2016年6月30日星期四

1：设计和调试的一些原则：

1. if的嵌套最好不要超过两层，当需要两层的时候，你应该将第二个if放到函数中
2. 将if看做段落，前后留出空行作为区分
3. If()中的布尔判断应该简单，如果复杂，应该放到一个变量中。
4. 最好的调试是使用print在想要检查的关键环节打印出来，从而检查哪里有错。

2：类的定义类似函数：

Eg：

#注意类的声明方法：

class Thing(object):

#self是需要有的 否则报错

def test(self,hi):

print hi

a = Thing() # 这是声明一个对象

a.test("hahaha") #调用类方法

3：关于数组（应该说是列表）：

1. pop()方法是从数组的最后的元素开始取
2. append()函数是添加元素，确切的说不应该是数组应该是list，因为长度不限
3. 列表下标从0开始！！！
4. 特别注意：-1是最后一个，下标从-1开始 ，从后向前遍历
5. print "stuff[-1] == ",stuff[-1]
6. print "stuff[-2] == ",stuff[-2]
7. listName[i:j]：功能是取列表下标从i到j的元素，不包括j。类似range()

Eg：

test\_things = "Apple Orange Crows Telephone Light Suger"

print "There is not 10 things in that list,let's fix it."

stuff = test\_things.split(' ')

more\_stuff = ["Mon","Tues","Wed","Thris","Fir","Sat","Sun","MOON"]

while len(stuff)!=10:

#注意：pop()方法是从后往前出，先出最后一个

next = more\_stuff.pop()

print "Adding ", next

#append()方法是增加

stuff.append(next)

print "There are %d elements in list " %len(stuff)

print "Here we go: ",stuff

#注意：下标从0开始！！！

print stuff[1]

#注意：-1是最后一个，下标从-1开始 ，从后向前遍历

print "stuff[-1] == ",stuff[-1]

print "stuff[-2] == ",stuff[-2]

print stuff.pop()