",
3    "transfer": {
4      "threshold": 1000000,
5      "low_fee": 7000,
6      "high_fee": 20000
7    },
8    "methods": [ "RTO (real-time)", "SKN", "RTGS", "BI FAST", "VA"],
9    "confirmation": {
10     "en": "yes",
11     "id": "ya"
12   }
13 }
```

Hasilnya



### 3. Penjelasan Kode

Program > Main(string[] args)

```

1  using System.Text.Json;
2
3  internal class Program
4  {
5      public class Transfer
6      {
7          public int threshold { set; get; }
8          public int low_fee { set; get; }
9          public int high_fee { set; get; }
10     }
11
12     public class Confirmation
13     {
14         public string en { set; get; }
15         public string id { set; get; }
16     }
17
18     public class BankTransferConfig
19     {
20         public string lang { set; get; }
21         public Transfer transfer { set; get; }
22         public List<String> methods { set; get; }
23         public Confirmation confirmation { set; get; }
24     }
25

```

- Membuat class untuk menampung masing-masing data dalam atribut file JSON

```

26     public class AppConfig
27     {
28         public BankTransferConfig config;
29         private const string fileConfigPath = "../../bank_transfer_config.json";
30
31         public AppConfig()
32         {
33             try
34             {
35                 ReadConfigFile();
36             }
37             catch
38             {
39                 SetDefault();
40                 WriteConfigFile();
41             }
42         }
43
44         public void ReadConfigFile()
45         {
46             string configJsonData = File.ReadAllText(fileConfigPath);
47             config = JsonSerializer.Deserialize<BankTransferConfig>(configJsonData);
48         }
49
50         public void WriteConfigFile()
51         {
52             JsonSerializerOptions options = new JsonSerializerOptions()
53             {
54                 WriteIndented = true
55             };
56
57             string data = JsonSerializer.Serialize(config);
58             File.WriteAllText(fileConfigPath, data);
59         }
60

```

- Membuat class AppConfig untuk melakukan konfigurasi. Membaca file config, membuat file config jika belum ada

```

61     public void SetDefault()
62     {
63         config = new BankTransferConfig();
64         config.lang = "en";
65         //config.transfer.threshold = 25000000;
66         config.transfer = new Transfer();
67         config.transfer.threshold = 25000000;
68         config.transfer.low_fee = 6500;
69         config.transfer.high_fee = 15000;
70         string[] dataMethod = { "RTO (real-time)", "SKN", "RTGS", "BI FAST" };
71         config.methods = new List<string>(dataMethod);
72         config.confirmation = new Confirmation();
73         config.confirmation.en = "yes";
74         config.confirmation.id = "ya";
75     }
76 }
77
78

```

- Menambahkan method untuk membuat data default untuk membuat fil config json ketika filnya belum ada

```

79     private static void Main(string[] args)
80     {
81         AppConfig cfg = new AppConfig();
82
83         if(cfg.config.lang == "en")
84         {
85             Console.Write("Please insert the amount of money to transfer : ");
86         }
87         else
88         {
89             Console.Write("Masukkan jumlah uang yang akan di-transfer : ");
90         }
91
92         int uang = int.Parse(Console.ReadLine());
93         int biaya = 0;
94
95         if(uang <= cfg.config.transfer.threshold)
96         {
97             biaya = cfg.config.transfer.low_fee;
98         }
99         else
100         {
101             biaya = cfg.config.transfer.high_fee;
102         }
103
104         if (cfg.config.lang == "en")
105         {
106             Console.WriteLine($"Transfer fee = {biaya}");
107             Console.WriteLine($"Total amount = {uang + biaya}");
108
109             Console.WriteLine("\nSelect transfer method : ");
110         }
111         else
112         {
113             Console.WriteLine($"Biaya transfer = {biaya}");
114             Console.WriteLine($"Total biaya = {uang + biaya}");
115             Console.WriteLine("Pilih metode transfer : ");
116         }
117

```

- Terakhir membuat main program dengan ketentuan yang sudah dijelaskan dalam soal
- Mulai dari input jumlah uang, menentukan biaya dan total biayanya

```
118     int i = 1;
119     foreach(String d in cfg.config.methods)
120     {
121         Console.WriteLine($"{i} {d}");
122         i++;
123     }
124
125     Console.Write("Pilihan : ");
126     int idxMethods = int.Parse(Console.ReadLine());
127
128     Console.WriteLine();
129
130     if(cfg.config.lang == "en")
131     {
132         Console.Write($"Please type {cfg.config.confirmation.en} to confirm the transaction : ");
133         string confirm = Console.ReadLine();
134
135         if(confirm == cfg.config.confirmation.en)
136         {
137             Console.WriteLine("The transfer is completed");
138         }
139         else
140         {
141             Console.WriteLine("Transfer is cancelled");
142         }
143     }
144     else
145     {
146         Console.Write($"Ketik {cfg.config.confirmation.id} untuk mengkonfirmasi transaksi : ");
147         string confirm = Console.ReadLine();
148
149         if (confirm == cfg.config.confirmation.id)
150         {
151             Console.WriteLine("Proses transfer berhasil");
152         }
153         else
154         {
155             Console.WriteLine("Transfer dibatalkan");
156         }
157     }
158 }
159 }
160 }
```

Ln 12

- Kemudian menggunakan foreach untuk menampilkan data array methods transfer
- Input pilihan pengguna
- Dan melakukan input konfirmasi beserta status transaksinya