**实验二 面向对象程序设计**

**一、实验目的**

1. 理解类的定义、继承等面向对象的的基本概念；
2. 掌握 C#语言定义类及其各种成员（字段，属性，方法）的方法；
3. 掌握方法覆盖的应用；
4. 掌握接口的定义和实现方法。

**二、实验要求**

根据要求，编写 C#程序，并将程序代码和运行结果写入实验报告。

**三、实验内容**

1. 设计编写一个控制台应用程序，练习类的继承。

1. 编写一个抽象类 People，具有”姓名”，”年龄”字段，”姓名”属性，Work 方法。
2. 由抽象类 People 派生出学生类 Student 和职工类 Employer，继承 People 类，并覆盖 Work 方法。
3. 派生类 Student 增加”学校”字段，派生类 Employer 增加”工作单位”字段。
4. 在 Student 和 Employer 实例中输出各自不同的信息。

**程序代码：**

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**namespace** 实训2**.**\_1

**{**

class Program

**{**

static void Main**(**string**[]** args**)**

**{**

Student stu **=** **new** Student**();**

Employer emp **=** **new** Employer**();**

stu**.**Name **=** "张三"**;**

stu**.**age **=** 20**;**

Console**.**Write**(**stu**.**Name **+** "，今年" **+** stu**.**age **+** "岁，"**);**

stu**.**work**();**

emp**.**Name **=** "李四"**;**

emp**.**age **=** 40**;**

Console**.**Write**(**emp**.**Name **+** "，今年" **+** emp**.**age **+** "岁，"**);**

emp**.**work**();**

**}**

**}**

//抽象类people

**abstract** class People

**{**

**public** string name**;**

**public** string Name

**{**

**get** **{** **return** name**;** **}**

**set** **{** name **=** **value;** **}**

**}**

**public** int age**;**

**public** **abstract** void work**();**

**}**

//派生出学生类

class Student **:** People

**{**

string school **=** "石家庄铁道大学"**;**

**public** **override** void work**()**

**{**

Console**.**WriteLine**(**"在" **+** school **+** "学习。"**);**

**}**

**}**

//派生出职工类

class Employer **:** People

**{**

string address **=** "石家庄铁道大学"**;**

**public** **override** void work**()**

**{**

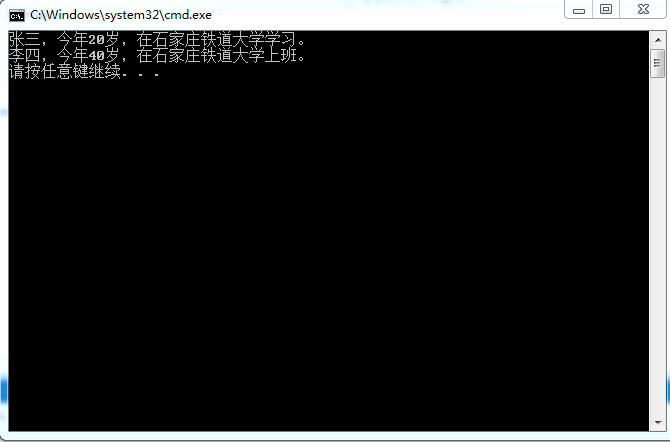
Console**.**WriteLine**(**"在" **+** address **+** "上班。"**);**

**}**

**}**

**}**

**运行结果截图：**



2. 编写一个控制台应用程序，输入正方形边长或者半径，计算其周长和面积并输出。

(1) 编写两个接口，接口 IShape 包含三个方法：initialize, getPerimeter, getArea。分

别进行初始化、获取边长和面积，其返回值均为 decimal。接口 IDisplayresult 显示计算结

果。

(2) 编写两个类，Square（正方形）和 Circle（圆形），实现 IShape 和 IDisplayresult

接口。

**程序源代码：**

**using** System**;**

**using** System**.**Collections**.**Generic**;**

**using** System**.**Linq**;**

**using** System**.**Text**;**

**using** System**.**Threading**.**Tasks**;**

**namespace** 实训2**.**\_2

**{**

class Program

**{**

static void Main**(**string**[]** args**)**

**{**

//正方形

Square s **=** **new** Square**();**

s**.**initialize**();**

s**.**getPerimeter**();**

s**.**getArea**();**

s**.**result**();**

//圆形

Circle c **=** **new** Circle**();**

c**.**initialize**();**

c**.**getPerimeter**();**

c**.**getArea**();**

c**.**result**();**

**}**

**}**

**interface** IShape

**{**

decimal initialize**();**

decimal getPerimeter**();**

decimal getArea**();**

**}**

**public** **interface** IDisplayResult

**{**

void result**();**

**}**

**public** class Square **:** IShape**,** IDisplayResult

**{**

**public** decimal a**;** //边长

**public** decimal zhouchang**;**

**public** decimal area**;**

**public** decimal initialize**()**

**{**

a **=** 0**;**

**return** a**;**

**}**

**public** decimal getPerimeter**()**

**{**

Console**.**WriteLine**(**"请输入正方形边长："**);**

a **=** decimal**.**Parse**(**Console**.**ReadLine**());**

**return** a**;**

**}**

**public** decimal getArea**()**

**{**

zhouchang **=** 4 **\*** a**;**

area **=** a **\*** a**;**

**return** area**;**

**}**

**public** void result**()**

**{**

Console**.**WriteLine**(**"周长为：" **+** zhouchang**);**

Console**.**WriteLine**(**"面积为：" **+** area**);**

**}**

**}**

**public** class Circle **:** IShape**,** IDisplayResult

**{**

**public** decimal r**;** //半径

**public** decimal zhouchang**;**

**public** decimal area**;**

**public** decimal initialize**()**

**{**

r **=** 0**;**

**return** r**;**

**}**

**public** decimal getPerimeter**()**

**{**

Console**.**WriteLine**();**

Console**.**WriteLine**(**"请输入圆形半径："**);**

r **=** decimal**.**Parse**(**Console**.**ReadLine**());**

**return** r**;**

**}**

**public** decimal getArea**()**

**{**

zhouchang **=** 2 **\*** 3 **\*** r**;**

area **=** 3 **\*** r **\*** r**;**

**return** area**;**

**}**

**public** void result**()**

**{**

Console**.**WriteLine**(**"周长为：" **+** zhouchang**);**

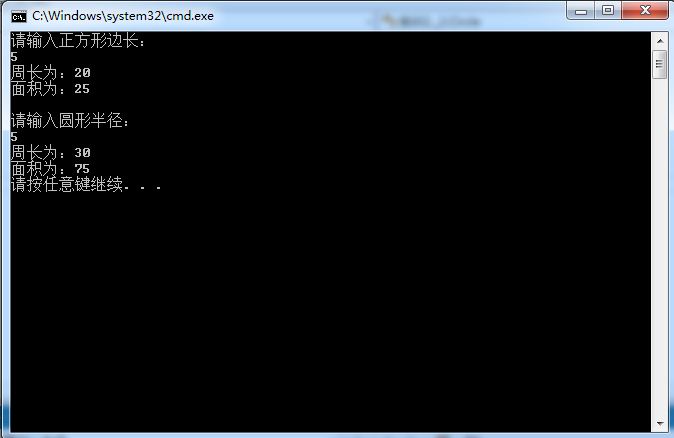
Console**.**WriteLine**(**"面积为：" **+** area**);**

**}**

**}**

**}**

结果截图：



**四、实验总结**

通过这次的实验，学会了怎样运用抽象类和接口，让我进一步的了解和掌握c#。抽象类和接口都是可以被继承，但是抽象类只能实现单继承，而接口能够可以实现多继承。