# **创意编程班第二周周末复盘总结**



## 复盘作业

复盘：1）整理目前学到的东西 2）整理学习所犯的错误以及当时的思维过程 3）整理对今后学习有指导意义的方法；

老规矩，我给一个模版

## 1.本周学习到的东西

每学到一个东西，就像你的工具集之中多了一件工具，以后你一定会使用到的，试着用自己的话说给组队的小伙伴听，你要如何让他听懂呢？

[1,2,”3”,”aaa”]

{“aa”:”bb”, “cc”:”dd”}

(1,2,3)

以上这三个数据结构有何异同？什么场景时候使用，说出自己的理解

[1,2,”3”,”aaa”]是一个列表，里面包含4个对象，对象可以是数字或字符串，甚至其他列表，它就像是一个菜单目录一样，这个目录上有你需要的所有元素。列表中的对象可以增加（append），可以修改，可以删除（del），在我们需要用如for循环在列表中的各个元素间递归使用时，就可以用上列表，需要注意的是，Python从0开始计数，所以列表中的第一个对象计数是0，第二个对象计数是1，以此类推。

{“aa”:”bb”, “cc”:”dd”}是一个字典，字典包括键和值，就像是名字和解释详情，用花括号扩起，每一对键值用冒号分割，每一组键值用逗号分割。键是不可变的（唯一的），值是可变的。字典里的元素可以增加、可以修改、可以删除、可以遍历，字典里的对象是没有顺序的。在我们除了需要知道名字以外，可能还需要知道每个名字所代表的详细信息，就可以用到字典。

(1,2,3)是一个元组，和列表类似，不同点在于元祖是不可变的，不能够被增加、修改或删除，元组内的对象用逗号分割，外用圆括号。访问元组内的对象时用方括号[ ]来指明对象的位置。在我们需要一组稳定的、绝对不会变的值时，可以用元组。只包含一个元素的元组需要在第一个对象后面加逗号，如lst(3,)。

try/except

这个编程语法存在的价值？说出你的理解

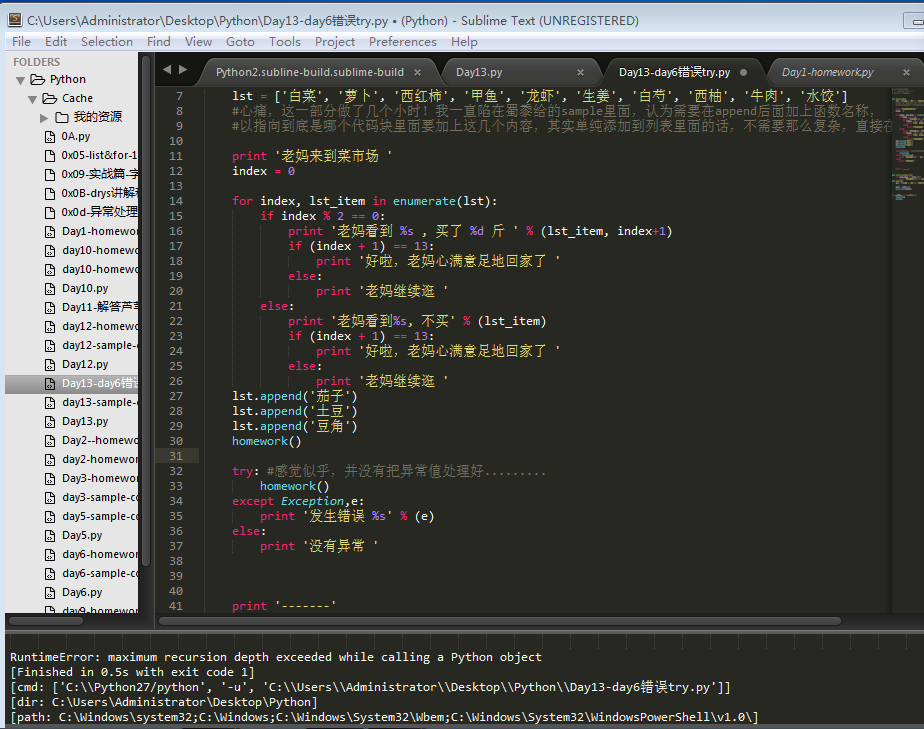
在程序发生异常时，程序会中止不再继续往下进行，但我们在编写程序时是希望程序拥有一定的容错能力，在出现某一个异常以后可以记录下来，然后程序继续往下进行而不受影响。try/excpet可以帮助我们找到异常并设置相关记录，使原本中止的程序能够继续往下进行。

“Don’t repeat yourself” 原则是什么

当自己的编写的程序代码中有三处以上的重复地方，就可以开始思考，需要这么繁杂吗，是不是有更简便的办法使得不需要总是重复打代码，这是一个不断自省的原则。

## 2.本周编程中所犯的错误以及思维过程

1）我在写异常处理的代码时，try/except似乎并没有让程序继续往下进行，显示出runtime的错误，但这一异常没有被处理好，程序没有执行我写的except语句后的指令。当时我以为所有的错误，包括语法错误和逻辑错误以及条件判断的中止等都是可以通过异常处理来解决的，实际上好像处理方法会有所不同，还是很疑惑，仍在尝试中，并没有得到答案。



---

