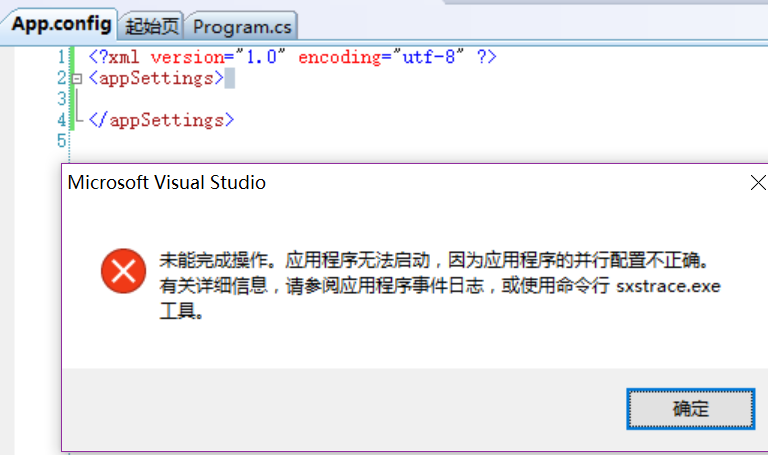
新建一个控制台程序，app.config没有设置对。能生成成功，但是无法运行



物流登陆程序报错，一查是类中的静态成员初始化异常

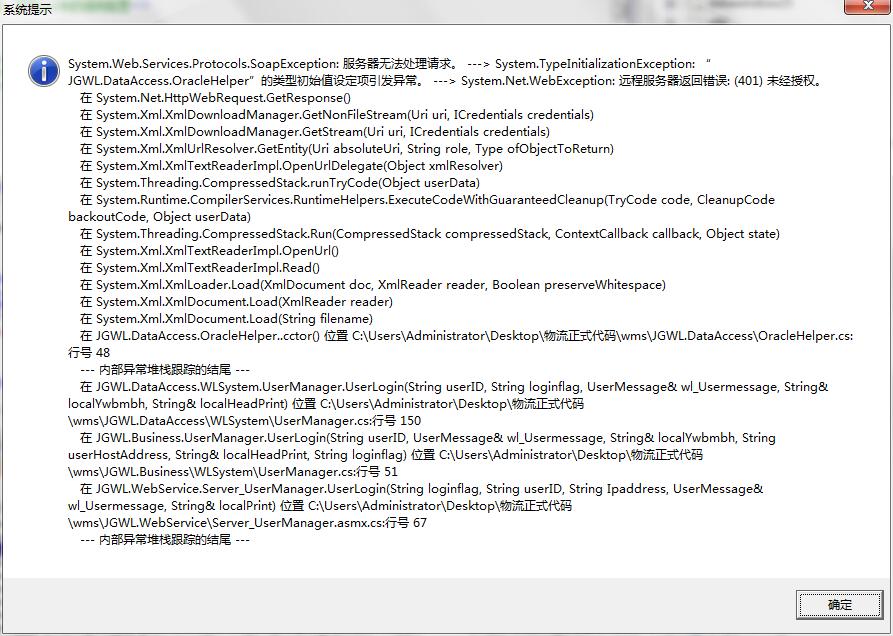
System.Web.Services.Protocols.SoapException: 服务器无法处理请求。 ---> System.TypeInitializationException: “JGWL.DataAccess.OracleHelper”的类型初始值设定项引发异常。 ---> System.Net.WebException: 远程服务器返回错误: (401) 未经授权。

public static XmlDocument dataAccessXml;

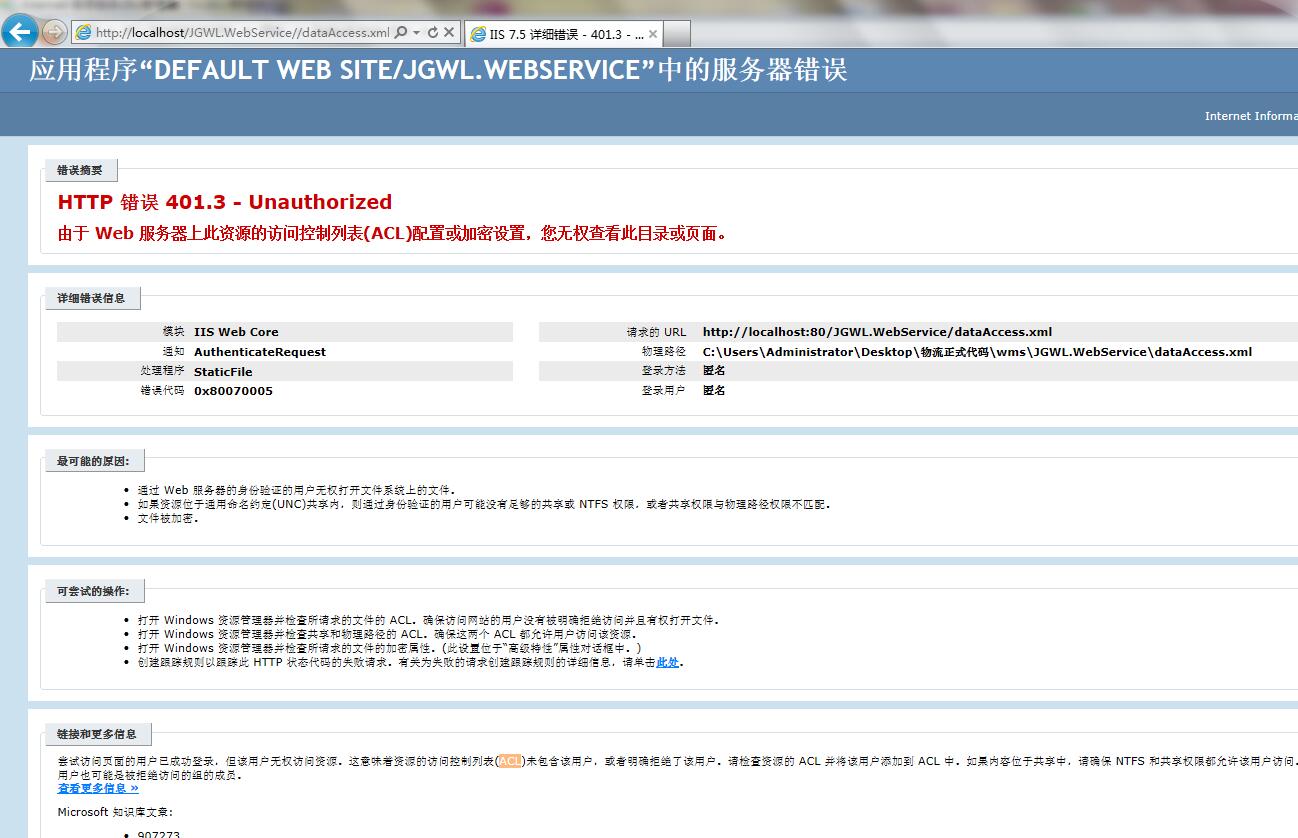
string path = "http://localhost/JGWL.WebService\\dataAccess.xml";

dataAccessXml = new XmlDocument();

dataAccessXml.Load(path);



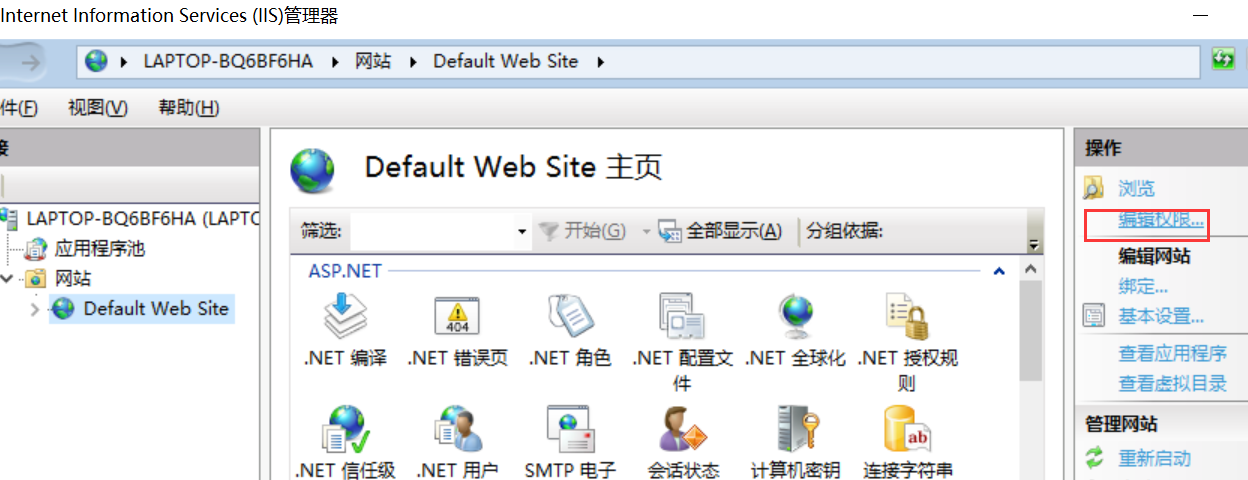
发现在iis中无法访问这个xml文件



网上查

在IIS中 依次执行如下操作：

网站——编辑权限——共享（为了方便可以直接将分享对象设置为everyone）——安全（直接勾选 everyone ）——应用——确定。



同时还右键xml文件，在属性里的安全设置了everyone的可写可读权限。最后可以了。但不知道是哪个方法解决的问题

SQLERRM是一则函数。oracle sqlerrm函数 sqlerrm函数返回指定[错误代码](http://baike.baidu.com/item/%E9%94%99%E8%AF%AF%E4%BB%A3%E7%A0%81)的错误信息。

sqlcode和sqlerrm是不能直接在sql语句中使用，必须先将其赋给变量后，才能在sql语句中使用，如下。

Error: PL/SQL: ORA-00904: "SQLERRM": 标识符无效

Line: 93

Text: update t\_changehw set errtext=sqlerrm where oldhw=c1.oldhw;

在函数中创建的timer也能每次都触发Tick事件

private void Form1\_Load(object sender, EventArgs e)

{

System.Windows.Forms.Timer timer1 = new Timer();

timer1.Interval = 1000;

timer1.Tick += new System.EventHandler(this.timer1\_Tick);

timer1.Enabled = true;

}

private void timer1\_Tick(object sender, EventArgs e)

{

MessageBox.Show("h");

}

在JGWL.WebService下创建Global.asax文件，程序启动时启动定时器。要有访问webservice的操作才会才会启动Application\_Start。访问RFserver这个webservice不会触发JGWL.WebService的Application\_Start。应用程序池的闲置超时时间到了以后工作进程关闭。定时器一起关闭

namespace JGWL.WebService

{

public class Global : System.Web.HttpApplication

{

protected void Application\_Start(object sender, EventArgs e)

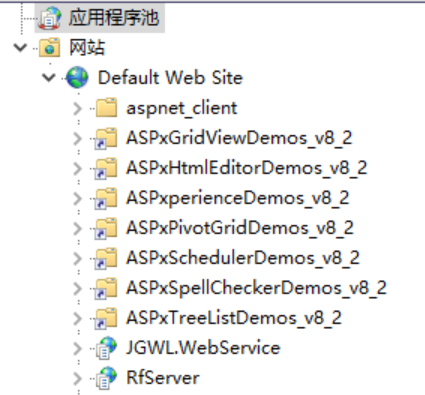
{

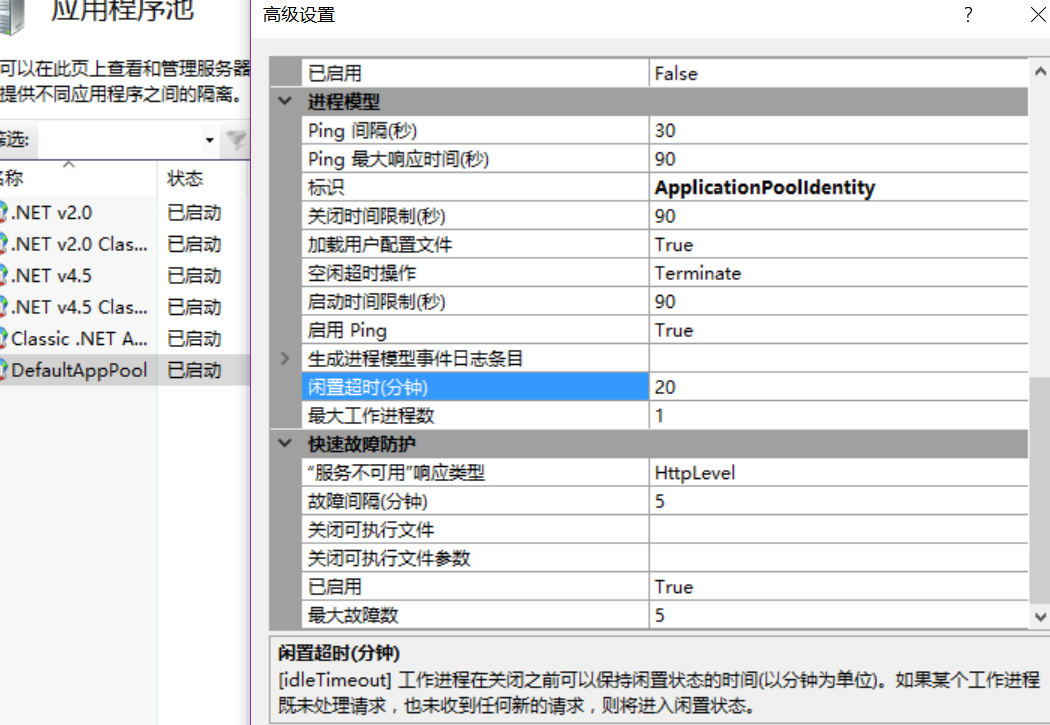
System.Timers.Timer evtTimer = new System.Timers.Timer(2000);

evtTimer.Elapsed += new System.Timers.ElapsedEventHandler(OnEvtTimer);

evtTimer.Enabled = true;

}





**Application.AddMessageFilter 方法 (IMessageFilter)**

Use a message filter to prevent specific events from being raised or to perform special operations for an event before it is passed to an event handler. Message filters are unique to a specific thread.

To prevent a message from being dispatched, the value parameter instance that you pass to this method must override the [PreFilterMessage](https://msdn.microsoft.com/zh-tw/library/system.windows.forms.imessagefilter.prefiltermessage.aspx) method with the code to handle the message. The method must return **false**.

As you will have seen from the .NET Windows namespace, mouse events are provided individually to the form or control that the mouse is over when the event occurs. So how can you detect a mouse event over any control or form regardless of where it occurs? The answer is to filter all messages sent to the application's Message Loop. suffice to say that to use this technique you implement the IMessageFilter interface which then receives notifications of a mouse event over any object.

# Control.Bounds 屬性

Gets or sets the size and location of the control including its nonclient elements, in pixels, relative to the parent control.

通过controls.add添加的控件的location是相对父控件。其他new出来的form相对于屏幕？