

Windowsサービスの作成手順 (VB.NET 2005)

Copyright (C) 2008 NonSoft. All Rights Reserved.

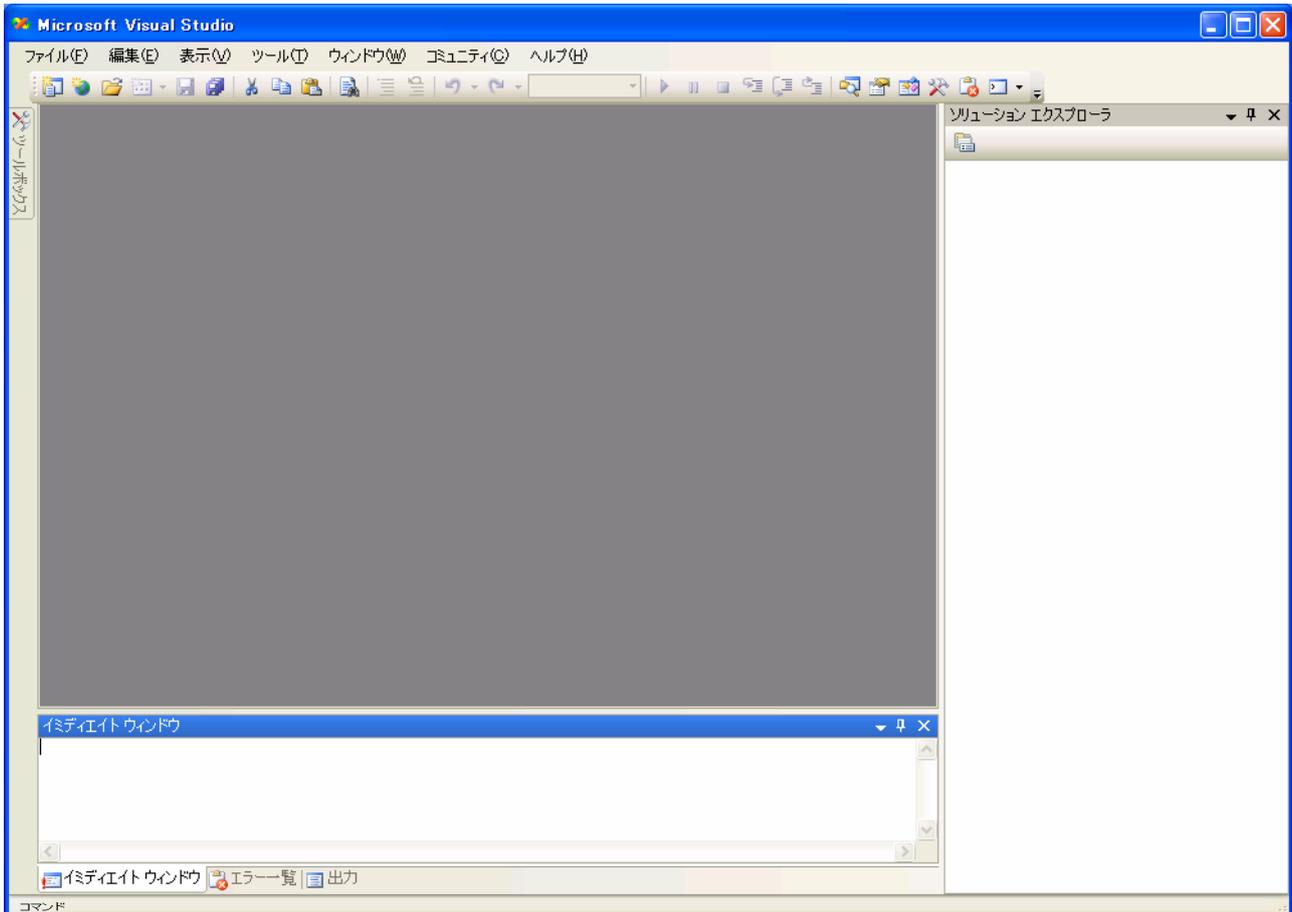
本手順書は、Windowsサービスの作成方法を実際の手順に沿って記述しました。

Windowsサービスは大きく分けて5つの手順で作成出来ます。

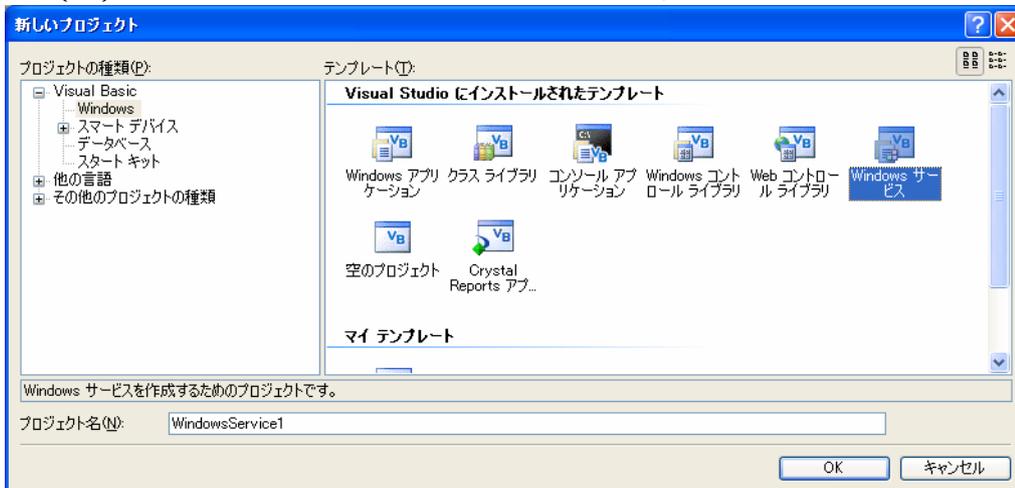
1. 「Windowsサービス」プロジェクトの作成
2. 「Windowsサービス」プロジェクトへの「インストーラーの追加」
3. 「セットアップ プロジェクト」の作成
4. ビルド
5. デバック

詳細な手順は以下に記述します。

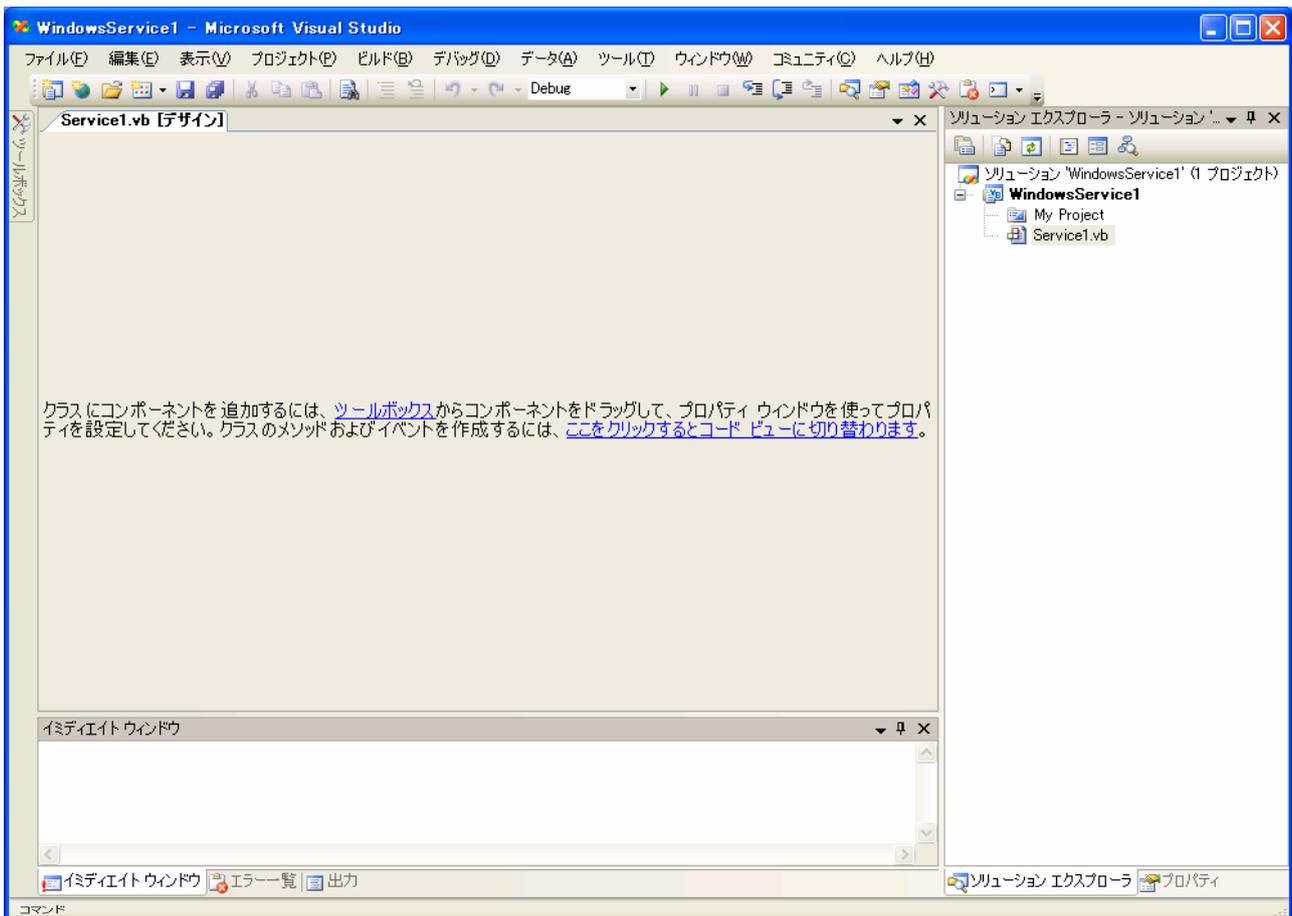
1. 「Windowsサービス」プロジェクトの作成
(1) 「Microsoft Visual Studio 2005」を起動



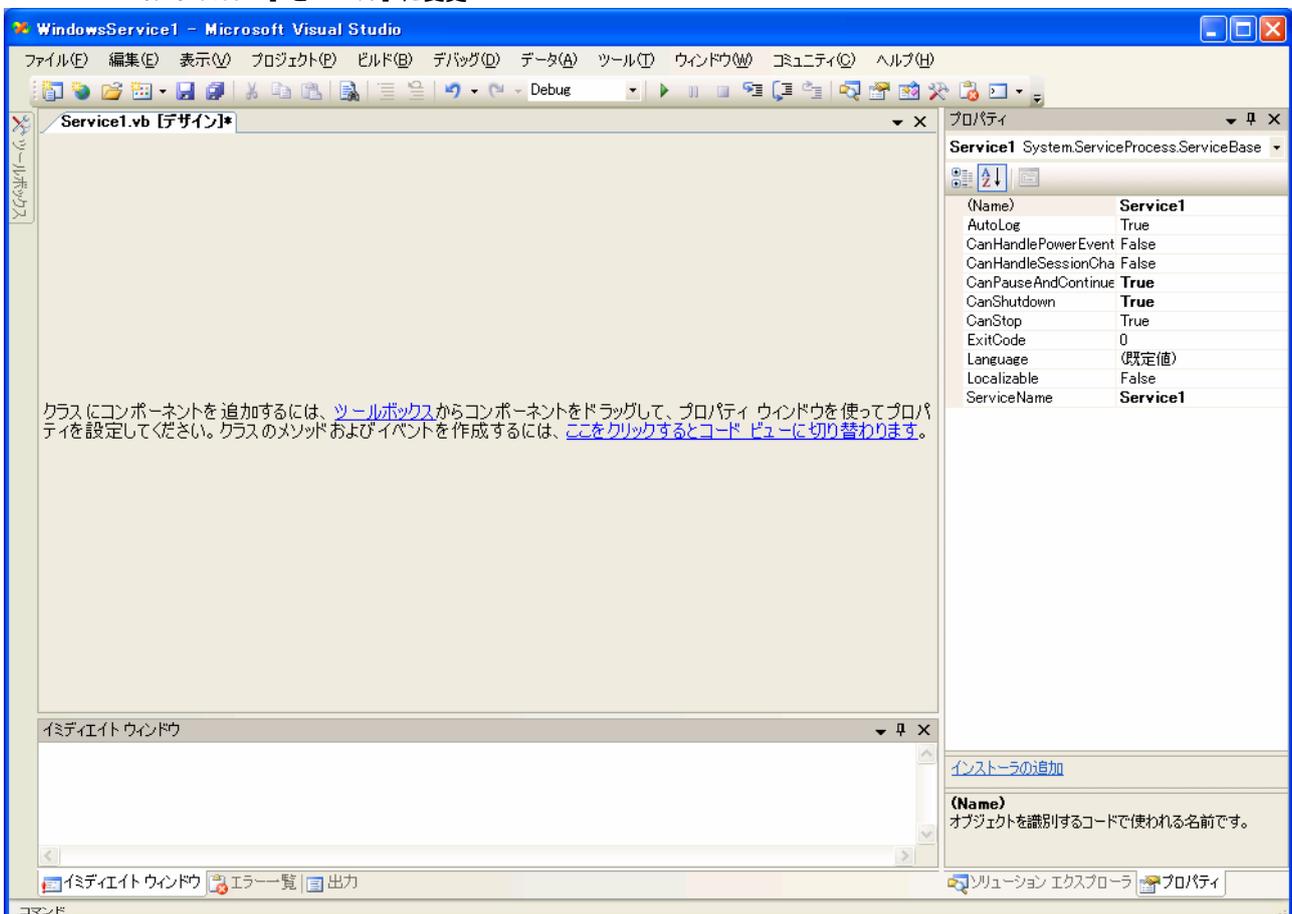
- (2) 「ファイル」メニュー～「新しいプロジェクト」を実行



- (3) 「Visual Basic」～「Windows」～「Windowsサービス」を選択し「OK」をクリック



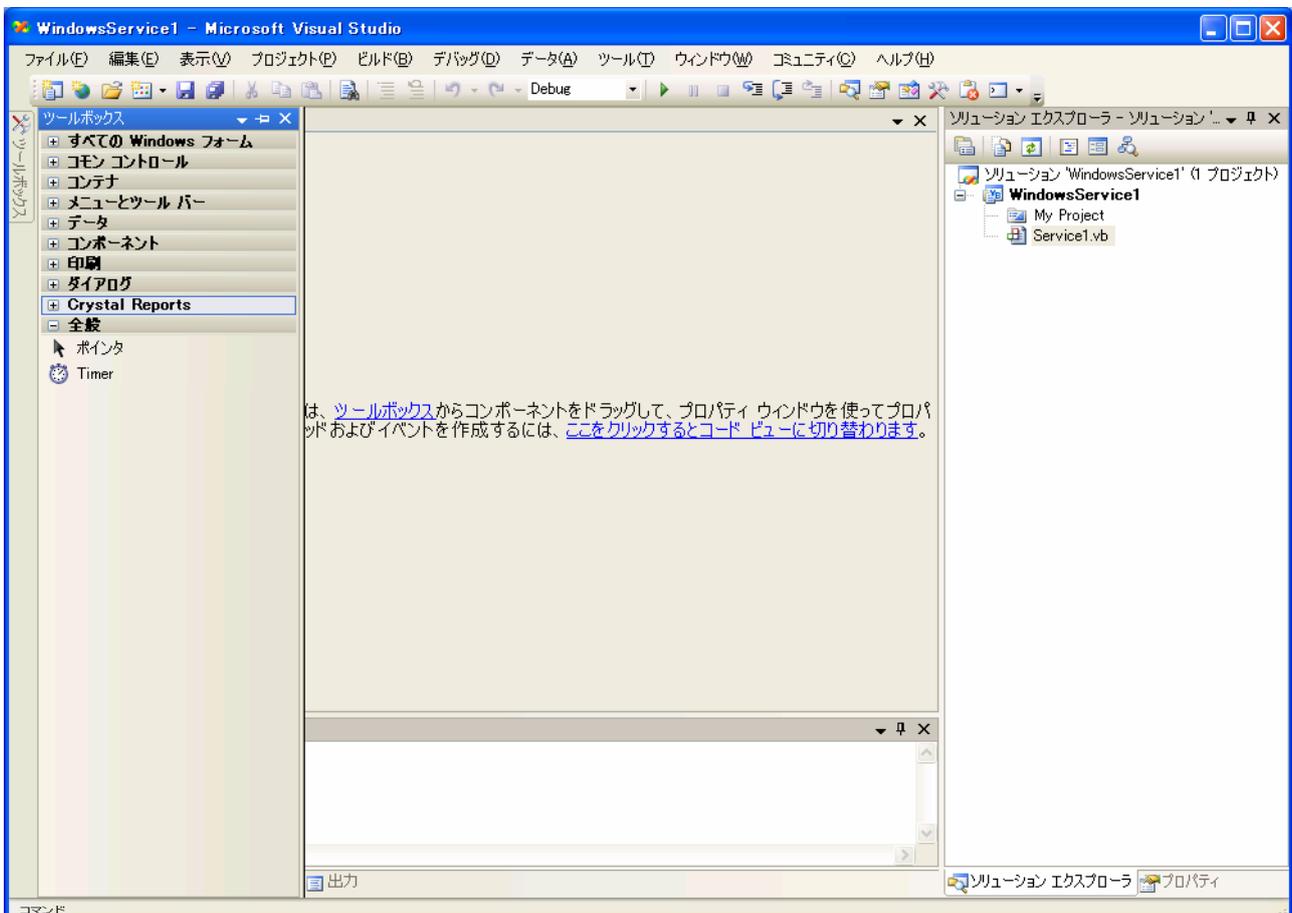
- (4) 「Service1.vb」のプロパティを確認
 「CanPauseAndContinue」を「True」に変更
 「CanShutdown」を「True」に変更



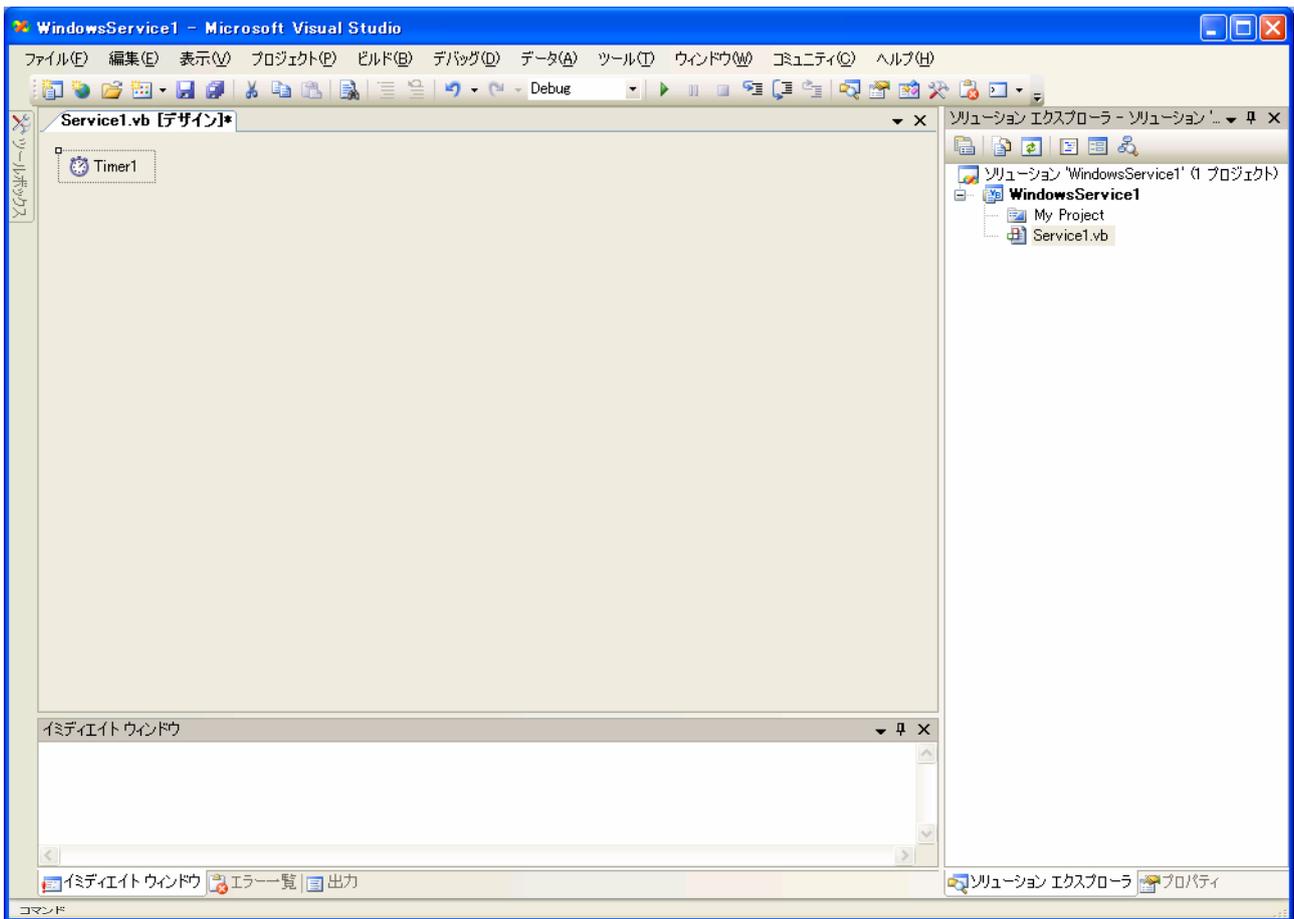
- (5) 「ツールボックス」の下の方の空いている所でマウス右クリックして「アイテムの選択」を実行



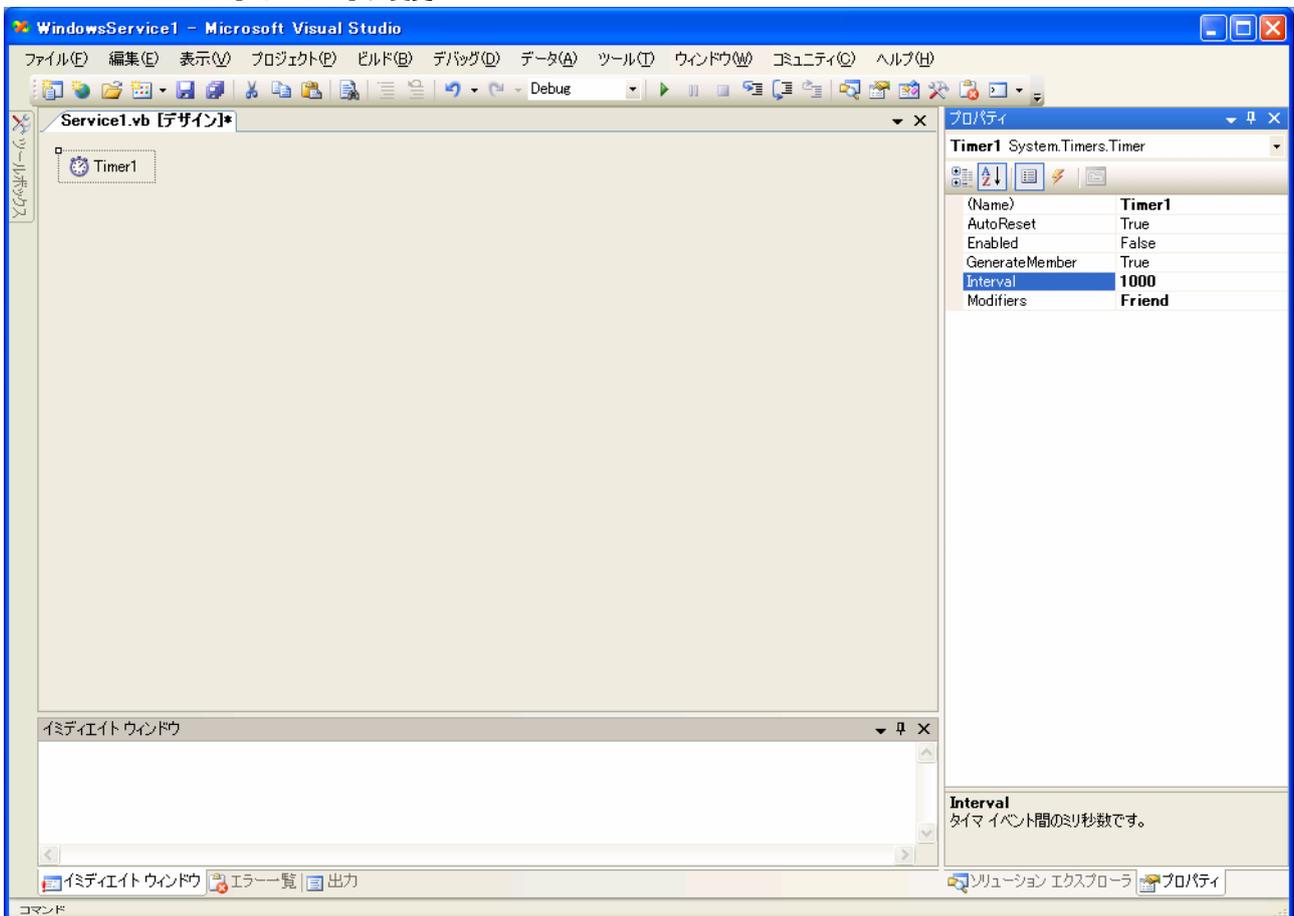
- (6) 「.NET Framework コンポーネント」の名前 = 「Timer」、名前空間 = 「System.Timers」をチェックし、「OK」をクリック



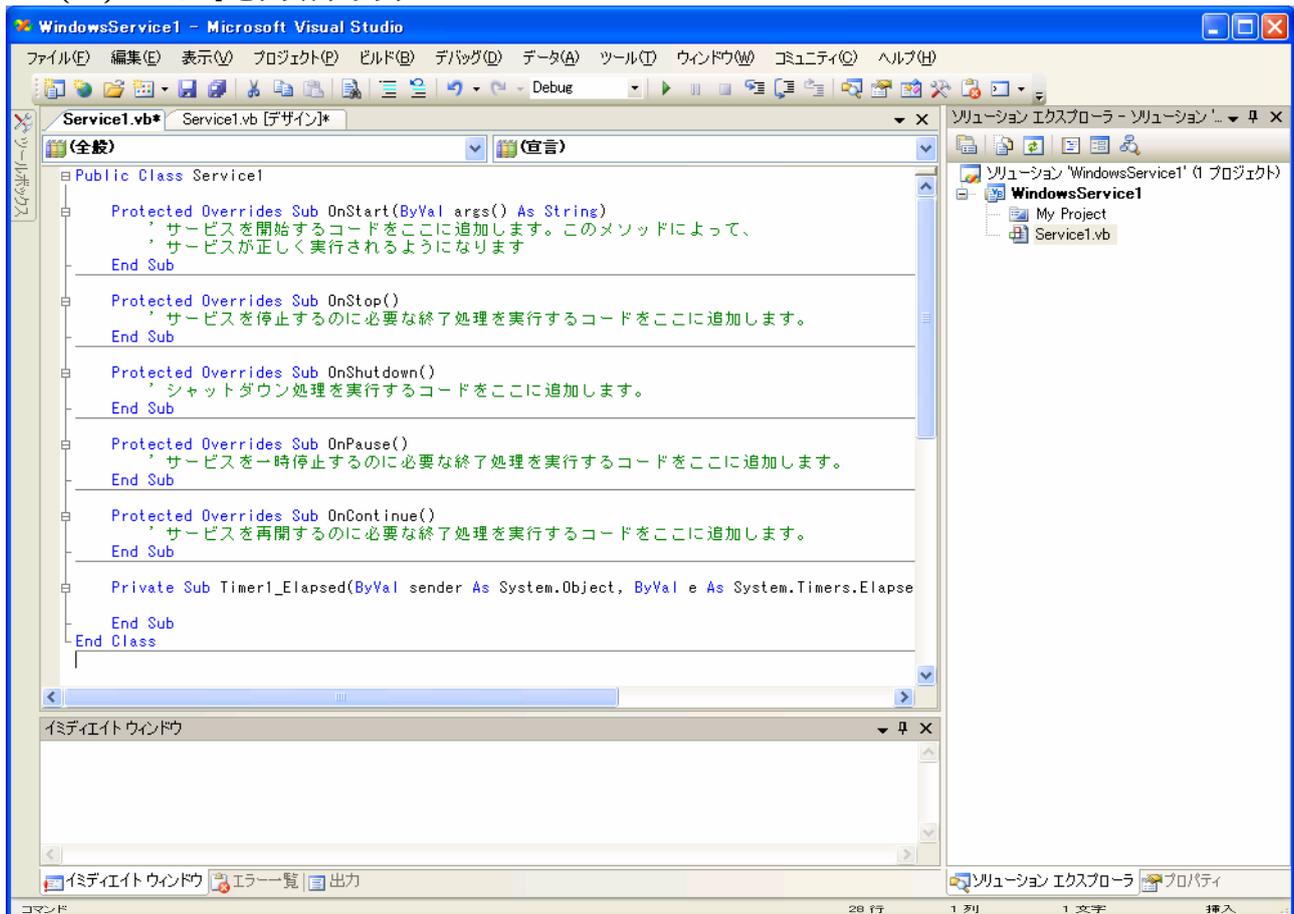
- (7) 「System.Timers.Timer」を「Service1.vb」へドラッグ&ドロップ



- (8) 「Timer1」のプロパティの確認
「Enabled」を「False」に変更
「Interval」を「1000」に変更



(9) 「Timer1」をダブルクリック



(10) 「Service1.vb」のコードを以下のように変更
(発生したイベント名をc:\temp\Service1.logに書くコードです。)

```
Public Class Service1

    Protected Overrides Sub OnStart(ByVal args() As String)
        ' サービスを開始するコードをここに追加します。このメソッドによって、
        ' サービスが正しく実行されるようになります
        ' デバック ( 「ツール」メニュー ~ 「プロセッサにアタッチ」 ) 操作する時間のスリープ
        System.Threading.Thread.Sleep(10000)
        ' トレースログ出力
        TraceLog("OnStart")
        ' タイマー開始
        Timer1.Enabled = True
    End Sub

    Protected Overrides Sub OnStop()
        ' サービスを停止するのに必要な終了処理を実行するコードをここに追加します。
        ' トレースログ出力
        TraceLog("OnStop")
        ' タイマー終了
        Timer1.Enabled = False
    End Sub

    Protected Overrides Sub OnShutdown()
        ' シャットダウン処理を実行するコードをここに追加します。
        ' トレースログ出力
        TraceLog("OnShutdown")
        ' タイマー終了
        Timer1.Enabled = False
    End Sub

    Protected Overrides Sub OnPause()
        ' サービスを一時停止するのに必要な終了処理を実行するコードをここに追加します。
        ' トレースログ出力
        TraceLog("OnPause")
        ' タイマー終了
        Timer1.Enabled = False
    End Sub

    Protected Overrides Sub OnContinue()
        ' サービスを再開するのに必要な終了処理を実行するコードをここに追加します。
        ' トレースログ出力
        TraceLog("OnContinue")
    End Sub
```

```

    ' タイマー開始
    Timer1.Enabled = True
End Sub

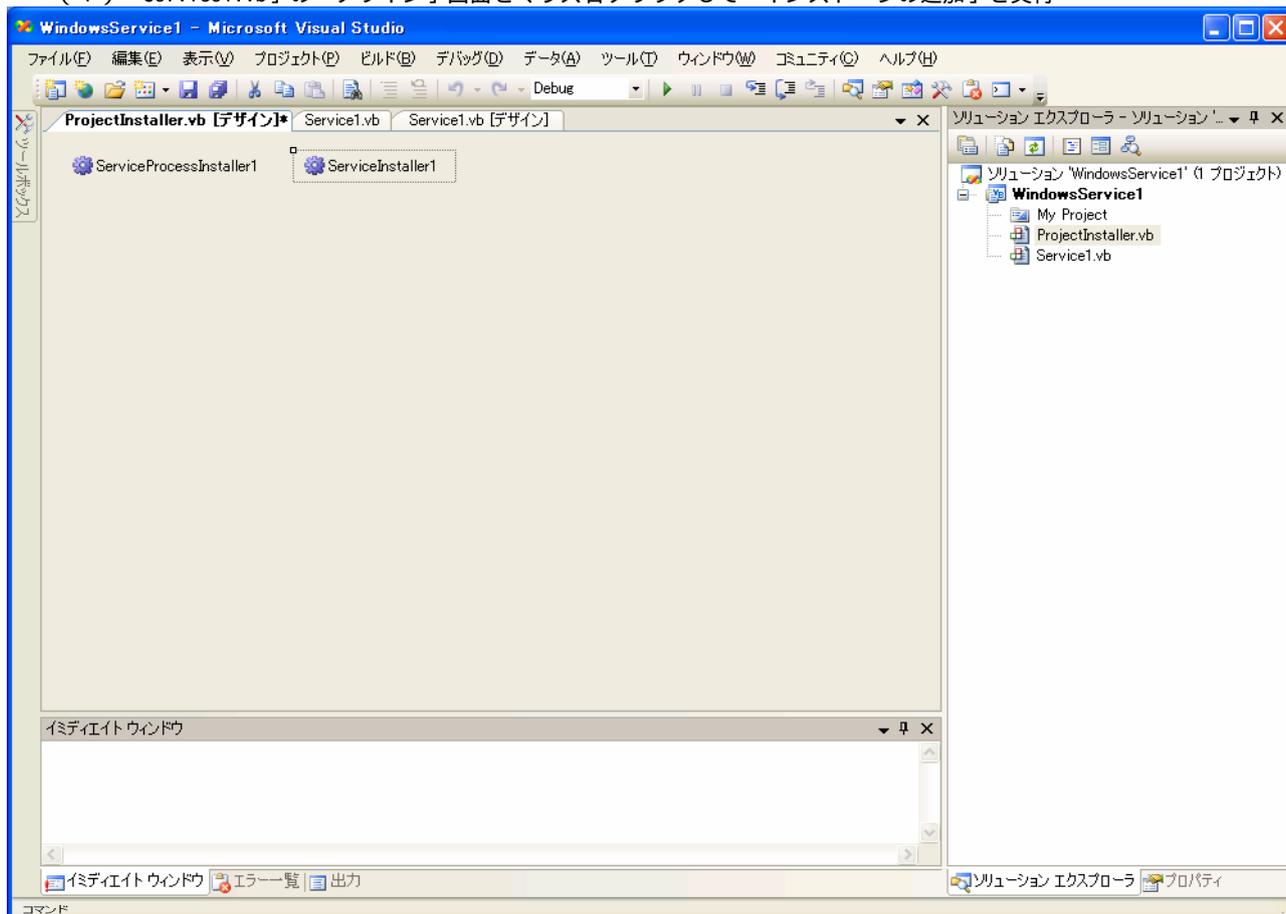
Private Sub Timer1_Elapsed(ByVal sender As System.Object, _
    ByVal e As System.Timers.ElapsedEventArgs) Handles Timer1.Elapsed
    ' タイマーの処理を実行するコードをここに追加します。
    ' トレースログ出力
    TraceLog("Timer1_Elapsed")
End Sub

Private Sub TraceLog(ByVal strMsg As String)
    ' トレースログ出力
    Dim sw As System.IO.StreamWriter = Nothing
    Try
        sw = New System.IO.StreamWriter("c:\temp\%Service1.log", _
            True, System.Text.Encoding.Default)
        sw.WriteLine(Now.ToString("yyyy/MM/dd HH:mm:ss") & " " & strMsg)
        sw.Flush()
    Catch ex As Exception
        Throw ex
    Finally
        If sw Is Nothing = False Then sw.Close()
    End Try
End Sub

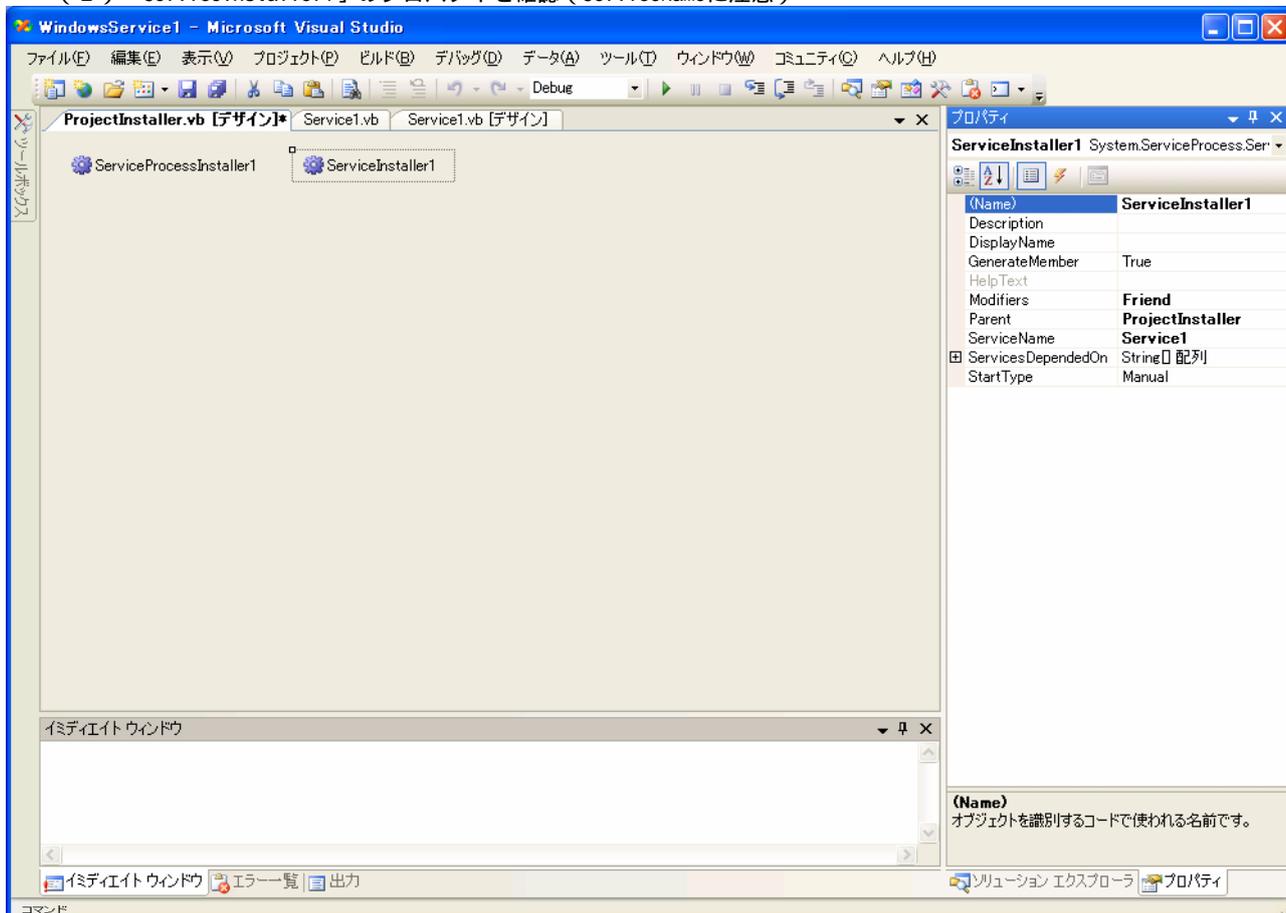
End Class

```

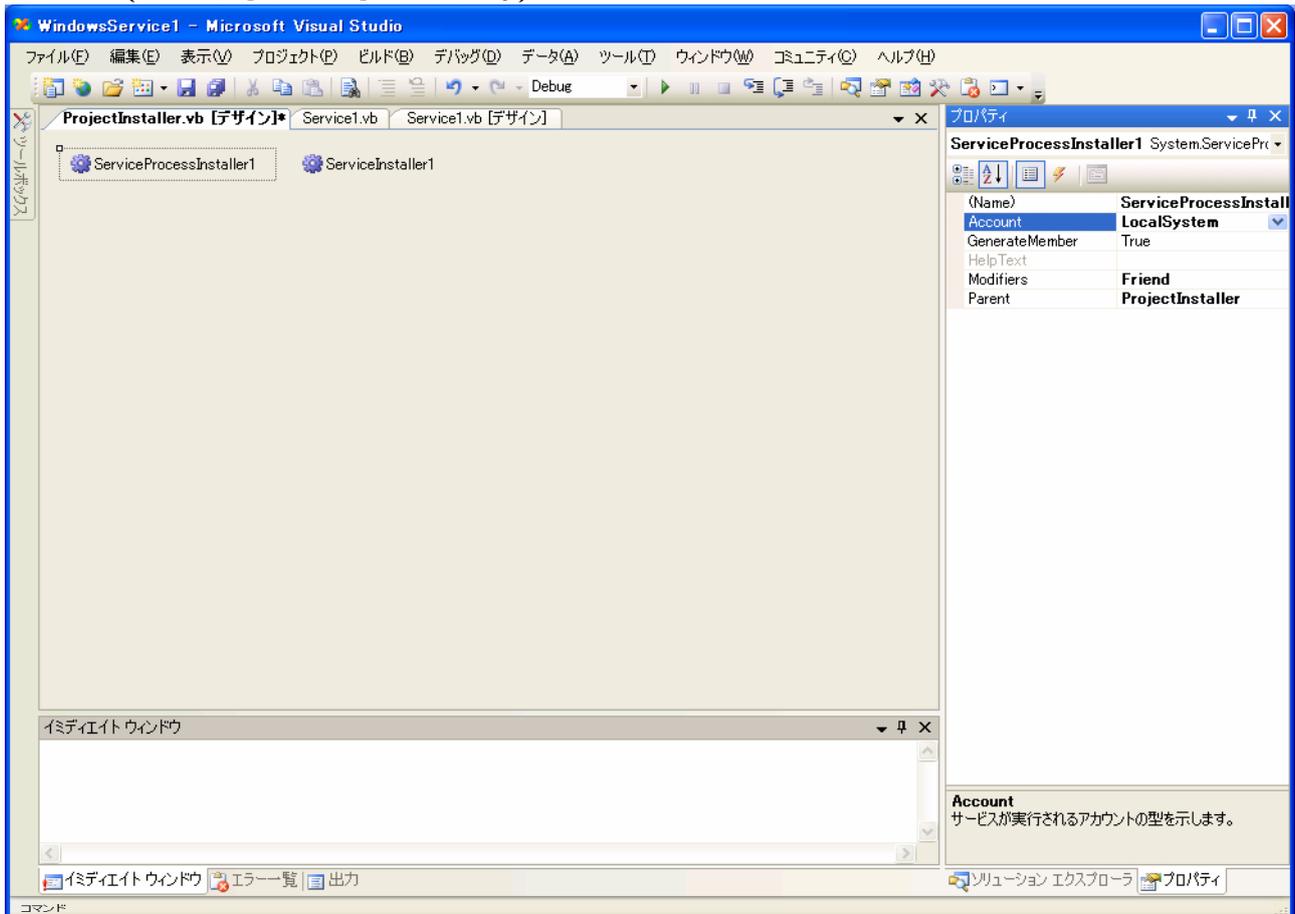
2. 「Windowsサービス」プロジェクトへの「インストーラーの追加」
(1) 「Service1.vb」の「デザイン」画面をマウス右クリックして「インストーラーの追加」を実行



- (2) 「ServiceInstaller1」のプロパティを確認 (ServiceNameに注意)

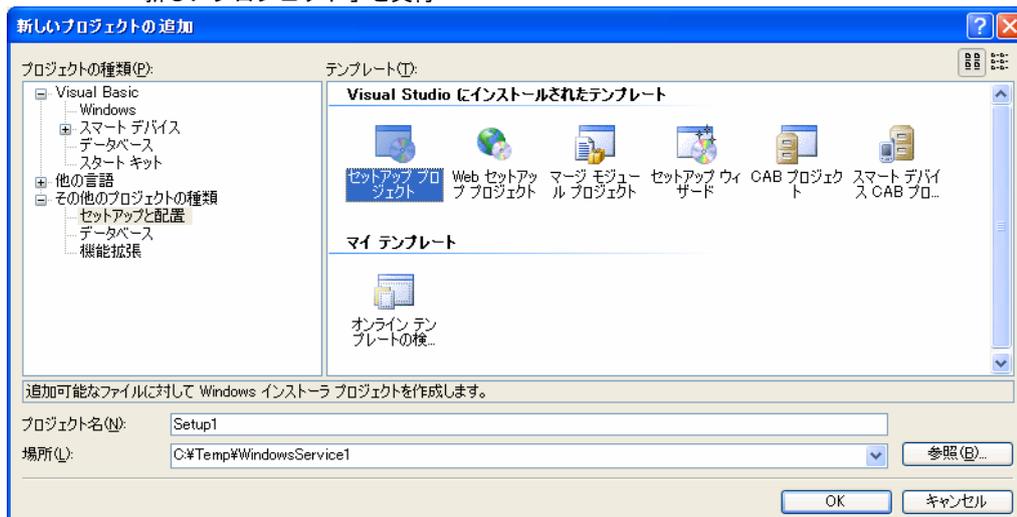


- (3) 「ServiceProcessInstaller1」のプロパティを確認、「Account」を「LocalSystem」に変更
(「Account」を「User」のままでも可)

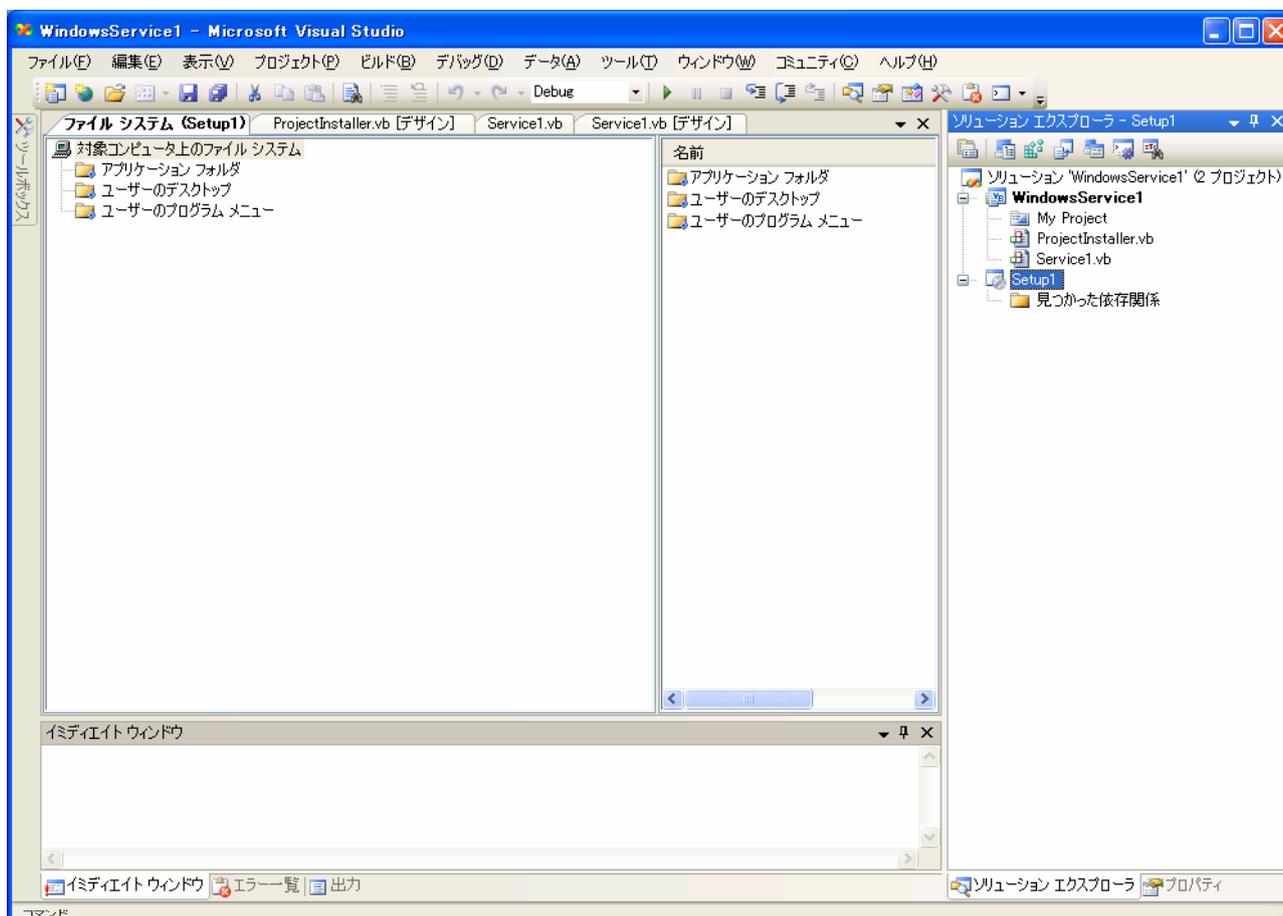


3. 「セットアップ プロジェクト」の作成

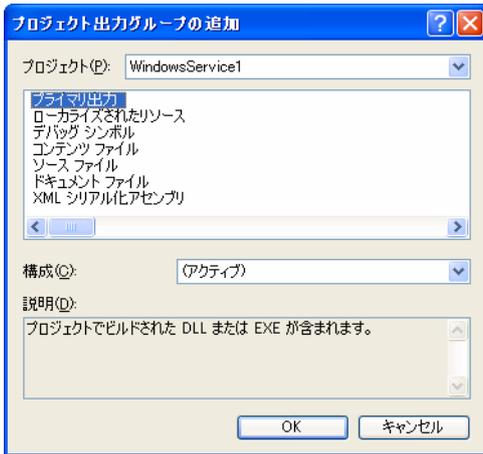
- (1) 「ソリューション エクスプローラ」の「WindowsService1」をマウス右クリックして「追加」～「新しいプロジェクト」を実行



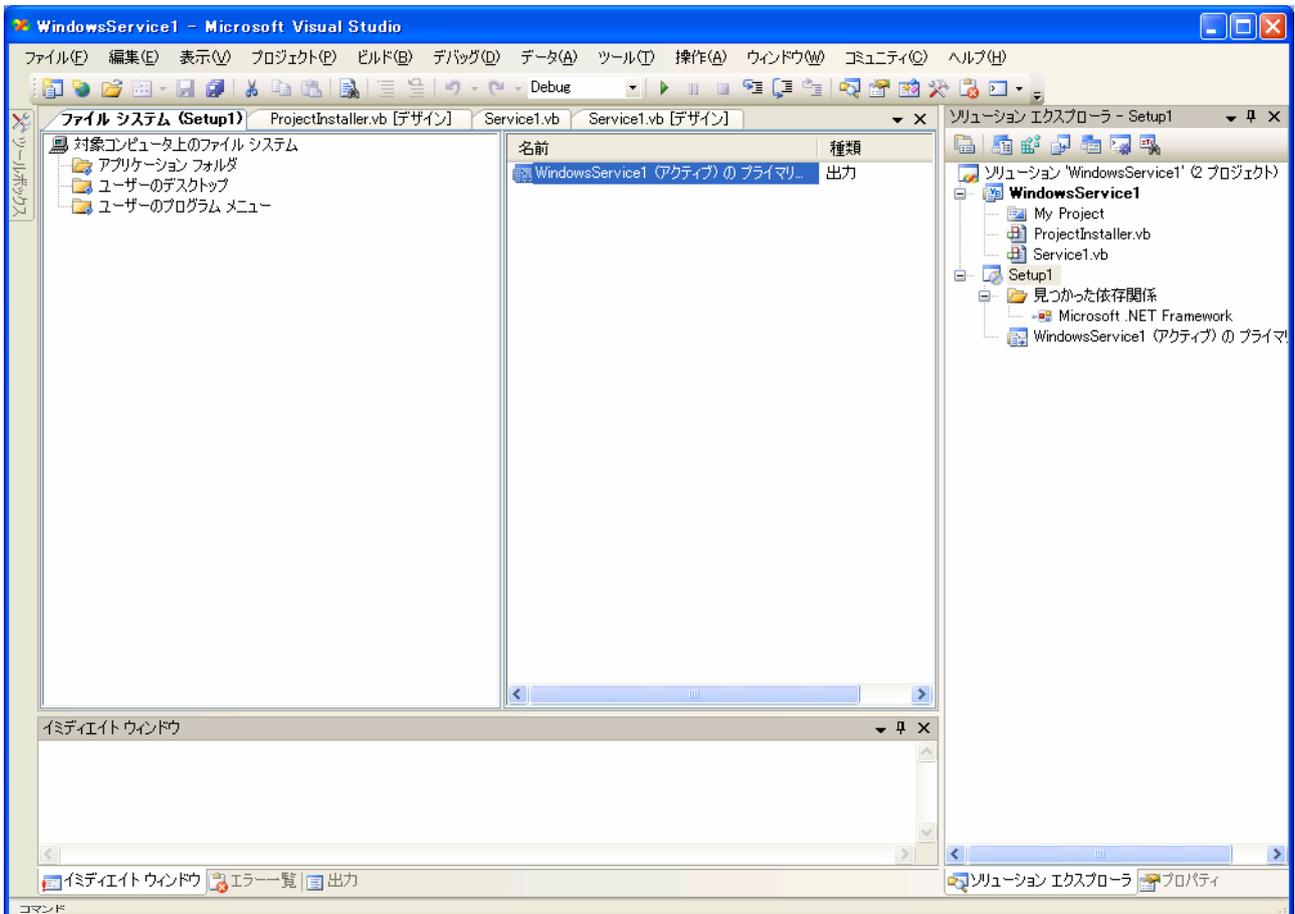
- (2) 「ファイル システム (Setup1)」タブの「その他のプロジェクトの種類」～「セットアップと配置」～「セットアップ プロジェクト」を選択し、「OK」をクリック



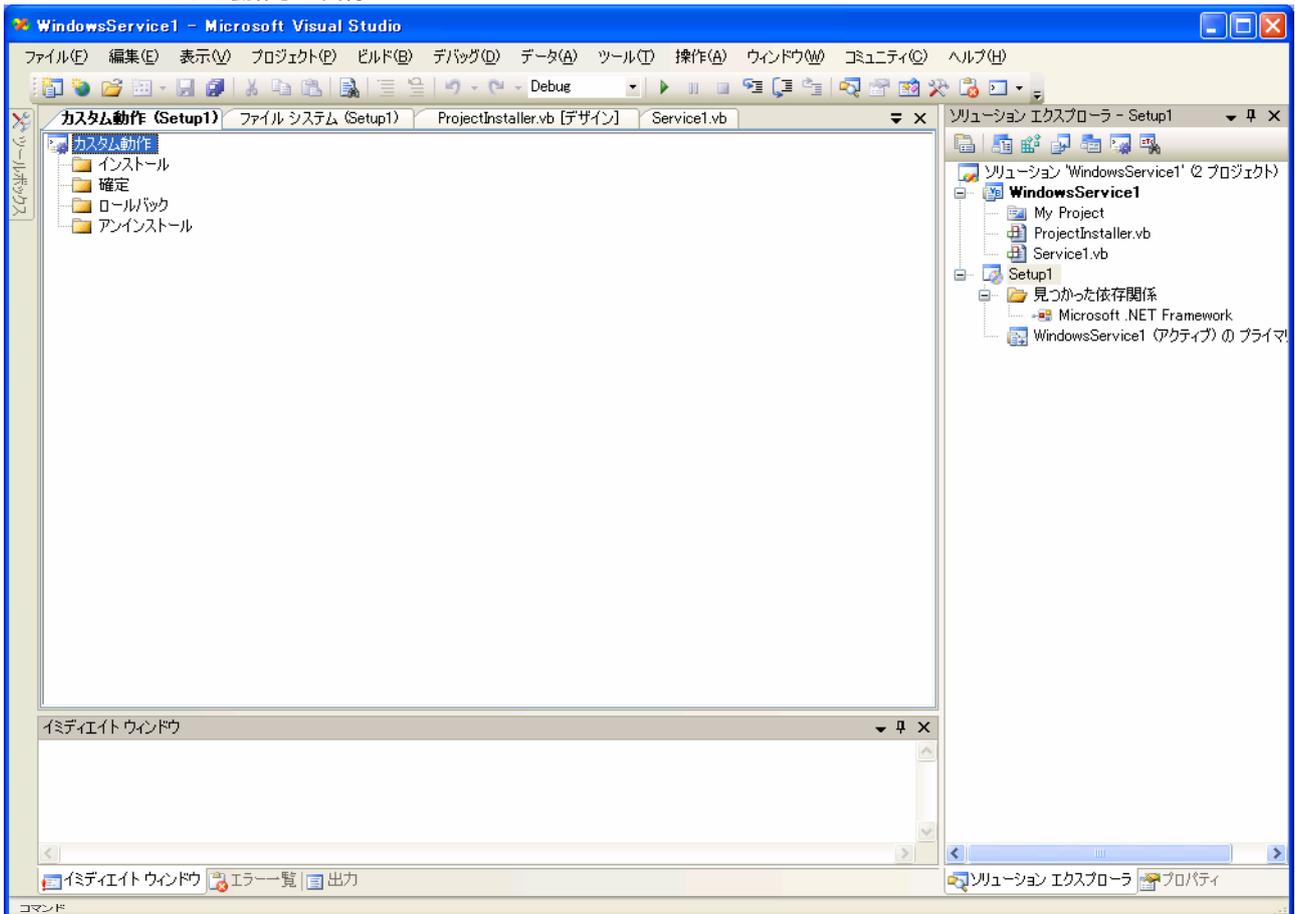
- (3) 「対象コンピュータ上のファイル システム」～「アプリケーション フォルダ」をマウス右クリックし、「追加」～「プロジェクト出力」を実行



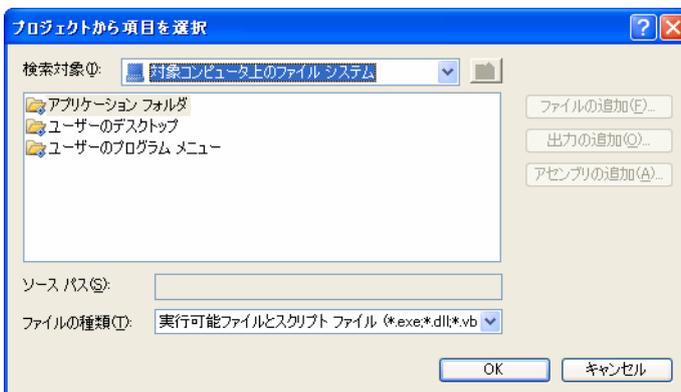
(4) 「プライマリ出力」を選択して「OK」をクリック



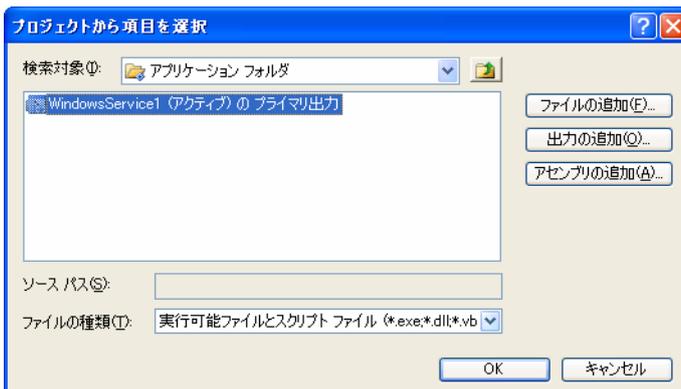
- (5) 「ソリューション エクスプローラ」の「Setup1」を選択して、「表示」メニューの「エディタ」～「カスタム動作」を実行



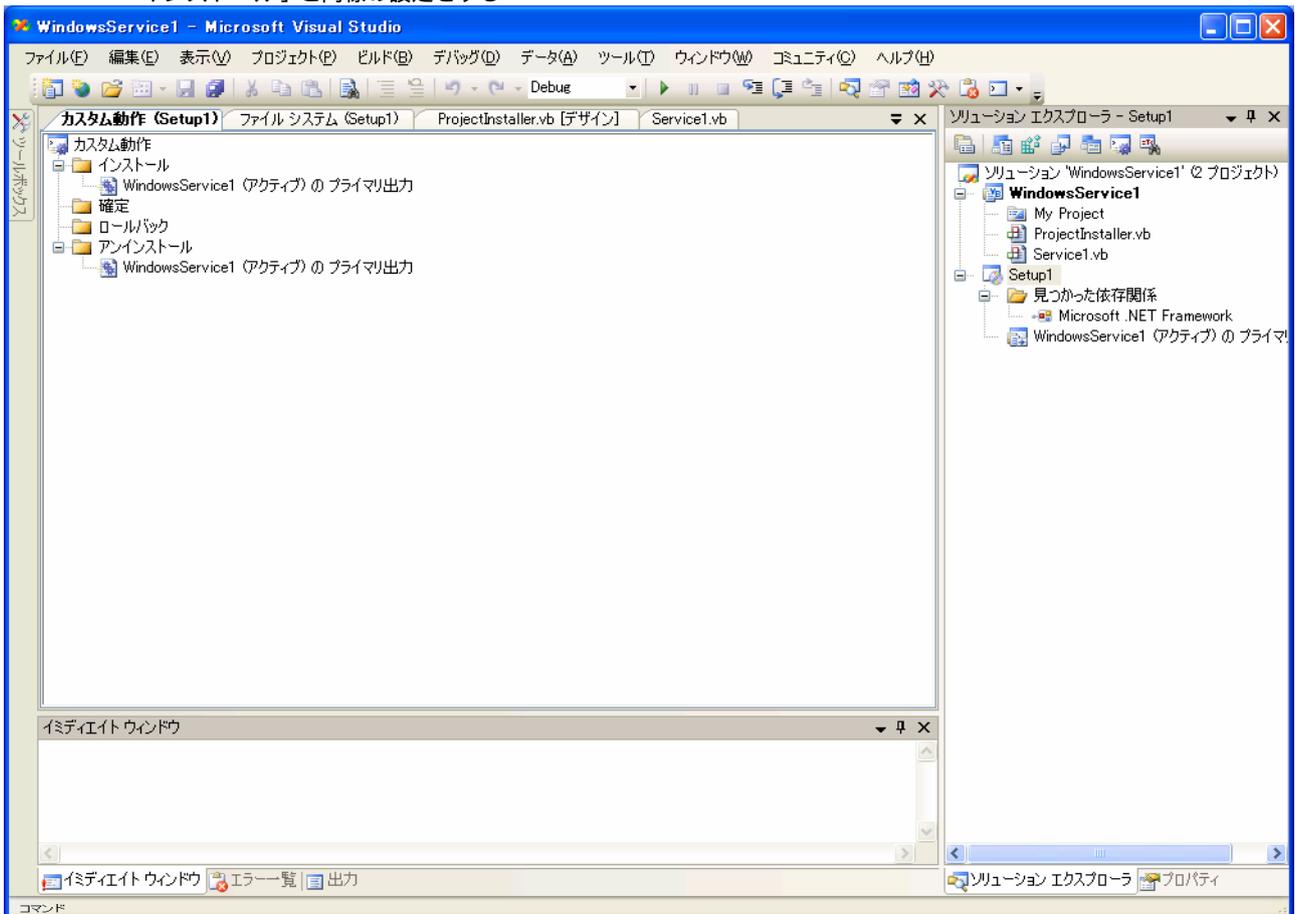
- (6) 「カスタム動作 (Setup1)」タブの「カスタム動作」～「インストール」をマウス右クリックし、「カスタム動作の追加」を実行



- (7) 「アプリケーション フォルダ」をダブルクリック



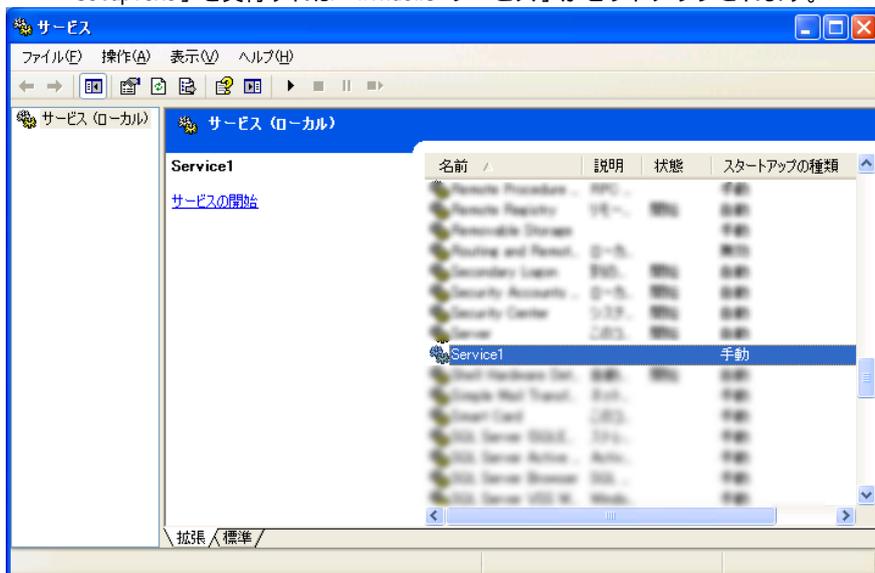
- (8) 「WindowsService1 (アクティブ) のプライマリ出力」を選択し、「OK」をクリック
「カスタム動作 (Setup1)」タブの「カスタム動作」～「アンインストール」についても
「インストール」と同様の設定をする



4. ビルド

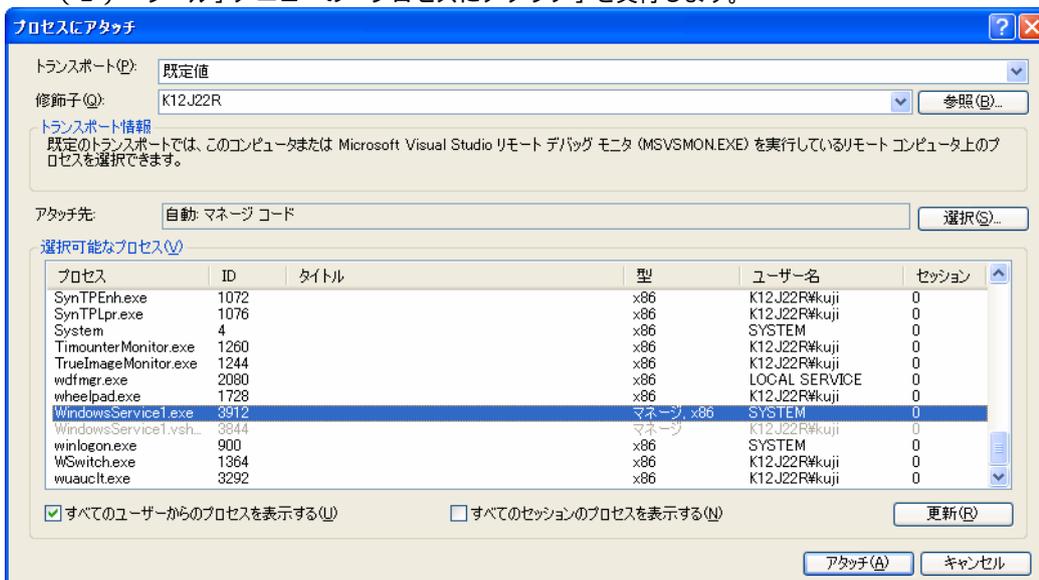
- (1) 「WindowsService1」プロジェクトと「Setup1」プロジェクトをリビルド
以上で「Windows サービス」プログラムと「セットアップ」の完成です。

「Setup1」プロジェクトに作成された「Debug」フォルダまたは「Release」フォルダに作成された「setup.exe」を実行すれば「Windows サービス」がセットアップされます。



5. デバッグ

- (1) 「Service1」サービスを開始します。
今回のサンプルではOnStartにデバック用に以下のようなスリープを記述しています。
System.Threading.Thread.Sleep(10000)
10秒以内に2の操作を行なうとOnStartのブレークポイントで停止させる事が出来ます。
- (2) 「ツール」メニューの「プロセスにアタッチ」を実行します。



- (3) 「すべてのユーザからのプロセスを表示する」をチェックし、一覧表示の中から「WindowsService1.exe」を選択し、「アタッチをクリックします」