复习:

见思维导图

1.Python中的逻辑结构

①顺序执行

②选择执行:

if...elif...else...

没有switch..case..

③循环执行:

for...in...

while...

没有do..while...

注意: if/for/while后面不使用{ }, 而用 : 代替; 下面一句必须有缩进(可以是TAB或若干空格); 再下面一句如果也是执行主体必须有同样的缩进!

循环语句中，可以使用关键字：break(打断循环) 和 continue(继续下一次循环)

|  |  |  |
| --- | --- | --- |
| if( ):  pass | if( ):  pass  else:  pass | if( ):  pass  elif( ):  pass  else:  pass |
| while( ):  pass | | |
| for 临时变量 in list/tuple/range/set/dict :  pass | | |

练习:创建变量表示用户的积分,如果达到了10000分,输出"黑金用户"; 否则如果达到了5000分,输出"黄金用户"; 否则如果达到1000分,输出"白银用户"; 否则输出"普通用户"

练习：创建一个空数组，用于保存员工姓名；提示“请输入下一位员工姓名(输入end以结束输入：”,程序读取用户输入，保存入空数组；直到用户输入了end，程序输出已经保存的所有员工。 提示：死循环+break

练习：使用while和for输出九九乘法表

|  |
| --- |
| 面试题：请谈谈“面向过程编程”和“面向对象编程”的区别？  **面向过程：**分析业务逻辑，理清其中的流程，分割为若干过程——function  **面向对象：**分析业务逻辑，理清其中的数据对象及关系，封装为若干对象，对象间互相发消息——object |

2.Python中使用函数

定义：函数用于封装一批需要反复执行的代码

**创建函数：**

def 函数名( 形参列表 ) :

函数主体

return 返回值

**调用函数：**

变量 = 函数名( 实参列表 )

注意：Python中实参的个数必须与形参个数完全一样！！！

**默认值参数：**

def add(n1, n2=200):

pass

调用可以有两种形式：

add(10) #n1=10; n2=200

add(10, 20) #n1=10; n2=20

注意：有默认值的形参必须声明在所有无默认值的形参之后！

**可变数量的参数：**

def add( \*nums ):

pass #此处的nums是一个tuple

add(10, 30, 50)

-----------------------------------------------------

def add( \*\*money ):

pass #此处的money是一个dict

add(jan=1000, feb=3000, may=5000)

**关键字参数：**

如果函数的形参过多，因为顺序很重要，不方便记忆，实参列表中可以使用关键字参数：每个实参都有一个名字——该实参对应的形参名

def connect(host, user, pwd, port, dbname):

pass

connect(user='root', pwd='123456', port=3306, dbname='xz', host='127.0.0.1')

练习：创建函数getLeapYears(start, end)，返回一个数组，其中包含指定范围(start~end)之间所有的闰年

练习：创建函数getPrimeNumber(start, end)，返回一个数组，其中包含指定范围(start~end)之间所有的质数

3.变量的作用域问题 —— 难点

Python中，变量的作用域分为四类：简称LEGB

①Local：本地作用域，指定义在函数内的变量，只能在当前函数内使用

②Embeded：嵌入作用域，或称为闭包作用域，指定义在外部函数，但要供内部函数在外部使用的变量

③Global：全局作用域，指定义在任意函数外面，可供当前脚本中的任意函数使用

④Built-In：内置作用域，指Python解释器提供的变量和函数，可以在任意脚本中使用

**总结：四种作用域下若都有某个变量，其优先级为：Local > Embeded > Global > Built-In**

|  |
| --- |
| 面试题：类和对象的关系？  面试题：面向对象三大基本特征？ |

4.面向对象编程

class：类，是一组事物共性的抽象——类是抽象的，没有具体值的

class = attributes + methods

object：对象，是从属于某个类的一个具体实例——对象是具体的

面向对象的三个基本特征：

**①封装**：把原本离散的数据和操作包装为一个整体——定义class

class Emp :

def \_\_init\_\_(**self**, age, ename) : #构造方法

self.age = age #封装属性

self.ename = ename #封装属性

def printInfo(**self**): #封装方法

print(self.ename, self.age)

e2 = Emp(20, 'Tom') #构建实例不要使用new关键字

**②继承**：让一个类自动获取另一个类中全部的成员，从而提高代码可维护性

class Emp:

pass

class Coder**( Emp )**: #Python支持多继承

def \_\_init\_\_(self)

super().\_\_init\_\_() #在子类构造方法内调用父类构造方法

c3 = Coder( ) #创建子类对象，其体内有一个父类的对象

**③多态**：一个方法，根据对象的不同(父或子)，可能运行出多种结果

class Emp:

def printInfo():

pass

class Coder(Emp):

def printInfo(): #子类重写/覆盖了父类的同名方法

pass

-------------------------------------

e = Emp()

e.printInfo() #第一次调用

e = Coder()

e.printInfo() #第二次调用，与第一次调用结果不同

说明：

①Python中没有private/protected/public访问修饰符，如果想让一个属性只能在本类内部使用不能在类外使用：只需要在属性名前添加两个下划线即可；例如： self.\_\_age = 30

②Python中没有static关键字，想实现类似的效果，可使用“类属性”——直接声明在类内（不是方法内），且不用self.来访问：

class Emp:

location = '中国' #类属性，访问方法：Emp.location

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

self.age = 30 #实例属性

练习：画图板应用中需要一个类Shape，有属性fillColor、borderColor，以及两个方法getSize()、getPerimeter()；继续创建类Circle和Rectangle，都继承自Shape，但是分别多了属性：半径(r)、宽(with)和高(height)，试着创建这三个类

5.Python中的异常处理

Exception：异常，可能发生也可能不发生错误！如果没发生程序可以正常执行下去；反之若发生了就可能导致程序终止执行。

示例：请用户从键盘输入两个数字，程序计算出数字的商，输出在屏幕上

try:

#可能发生异常的语句

#...

except XxError as e:

print(e)

except YyError as e:

print(e)

except Exception as e:

print(e)

finally:

#无法异常发生与否都要执行的收尾处理

|  |
| --- |
| 如果需要根据业务逻辑，可以手工抛出错误： raise Exception('错误消息')  注意：手工抛出的错误也必须被try..except... |

课后任务：

(1)扩展项目：“MFresh净美仕官网”（http://www.codeboy.com/mfresh）

项目描述：旧有项目的重构

净美仕官网原本是用PHP编写的前后端没有分离的项目；请使用前后端分离技术实现该项目的重构，要求前端子系统使用Vue.js/Node.js，后端子系统使用Python/Django实现