

Windows Service in C#

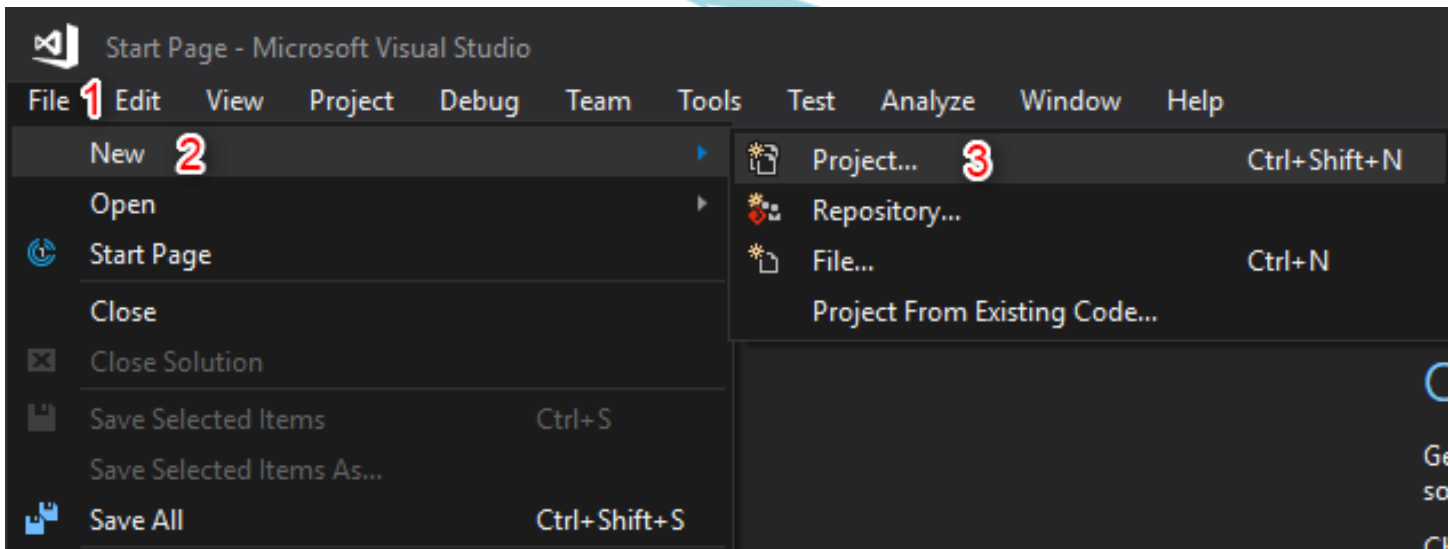
Windows Services normally start when the OS boots and runs an application in the background. Windows Services executes applications in its own session. It either starts automatically or we can manually pause, stop, and restart it.

Steps to Create Windows service

Step 1 :

Open Visual Studio, go to
File > New and select Project.

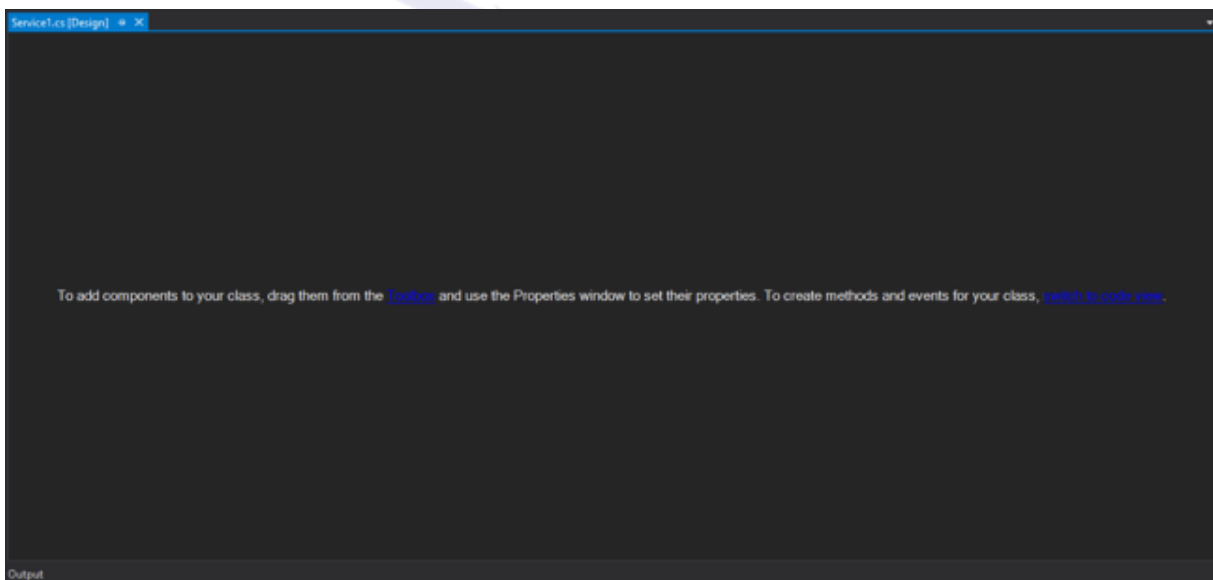
Now select a new project from the Dialog box and select "Window Service" and click on the OK button.



Step 2 :

Go to Visual C# -> "Windows Desktop" -> "Windows Service," give your project name and then click OK

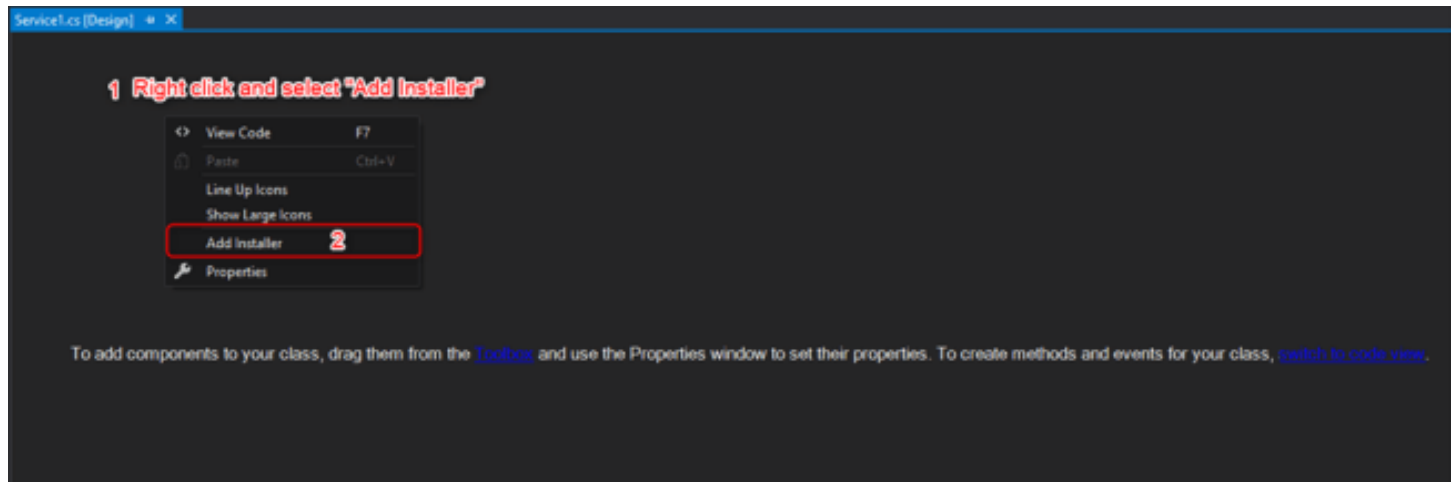
Once you click OK, the below screen will appear, which is your service



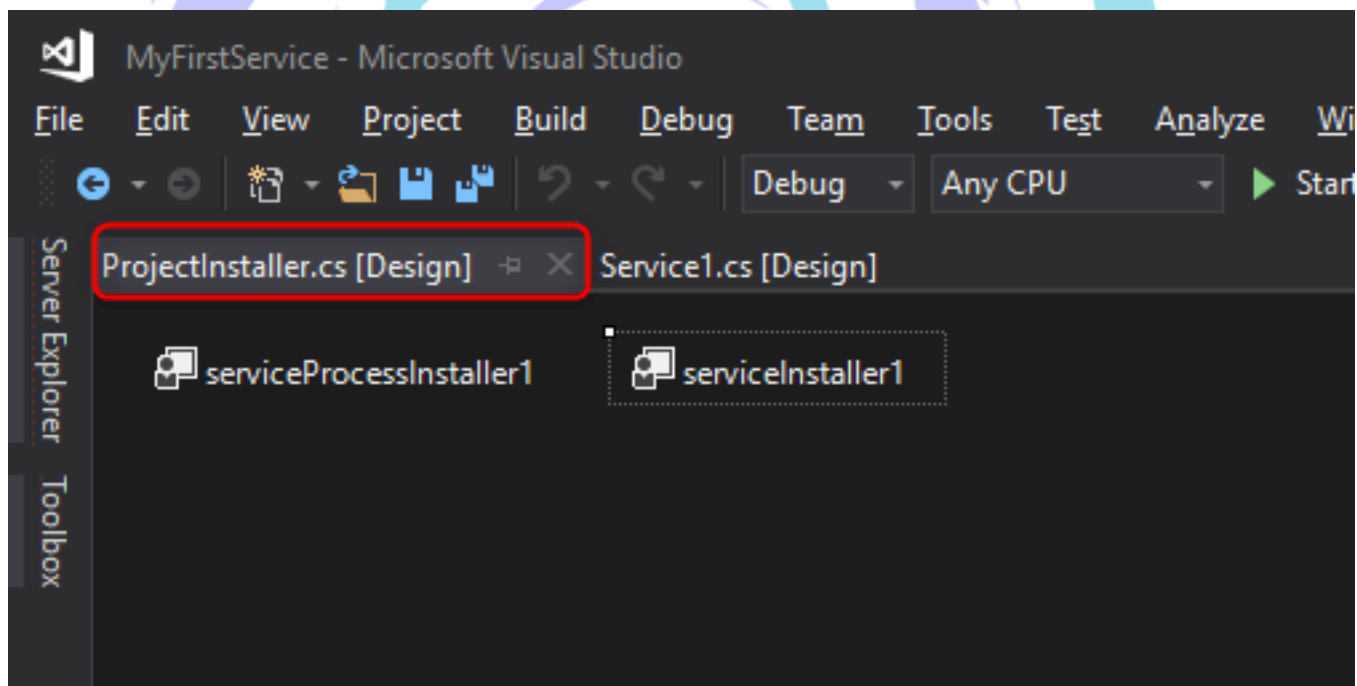
Step 3 :

Right-click on the blank area and select "Add Installer."
Adding Installers to the Service

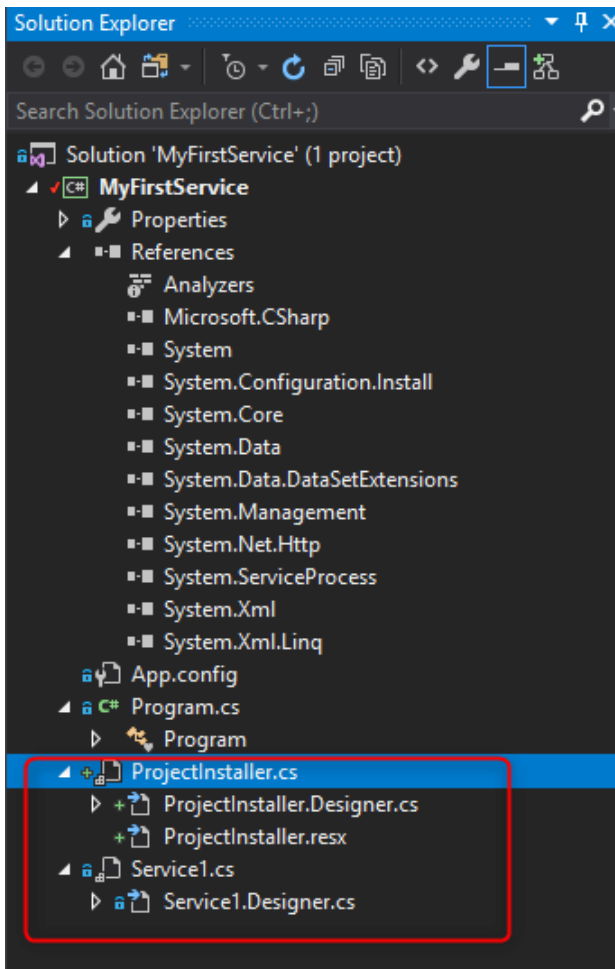
Before you can run a Windows Service, you need to install the Installer, which registers it with the Service Control Manager.



After Adding Installer, ProjectInstaller will add in your project and ProjectInstaller.cs file will be open.

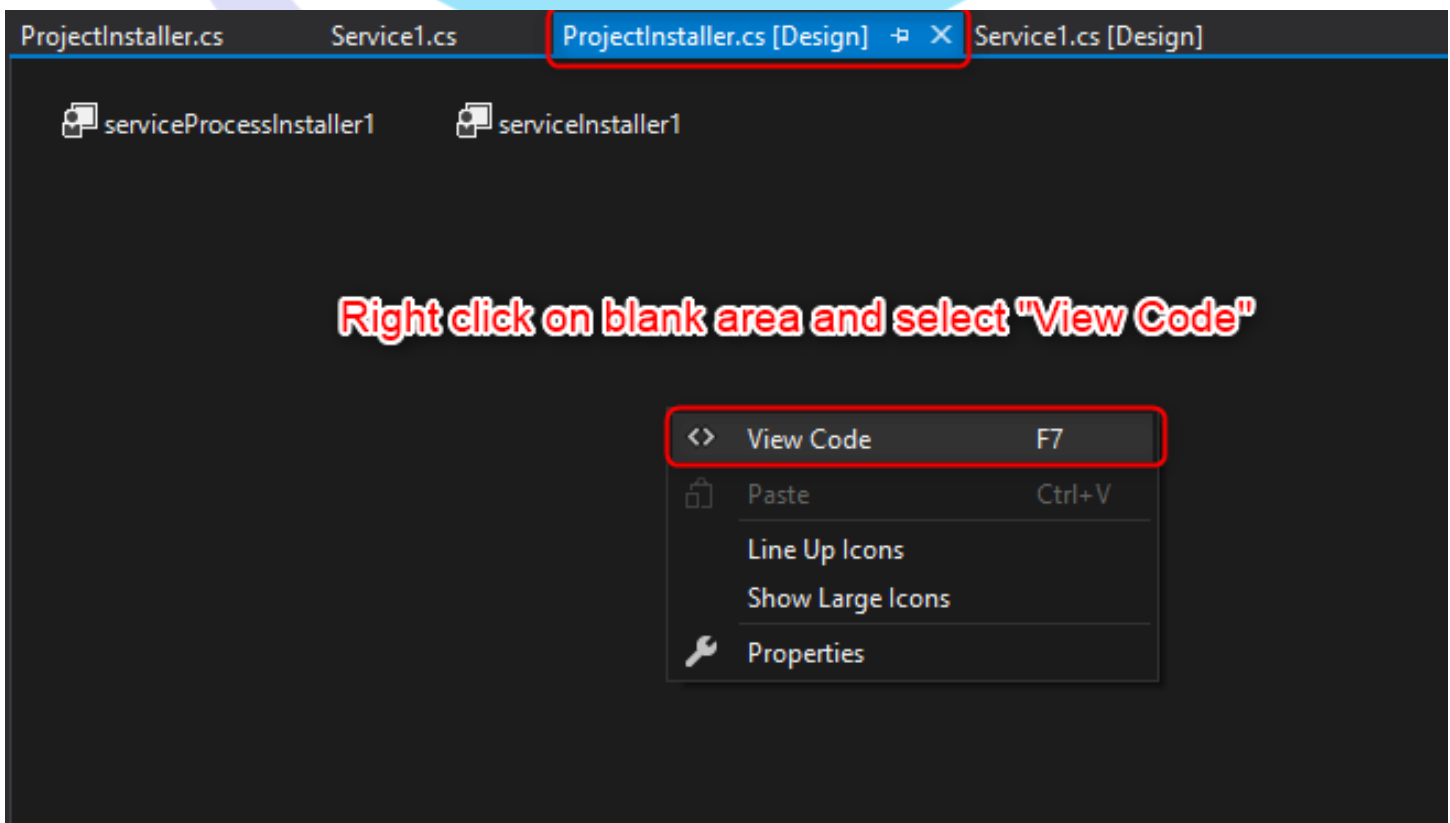


The Solution Explorer looks like this:



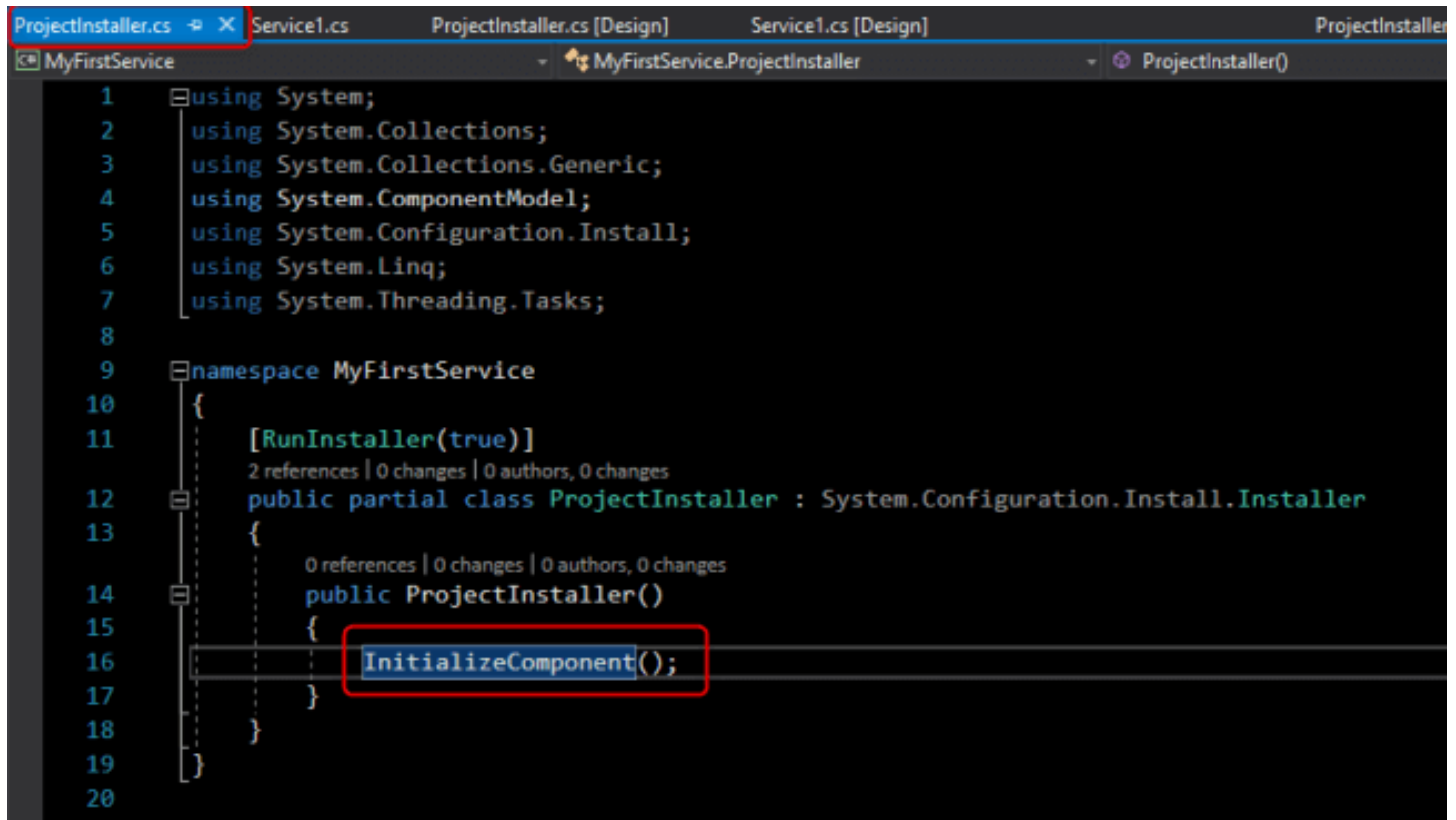
Step 4 :

Right click on blank area and select "View Code"



Step 5 :

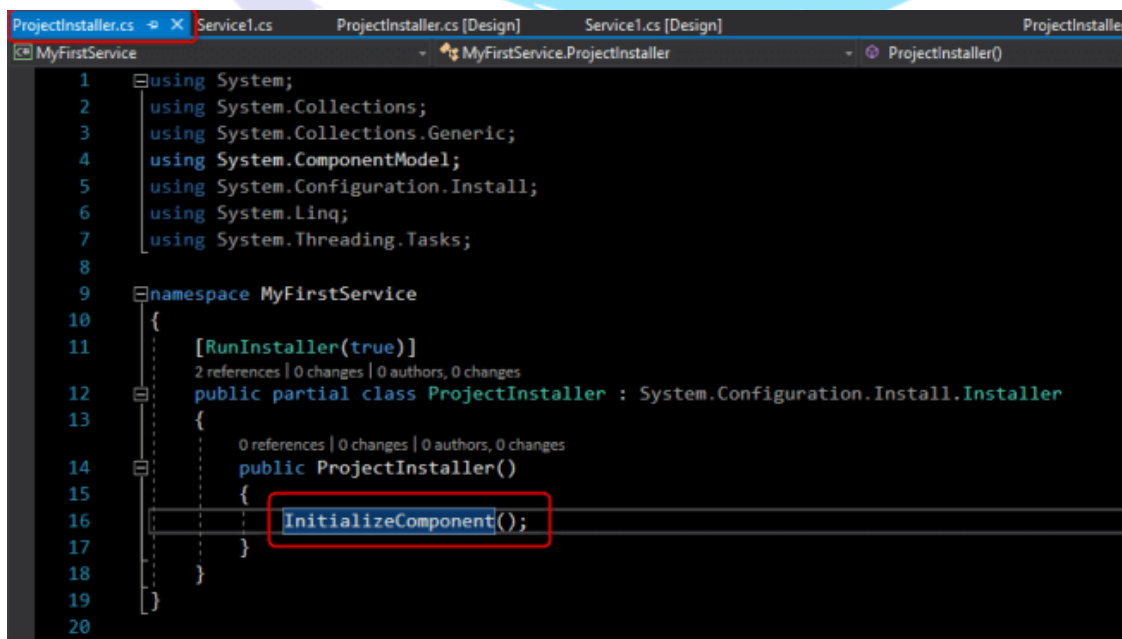
It has a Constructor, which contains the InitializeComponent method. The InitializeComponent method contains the logic which creates and initializes the user interface objects.



```
1  using System;
2  using System.Collections;
3  using System.Collections.Generic;
4  using System.ComponentModel;
5  using System.Configuration.Install;
6  using System.Linq;
7  using System.Threading.Tasks;
8
9  namespace MyFirstService
10 {
11     [RunInstaller(true)]
12     public partial class ProjectInstaller : System.Configuration.Install.Installer
13     {
14         public ProjectInstaller()
15         {
16             InitializeComponent();
17         }
18     }
19 }
20
```

Step 6 :

Select the InitializeComponent method and press the F12 key to go the definition.



```
1  using System;
2  using System.Collections;
3  using System.Collections.Generic;
4  using System.ComponentModel;
5  using System.Configuration.Install;
6  using System.Linq;
7  using System.Threading.Tasks;
8
9  namespace MyFirstService
10 {
11     [RunInstaller(true)]
12     public partial class ProjectInstaller : System.Configuration.Install.Installer
13     {
14         public ProjectInstaller()
15         {
16             InitializeComponent();
17         }
18     }
19 }
20
```

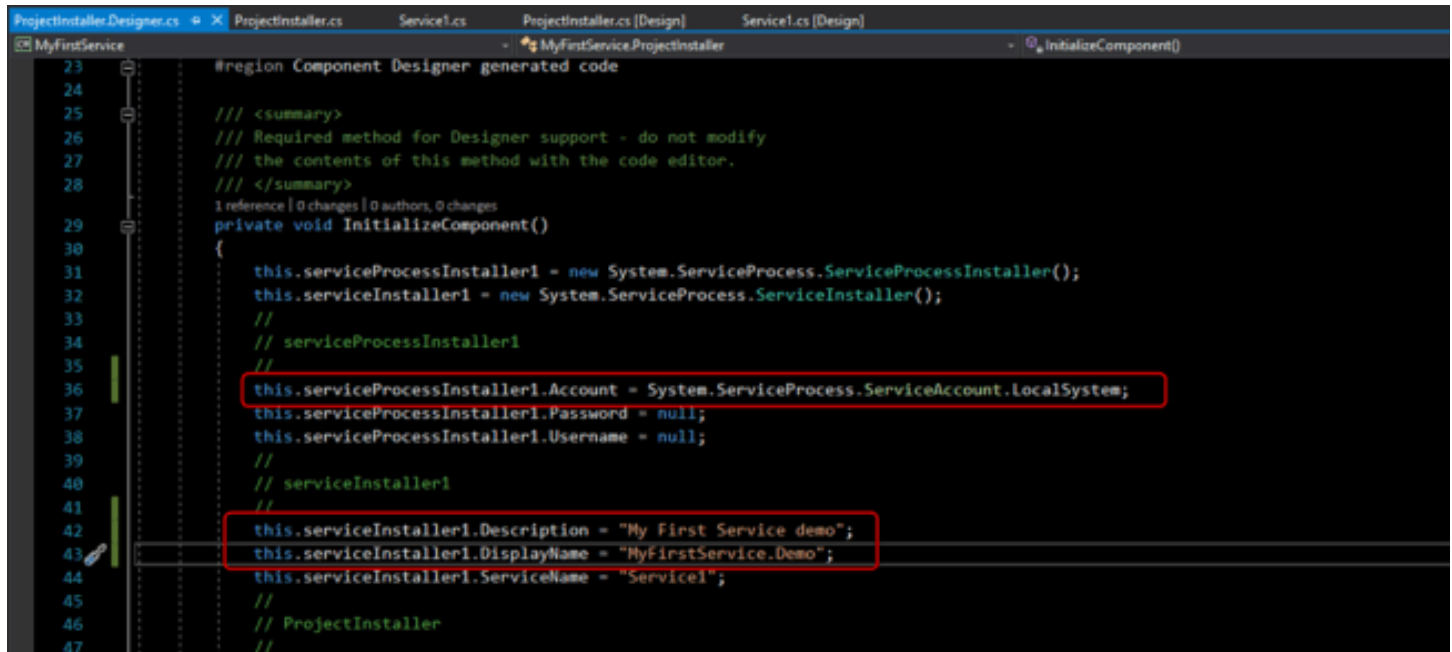
Step 7 :

Now add the below line while installing the service:

```
this.serviceProcessInstaller1.Account = System.ServiceProcess.ServiceAccount.LocalSystem;
```

```
this.serviceInstaller1.Description = "Marvellous Service";
```

```
this.serviceInstaller1.DisplayName = "Marvellous Service";
```

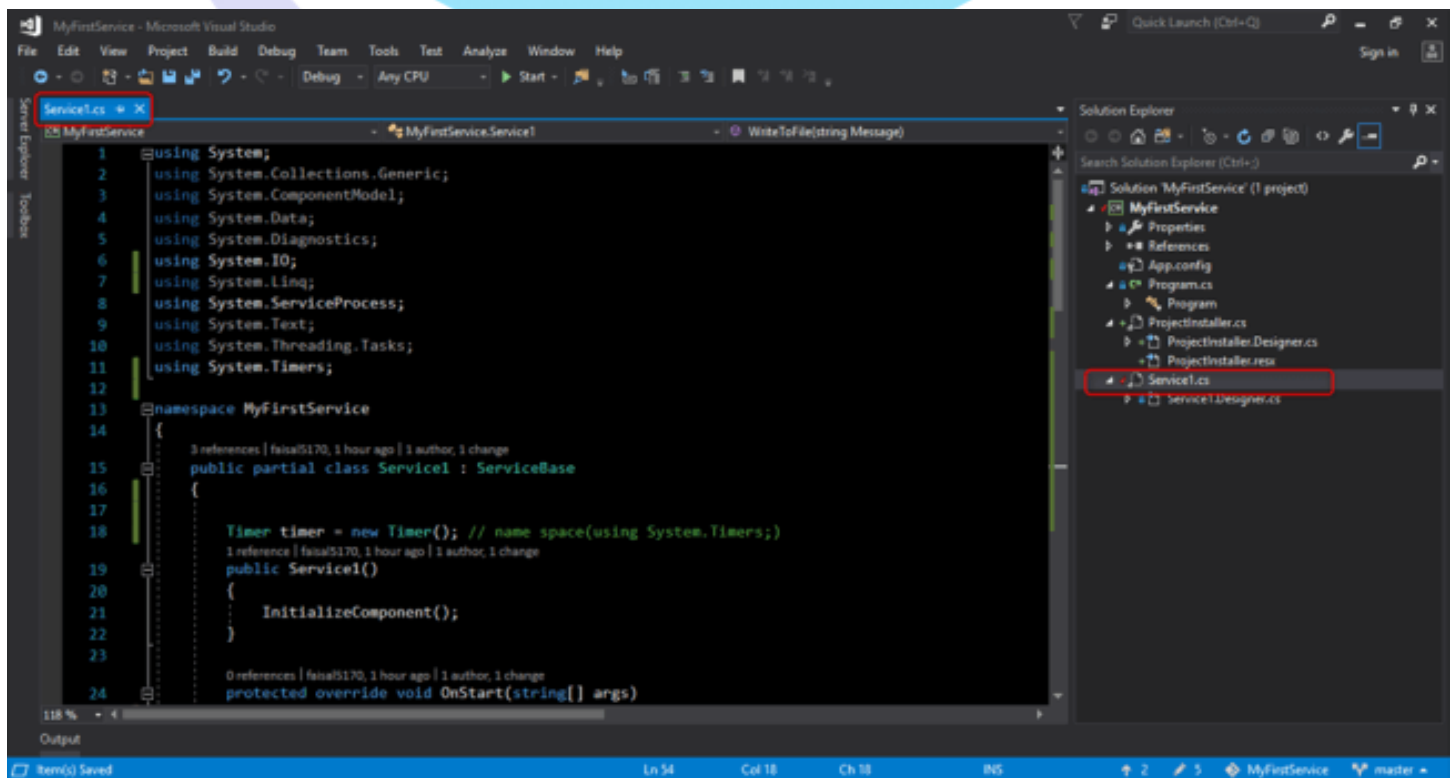


```

23  #region Component Designer generated code
24
25  /// <summary>
26  /// Required method for Designer support - do not modify
27  /// the contents of this method with the code editor.
28  /// </summary>
29  1 reference | 0 changes | 0 authors, 0 changes
30  private void InitializeComponent()
31  {
32      this.serviceProcessInstaller1 = new System.ServiceProcess.ServiceProcessInstaller();
33      this.serviceInstaller1 = new System.ServiceProcess.ServiceInstaller();
34      //
35      // serviceProcessInstaller1
36      //
37      this.serviceProcessInstaller1.Account = System.ServiceProcess.ServiceAccount.LocalSystem;
38      this.serviceProcessInstaller1.Password = null;
39      this.serviceProcessInstaller1.Username = null;
40      //
41      // serviceInstaller1
42      //
43      this.serviceInstaller1.Description = "My First Service demo";
44      this.serviceInstaller1.DisplayName = "MyFirstService.Demo";
45      this.serviceInstaller1.ServiceName = "Service1";
46      //
47      // ProjectInstaller
  
```

Step 8 :

In this step, we will write the code which gets executed when our service starts.



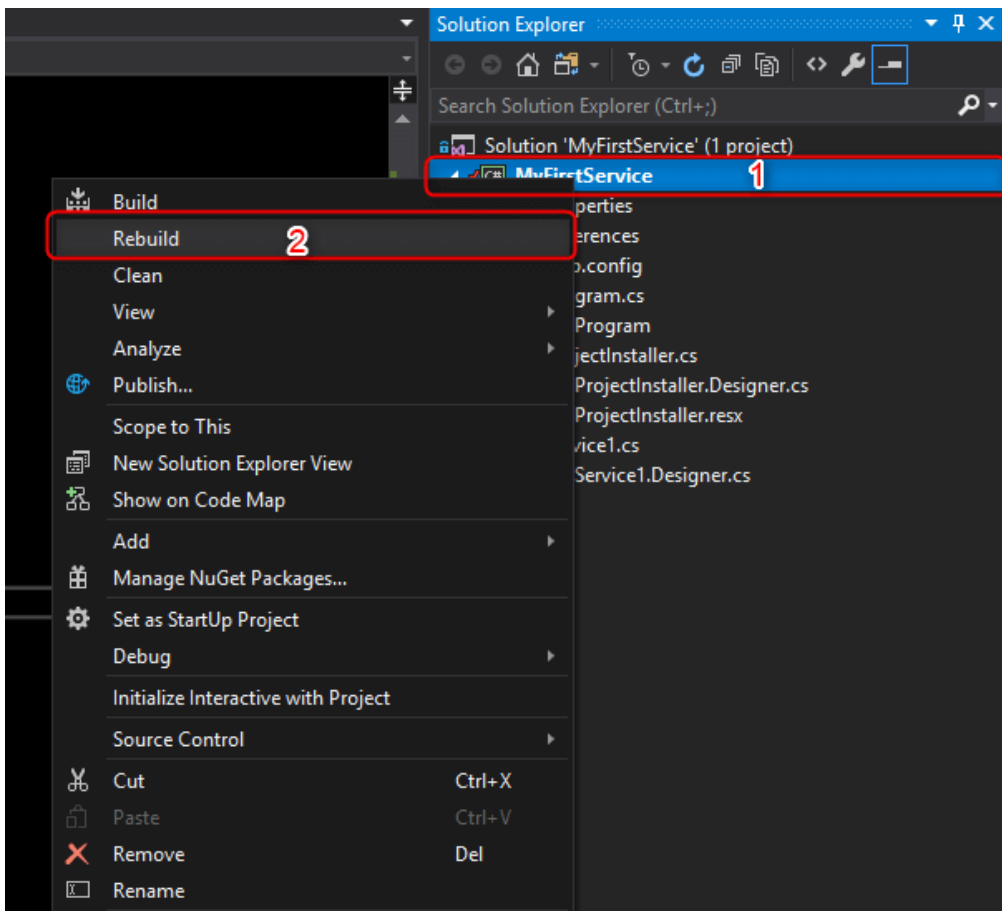
```

1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;
5  using System.Diagnostics;
6  using System.IO;
7  using System.Linq;
8  using System.ServiceProcess;
9  using System.Text;
10 using System.Threading.Tasks;
11 using System.Timers;
12
13 namespace MyFirstService
14 {
15     3 references | faisal5170, 1 hour ago | 1 author, 1 change
16     public partial class Service1 : ServiceBase
17     {
18         Timer timer = new Timer(); // name space(using System.Timers;)
19         1 reference | faisal5170, 1 hour ago | 1 author, 1 change
20         public Service1()
21         {
22             InitializeComponent();
23         }
24
25         0 references | faisal5170, 1 hour ago | 1 author, 1 change
26         protected override void OnStart(string[] args)
  
```

Step 9 :

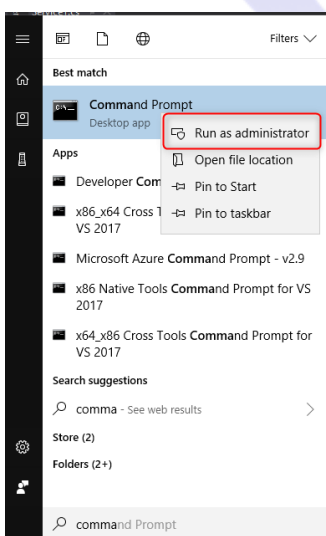
Rebuild Your Application

Right-click on your project or solution and select Rebuild.



Step 10 :

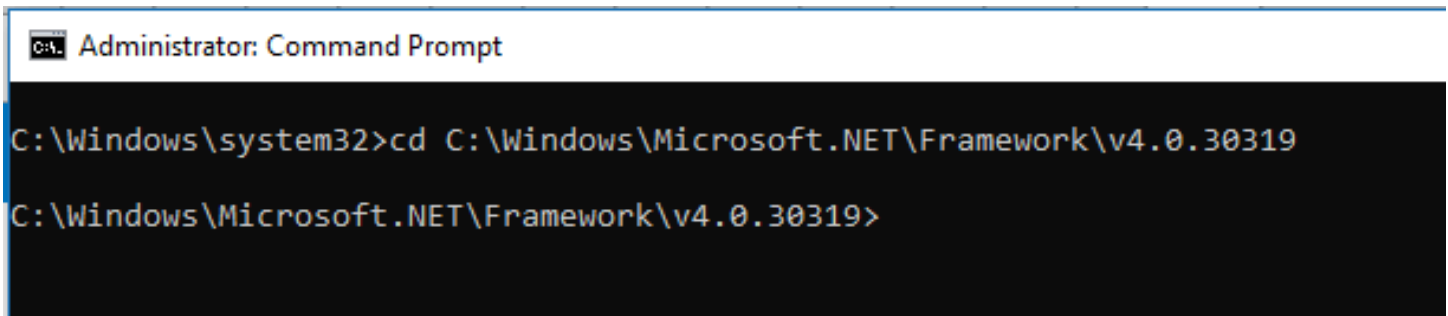
Search "command Prompt" and run the program as an administrator:



Step 11 :

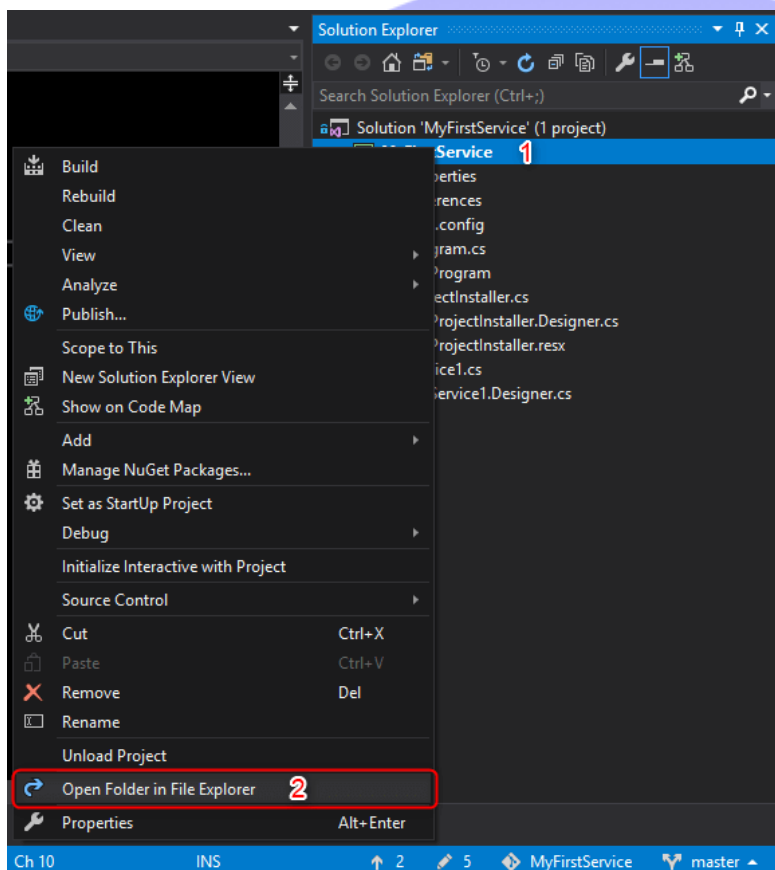
Fire the below command in the command prompt and press Enter.

```
cd C:\Windows\Microsoft.NET\Framework\v4.0.30319
```



Step 12 :

Now Go to your project source folder > bin > Debug and copy the full path of of the Windows Service.exe file



Step 13 :

Open the command prompt and fire the below command and press enter.
Syntax

InstallUtil.exe + Your copied path + \your service name + .exe
Our Path

InstallUtil.exe

C:\Users\Marvellous\source\repos\Marvellous\Marvellous\bin\Debug\Marvellous.exe

Step 14 :

Open services by following the below steps:

1. Press Window key + R.
2. Type services.msc
3. Find your Service.

