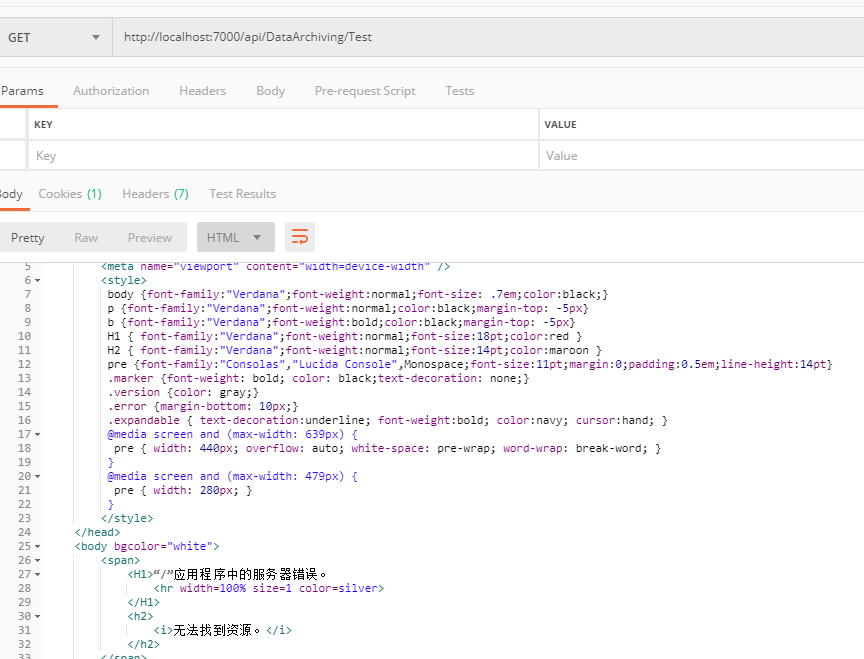
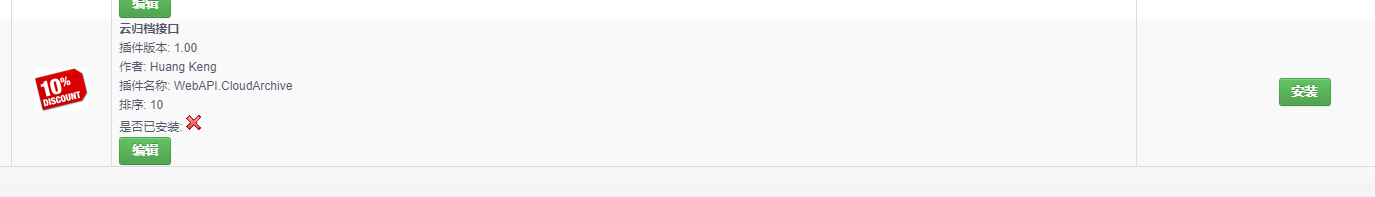
能力开放平台——正式动工

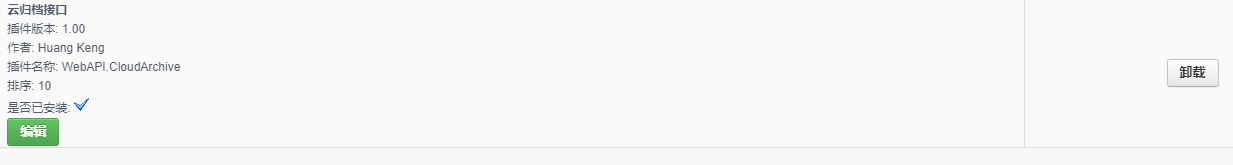
1.

接口无法找到资源，

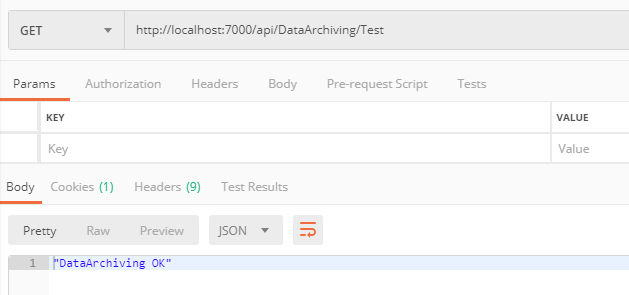


安装插件：





接口通了，开森。



2.

安装代码检查器。

除了eWorld.Entity之外，都给加上。

3.

添加第一个方法：

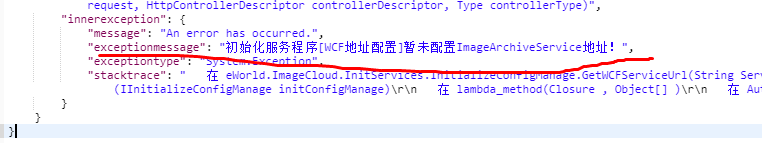
"error:System.NullReferenceException: 未将对象引用设置到对象的实例。\r\n 在 eWorld.Plugin.WebAPI.CloudArchive.Controllers.PublicController.Hua()"

应该是云归档的wcf没有配置地址。感觉。

4.

调用crm的wcfClient没有问题，ImageArchive就有问题了。

查了错误，是没有配置。



直接去源代码里面找：

/// <summary>

/// 获取WCF地址，如CRMService

/// </summary>

/// <param name="ServiceName">服务名称</param>

/// <param name="ReturnCompleteWCFUrl">是否返回完整的WCF地址，false仅返回IP</param>

/// <returns></returns>

public string GetWCFServiceUrl(string ServiceName, bool ReturnCompleteWCFUrl = true)

{

if (\_cacheWCFConnectSetting == null || \_cacheWCFConnectSetting.Count <= 0)

\_cacheWCFConnectSetting = InvokeApiGetWCFConnectString();

var currentWCFItem = \_cacheWCFConnectSetting.FirstOrDefault(s => s.Name == ServiceName);

if (currentWCFItem == null || string.IsNullOrEmpty(currentWCFItem.Url))

没有配置，是这个报错

throw new Exception(string.Format("初始化服务程序[WCF地址配置]暂未配置{0}地址！", ServiceName));

Uri hostUri = new Uri(currentWCFItem.Url);

if (!ReturnCompleteWCFUrl)

return hostUri.ToString();

else

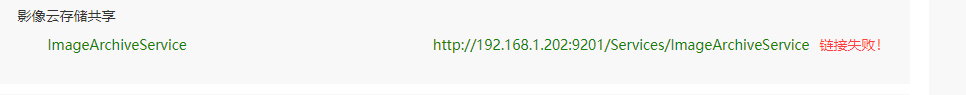
{

Uri retUri = new Uri(hostUri, string.Format("/Services/{0}.svc", ServiceName));

return retUri.ToString();

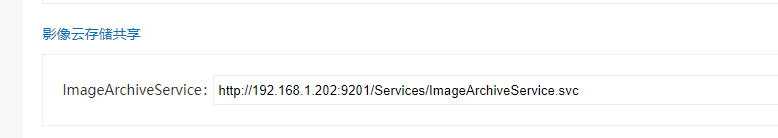
}

}



应该是这个的缘故，我改下配置。

加了svc



去202获取这个记录数据的xml文件，对的啊，有值的。

http://192.168.1.202:9201/Services/ImageArchiveService.svc

//初始化配置工具、WCF客户端调用

builder.RegisterType<InitializeConfigManage>().As<IInitializeConfigManage>().WithParameters(new List<NamedParameter>()

{

new NamedParameter("systemCode",1100),

new NamedParameter("initServiceSystemKey","bac11f3bd919f9fb")

}).SingleInstance();

知道原因了：

public enum SystemCodeEnum

{

门户 = 10,

公共服务 = 20,

远程医疗 = 100,

网络医院 = 200,

影像存储 = 300,

医学教育 = 400,

典型病例 = 500,

影像质控 = 600,

科研协同 = 700,

智能诊断 = 800,

影像大数据 = 900,

客户关系管理 = 1100,

运营管理 = 1200,

机构管理 = 1210,

在线监控 = 1300,

数据挖掘 = 1400,

能力开放 = 9100

}

这里是根据code的编码获取的，我这里给的是1100，所以我要换成300才行。

用军哥的1100获取的结果：

{

"data": [

{

"Name": "CommonService",

"Url": "http://192.168.1.202:9002/Services/CommonService.svc"

},

{

"Name": "MediaService",

"Url": "http://192.168.1.202:9002/Services/MediaService.svc"

},

{

"Name": "CRMService",

"Url": "http://192.168.1.202:9101/Services/CRMService.svc"

},

{

"Name": "PaymentService",

"Url": "http://192.168.1.202:9101/Services/PaymentService.svc"

}

],

"code": 0,

"msg": "操作成功"

}

用云归档的300获取的结果：

{

"data": [

{

"Name": "CommonService",

"Url": "http://192.168.1.202:9002/Services/CommonService.svc"

},

{

"Name": "MediaService",

"Url": "http://192.168.1.202:9002/Services/MediaService.svc"

},

{

"Name": "CRMService",

"Url": "http://192.168.1.202:9101/Services/CRMService.svc"

},

{

"Name": "ImageArchiveService",

"Url": "http://192.168.1.202:9201/Services/ImageArchiveService.svc"

}

],

"code": 0,

"msg": "操作成功"

}

开始报新的错：



两个方法重名了：

"exceptionmessage": "An exception was thrown while invoking the constructor 'Void .ctor(eWorld.ImageCloud.InitServices.IInitializeConfigManage)' on type 'eWorldWCFClient`1'. ---> 同一个协定中不能存在两个名称相同的操作，类型为 eWorld.IDCAS.WCFServerContract.ServerContract.IImageArchiveService 的方法 GetPatientExamDocumentList 和 GetPatientExamDocumentList 违反了此规则。可以通过更改方法名称或使用 OperationContractAttribute 的 Name 属性更改其中一个操作的名称。 (See inner exception for details.)",

暂时在方法后面加一个WithShort

/// <summary>

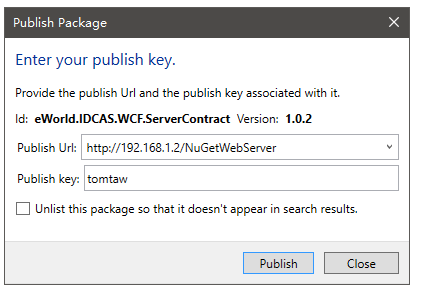
/// 获取文档信息(使用短链)

/// </summary>

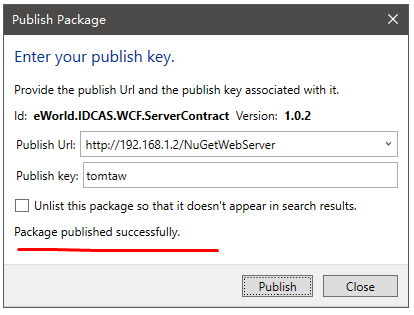
/// <param name="query">请求参数</param>

/// <returns>检查文档列表信息</returns>

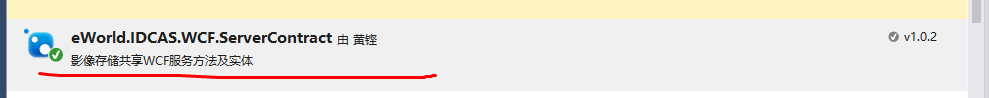
BaseListReturnResult<DocumentViewDTO> GetPatientExamDocumentListWithShort (DocumentQuery query);



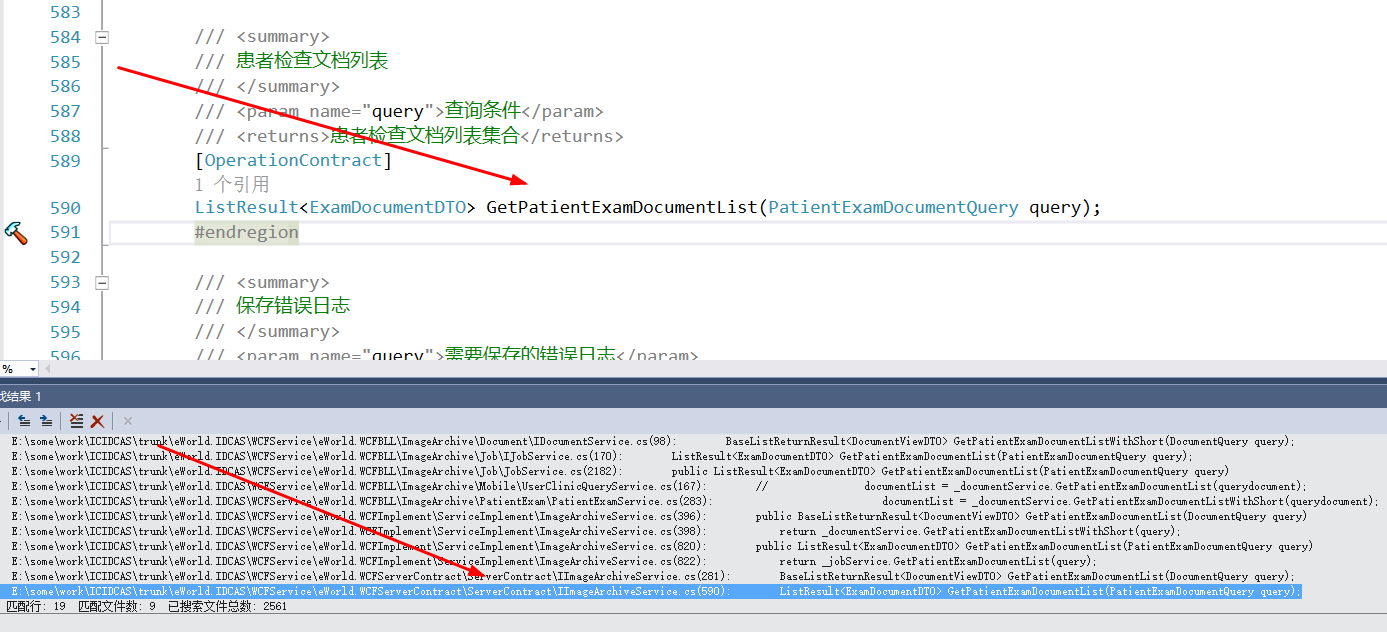
应该是成功了：



好了

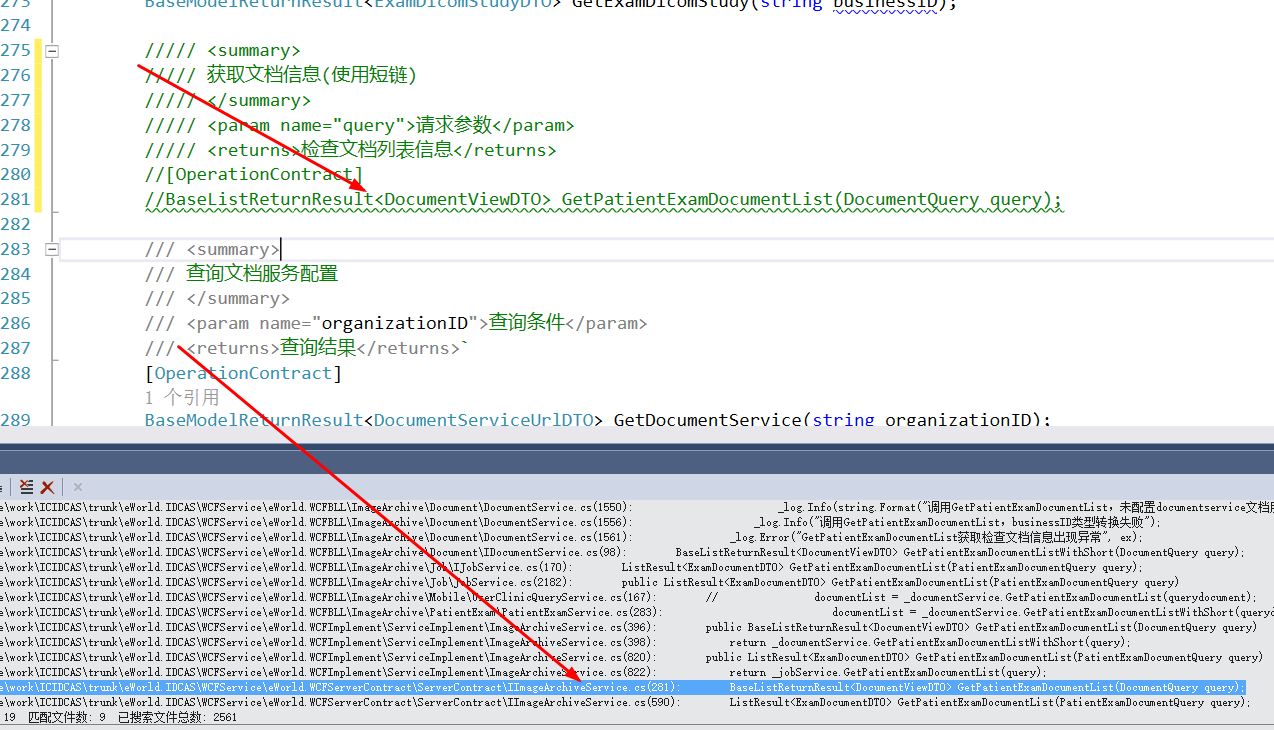


还有重名的，在ImageArchiveService中。先注释掉了。

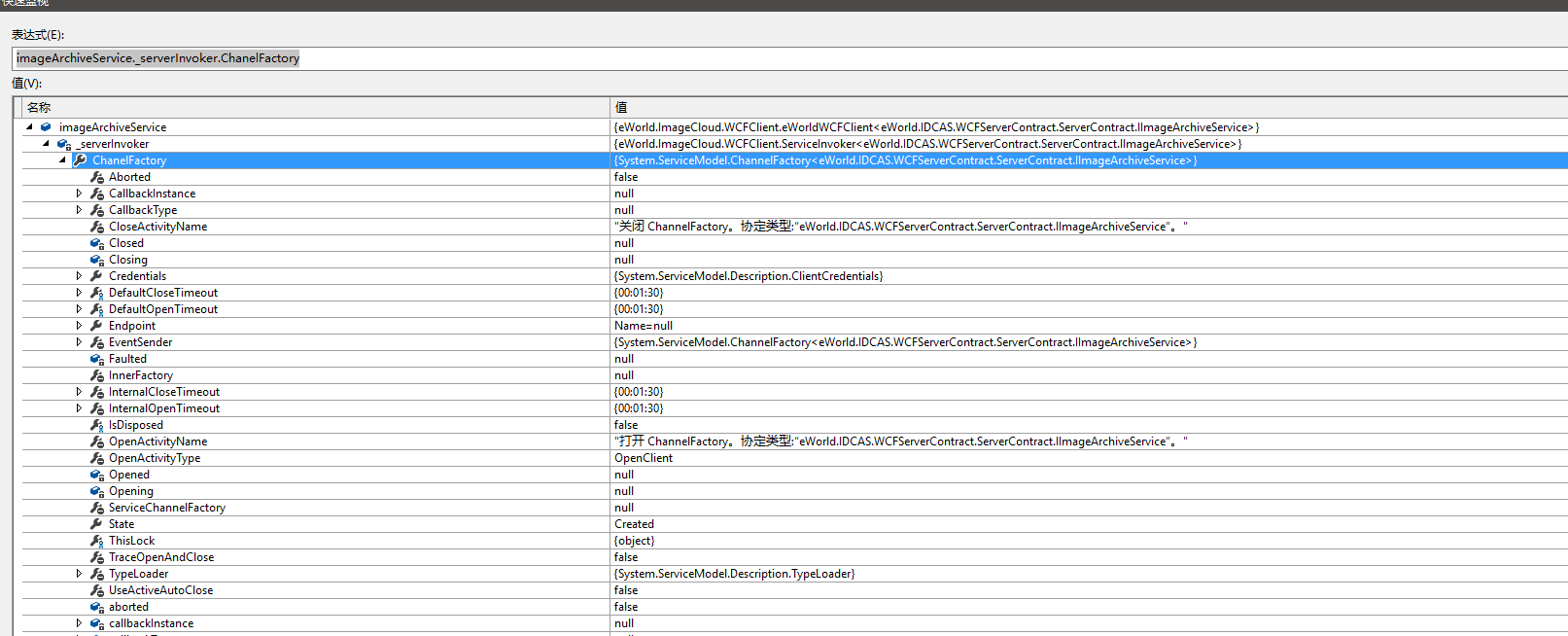


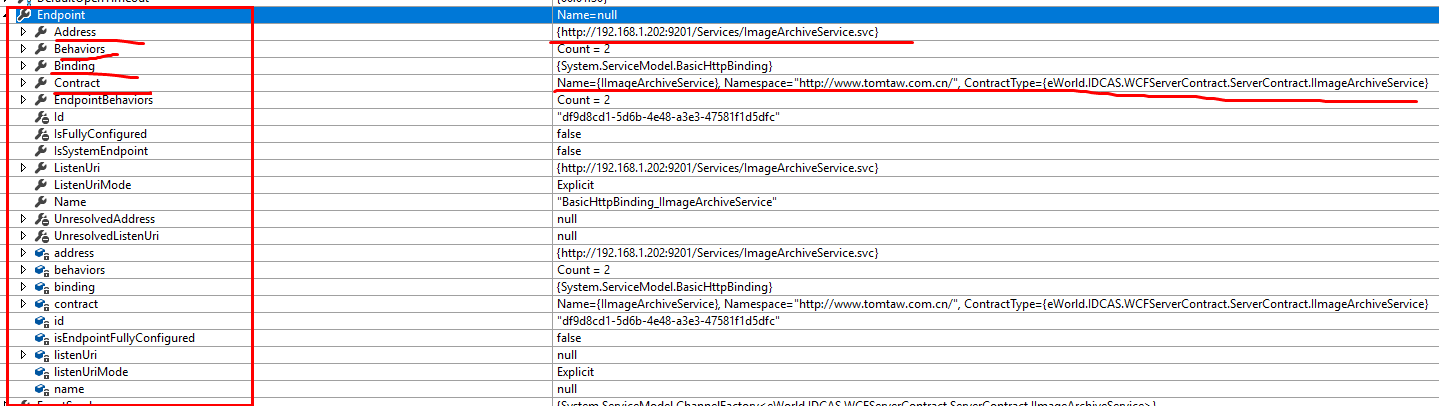
一共两个，一个是调用之前的DocumentService中的GetPatientExamDocumentList(DocumentQuery query)方法，（获取文档信息(使用短链)）

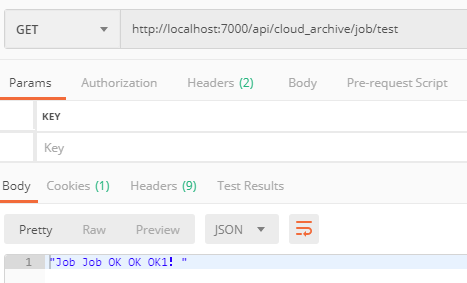
还有一个是调用JobService中的GetPatientExamDocumentList(PatientExamDocumentQuery query)方法。



来了来了，开森







Good Job！

调用的时候发生报错：



修改一下地址：

<http://localhost:9301/Services/ImageArchiveService.svc>

<http://192.168.1.202:9201/Services/ImageArchiveService.svc>

地址变成了本地，但是报错啊：

None of the constructors found with

'Autofac.Core.Activators.Reflection.DefaultConstructorFinder'

on type 'eWorld.IDCAS.WCF.BLL.Common.Configuration.DataDicInfoService'

can be invoked with the available services and parameters:Cannot resolve parameter 'eWorld.Core.Data.ICommonRepository`1[eWorld.Entity.Domain.Common.Configuration.DataDicInfo] datadicinfoRepository'

of constructor

'Void .ctor(eWorld.Core.Data.ICommonRepository`1[eWorld.Entity.Domain.Common.Configuration.DataDicInfo]

使用可用的服务和参数可以调用在

'eWorld.IDCAS.WCF.BLL.Common.Configuration.DataDicInfoService'类型中使用'Autofac.Core.Activators.Reflection.DefaultConstructorFinder'找到的构造函数：

无法解析参数

'eWorld.Core.Data.ICommonRepository`1 [eWorld.Entity.Domain.Common.Configuration.DataDicInfo] datadicinfoRepository'的构造函数

'Void.ctor（eWorld.Core.Data.ICommonRepository .1

[eWorld.Entity.Domain.Common.Configuration.DataDicInfo]]

"None of the constructors found with 'Autofac.Core.Activators.Reflection.DefaultConstructorFinder' on type 'eWorld.IDCAS.WCF.BLL.Common.Configuration.DataDicInfoService' can be invoked with the available services and parameters:\r\nCannot resolve parameter 'eWorld.Core.Data.ICommonRepository`1[eWorld.Entity.Domain.Common.Configuration.DataDicInfo] datadicinfoRepository' of constructor 'Void .ctor(eWorld.Core.Data.ICommonRepository`1[eWorld.Entity.Domain.Common.Configuration.DataDicInfo])'."

5.

6.

7.

8.