



4.5 异常处理

■ **异常**：程序运行时的**错误**，对应一个**Python对象**。

□ try-except语句

try:

语句块

except 异常1 as 错误原因:

出现异常1后的处理代码

except 异常2 as 错误原因:

出现异常2后的处理代码

- 在程序运行时，解释器尝试执行try语句块中的所有代码。
- 如果语句块被执行完后没有异常发生，就会忽略except后的代码。
- 当某个except所指定的异常发生后，会忽略try子句中剩余的语句，直接跳转到对应异常的处理代码处执行。



例：异常处理

```
try:
    aList = [0,1,2]
    print(aList[3])
    print("try语句块继续执行中.....")
except IndexError as e:
    print(e)
    print("异常已处理")

print("程序继续执行中.....")
```

运行结果：

```
list index out of range
异常已处理
程序继续执行中.....
```

```
try:
    aList = [0,1,2]
    print(aList[2])
    print("try语句块继续执行中.....")
except IndexError as e:
    print(e)
    print("异常已处理")

print("程序继续执行中.....")
```

运行结果：

```
2
try语句块继续执行中.....
```



□ Python中常见的异常

- **IOError**: 输入/输出异常; (基本上是无法打开文件)
- **ImportError**: 无法导入模块或包; (出现这个异常基本上是路径问题或名称错误)
- **IndentationError**: 缩进错误;(代码没有正确对齐)
- **NameError**: 没有声明、或初始化对象
- **KeyError**: 试图访问字典里不存在的键
- **AttributeError**: 试图访问一个对象没有的属性
- **TypeError**: 类型不匹配
- **ValueError**: 传入一个调用者不期望的值, 即使值的类型是正确的



4.5 异常处理

Python异常类的继承关系

```
1 BaseException
2 +-- SystemExit
3 +-- KeyboardInterrupt
4 +-- GeneratorExit
5 +-- Exception
6     +-- StopIteration
7     +-- ArithmeticError
8         +-- FloatingPointError
9         +-- OverflowError
10        +-- ZeroDivisionError
11    +-- AssertionError
12    +-- AttributeError
13    +-- BufferError
14    +-- EOFError
```

```
1 BaseException
2 +-- SystemExit
3 +-- KeyboardInterrupt
4 +-- GeneratorExit
5 +-- Exception
6     +-- StopIteration
7     +-- ArithmeticError
8         +-- FloatingPointError
9         +-- OverflowError
10        +-- ZeroDivisionError
11    +-- AssertionError
12    +-- AttributeError
13    +-- BufferError
14    +-- EOFError
15    +-- ImportError
16    +-- LookupError
17        +-- IndexError
18        +-- KeyError
19    +-- MemoryError
20    +-- NameError
21        +-- UnboundLocalError
22    +-- OSError
23        +-- BlockingIOError
24        +-- ChildProcessError
25        +-- ConnectionError
26            +-- BrokenPipeError
27            +-- ConnectionAbortedError
28            +-- ConnectionRefusedError
29            +-- ConnectionResetError
30        +-- FileExistsError
31        +-- FileNotFoundError
32        +-- InterruptedError
33        +-- IsADirectoryError
34        +-- NotADirectoryError
35        +-- PermissionError
36        +-- ProcessLookupError
37        +-- TimeoutError
38    +-- ReferenceError
39    +-- RuntimeError
40        +-- NotImplementedError
41    +-- SyntaxError
42        +-- IndentationError
43        +-- TabError
44    +-- SystemError
45    +-- TypeError
46    +-- ValueError
47        +-- UnicodeError
48            +-- UnicodeDecodeError
49            +-- UnicodeEncodeError
50            +-- UnicodeTranslateError
51    +-- Warning
52        +-- DeprecationWarning
53        +-- PendingDeprecationWarning
54        +-- RuntimeWarning
55        +-- SyntaxWarning
56        +-- UserWarning
57        +-- FutureWarning
58        +-- ImportWarning
59        +-- UnicodeWarning
60        +-- BytesWarning
61        +-- ResourceWarning
```



□ Python异常类的继承关系

```
1 BaseException
2 +-- SystemExit
3 +-- KeyboardInterrupt
4 +-- GeneratorExit
5 +-- Exception
6     +-- StopIteration
7     +-- ArithmeticError
8         +-- FloatingPointError
9         +-- OverflowError
10        +-- ZeroDivisionError
11 +-- AssertionError
12 +-- AttributeError
13 +-- BufferError
14 +-- EOFError
```

□ Exception

- 是所有非系统退出类异常类的基类
- 在编程时，可以通过捕获它，来避免程序遇到错误而退出

```
try:
    语句块
except exception as e:
    异常处理语句块
```



□ finally子句：无论异常是否发生，都会执行

```
try:  
    语句块  
except 异常 as 错误原因:  
    出现异常后的处理代码  
finally:  
    语句块
```

- 如果try语句块中没有出现任何运行时错误，会跳过except语句块执行finally语句块的内容。
- 如果出现异常，则会先执行except语句块的内容，再执行finally语句块的内容。
- finally语句块经常用于**关闭资源**等清理工作。



例: try-except-finally语句

```
try:
    print("try语句块开始")
    f = open('c:/test.txt')
    print(f.read())
    print("try语句块结束")
except IOError as e:
    print("except子句")
    print(e)
finally:
    print("finally子句")
    f.close()
```

运行结果:

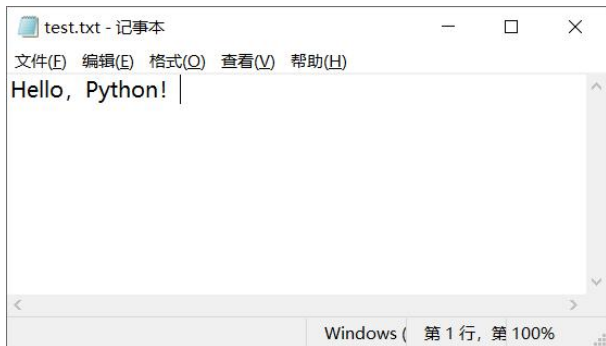
```
try语句块开始
except子句
[Errno 2] No such file or directory: 'c:/test.txt'
finally子句
```

“找不到文件” 错误



4.5 异常处理

在c盘根目录下创建文件test.txt,
并且写入 "Hello, Python! "



```
try:
    print("try语句块开始")
    f = open('c:/test.txt')
    print(f.read())
    print("try语句块结束")
except IOError as e:
    print("except子句")
    print(e)
finally:
    print("finally子句")
    f.close()
```

运行结果:

```
try语句块开始
Hello, Python!
try语句块结束
finally子句
```

