using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

namespace ConsoleApp1

{

#region Thread Background Example

//public class Program

//{

// static void Something()

// {

// for (int i = 0; i < 20; i++)

// {

// Console.WriteLine($"Downloading {i}");

// Thread.Sleep(500);

// }

// }

// static void Main(string[] args)

// {

// Thread thread = new Thread(() =>

// {

// Something();

// });

// thread.Start();

// thread.IsBackground = true;

// Thread thread1 = new Thread(() =>

// {

// thread.Join();

// for (int i = 0; i < 100; i++)

// {

// Thread.Sleep(30);

// Console.WriteLine("AAAAAAAAAAAAAAAAAAAAAAAAAAA");

// }

// });

// thread1.Start();

// var name = Console.ReadLine();

// var surname = Console.ReadLine();

// Console.WriteLine($"Welcome {name} {surname}");

// for (int i = 0; i < 50; i++)

// {

// Console.WriteLine($"Main Downloading . . . . {i}");

// Thread.Sleep(20);

// }

// }

//}

#endregion

#region BeginInvoke

//class Program

//{

// delegate string SomeSpecialDelegate();

// public static string SomeTask()

// {

// Console.WriteLine("Some process going");

// Console.WriteLine($"Some task ID : {Thread.CurrentThread.ManagedThreadId}");

// for (int i = 0; i < 10; i++)

// {

// Thread.Sleep(100);

// Console.WriteLine("New Thread Working");

// }

// return "Data";

// }

// static void Main(string[] args)

// {

// //var del = new SomeSpecialDelegate(SomeTask);

// //Console.WriteLine($"Main Thead Id : {Thread.CurrentThread.ManagedThreadId}");

// //var ar= del.BeginInvoke(null,null);

// //Console.WriteLine("RESULT");

// //for (int i = 0; i < 10; i++)

// //{

// // Thread.Sleep(100);

// // Console.WriteLine("Main Working");

// //}

// //var result = del.EndInvoke(ar);

// //Console.WriteLine(result);

// #region With Func

// //var func = new Func<string>(SomeTask);

// //Console.WriteLine($"Main Thead Id : {Thread.CurrentThread.ManagedThreadId}");

// //var ar = func.BeginInvoke(null, null);

// //Console.WriteLine("RESULT");

// //for (int i = 0; i < 10; i++)

// //{

// // Thread.Sleep(100);

// // Console.WriteLine("Main Working");

// //}

// //var result = func.EndInvoke(ar);

// //Console.WriteLine(result);

// #endregion

// }

//}

#endregion

#region ThreadPool

class Program

{

//static void AsyncOperation(object state)

//{

// Console.WriteLine(state.ToString());

// Thread.Sleep(5000);

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

//}

//static void Main(string[] args)

//{

// ThreadPool.GetMaxThreads(out int a, out int b);

// Console.WriteLine(a);

// Console.WriteLine(b);

// //ThreadPool.QueueUserWorkItem(AsyncOperation, "My Data 123");

// //Console.ReadLine();

//}

static void FunctionThread(object state)

{

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

Thread.Sleep(10000);

}

static void Main(string[] args)

{

//Thread thread = new Thread(FunctionThread);

//for (int i = 0; i < 600; i++)

//{

// thread = new Thread(FunctionThread);

// thread.Start();

//}

for (int i = 0; i < 600; i++)

{

ThreadPool.QueueUserWorkItem(FunctionThread, null);

}

Console.ReadLine();

}

}

#endregion

}