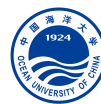


海纳百川
取则行远



中国海洋大学
OCEAN UNIVERSITY OF CHINA

Week3

左昊天

2024-09-06

Github 仓库地址

1 Python 基础

1.1 输出语句

1.1.1 单行输出

```
1 print("这是输出的语句")
```

print 中可以用单引号或双引号，但若想输出的内容中存在同样的引号，会和输出语句的引号产生歧义。

解决方法 1: print 语句和内容用不同的引号

解决方法 2: 在内容的引号前加转义字符\

```
1 print("I'm fine.")
2 print('Let\'s go!')
```

1.1.2 多行输出

使用三引号进行多行输出

```
1 print('''我相信能再次看到蓝天，
2 鲜花挂满枝头''')
```

```
这是输出的语句
I'm fine.
Let's go!
我相信能再次看到蓝天，
鲜花挂满枝头。
```

1.2 解一元二次方程

乘方为 ** 根号可以用 $\sqrt{\quad}$ 或使用 math 库

```
1 import math
2 a=int(input("请输入a的值"))
3 b=int(input("请输入b的值"))
4 c=int(input("请输入c的值"))
5 print((-b+math.sqrt(b**2-4*a*c))/(2*a))
6 print((-b-math.sqrt(b**2-4*a*c))/(2*a))
```

```
请输入a的值1
请输入b的值-3
请输入c的值2
2.0
1.0
```

1.3 for 循环与字典

1.3.1 普通循环

```
1 total=0
2 for i in range(1,11):
3     total=total+i
4 print(total)
```

注：1、python 的缩进非常严格，通过缩进判断循环是否结束。

2、range 的范围左闭右开

```
\系统开发工具基础\1\tool_class_2024_sum\week3\python\for1.py'
55
```

1.3.2 对字典的循环

```
1 a = {"小王":"123456789","小李":"15462847","小张":
      "4535415"}
2 for name,phone in a.items():
3     print(name+phone)
```

```
小王123456789
小李15462847
小张4535415
```

关于字典：

1、字典是键值对

2、a.keys() 返回所有键，a.values() 返回所有值，a.items() 返回所有键值对

1.4 while 循环

```
1 i=0
2 total=0
3 while i<11:
4     total=total+i
```

```
5     i=i+1
6     print(total)
```

1.5 格式化字符串

方法一：

```
1 name="小王"
2 print(f"你好{name}")
```

方法二：

```
1 name="小王"
2 print("你好{0}".format(name))
```

```
你好小王
你好小王
```

1.6 函数

```
1 def calculate(a):
2     s=a*a
3     return s
4 a=float(input("请输入正方形的边长"))
5 s=calculate(a)
6 print(f"边长为{a}的正方形的面积为: {s}")
```

```
请输入正方形的边长3
边长为3.0的正方形的面积为: 9.0
```

1.7 类

1.7.1 创建类以及类的实例化

```
1 class Student:
2     def __init__(self,name,number,grade): #构造函数
3         self.name = name
4         self.number = number
5         self.grade = grade
6     def showStudent(self):
```

```
7         print("姓名: "+self.name+" 学号: "+self.number+"
              年级: "+self.grade)
8
9 wang = Student("小王","1","9") #对象的初始化
10 wang.showStudent()
```

注: 构造函数的下划线是两个下划线并非一个!

姓名: 小王 学号: 1 年级: 9

1.7.2 类的继承

```
1 class People:
2     def __init__(self,name,age): #构造函数
3         self.name = name
4         self.age = age
5
6 class Student(People):
7     def __init__(self,name,age,number,grade): #构造函数
8         super().__init__(name,age)
9         self.number = number
10        self.grade = grade
11    def showStudent(self):
12        print("姓名: "+self.name+" 年龄: "+self.age+" 学
              号: "+self.number+" 年级: "+self.grade)
13
14 wang = Student("小王","14","1","9") #对象的初始化
15 wang.showStudent()
```

解释:

1、`super()`会返回当前类的父类,可以用此函数调用父类的构造函数。

2、`class Student(People)`中括号的内容表示继承的父类。

姓名: 小王 年龄: 14 学号: 1 年级: 9

1.8 文件的读写

打开文件的代码:

```
1 open("./test.txt","r",encoding="utf-8")
```

- 1、其中第一个引号中表示文件的地址
- 2、第二个引号中表示要对文件进行什么操作
- 3、encoding 表示文件的编码方式

第二个引号中的内容	可对文件进行的操作
r	只读
w	只写（覆盖之前的内容）
a	只写（在已有内容后面添加）
r+	读和写（覆盖之前的内容）
a+	读和写（追加写）

1.8.1 文件的读

.read() 返回文件全部内容
.readline() 返回文件的一行内容
.readlines() 返回文件所有内容组成的列表

1.8.2 实例

```
1 f=open("C:\\Users\\19355\\Desktop\\大二暑假\\系统开发工  
   具基础\\1\\tool_class_2024_sum\\week3\\python\\test.  
   txt","a",encoding="utf-8")  
2 f.write("—— 流浪地球")  
3 f.close()  
4  
5 f=open("C:\\Users\\19355\\Desktop\\大二暑假\\系统开发工  
   具基础\\1\\tool_class_2024_sum\\week3\\python\\test.  
   txt","r",encoding="utf-8")  
6 print(f.read())  
7 f.close()
```

我相信能再次看到蓝天，
鲜花挂满枝头。—— 流浪地球

2 Python 视觉应用