

Python学习内容(八上)

第一单元Python程序设计基础

一、初识Python

1. Python语言的简介与特点

它是一种跨平台、开源、解释型的高级编程语言，它的创始人是计算机程序员吉多·范罗苏姆（Guido van Rossum, 荷兰人），于1991年推出。

它的特点为语言简洁、黏性扩展、类库丰富

2. 开发环境为Python自带的IDLE，首先出现的交互式环境，只要在三个大于号（命令提示符）后即可输入命令，回车就出现命令运行结果。

3. 掌握两个函数，输出函数print()和输入函数input()

例如：>>>print("Hello world!")

注意：引号与括号必须为英文字符 ‘#’起注释的作用

4. 创建程序文件及运行，学会用F5键

```
name=input("请输入你的姓名：")
age=input("请输入你的年龄：")
print("大家好！我叫",name, "，今年",age,"岁。")
```

二、常用数据类型

1. 常用数据类型

数据类型名称	数据表示形式
整型(int)	数学中的整数，如0，1，-2等
浮点型(float)	数学中的小数，如2.0，3.14，
字符串型(str)	用引号作业定界符，如"ABC"，'快乐'等
列表(list)	用方括号作业定界符，如[1,2,3]，[1,'a']
布尔型(bool)	只有两种值，True和False（ 注意首字母大写 ）

2. 列表的操作

(1) 列表创建，用[]

(2) 列表访问：正向索引与反向索引 **字符串也适用该方法**

(3) 列表切片访问，格式为：[start:end:step]

3. 变量与赋值语句，格式：变量名=表达式，其中“=”为赋值号

变量名的命名规则：（1）只能包含字母、数字和下划线。注意字母区分大小写

(2) 不能以数字开头，不能与关键字同名

关键字是预先保留的标识符，因此又称为保留字，在Python中共有35个。详见P11页

三、算术运算符和表达式

1. 算术运算符

掌握各运算符的意义，以及优先级

试算：5+6%4**2//4/(6-4)

习题

```
x=int(input("输入3位数的自然数："))
a=x//100          # 百位上的数字
_____         # 十位上的数字
c=x%10           # 个位上的数字
print(a,b,c,sep=',')
```

第二单元 Python程序基本结构

Python程序的基本结构只有三种，即顺序结构、分支结构、循环结构。

四、顺序结构程序设计

顺序结构是程序设计中最简单、最基本的结构方式。程序只需要根据问题解决的步骤依次编写程序代码即可。

例如，编写一个“用海伦公式来解三角形面积”。基本过程如下

(一) 问题分析

用海伦公式求解三角形面积，需要先知道三条边的长度，然后根据公式计算机面积。

假设三角形的三条边长分别为a, b, c，则三角形的面积s可以用如下公式

$$s = \sqrt{p(p-a)(p-b)(p-c)} \quad \text{其中 } p = \frac{a+b+c}{2}$$

(二) 设计算法

(三) 编写代码

```
# 输入三角形的三边长
a=float(input('输入三角形第一边长：'))
b=float(input('输入三角形第二边长：'))
c=float(input('输入三角形第三边长：'))
p=(a+b+c)/2    # 计算三角形的半周长
```

```
s=(p*(p-a)*(p-b)*(p-c))**0.5 # 计算三角形的面积
print('三角形面积为: ',s)
```

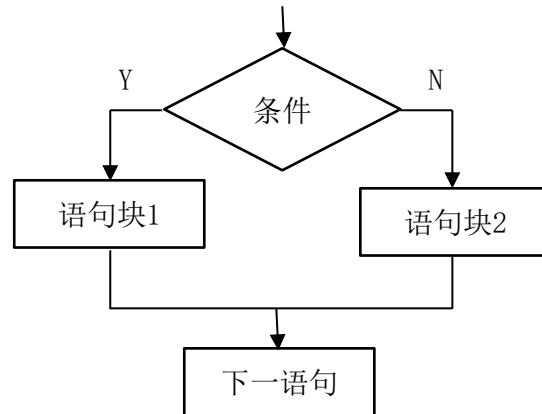
(四) 调试运行

参考书本P19-P20

五、分支结构程序设计

1. if条件语句

```
if <条件>:
    <语句块1>
[else:
    <语句块2>]
```



示例:

```
if a>=b:
    print('两个数中较大为: ',a)
else:
    print('两个数中较大为: ',b)
```

特别注意:

(1) <>与[]的区别:

if条件语句格式中,尖括号“<>”表示里面的内容是必须的,方括号“[]”表示里面的内容是可选的。在程序设计中,不要将尖括号和方括号输入到语句中。

(2) 在Python中,代码的缩进非常重要,缩进是体现代码逻辑关系的重要方式。同一层次的代码块必须保证相同的缩进量,一般以4个空格作为1个缩进单位,可以按4个空格键或按1次Tab键。

2. 关系运算符与逻辑运算符 (P24)

3. 思考: 输入三条边的长度后先判断它是否能组成三角形,若能组成三角形则输出面积,否则输出不能构成三角形

```
# 输入三角形的三边长
a=float(input('输入三角形第一边长: '))
b=float(input('输入三角形第二边长: '))
c=float(input('输入三角形第三边长: '))
if a+b>c and a+c>b and b+c>a:
```

```
p=(a+b+c)/2    # 计算三角形的半周长
s=(p*(p-a)*(p-b)*(p-c))**0.5 # 计算三角形的面积
print('三角形面积为: ',s)
else:
    print('输入的三条边, 不能构成三角形!')
```

六、多分支结构程序设计

1. if-elif条件语句

基本格式:

```
if <条件>:
    <语句块1>
elif <条件>:
    <语句块2>
.....
elif <条件n>:
    <语句块n>
[else:
    <语句块n+1>]
```

范例：班级通讯录

```
# 班级通讯录信息
xh=input('请输入想要查看的同学两位数学号:')
if xh=='01': # 通过input()输入的信息类型是字符串型, 因此01要加引号
    print('大家好, 我是1号同学, 很高兴与大家成为同学。')
elif xh=='02':
    print('大家好, 我是2号同学, 转眼又快同学两年了。')
else:
    print('输入的学号查找不到, 请输入正确的学号!')
```

具体应用可以在：成绩等级、运动会比赛得分、浙江省汽车牌号所在地查询

七、循环结构——for循环

循环结构是一种十分重要的程序控制结构，其特点是，在给定条件成立时，反复执行某程序段，直到条件不成立为止。给定的条件称为循环条件，反复执行的程序段称为循环体。

1. for循环语句

```
for <变量> in <序列>:  
    <循环体>  
[else:  
    语句块]
```

<变量>每经过一次循环，就会得到序列中的一个元素，并通过循环体处理它，<序列>通常是字符串、列表、range对象实例等。当序列中的元素全部遍历完成后，程序就会自动退出循环，继续执行else子句的语句块。红色标注的这句话比较难懂，需要后面学了continue、break学习后再进行举例说明。

举例：

(1) 遍历字符串

```
for i in "Python3":  
    print(i)
```

(2) 遍历列表

```
ls=['apple','banana','peach']  
for i in ls:  
    print(i)
```

(3) 遍历range()构建的数字序列

```
for i in range(10):  
    print(i)  
range()
```

2. range()函数

格式：range([start,]stop[, step])

功能：创建一个整数列表

参数：起始值（缺省为0），终值（但不包括），步长（缺省为1）

范例:

```
#for循环
for i in range(5):
    xh=input('请输入想要查看的同学两位数学号:')
    if xh=='01': # 通过input()输入的信息类型是字符串型,因此01要加引号
        print('大家好,我是1号同学,很高兴与大家成为同学。')
    elif xh=='02':
        print('大家好,我是2号同学,转眼又快同学两年了。')
    else:
        print('输入的学号查找不到,请输入正确的学号!')
```

3. len()函数

返回对象(字符、列表等)长度个数

```
#len()函数
lst=['apple','banana','peach']
n=len(lst)
for index in range(n):
    print("当前水果:",lst[index],sep="")
```

4. 寻找水仙花数(P44)

水仙花数是指一个三位数,它每个数位上的数的3次幂之和等于它本身。例如:

153是水仙数,因为 $1^3+5^3+3^3=153$ 。在如下程序的划线处填空。

```
lst=[]
for _____
    a=i%10      # 个位
    b=i//10%10 # 十位
    c=i//100   # 百位
    if (a**3+b**3+c**3)==i:
        ls.append(i)
print(lst)
```

八、循环结构——while循环

for循环语句是通过遍历某一序列对象来构建循环，循环结束的条件是对象遍历完成。但是，很多应用难以确定遍历结构，这就是需要根据条件控制循环。在Python中提供了while语句，实现条件循环。

1. while循环，也称为条件循环，格式如下：

```
while <条件>:  
    <循环体>  
[else:  
    <语句块>]
```

```
xh=input('请输入想要查看的同学两位数学号：')  
while xh!="0":  
    if xh=='01': # 通过input()输入的信息类型是字符串型，因此01要加引号  
        print('大家好，我是1号同学，很高兴与大家成为同学。')  
    elif xh=='02':  
        print('大家好，我是2号同学，转眼又快同学两年了。')  
    elif xh=='03':  
        print('大家好，我是3号同学。')  
    elif xh=='04':  
        print('大家好，我是4号同学。')  
    elif xh=='05':  
        print('大家好，我是5号同学。')  
    elif xh=='06':  
        print('大家好，我是6号同学。')  
    elif xh=='07':  
        print('大家好，我是7号同学。')  
    elif xh=='08':  
        print('大家好，我是8号同学。')  
    else:  
        print('输入的学号查找不到，请输入正确的学号！')  
    xh=input('\n请输入想要查看的同学两位数学号：')
```

2. break语句与continue

```
user=""
pwd=""
while True:
    if user!='admin':
        user=input('请输入账号: ')
        continue
    pwd=input('请输入密码: ')
    if pwd=="Python@16":
        break
print("账号和密码正确, 登录成功!")
```

3. else子句

```
#else子句
for i in range(5):
    a=input("请输入一个整数: ")
    if a=="0":
        break
else:
    print("你已经完成了5个数的输入。")
print("请进入下一环节!")
```


第三单元 算法与程序设计

九、Python的函数

1. 函数的概念

函数是一段具有特定功能、可重复使用的语句块。函数用函数名来表示，并通过函数名进行功能调用。在Python中主要有两类函数：系统内建函数和用户自定义函数。

内建函数又称内置函数，是指Python解释器内置的函数。

Python常用的内置函数

函数	格式和功能描述
input()	从标准输入读取字符串
print()	可以一次输出多个对象
int()	将数字或字符串转换为整型
float()	将整数和字符串转换为浮点型
str()	返回对象的字符串格式
len()	返回对象的字符或项目长度

2. 自定义函数

def <函数名>([参数列表]):

<函数体>

[return<返回值>]

```
def year(year):
    #year=2012
    if year%4==0 and year%100!=0 or year%400==0:
        s=str(year)+"年是闰年!"
    else:
        s=str(year)+"年不是闰年!"
    return s
a=year(2000)
print(a)
print("-----")
for i in range(5):
    print('我认为', year(2000), sep="")
```

十、Python的模块

使用函数可将代码块与主程序分离，使程序更加容易理解。另外，还可以将函数存储在称为模块的独立文件中，既可隐藏程序代码细节，又可在不同程序中重复使用函数。

1. 模块的概念

模块是用来把函数、变量，以及其他东西组织成更大的、更强的程序。在Python提供的内置函数有math、random、time、turtle等

2. 模块的分类

- (1) 系统内置模块
- (2) 自定义模块
- (3) 第三方的开源模块

3. 模块导入

方法一: `import <模块> as [<别名>]`

`<模块>.<函数名>()`

例如: `import turtle`

`turtle.forward(100)`

方法二: `from <模块> import <函数名>`

`<函数名>()`

例如: `from turtle import *`

`forward(100)`

4. 常用内置模块math和random

(1) math模块

名称	数学表示	含义
圆周率pi	π	π 的近似值，15位小数
自然常数e	e	e的近似值，15位小数
<code>pow(x, y)</code>	x^y	计算x的y次幂
<code>sqrt(x)</code>	$\sqrt{\quad}$	计算x的平方根

(2) random模块

名称	含义
<code>seed(x)</code>	给随机数一个种子值，默认随机种子是系统时钟
<code>random()</code>	随机生成一个0至1（不含1）之间的实数
<code>uniform(a, b)</code>	随机生成一个a至b之间的实数
<code>randint(a, b)</code>	随机生成一个a至b之间的整数

十一、枚举算法

枚举算法也称穷举法，基本思想是把问题所有的可能解一一列举出来，并判断每一个列举出的可能解是否为正确的解。枚举算法的步骤：

- (1) 确定枚举对象、范围和判定条件。
- (2) 逐一枚举可能的解，并验证每个解是否为问题的解。

范例：

某地有一车主做了好事就走开了，其车牌号码前两位是26，末位是5，即车牌号码是“浙Z26 * * 5”；并称车号同时35和15的倍数。如何快速确定中间两位数字，你能帮忙吗？

```
#枚举算法，找车牌
a=0
for i in range(26005, 26996):
    if i%35==0 and i%15==0:
        a=a+1
        print("该车的车牌是浙Z", i, sep="")
print("共有", a, "个车牌")
```