using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

namespace ConsoleApp1

{

#region InterLocked

//class Counter

//{

// public static int Count = 0;

// public static int Count2 = 0;

//}

//public class Program

//{

// static void Main(string[] args)

// {

// Thread[]threads=new Thread[5];

// for (int i = 0; i < 5; i++)

// {

// threads[i] = new Thread(() =>

// {

// for (int k = 0; k < 1000000; k++)

// {

// Interlocked.Increment(ref Counter.Count);

// //Counter.Count++;

// if (k % 2 == 0)

// {

// Counter.Count2++;

// }

// }

// });

// }

// for (int i = 0; i < 5; i++)

// {

// threads[i].Start();

// }

// for (int i = 0; i < 5; i++)

// {

// threads[i].Join();

// }

// Console.WriteLine(Counter.Count);

// }

//}

#endregion

#region Monitor

//class Counter

//{

// public static int Count = 0;

// public static int Count2 = 0;

//}

//class Program

//{

// static object obj = new object();

// static void Main(string[] args)

// {

// //Thread[] threads = new Thread[5];

// //for (int i = 0; i < 5; i++)

// //{

// // threads[i] = new Thread(() =>

// // {

// // Console.WriteLine(Thread.CurrentThread.ManagedThreadId);

// // for (int k = 0; k < 1000000; k++)

// // {

// // try

// // {

// // Monitor.Enter(obj);

// // Counter.Count++;

// // }

// // finally

// // {

// // Monitor.Exit(obj);

// // }

// // }

// // });

// //}

// //for (int i = 0; i < 5; i++)

// //{

// // threads[i].Start();

// //}

// //for (int i = 0; i < 5; i++)

// //{

// // threads[i].Join();

// //}

// //Console.WriteLine(Counter.Count);

// //Thread[] threads = new Thread[5];

// //for (int i = 0; i < 5; i++)

// //{

// // threads[i] = new Thread(() =>

// // {

// // Console.WriteLine(Thread.CurrentThread.ManagedThreadId);

// // for (int k = 0; k < 1000000; k++)

// // {

// // lock (obj)

// // {

// // Counter.Count++;

// // }

// // }

// // });

// //}

// //for (int i = 0; i < 5; i++)

// //{

// // threads[i].Start();

// //}

// //for (int i = 0; i < 5; i++)

// //{

// // threads[i].Join();

// //}

// //Console.WriteLine(Counter.Count);

// }

//}

#endregion

#region Mutex

#region Mutex 1

//class Program

//{

// static Mutex mutexObj = new Mutex();

// static int x = 0;

// static void Main(string[] args)

// {

// for (int i = 0; i < 5; i++)

// {

// Thread t = new Thread(Count);

// t.Name = " Threads " + i;

// t.Start();

// }

// }

// private static void Count()

// {

// mutexObj.WaitOne();

// // x = 0;

// for (int i = 0; i < 9; i++)

// {

// ++x;

// Console.WriteLine($"{Thread.CurrentThread.ManagedThreadId} x : {x}");

// Thread.Sleep(10);

// }

// mutexObj.ReleaseMutex();

// }

//}

#endregion

#region Mutex 2 Global

class Program

{

static void Main(string[] args)

{

string mutexName = "MyMutex";

using (var m = new Mutex(false, mutexName))

{

if (!m.WaitOne(4000,false))

{

Console.WriteLine("Second instance running");

}

else

{

Console.WriteLine("Run amazing codes");

Console.ReadKey();

m.ReleaseMutex();

}

}

}

}

#endregion

#endregion

}