**计算机与信息学院实验报告**

**实验七 面向对象程序设计**

1. 实验目的和要求
2. 掌握面向对象编程方法。
3. 掌握类属性、对象属性的不同用法。
4. 掌握类的创建和使用方法、构造方法的使用方法
5. 掌握类的运算符重载
6. 实验内容及原理
7. **类和对象的属性（Pyprog01.py，本题20分）**

**题目描述：**编写一个学生类Student，定义3个属性name、age和id，分别表示学生的姓名、年龄和学号。第一个学生的学号为1，以后每生成一个学生对象，学号增加1。初始化学生对象时，需要提供姓名和年龄。每个学生对象生成以后需要调用自定义的info()方法输出姓名、年龄和学号。

**输入格式：**第一行输入第一个学生的姓名，第2行输入第一个学生的年龄;按顺序输入学生的信息，直到输入q或Q结束输入。

**输出格式：**第一个学生输入后打印第一个学生的姓名、年龄和学号。按顺序输出学生的信息。

**输入输出样例1:**

|  |  |
| --- | --- |
| 输入 | 输出 |
| zhangsan  8  lisi  9  q | My name is zhangsan, age is 8, id is 1  My name is lisi, age is 9, id is 2 |

**输入输出样例2:**

|  |  |
| --- | --- |
| 输入 | 输出 |
| stu1  10  stu2  10  stu3  11  q | My name is stu1, age is 10, id is 1  My name is stu2, age is 10, id is 2  My name is stu3, age is 11, id is 3 |

1. **求三角形的面积和周长（Pyprog02.py，本题20分）**

**题目描述：**定义一个三角形类Triangle，属性a、b、c分别表示3条边，方法get\_area()和get\_cal()分别用来求面积和周长。  
**输入格式：**3行输入3条边长

**输出格式：**若3条边能构成三角形，输出“The perimeter is \*. The area is ?.”\*和？分别是三角形的周长和面积，保留2位数。若不能构成三角形，则输出“不能构成三角形！”

**输入输出样例1:**

|  |  |
| --- | --- |
| 输入 | 输出 |
| 3  4  5 | The perimeter is 24.00. The area is 2.45. |

**输入输出样例2:**

|  |  |
| --- | --- |
| 输入 | 输出 |
| 3  3  6 | 不能构成三角形！ |

1. **继承（Pyprog03.py，本题20分）**

#### 题目描述：设计一个基类Shape，包括：名为sName的属性（图形名称）；构造函数应对sName属性进行初始化。设计Shape的子类Rectangle, 包括：长，宽两个属性；构造函数调用Shape的构造函数，并初始化长，宽两个属性；getArea()方法计算并返回矩形面积。设计Shape的子类Circle,包括：半径属性；构造函数调用Shape的构造函数，并初始化半径属性；getArea()方法计算并返回圆形面积。打开Pyprog03.py补充完整代码。

**输入格式：**第一行输入1个字符串；第二行输入一个数字或是两个数字,数字用空格分隔。

**输出格式：**若输入一个数字，则打印圆面积；若是输入两个数字，则打印长方形面积。

**输入输出样例1:**

|  |  |
| --- | --- |
| 输入 | 输出 |
| ff  3 | The area of Circle ff is 28.27 |

**输入输出样例2:**

|  |  |
| --- | --- |
| 输入 | 输出 |
| aa  3.0 4 | The area of Recangle aa is 12.00 |

1. **运算符重载（Pyprog04.py，本题20分）**

**题目描述**：请修改代码Pyprog04.py的Point类，通过运算符重载使其支持如下运算符操作，其中p1和p2为Point类，n为数值。

1. pl+p2：用两个点的x、y坐标分别相加，返回一个新的Point对象；
2. pl-p2：用两个点的x、y坐标分别相减，返回一个新的Point对象；
3. p\*n：用点的x、y坐标分别乘数值n，返回一个新的Point对象；
4. p/n：用点的x、y坐标分别除以数值n，返回一个新的Point对象。
5. print(p1)：如p1的x，y坐标为2,3,则打印内容为“Point(2,3)”

**输入格式：第一行输入两个数字，空格分隔；同样第二行输入2个数字**

**输出格式：**输出两个Point类的加、减，Point类和数字n的乘、除。如Point(3,6)+Point(4.5,1)输出位Point(7.5,7);

**输入输出样例1:**

|  |  |
| --- | --- |
| 输入 | 输出 |
| 3 6  4.5 1  5 | Point(7.5,7)  Point(-1.5,5)  Point(15,30)  Point(0.6,1.2) |

**输入输出样例2:**

|  |  |
| --- | --- |
| 输入 | 输出 |
| 5.5 3  5 7  2 | Point(10.5,10)  Point(0.5,-4)  Point(11.0,6)  Point(2.75,1.5) |

1. 实验设备与环境

⑴硬件：微型计算机、局域网；⑵软件：Python

1. 方法与实验步骤

（1）题目1：

### 请在此两行之间###

### 你的代码写在这里

class Student:

def \_\_init\_\_(self,name,age):

self.name=name

self.age=age

def info(self):

print(f"My name is {self.name}, age is {self.age}, id is {ID}")

ID=0

###################

name=input()

stu=[]

while name != 'q' and name != 'Q' :

ID+=1

age=eval(input())

a=Student (name,age)

a.info()

stu.append(a)

name=input().strip()

1. 题目2：

import math

### 请在此两行之间###

### 你的代码写在这里

class Triangle:

def \_\_init\_\_(self,a,b,c):

self.a=a

self.b=b

self.c=c

def get\_area(self):

p=(self.a+self.b+self.c)/2

s=math.sqrt(p\*(p-self.a)\*(p-self.b)\*(p-self.c))

return s

def get\_cal(self):

C=self.a+self.b+self.c

return C

###################

try:

a = float ( input ())

b = float ( input ())

c = float ( input ())

if a >0 and b >0 and c >0 and a+b > c and a + c > b and b + c > a :

triangle = Triangle ( a , b ,c )

print ("The perimeter is {:.2f}. The area is {:.2f}.". format ( triangle . get\_cal(), triangle . get\_area ()))

else :

print ("不能构成三角形！")

except:

print("输入有误！")

1. 题目3：

import math

### 请在此两行之间###

### 你的代码写在这里

class Shape:

def \_\_init\_\_(self,sName):

self.sName=sName

class Rectangle(Shape):

def \_\_init\_\_(self,sName,width,length):

super().\_\_init\_\_(sName)

self.length=length

self.width=width

def getArea(self):

return self.length\*self.width

class Circle(Shape):

def \_\_init\_\_(self,sName,radius):

super().\_\_init\_\_(sName)

self.radius=radius

def getArea(self):

return math.pi\*(self.radius\*\*2)

###################

try:

name=input()

s = input() .split()

if len(s)==2:

w,h =map(eval,s)

r1 = Rectangle(name,w,h)

print("The area of Recangle {} is {:.2f}".format(r1.sName,r1.getArea()))

elif len(s)==1:

r = eval(s[0])

c1 = Circle(name,r)

print("The area of Circle {} is {:.2f}" .format (c1.sName,c1.getArea()))

else:

print("输入边个数有误！")

except:

print("程序出错")

（4）题目4：

### 请在此两行之间###

### 你的代码写在这里

class Point:

def \_\_init\_\_(self,x,y):

self.x=x

self.y=y

def \_\_str\_\_(self):

return f"Point({self.x},{self.y})"

def \_\_add\_\_(self,other):

x=self.x+other.x

y=self.y+other.y

return Point(x,y)

def \_\_sub\_\_(self,other):

x=self.x-other.x

y=self.y-other.y

return Point(x,y)

def \_\_mul\_\_(self,n):

x=self.x\*n

y=self.y\*n

return Point(x,y)

def \_\_truediv\_\_(self,n):

x=self.x/n

y=self.y/n

return Point(x,y)

###################

try:

x1,y1=map(eval,input().split())

x2,y2=map(eval,input().split())

n=eval(input())

a =Point (x1,y1)

b =Point (x2,y2)

#输出结果都为Point对象

print (a+ b)

print (a - b)

print (a \*n)

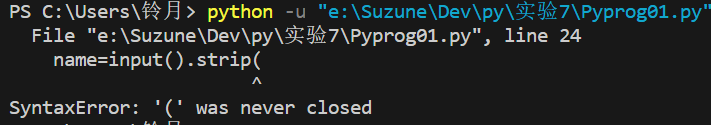
print (a /n)

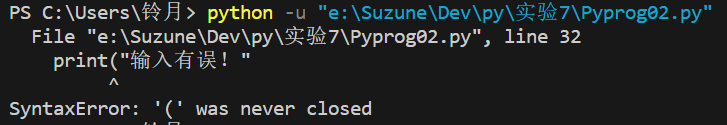
except:

print("你的输入有误")

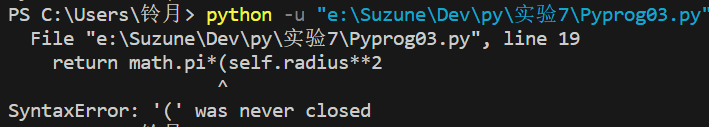
1. 实验调试过程

（1）题目1调试过程

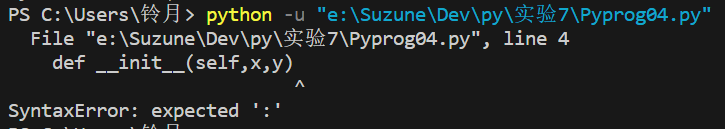
（2）题目2调试过程



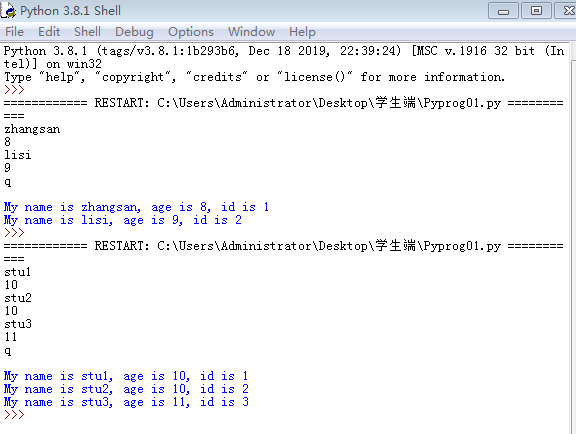
（3）题目3调试过程



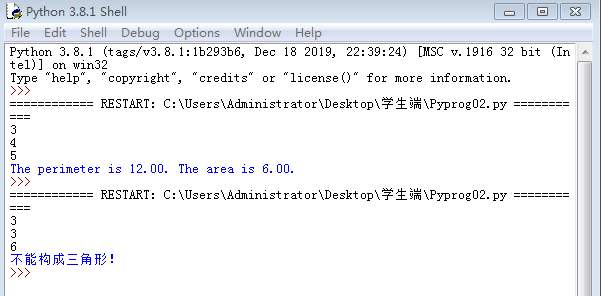
1. 题目4调试过程



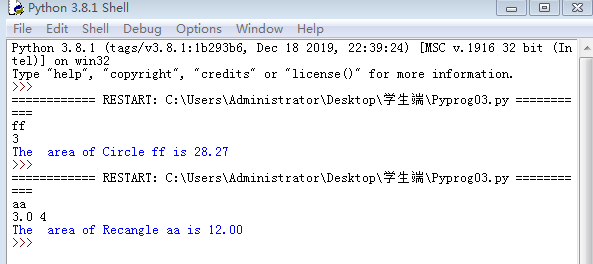
1. 实验结果分析
2. 题目1实验结果



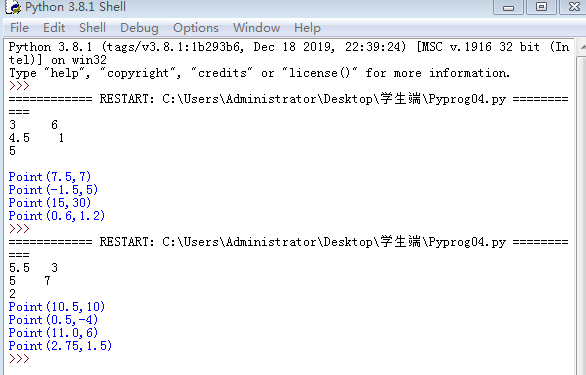
1. 题目2实验结果



1. 题目3实验结果



1. 题目4实验结果



1. 建议、质疑和问题

无