



2017 Winter

UClass CSC148

Week 3



UCLASSKF

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:

Question 1

What is the output produced by the following code?

```
def f2(x):
    if x % 2 == 1:
        raise ArithmeticError
    return x // 2

def f1(x):
    while x > 0:
        try:
            print(f2(x))
        except ArithmeticError:
            print('caught')
        finally:
            x = x - 1

f1(5)
```

Question 2

What is the output produced by parts a to e in the following code? Indicate the type of error followed by its corresponding print statement.

```
def division(a, b):
    try:
        answer = a / b
    except ZeroDivisionError:
        print('Cannot divide by zero')
    except TypeError:
        print('Wrong type')
    except NameError:
        print('Name not defined inside')
    else:
        print('Answer available')
    finally:
        print('End of program')

if __name__ == '__main__':
    # Part a
    division(6, 0)

    # Part b
    division(0, 42)

    # Part c
    division(100, 50)

    # Part d
    division('science', 1)

    # Part e
    try:
        division(x, 1)
    except NameError:
        print('Name not defined outside')
```

课件及 py 文件

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