

Python语言基础与应用

高级特性 / 例外处理

陈斌 北京大学 gischen@pku.edu.cn



例外处理

- › 代码错误类型
- › 捕捉错误

代码错误类型

› 代码运行可能会意外各种错误

程序的逻辑错误、用户输入不合法等都会引发异常，但它们不会导致程序崩溃

可以利用python提供的异常处理机制，在异常出现时及时捕获并从内部消化掉



代码错误类型

› 语法错误: `SyntaxError`

```
>>> print "Hello World"  
SyntaxError: Missing parentheses in call to 'print'
```

› 除以0错误: `ZeroDivisionError`

```
>>> 4 / 0  
Traceback (most recent call last):  
  File "<pyshell#9>", line 1, in <module>  
    4 / 0  
ZeroDivisionError: division by zero
```

代码错误类型

› 列表下标越界：IndexError

```
>>> alist = [1, 2, 3, 4]
>>> alist[4]
Traceback (most recent call last):
  File "<pyshell#8>", line 1, in <module>
    alist[4]
IndexError: list index out of range
```

› 类型错误：TypeError

```
>>> mix = ['Connor', 'Bruce', 'John', 'Alice', 7, 3, 9]
>>> mix.sort()
Traceback (most recent call last):
  File "<pyshell#21>", line 1, in <module>
    mix.sort()
TypeError: '<' not supported between instances of 'int' and 'str'
```

代码错误类型

› 访问变量不存在: NameError

```
>>> x
Traceback (most recent call last):
  File "<pyshell#12>", line 1, in <module>
    x
NameError: name 'x' is not defined
```

› 字典关键字不存在: KeyError

```
>>> adict = {"A":1, "B":2, "C":3, "D":4}
>>> adict["E"]
Traceback (most recent call last):
  File "<pyshell#17>", line 1, in <module>
    adict["E"]
KeyError: 'E'
```

代码错误类型

› 未知的变量属性: AttributeError

```
>>> alist = [1, 2, 3, 4]
>>> alist.sorted()
Traceback (most recent call last):
  File "<pyshell#19>", line 1, in <module>
    alist.sorted()
AttributeError: 'list' object has no attribute 'sorted'
```

› 以上这些错误会引起程序中止退出，
如果希望掌控意外，就需要在可能出
错误的地方设置陷阱捕捉错误

捕捉错误

› try-except语句

try:

 <检测语句>

except <错误类型> [as e]:

 <处理异常>

try: # 为缩进的代码设置陷阱

except: # 处理错误的代码

› 针对不同异常可以设置多个except

捕捉错误

› try-finally语句

try:

 <检测语句>

except <错误类型> [as e]:

 <处理异常>

finally:

 <语句块>

finally: # 无论出错否，都执行的代码

如果try语句块运行时没有出现错误，会跳过except语句块执行finally语句块的内容

捕捉错误

› else语句

try:

 <检测语句>

except <错误类型> [as e]:

 <处理异常>

else:

 <语句块>

else: # 没有出错执行的代码

捕捉错误

```
1 try:
2     print('try...')
3     r = 10 / 'xyz'
4     print('result:', r)
5 except TypeError as e:
6     print('TypeError:', e)
7 except ZeroDivisionError as e:
8     print('ZeroDivisionError:', e)
9 else:
10    print('no error!')
11 finally:
12    print('finally...')
13 print('END')
```

```
try...
TypeError: unsupported operand type(s) for /: 'int' and 'str'
finally...
END
```