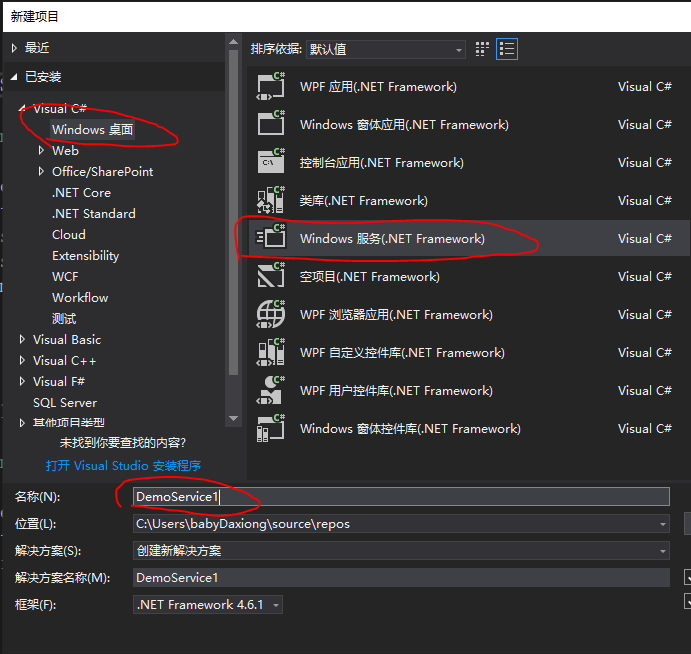
C#制作服务 参考https://www.cnblogs.com/mq0036/p/7875864.html

另外，通用的window把已有的exe程序包装成服务的示例[Windows下把EXE 程序变成服务运行\_过期罐头-CSDN博客\_windows把exe程序做成服务](https://blog.csdn.net/lys1220/article/details/90052262?utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromMachineLearnPai2%7Edefault-1.control&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromMachineLearnPai2%7Edefault-1.control)

# 制作服务

## 步骤1：



## 步骤2：

将Service1.cs文件名重命名改为MyService.cs

## 步骤3：

选中MyService.cs ，按住 ctrl+alt+数字0，进入对应的代码编辑器内，编写如下代码，主要实现两个函数，已用绿色标注。

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Diagnostics;

using System.IO;

using System.Linq;

using System.ServiceProcess;

using System.Text;

using System.Threading.Tasks;

namespace DemoService1

{

public partial class MyService : ServiceBase

{

public MyService()

{

InitializeComponent();

}

string filePath = @"H:\springtemp\MyServiceLog.txt";

protected override void OnStart(string[] args)

{

using (FileStream stream = new FileStream(filePath, FileMode.Append))

using (StreamWriter writer = new StreamWriter(stream))

{

writer.WriteLine($"{DateTime.Now},服务启动！");

}

}

protected override void OnStop()

{

using (FileStream stream = new FileStream(filePath, FileMode.Append))

using (StreamWriter writer = new StreamWriter(stream))

{

writer.WriteLine($"{DateTime.Now},服务停止！");

}

}

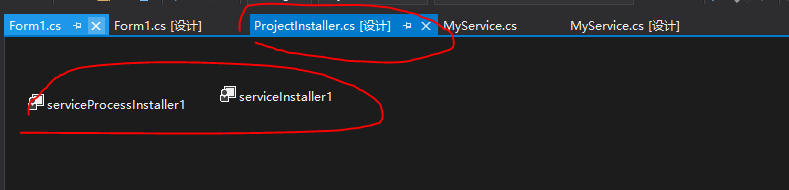
}

}

## 步骤4：

双击MyService.cs，回到设计状态，右键设计界面，选择“添加安装程序”。

此时新增了一个ProjectInstaller.cs，同时会生成两个组件，分别为“serviceInstaller1”及“serviceProcessInstaller1”



## 步骤5：

设置服务安装相关参数，

* 点击“serviceInstaller1”，在“属性”窗体将ServiceName改为MyService，Description改为我的服务，StartType保持为Manual。
* 点击“serviceProcessInstaller1”，在“属性”窗体将Account改为LocalSystem（服务属性系统级别）

这些都会对应之后的服务中看到的信息(在windows的cmd中输入services.msc)

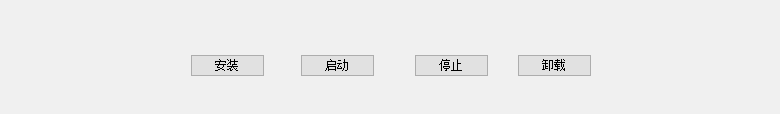
## 步骤6：

右键项目，点击生成，就可以看到有一个exe程序。至此一个服务程序就开发好了。但是目前还不能使用，缺少一个调用安装。可以使用installutil.exe工具来手工安装。 也可以我们自己开发一个工具来帮助安装，接下来我们自己做一个.net服务安装小工具。

# 制作安装该服务的工具程序

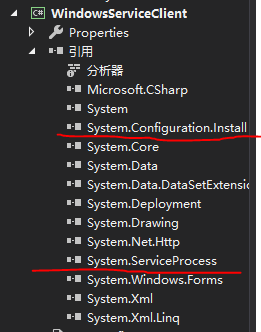
## 步骤1：

新建一个winform程序，在Form1界面上添上如下四个按钮



## 步骤2：

引入如下两个库



## 步骤3：

编写代码如下

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.ServiceProcess;

using System.Configuration.Install;

using System.Collections;

namespace WindowsServiceClient

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

string serviceFilePath = $"{Application.StartupPath}\\DemoService1.exe";

string serviceName = "MyService";

private void button1\_Click(object sender, EventArgs e)

{

if (this.IsServiceExisted(serviceName)) this.UninstallService(serviceName);

this.InstallService(serviceFilePath);

}

private void button2\_Click(object sender, EventArgs e)

{

if (this.IsServiceExisted(serviceName)) this.ServiceStart(serviceName);

}

private void button3\_Click(object sender, EventArgs e)

{

if (this.IsServiceExisted(serviceName)) this.ServiceStop(serviceName);

}

private void button4\_Click(object sender, EventArgs e)

{

if (this.IsServiceExisted(serviceName))

{

this.ServiceStop(serviceName);

this.UninstallService(serviceFilePath);

}

}

//判断服务是否存在

private bool IsServiceExisted(string serviceName)

{

ServiceController[] services = ServiceController.GetServices();

foreach (ServiceController sc in services)

{

if (sc.ServiceName.ToLower() == serviceName.ToLower())

{

return true;

}

}

return false;

}

//安装服务

private void InstallService(string serviceFilePath)

{

using (AssemblyInstaller installer = new AssemblyInstaller())

{

installer.UseNewContext = true;

installer.Path = serviceFilePath;

IDictionary savedState = new Hashtable();

installer.Install(savedState);

installer.Commit(savedState);

}

}

//卸载服务

private void UninstallService(string serviceFilePath)

{

using (AssemblyInstaller installer = new AssemblyInstaller())

{

installer.UseNewContext = true;

installer.Path = serviceFilePath;

installer.Uninstall(null);

}

}

//启动服务

private void ServiceStart(string serviceName)

{

using (ServiceController control = new ServiceController(serviceName))

{

if (control.Status == ServiceControllerStatus.Stopped)

{

control.Start();

}

}

}

//停止服务

private void ServiceStop(string serviceName)

{

using (ServiceController control = new ServiceController(serviceName))

{

if (control.Status == ServiceControllerStatus.Running)

{

control.Stop();

}

}

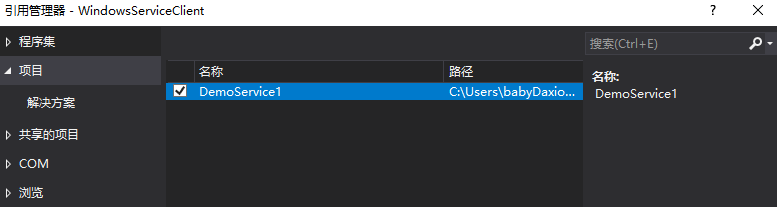
}

}

}

## 步骤4

添加刚开发好的demoService1，这样生成的服务程序就和服务安装工具在同一个目录下了。



## 步骤5：

网上的那篇文章说，需要添加UAC程序清单，修改管理员权限。我没有这样子做，而是直接生成程序，然后右键以管理员权限执行即可。

打开services.msc，然后运行WindowsServiceClient.exe，

* 点击“安装”，可以看到服务



* 点击启动，这个动作也可以直接选中服务进行操作。



* 点停止，这个动作也可以直接选中服务进行操作。



* 卸载，在服务中刷新一下，就找不到MyService了。