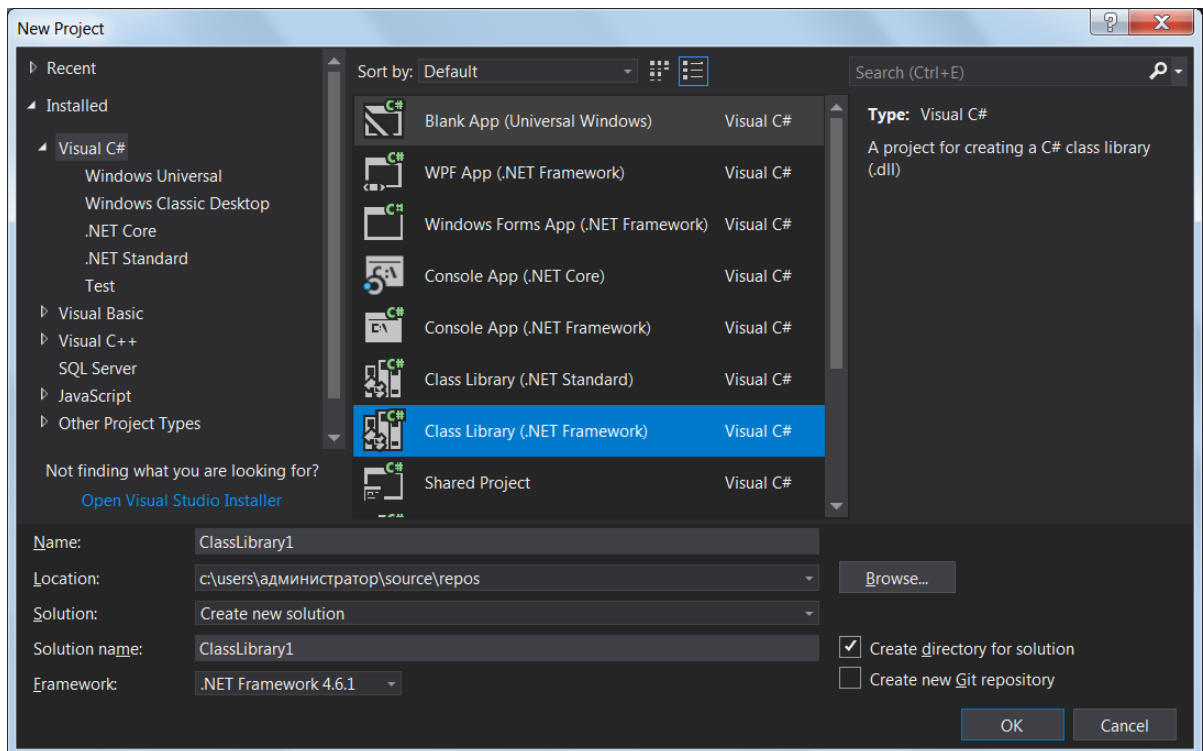


Создание dll-библиотеки в C#

Подключаемая dll-библиотека представляет из себя отдельный самостоятельный модуль, который может в готовом виде использоваться в других проектах. Для создания библиотеки откройте новый проект **Class Library (.NET Framework)**:

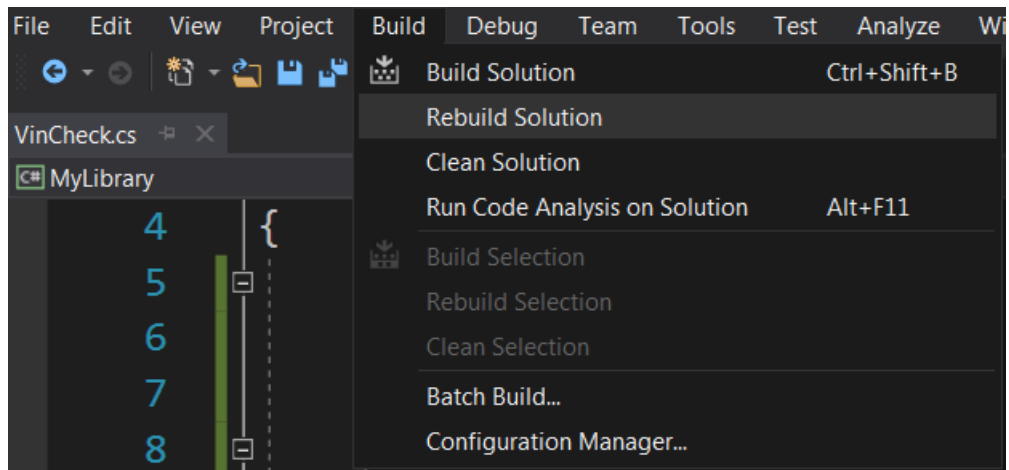


Реализуйте требуемую логику в **статических** методах **статического** класса:

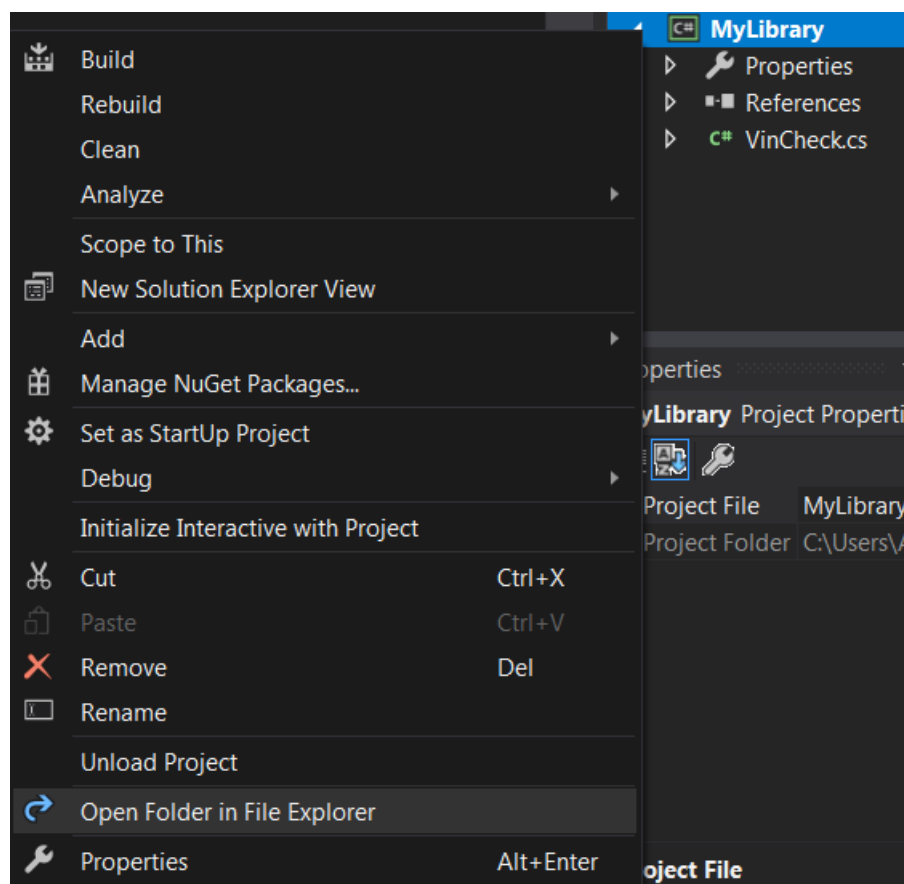
```
public static class VinCheck
{
    public static bool CheckVIN(String vin) {
        if (vin.Length != 17)
            return false;
        String serial_number = vin.Substring(11);
        try {
            int x = Int32.Parse(serial_number);
        } catch {
            return false;
        }
        return true;
    }

    public static String GetArea(String vin) {
        char first = vin.ToCharArray()[0];
    }
}
```

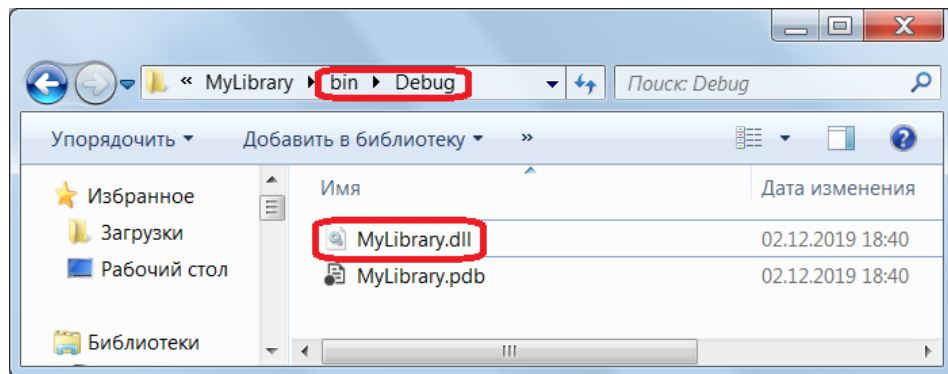
После того, как необходимая программная логика реализована, пересоберите проект через пункт меню **Build – Rebuild Solution**:



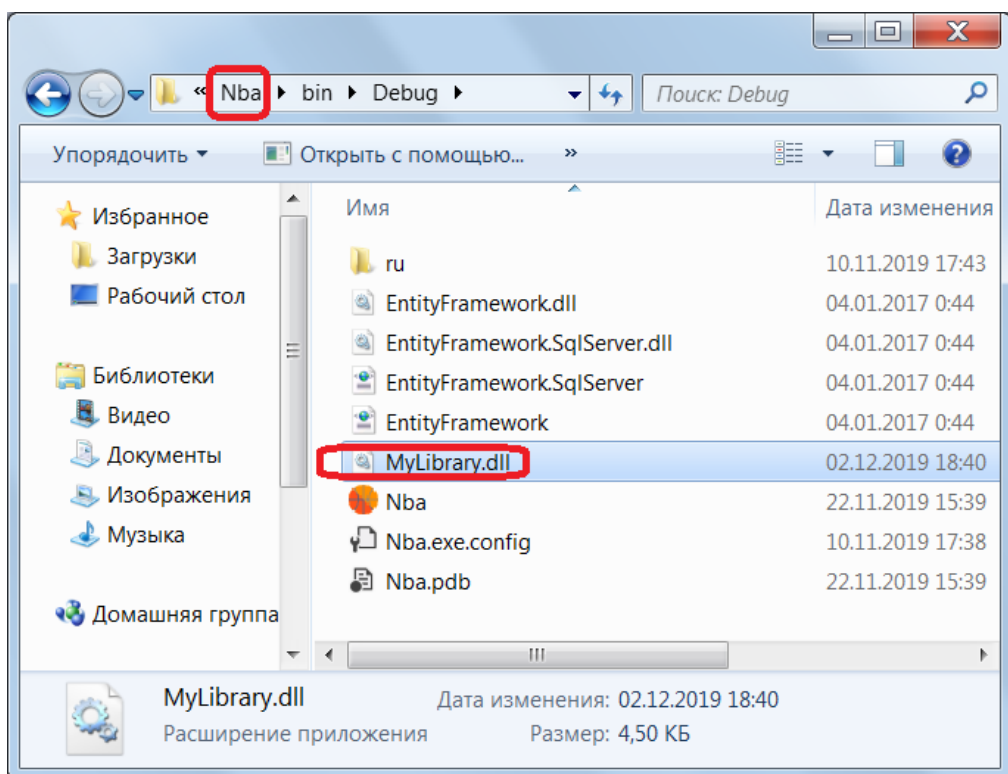
Результатом сборки должен стать dll-файл, который появится в директории проекта. Доступ к директории проекта можно получить, нажав правой кнопкой мыши на проекте и выбрав пункт **Open Folder in File Explorer**:



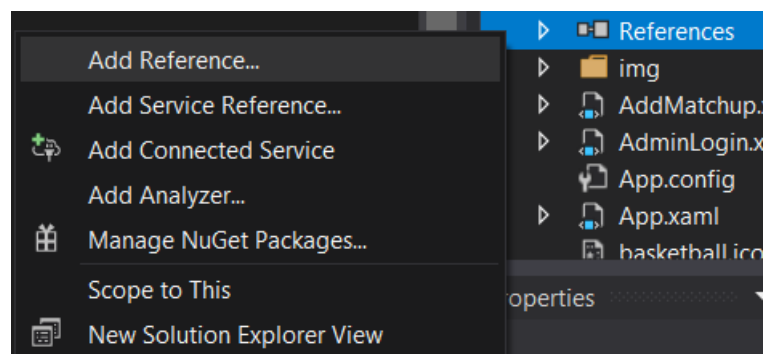
Dll-файл будет находиться в папке **bin\debug**:



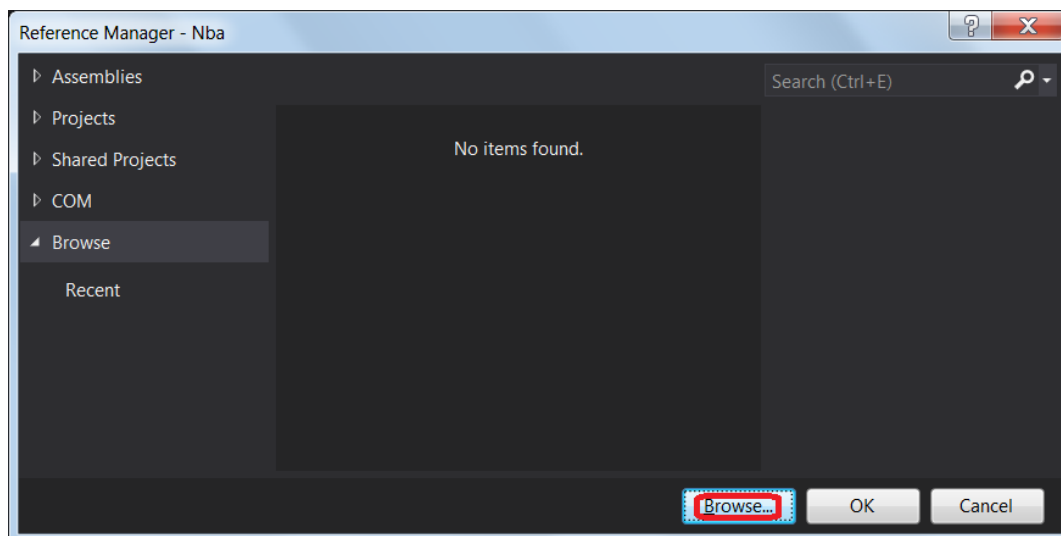
Самый простой способ подключения библиотеки заключается в копировании данного файла в директорию **bin\debug** другого проекта:



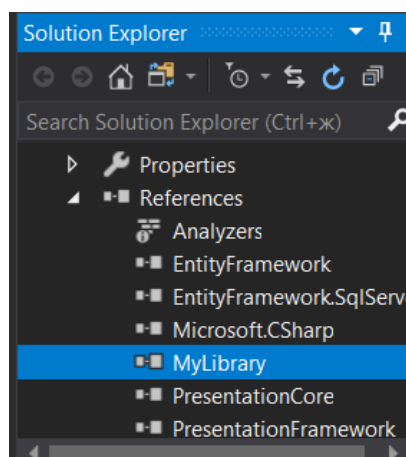
После копирования dll-файла необходимо добавить в проект ссылку на него через **Visual Studio**. Находясь в окне WPF-проекта, нажмите правой кнопкой мыши на раздел **References** и выберите пункт **Add Reference**:



Далее нажмите кнопку **Browse** и выберите dll-файл, который был скопирован в папку **bin\debug** текущего проекта на предыдущем шаге:



При успешном подключении библиотеки она появится в разделе **References**:



После этого можно импортировать библиотеку в код проекта через директиву **using**:

```
using System.Windows.Navigation;
using System.Data.Entity;
using MyLibrary;

namespace Nba {

    public partial class AddMatchup : Page {
        int seasonId;
        int matchupTypeId;
        int teamAwayId;
        int teamHomeId;
```

Теперь в коде другого проекта можно работать с методами созданной библиотеки:

```
teamHomeViewSource = ((CollectionViewSource)(FindResource  
DataContext = this;  
bool x = VinCheck.CheckVIN("asfaf123adf67");  
}  
  
private void Page_Loaded(object sender, RoutedEventArgs e) {  
    //Обновить заголовок окна  
    Window.GetWindow(this).Title = "Add a new matchup for re
```