

UCLASSKF

Week 3



UClass CSC148

2017 Winter

Exception：

简单的句法：

try:

<语句> #运行别的代码

except <名字>：

<语句> #如果在try部份引发了'name'异常

else:

<语句> #如果没有异常发生

try的工作原理是，当开始一个try语句后，python就在当前程序的上下文中作标记，这样当异常出现时就可以回到这里，try子句先执行，接下来会发生什么依赖于执行时是否出现异常。

* 如果当try后的语句执行时发生异常，python就跳回到try并执行第一个匹配该异常的except子句，异常处理完毕，控制流就通过整个try语句（除非在处理异常时又引发新的异常）。
* 如果在try后的语句里发生了异常，却没有匹配的except子句，异常将被递交到上层的try，或者到程序的最上层（这样将结束程序，并打印缺省的出错信息）。
* 如果在try子句执行时没有发生异常，python将执行else语句后的语句（如果有else的话），然后控制流通过整个try语句。

还可以在同一个except使用多个error

... **except** (RuntimeError, TypeError, NameError):

... **pass**

3个注意点

1.可以将exception变成一个变量, ex1.py

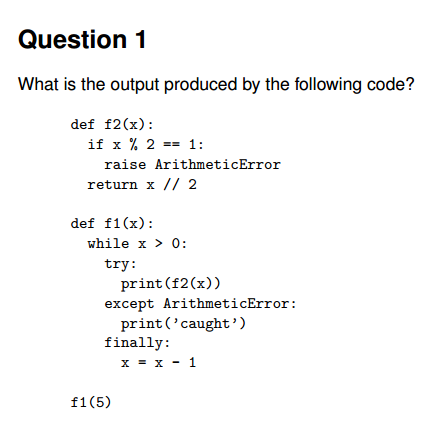
2.自定义exception, ex2.py

3.使用finally来结束try，在任何情况finally都会最后运行，ex3.py

**List Comprehensions:**

* [<expression> for <variable> in <iterable>], lc.py,ex1
* Zip,将两个iterable加到一起， 注意，两个size不一样也可以，但是只到小的iterable结束为止， ex2
* List comprehensions with zip, ex3
* [<expression> for <variable1> in <iterable1>, for <variable2> in <iterable2>]. Ex4
* [<expression> for <variable> in <iterable> if <condition>] , ex5
* Filter, 需要一个function，ex6
* Any, 任意一个是true即可
* If/else，ex7

Example:





课件及py文件

https://github.com/tangkaiq/uclass/tree/master/w3