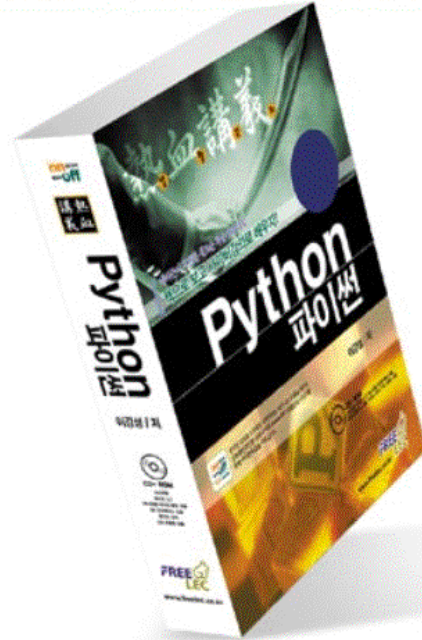


熱血講義

프리렉의 열혈강의 시리즈

Python 파이썬



1

파이썬 (Python)

Python

❖ 13

:

(gslee@mail.gwu.ac.kr₂)



1.
 1. try/except/else
 2. try/finally
 - 3.
 - 4.
- 2.



- (syntax error)

➤ .

- (exception)

➤

➤ 0 , ,
..

(1)

```
>>> 5 / 0
Traceback (most recent call last):
  File "<pyshell#2>", line 1, in ?
    5 / 0
ZeroDivisionError: integer division or modulo by zero

>>> 4 + spam*3
Traceback (innermost last):
  File "<pyshell#23>", line 1, in ?
    4 + spam*3
NameError: spam

>>> '2' + 2
Traceback (innermost last):
  File "<pyshell#27>", line 1, in ?
    '2' + 2
TypeError: illegal argument type for built-in operation
```

(2)

```
>>> L = [1,2,3]
>>> L[5]
Traceback (most recent call last):
  File "<pyshell#4>", line 1, in ?
    L[5]
IndexError: list index out of range

>>> D = {'one':1}
>>> D['two']
Traceback (most recent call last):
  File "<pyshell#6>", line 1, in ?
    D['two']
KeyError: two
```

13-1 try / except / else

try/except/else

```
try:
```

```
1
```

```
except
```

```
1:
```

```
2
```

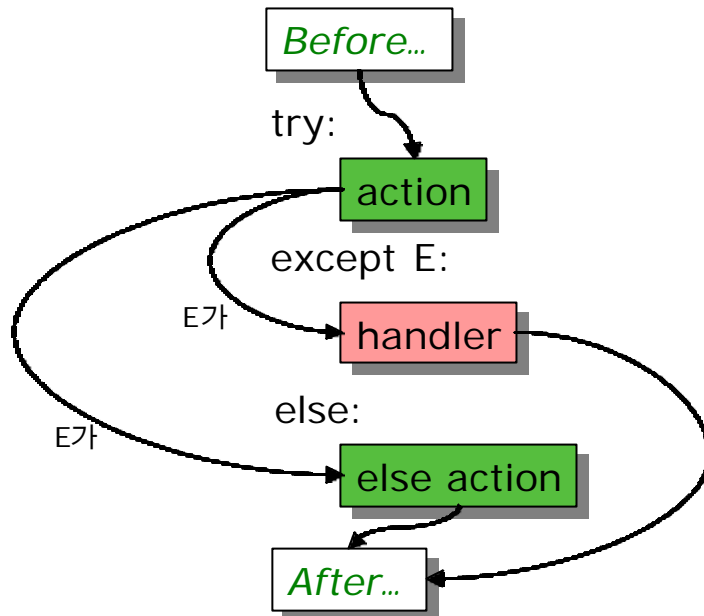
```
except
```

```
2:
```

```
3
```

```
else:
```

```
4
```



7

13-1 try / except / else

try/except/else

```
try:
```

```
    print 1.0/x
```

```
except ZeroDivisionError:
```

```
    print 'has no inverse'
```

```
#-----
```

```
name = 'notexistingfile'
```

```
try:
```

```
    f = open(name, 'r')
```

```
except IOError:
```

```
    print 'cannot open', name
```

```
else:
```

```
    print name, 'has', len(f.readlines()), 'lines'
```

```
    f.close()
```

```
#-----
```

```
try:
```

```
    spam()
```

```
except NameError, x:
```

```
    print 'name', x, 'undefined'
```

8

13-1 try / except / else

```
#
def this_fails():
    x = 1 / 0

try:
    this_fails()
except ZeroDivisionError, msg:
    print 'Run-timeerror:', msg

#
try:
    some_job()
except ZeroDivisionError:
    ...
except KeyError:
    ...
except (TypeError, IOError):
    ...
else:
    ...
```

9

13-2 try/finally

• finally

(except

)

```
#
f = open(filename, "w")
try:
    do_something_with(f)
finally:
    f.close()
```

(?)

10

13-3



13-3

Exception	가
StandardError	SystemExit
ArithmeticError	OverflowError, ZeroDivisionError, FloatingPointError
LookupError	IndexError, KeyError
EnvironmentError	(IOError, OSError)

13-4



```
def dosomething():
    a = 1/0      # ZeroDivisionError

try:
    dosomething()
except ArithmeticError:
    print 'Arithmetic Exception occurred'
```

13-5 raise



```
MyException = 'my exception'

def dosomething():
    raise MyException, 123

try:
    dosomething()
except MyException, param:
    print 'Exception occurred', param
```

Exception occurred 123

13

13-5 raise



```
class SquareSeq:
    def __init__(self, n):
        self.n = n
    def __getitem__(self, k):
        if k >= self.n or k < 0:
            raise IndexError
        return k*k
    def __len__(self):
        return self.n
```

14

13-5



```
class MyException(Exception):
    pass

try:
    raise MyException
except MyException:
    print 'caught'
```

15

13-6 assert



- assert ,
- '가 'raise
- AssertionError, '
- -O

```
>>> assert 10 > 4 * 2, '10 is bigger'
>>> assert 10 > 4 * 3, '10 is bigger'
Traceback (most recent call last):
  File "<pyshell#1>", line 1, in ?
    assert 10 > 4 * 3, '10 is bigger'
AssertionError: 10 is bigger
```

16