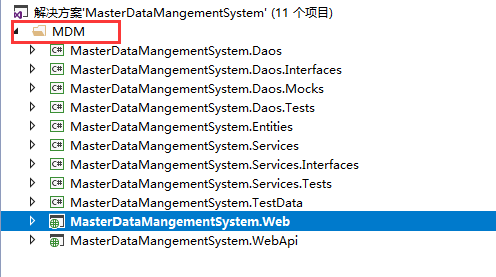
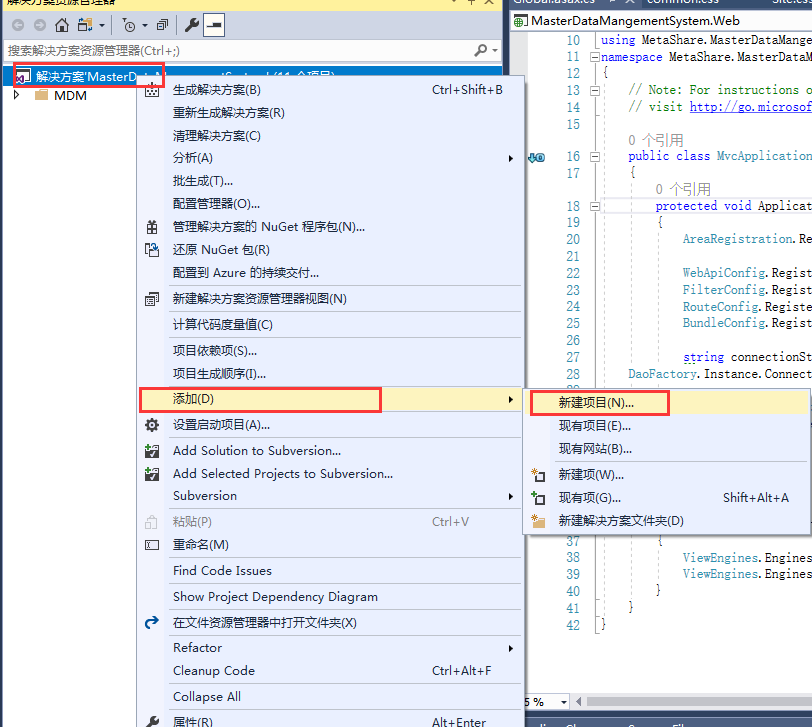
**MDM外挂项目研究方案**

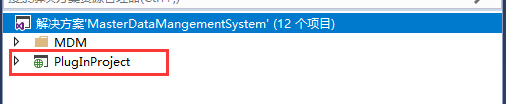
**(针对研究方案中的4、5、6步，页面及相关的css、js文件需要同步到外挂项目PlugInProject对应的文件夹位置，目前没有想到好的解决方案，目前可以想到的是后面生成的MDM文件替换PlugInProject项目里面相关文件时保留修改过的文件)**

1. 新建解决方案文件夹MDM，将MDD生成MasterDataMangementSystem的项目放到MDM中



1. 在同一个MDM解决方案的同级目录下面添加新项目：外挂项目（例如：PlugInProject）





1. 向外挂项目PlugInProject里面添加引用，需要添加的引用如下：

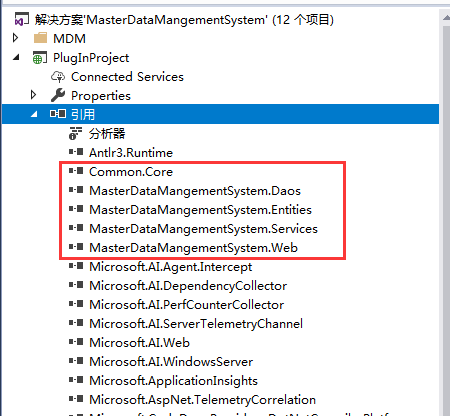
Common.Core.dll

MasterDataMangementSystem.Daos

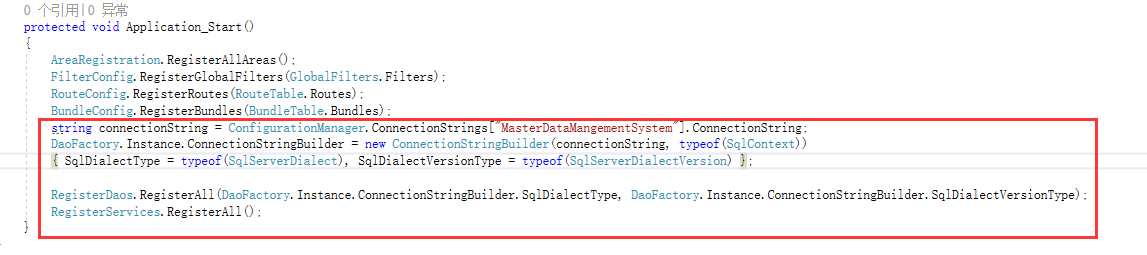
MasterDataMangementSystem.Entities

MasterDataMangementSystem.Services

MasterDataMangementSystem.Web



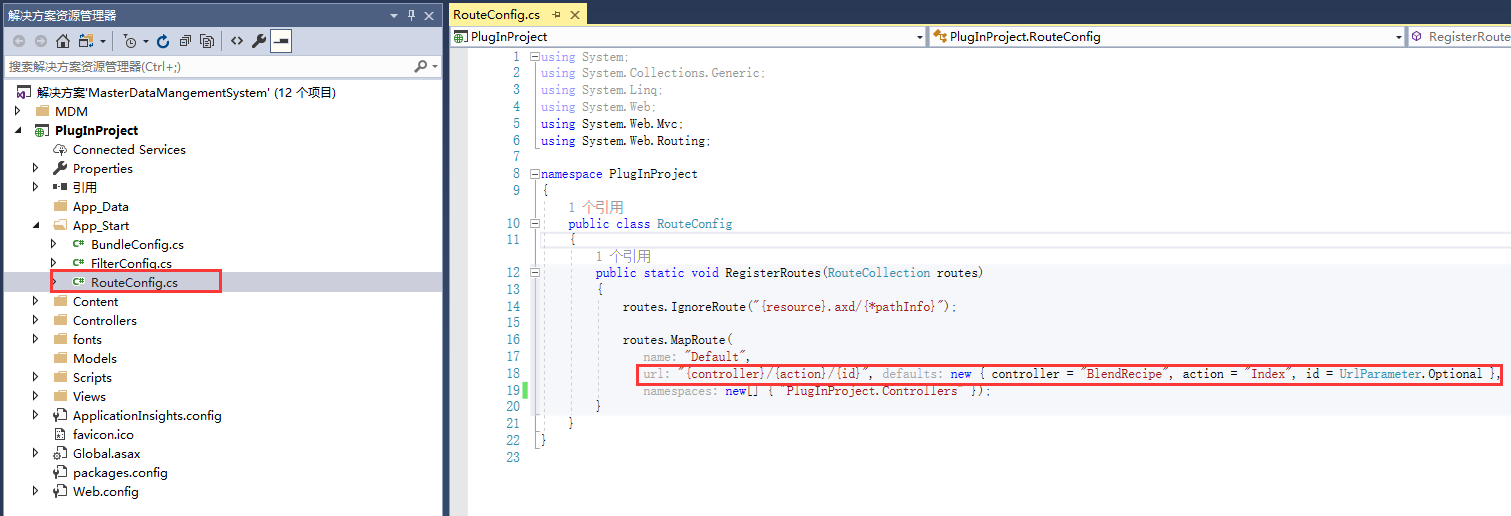
1. 将PlugInProject项目里的Views文件替换成MasterDataMangementSystem.Web项目下的Views文件
2. 将PlugInProject项目里的Content文件替换成MasterDataMangementSystem.Web项目下的Content文件
3. 将PlugInProject项目里的Scripts文件替换成MasterDataMangementSystem.Web项目下的Scripts文件
4. 在PlugInProject项目的Global.asax文件的Application\_Start()方法添加数据库连接以及Dao层和Service层的注册，



1. 在PlugInProject项目的配置文件Web.config添加数据库的连接配置参数



1. 在PlugInProject项目的RouteConfig.cs文件中修改项目运行起来的默认方法

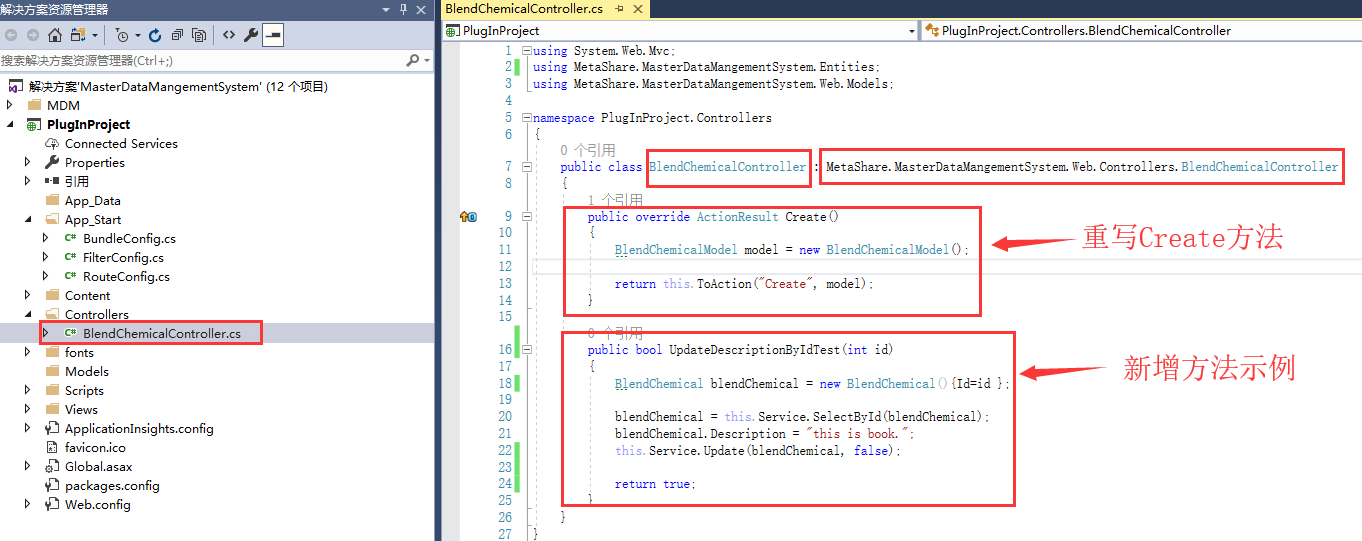


1. 将项目MasterDataMangementSystem运行，之后运行外挂项目PlugInProject，就可以在外挂

项目上管理MasterDataMangementSystem项目的数据和页面样式。

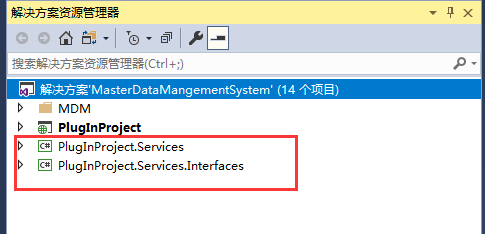
**下面的方案是在外挂项目PlugInProject上扩展MDM某个Controller（重写Controller某个原有方法，或者根据具体业务需要新增方法）以BlendChemicalController为例：**

1. 首先在PlugInProject项目的Controllers文件夹下面新建BlendChemicalController类(与需要扩展的MDM的类同名)，继承MDM的BlendChemicalController，如下图所示，如需在原有的方法里面增加业务，可以通过重写方法来实现，也可以根据具体业务需求增加新的方法。



**下面的方案是在外挂项目PlugInProject上扩展MDM某个Service（重写Service某个原有方法，或者根据具体业务逻辑处理需要新增方法）以BlendChemicalService为例：**

1. 在当前解决方案下添加名称为PlugInProject.Services和PlugInProject.Services.Interfaces的项目



1. 在PlugInProject.Services.Interfaces项目添加如下引用：

Common.Core.dll

MasterDataMangementSystem.Entities

MasterDataMangementSystem.Services.Interfaces

1. 在PlugInProject.Services项目添加如下引用：

Common.Core.dll

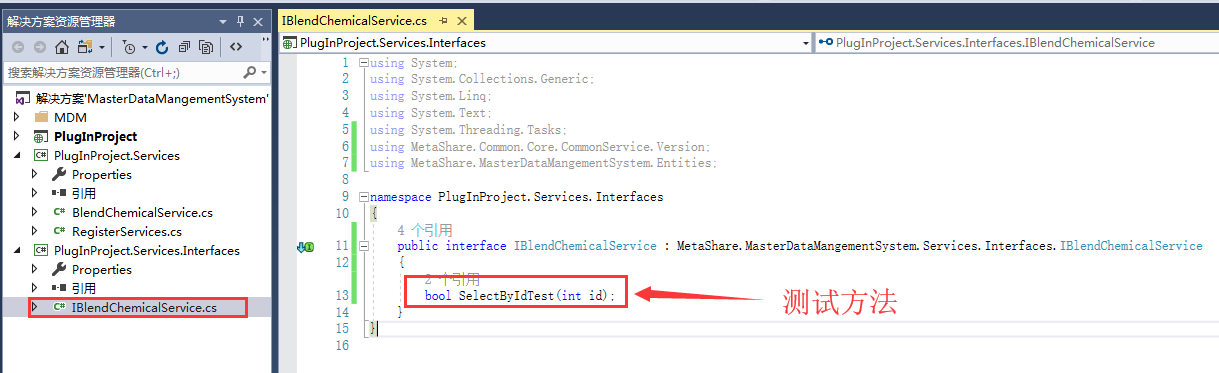
MasterDataMangementSystem.Entities

MasterDataMangementSystem.Services

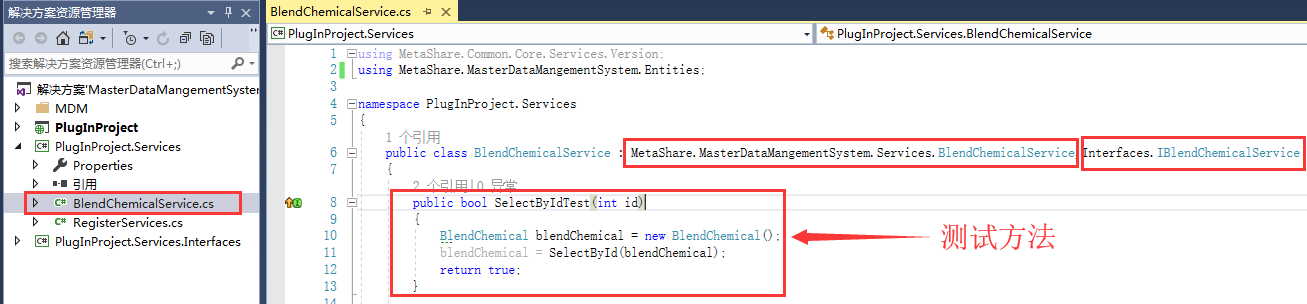
MasterDataMangementSystem.Services.Interfaces

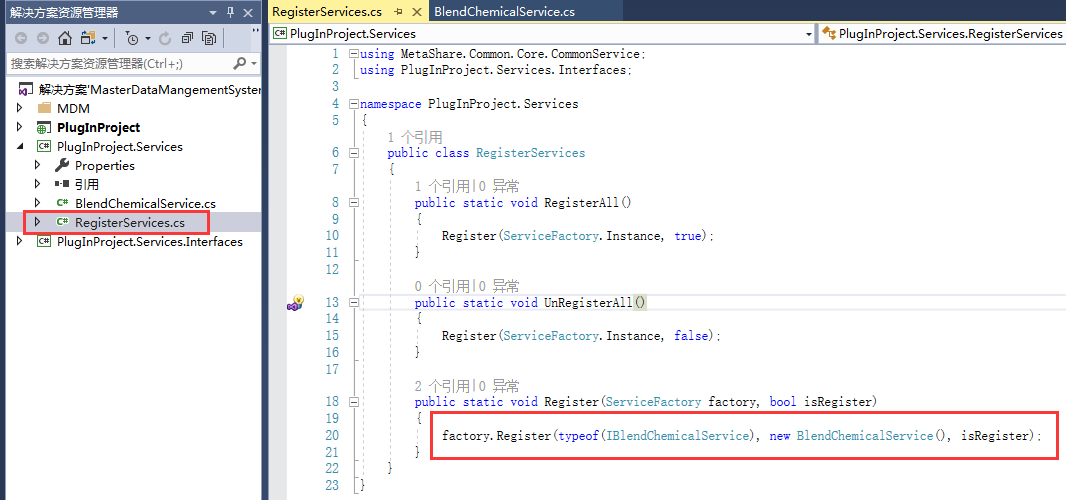
PlugInProject.Services.Interfaces

1. 在PlugInProject.Services.Interfaces加接口IBlendChemicalService.cs



1. 在PlugInProject.Services加BlendChemicalService.cs和RegisterServices.cs，在RegisterServices里面注册BlendChemicalService



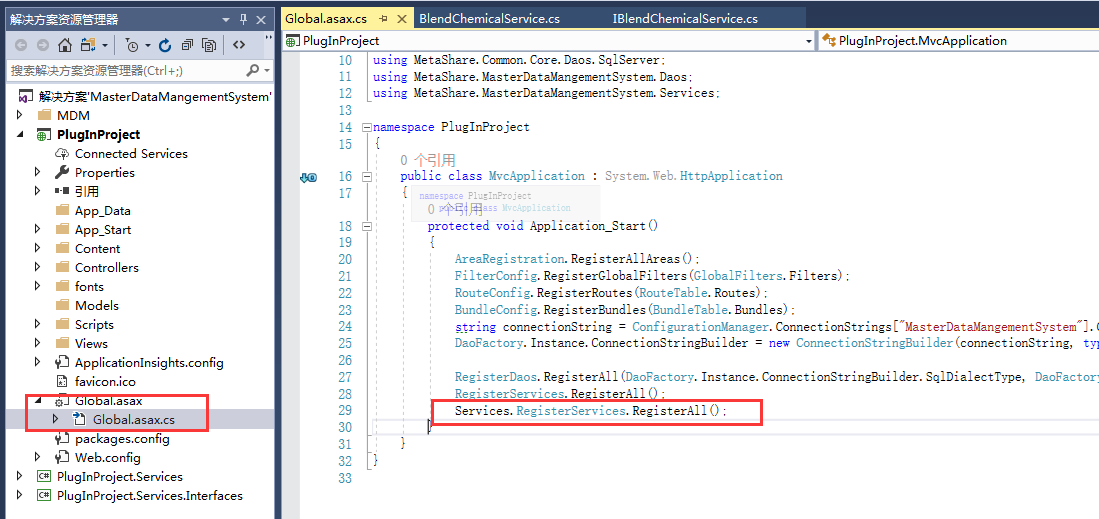


1. 在项目PlugInProject里面添加引用：

PlugInProject.Services

PlugInProject.Services.Interfaces

1. 在PlugInProject项目的Global.asax添加PlugInProject.Services的注册



1. 在PlugInProject项目的BlendChemicalController写一个构造函数，定义\_service为扩展后的IBlendChemicalService，就可以调用扩展出的方法了

