-----------------------------------App.config --------------------

<?xml version="1.0" encoding="utf-8" ?>

<configuration>

  <appSettings>

    <add key="NNFConIp" value="192.168.168.36" />

    <!--<add key="NNFConIp" value="192.168.168.36" />-->

    <add key="NNFConPUBPort" value="8003" />

    <add key="NNFConSUBPort" value="8002" />

    <!--<add key="DataConIp" value="192.168.168.36" />-->

    <add key="DataConIp" value="192.168.168.36" />

    <add key="DataConSUBPort" value="7070" />

    <add key="LanIp" value="192.168.168.75" />

    <add key="McastIp" value="192.168.168.97" />

    <add key="Mcastport" value="7171" />

    <add key="INDEXIP" value="233.1.2.5" />

   <add key="INDEXPORT" value="34074" />

    <add key="Interval" value="15000000" />

    <add key="APPTYPE" value="2LIOC" />

    <add key="ClientId" value="201010100282100"/>

    <add key="NNFPassword" value="11111111" />

    <add key="ClientSettingsProvider.ServiceUri" value="" />

  </appSettings>

</configuration>

====================================================Global.cs=======================

using System;

using AMS.Profile;

using System.Data;

using System.Configuration;

using System.Collections.Concurrent;

    public sealed class Global

    {

        internal string DataConIp;

        internal bool ReloginFarmloader;

        internal string DataConSUBPort;

        internal string NNFConIp;

        internal string NNFConPUBPort;

        internal string NNFConSUBPort;

        internal string LanIp;

        internal string McastIp;

        internal string Mcastport;

        internal string INDEXIP;

        internal string INDEXPORT;

        internal long ClientId;

        internal string NNFPassword;

        internal bool SignInStatus;

        internal bool Pass\_bool;

        internal bool warningvar;

        internal bool Fopairbool;

        internal bool Relogin;

        private static readonly Global instance = new Global();

       public static  Global Instance

        {

            get

            {

                return instance;

            }

        }

        private Global()

        {

            var config = new Config { GroupName = null };

            DataConIp = config.GetValue("appSettings", "DataConIp", null);

            DataConSUBPort = config.GetValue("appSettings", "DataConSUBPort", null);

            NNFConIp = config.GetValue("appSettings", "NNFConIp", null);

            NNFConPUBPort = config.GetValue("appSettings", "NNFConPUBPort", null);

            NNFConSUBPort = config.GetValue("appSettings", "NNFConSUBPort", null);

            LanIp = config.GetValue("appSettings", "LanIp", null);

            ClientId = Convert.ToInt64(config.GetValue("appSettings", "ClientId", null));

            McastIp = config.GetValue("appSettings", "McastIp", null);

            Mcastport = config.GetValue("appSettings", "Mcastport", null);

            McastIp = config.GetValue("appSettings", "McastIp", null);

            INDEXIP = config.GetValue("appSettings", "INDEXIP", null);

            INDEXPORT = config.GetValue("appSettings", "INDEXPORT", null);

           // APPTYPE = config.GetValue("appSettings", "APPTYPE", null);

           // interval = Convert.ToInt32(config.GetValue("appSettings", "Interval", null));

        }

    }

=====================================Program.cs==================

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApplication3

{

    class Program

    {

        static void Main(string[] args)

        {

            Console.WriteLine(Global.Instance.INDEXPORT);

            Console.WriteLine(Global.Instance.LanIp);

            Console.WriteLine(Global.Instance.INDEXIP);

            Console.WriteLine("dddddddd");

            Console.ReadLine();

        }

    }}