Nama : Muhammad Rizky Fajri

NIM : 1302204043

Link : https://github.com/Rizkyfajri/mod7\_1302204043.git

|  |  |
| --- | --- |
| Program.cs | // See https://aka.ms/new-console-template for more information  //Console.WriteLine("Hello, World!");  using Newtonsoft.Json;  namespace mod7\_1302204043  {  class Program  {  static void Main(string[] args)  {  Config config = new Config();  try  {    BankTransferConfig model = config.ReadConfig();  Console.WriteLine("Membaca file");  //BankTransferConfig model = new BankTransferConfig();  string message = (model.lang == "id") ? "masukan data anda": "Please insert your data";  //string message1 = (model.lang == )  Console.WriteLine(message);  //Console.WriteLine(model.transfer.Threshold);  int tempt\_input = Convert.ToInt32(Console.ReadLine());  if (tempt\_input < model.transfer.Threshold)  {  Console.WriteLine(model.transfer.Low\_fee);  }  else  {  Console.WriteLine(model.transfer.High\_fee);  }  string transfer\_met = (model.lang == "id") ? "pilih metode transfer" : "select transfer method";  Console.WriteLine(transfer\_met);  int a = 4;  //string[] tipe\_trans = { model.Method };  //List[] = model.Met hod  //Console.WriteLine(model.Method);  //while (a < )  //{  // Console.WriteLine(model.transfer.)  //}  var tipe\_trans = model.Method;  tipe\_trans.ForEach(t =>Console.WriteLine(t));  Console.ReadLine();  string konfirmasi = (model.lang == "id") ? "konfirmasi" : "confirm";  Console.WriteLine(konfirmasi);  if (model.lang == "id")  {  Console.WriteLine(model.confirm.id);  }  else  {  Console.WriteLine(model.confirm.en);  }  }  catch (Exception ex)  {  Console.WriteLine("file config belum ada");  config.WriteConfig();  Console.WriteLine("file config sudah ada");  }  }  public class Config  {  public BankTransferConfig ReadConfig()  {  BankTransferController controller = new BankTransferController();  return controller.Readjson();  }  public void WriteConfig()  {  //ini tulis cofig manual terus disimpan di direktori  BankTransferConfig model = new BankTransferConfig()  {  lang = "en",  transfer = new TransferModul()  {  Threshold = 25000000,  Low\_fee = 6500,  High\_fee = 15000  },  Method = new List<string>()  {  "RTO (real time)",  "SKN",  "RTGS",  "BI FAST"  },  confirm = new ConfirmModul()  {  en = "yes",  id = "ya"  }  };  BankTransferController controller = new BankTransferController();  controller.WriteJson(model);    string defaultValue = JsonConvert.SerializeObject(model, Formatting.Indented);  }  }  }  } |
| BankTransferConfig.cs | using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using Newtonsoft.Json;  namespace mod7\_1302204043  {  public class BankTransferConfig  {  public string lang { get; set; }  public TransferModul transfer { get; set; }  public List<string> Method { get; set; }  public ConfirmModul confirm { get; set; }  }  public class TransferModul  {  public int Threshold { get; set; }  public int Low\_fee { get; set; }  public int High\_fee { get; set; }  }  public class ConfirmModul  {  public string en { get; set; }  public string id { get; set; }  }  } |
| BankTransferController.cs | using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using Newtonsoft.Json;  namespace mod7\_1302204043  {  public class BankTransferController  {  public void WriteJson(BankTransferConfig data)  {  string file = JsonConvert.SerializeObject(data,Formatting.Indented);  File.WriteAllText(@"D:\Programs\Universitas\semester 4\Praktikum\_KPL\Model\_2\Project\_7\mod7\_1302204043\mod7\_1302204043\Bank\_Transfer\_Config.json", file);  }  public BankTransferConfig Readjson()  {  dynamic data = File.ReadAllText(@"D:\Programs\Universitas\semester 4\Praktikum\_KPL\Model\_2\Project\_7\mod7\_1302204043\mod7\_1302204043\Bank\_Transfer\_Config.json");  //dynamic data = File.ReadAllText("D://Programs//Universitas//semester 4//Praktikum\_KPL//Model\_2//Project\_7//mod7\_1302204043//mod7\_1302204043//Bank\_Transfer\_Config.json");  BankTransferConfig config = JsonConvert.DeserializeObject<BankTransferConfig>(data);  return config;  }  }  } |

Text

Description automatically generated

Figure output program