

## 7-异常、模块、包

### 7-1 异常

语法：

```
1 try:
2     可能发生错误的代码
3 except:
4     如果出现异常执行的代码
```

捕获指定异常：

```
1 try:
2     print(num)
3 except NameError:
4     print('error')
5
6 #如果尝试执行的代码异常类型和要捕获的异常类型不一致，则无法捕获异常
7 #一般try下方只放一行尝试执行的代码
```

捕获多个异常：

```
1 try:
2     print(1/0)
3 except (NameError,ZeroDivisionError):
4     print('error')
```

捕获异常描述信息：

```
1 try:
2     print(1/0)
3 except (NameError,ZeroDivisionError) as result:
4     print(result)
5
6 #result:division by zero
```

捕获所有异常：

Exception是所有程序异常类的父类

```
1 try:
2     print(num)
3 except Exception as result:
4     print(result)
5
6 #result:name 'num' is not defined
```

异常的else：

表示没有异常要执行的代码。

```
1 try:
2     print(1)
3 except Exception as result:
4     print(result)
5 else:
6     print("没有异常")
```

异常的finally：

无论是否存在异常都要执行的代码。

```
1 try:
2     f = open('test.txt','r')
3 except Exception as result:
```

```

4     print(result)
5     f = open('test.txt', "w")
6 else:
7     print("no error")
8 finally:
9     f.close()
10
11 #result:No such file or directory: 'test.txt'

```

## 7-2 异常的传递

```

1 import time
2 # 需求:
3 #     1、尝试只读打开test.txt文件, 如果文件存在则读取文件内容, 文件不存在则提示用户即可
4 #     2、读取文件需求: 尝试循环读取内容, 读取过程中如果监测到用户意外终止程序, 则except捕获异常并提示用户
5 try:
6     f = open('test.txt', "r")
7     try:
8         while True:
9             content = f.readline()
10            if len(content) == 0:
11                break
12            time.sleep(2)
13            print(content)
14        except Exception as result:
15            print(result)
16    finally:
17        f.close()
18        print('关闭文件')
19 except Exception as result:
20     print(result)

```

## 7-3 自定义异常

```

1 #需求: 密码长度不足, 则报错
2 class ShortInputError(Exception):
3     def __init__(self, length, min_len):
4         self.length = length
5         self.min_len = min_len
6     def __str__(self):
7         return f'您输入的长度是{self.length}, 不能少于{self.min_len}个字符'
8
9 if __name__ == '__main__':
10     try:
11         con = input("请输入密码: ")
12         if len(con) < 6:
13             # 自定义异常语法 raise 异常类对象
14             raise ShortInputError(len(con), 6)
15         except Exception as result:
16             print(result)
17     else:
18         print("密码输入成功")

```

## 7-4 模块

## 模块定位顺序：

- 1、当前目录
- 2、如果不在当前目录，Python则搜索在shell变量PYTHONPATH下的每个目录
- 3、如果都找不到，Python会查看默认路径（/usr/local/lib/python/）。

## \_\_all\_\_：

当使用 `from xxx import *` 导入时，如果模块文件中有\_\_all\_\_变量，只能导入这个列表中的元素。

```
1 __all__ = ["testA"]
2 def testA():
3     print("testA")
4 def testB():
5     print("testB")
```

## 7-5 包

\_\_init\_\_.py:

```
1 __all__ = ["add", "sub"]
```

add.py:

```
1 def add(*args):
2     num = 0
3     for i in args:
4         num += i
5     return num
```

main.py:

```
1 from my_module import *
2 print(add.add(3, 5, 5))
```