

LabView OOP

基于labview 面向对象的一个简单的小例子

面向对象概述

将所有的事物尽可能地向上抽象，向上抽象是多个事物之间有共同规律，但也有不同属性。

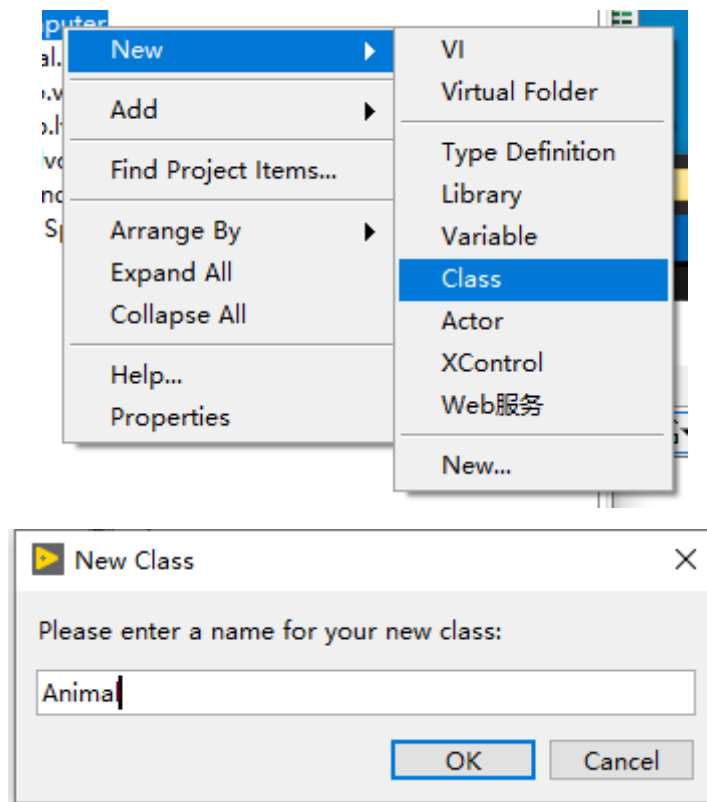
将共有属性向上提取到父类，特有属性分配给子类

这里拿动物进食来举例

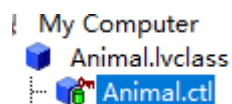
1.创建新项目

2.创建Animal父类

右键项目->New->Class

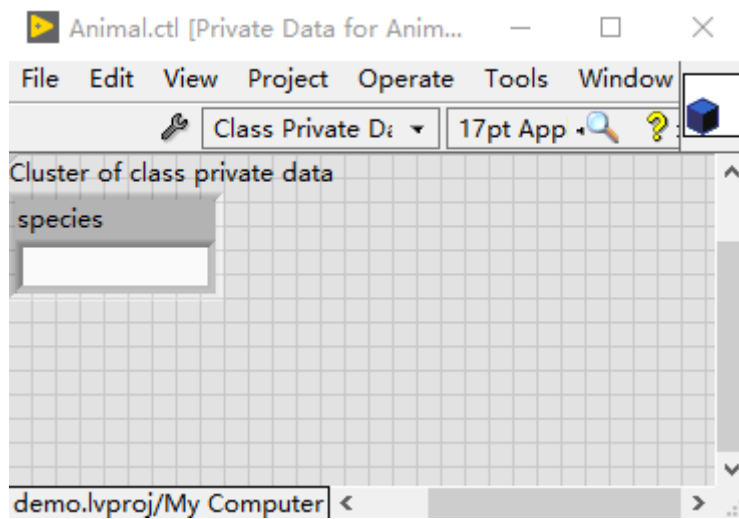


创建完成后



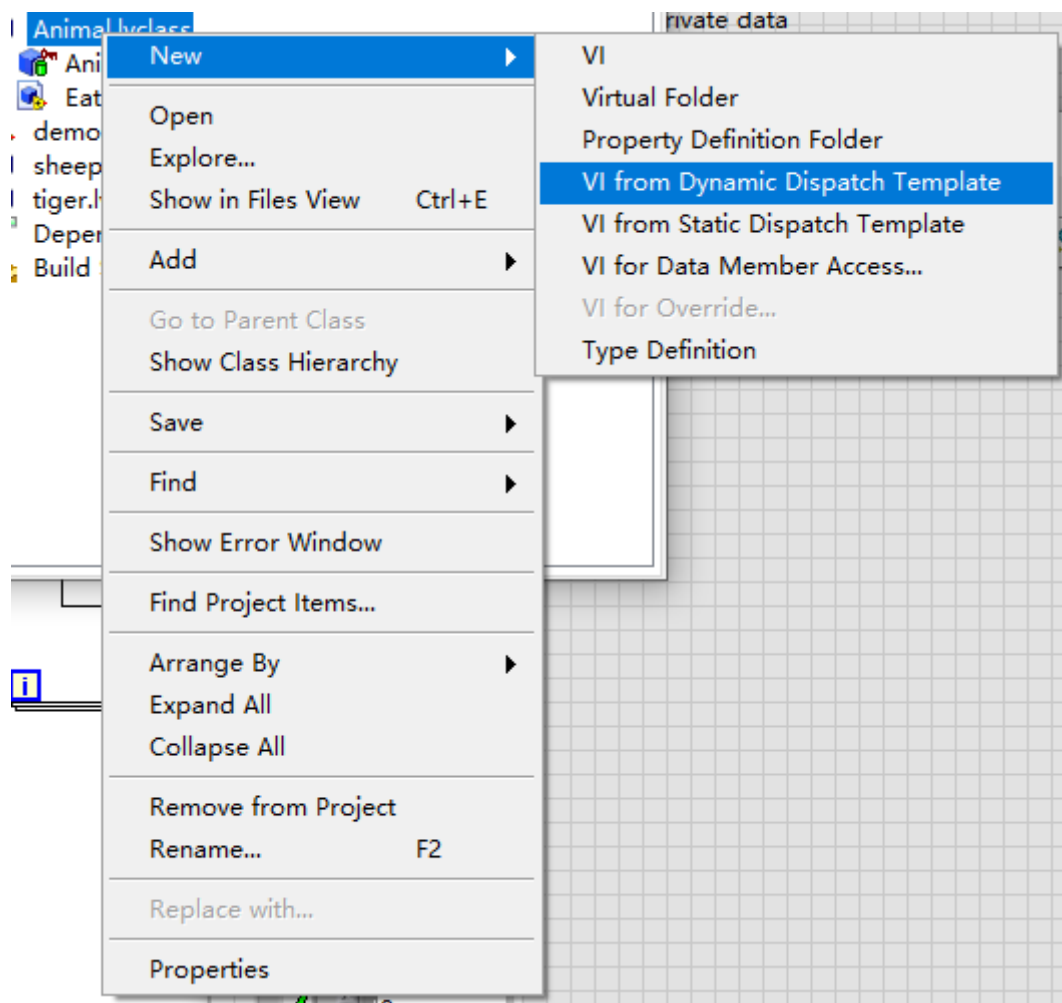
加入私有属性

双击Animal.cti，可在簇中加入该类的私有属性。注：根据概述所讲，该类的子类会继承类的私有属性，而无法通过该类的父类则无法获取这些属性，之后的方法同理。



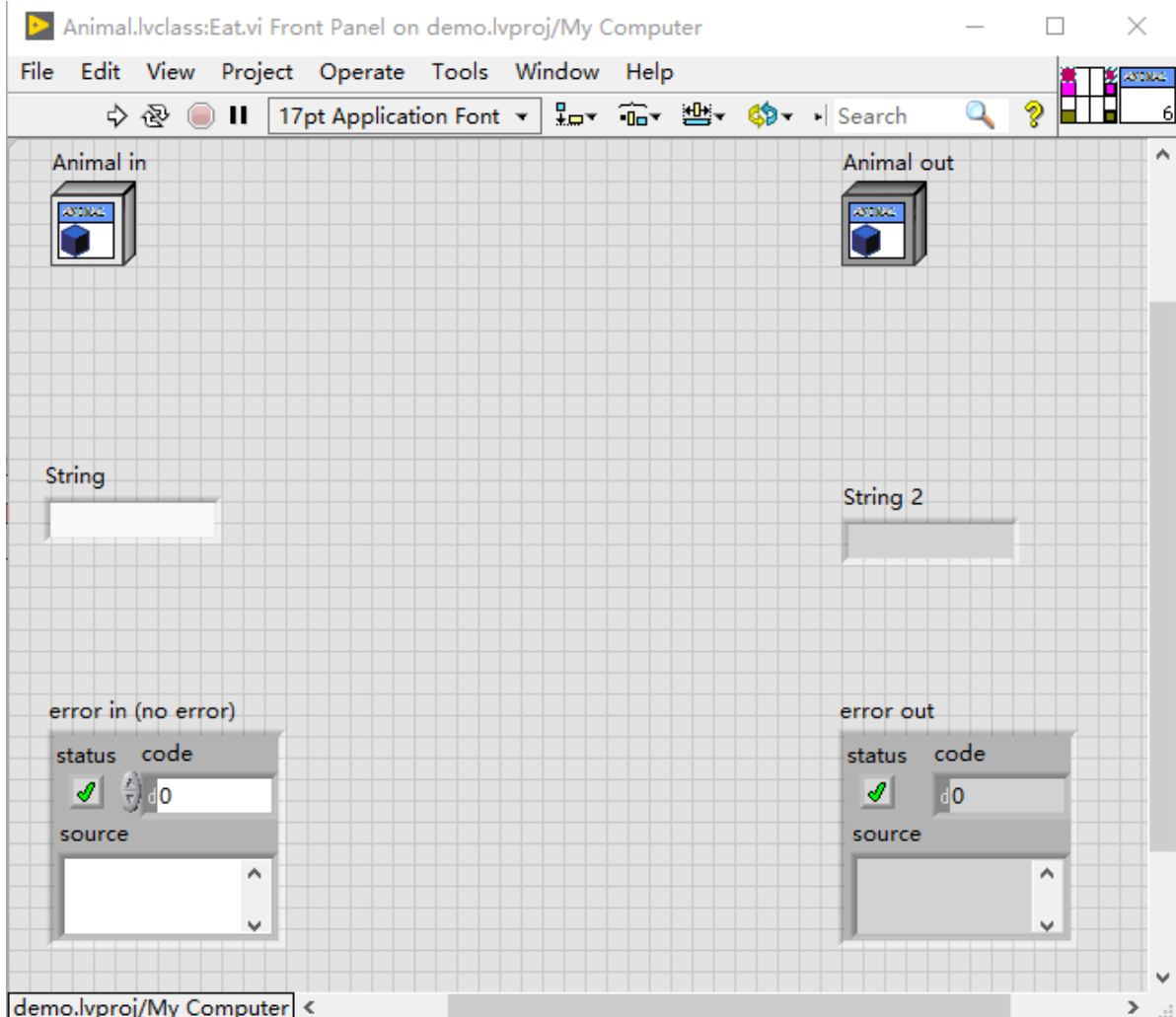
创建接口VI

右击类创建Animal的动态分配模板的VI，命名为Eat.vi，子类可以继承重写该VI

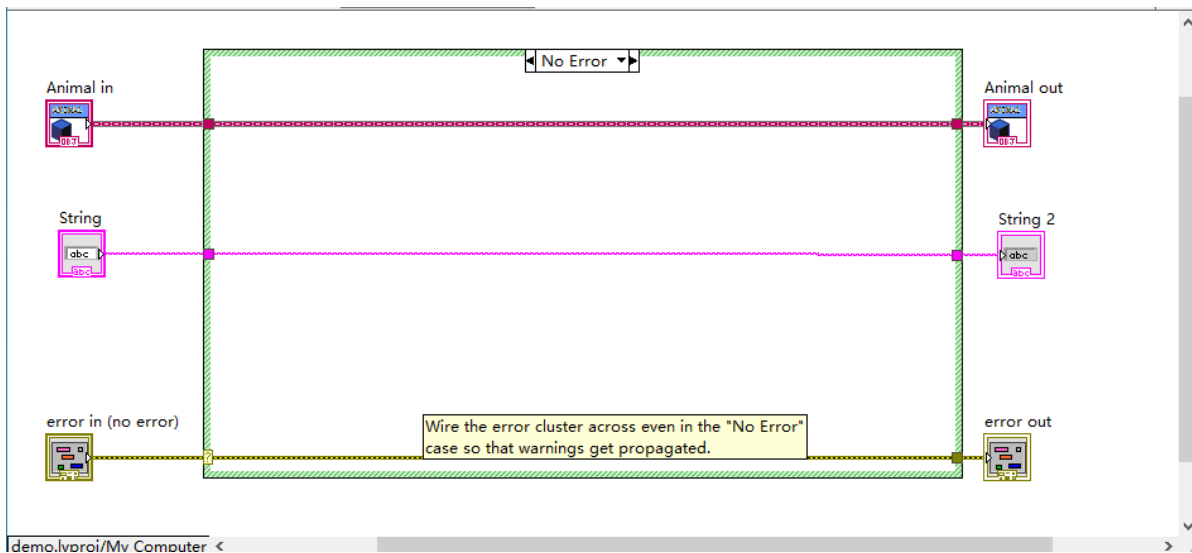


定义接口参数

创建一个字符串输入和字符串输出。



在程序面板中将两者连起来，两个字符串控件仅作为子类实现接口方法的参数定义



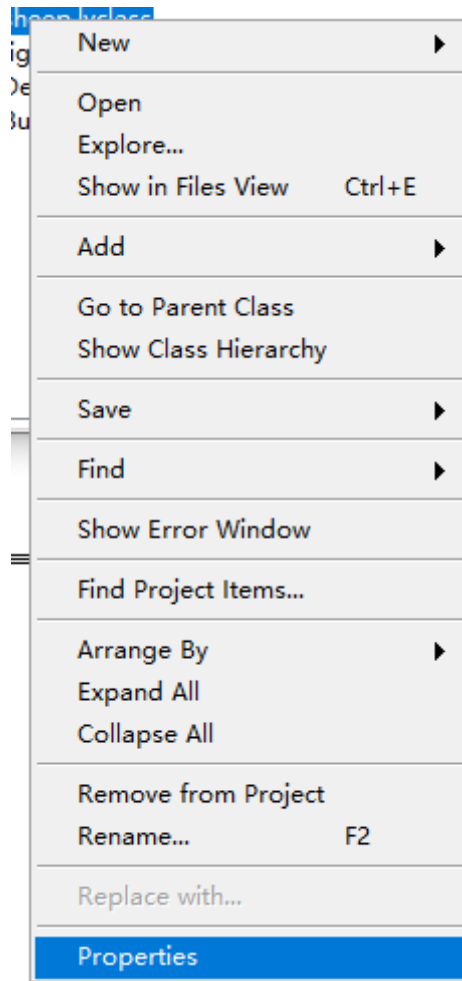
到此，Animal类已完成

2.创建sheep子类

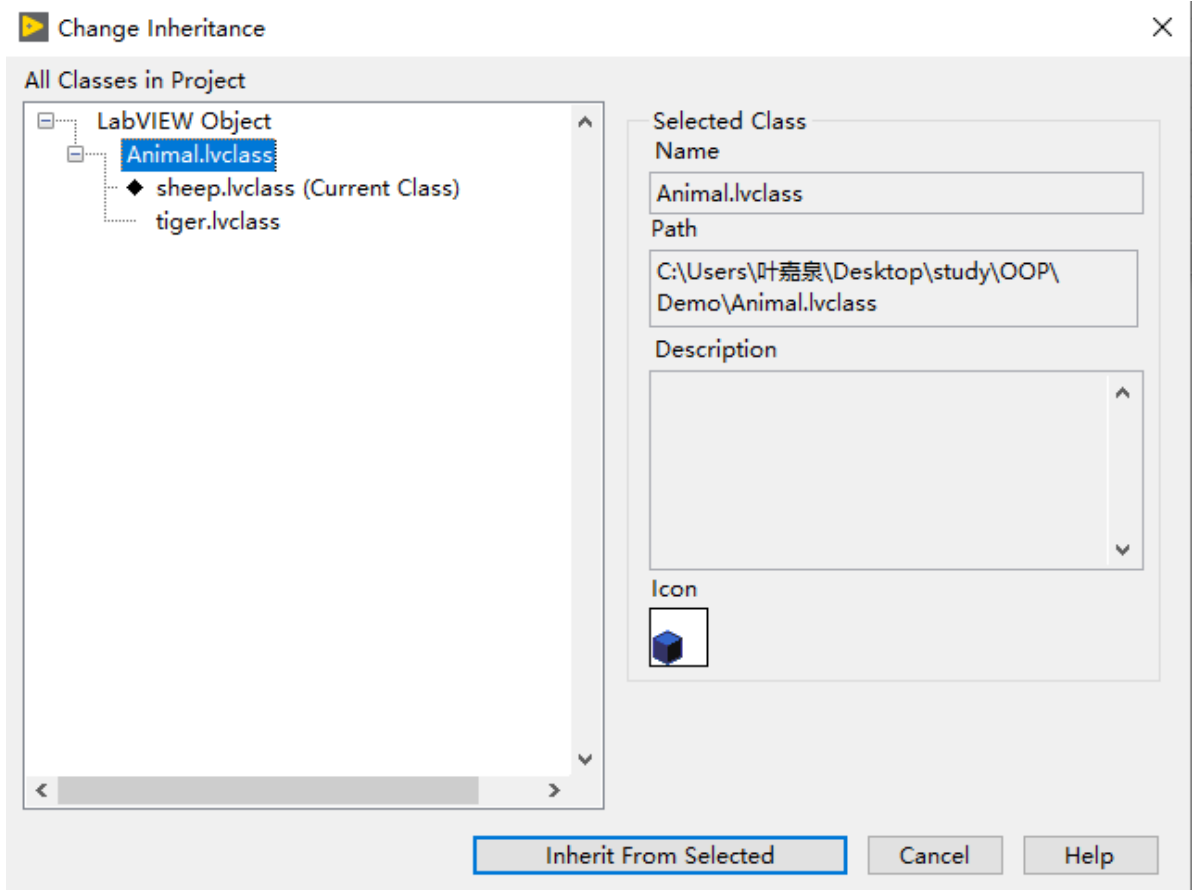
同理，右键项目->New->Class，新建一个Name为sheep的class

继承Animal类

右击sheep class, 打开属性



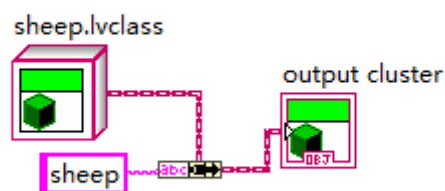
选中Inheritance面板, 点击Change Inheritance, 选中Animal.lvclass, 点击Inherit From Selected即可



创建Create VI

create VI的作用是生成一个新的实例，create中的操作一般为初始化类

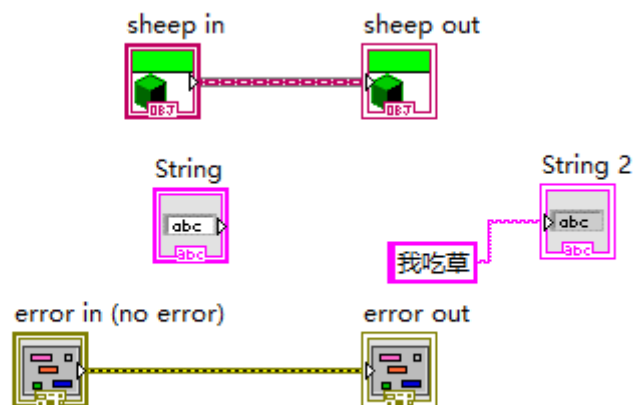
右键类创建新VI，打开程序面板，将sheep类拖入，可以进行一些对私有属性的赋值及其他操作，然后返回



创建override VI

右键类-> New -> VI for Override，会自动成接口VI的继承

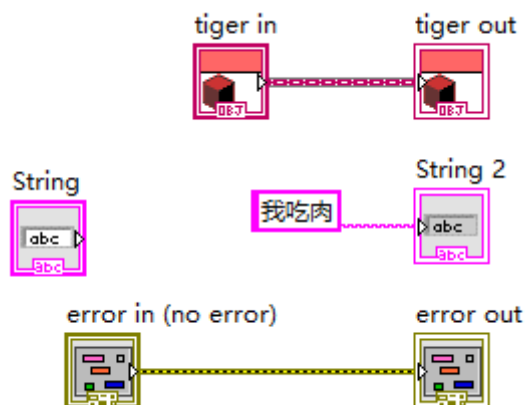
打开sheep类下的Eat.vi程序面板，这里给定将“我吃草”作为sheep类Eat接口的输出。注：这里体现的是类的不同实现方法



OK, sheep类已经完成

3.创建tiger子类

与sheep类相同的操作，需要改变的是tiger类的Eat.vi实现，这里将“我吃肉”作为输出



4.测试

新建demo.vi

将sheep类的Create Sheep.vi、tiger类的Create Tiger.vi和Animal类的Eat.vi拖入

两个类通过Animal.Eat()分别输出“我吃草”和“我吃肉”

