



예외처리

Update – 2018.06

Contents

- 오류 발생 알아보기
- try.. except..문
- try.. else 문
- try .. finally 문
- raise 문

오류발생 알아보기

- 오류는 어떤 때 발생하는가?
: 파일이 없는 경우 발생

```
>>> f = open("나없는파일", 'r')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
FileNotFoundError: [Errno 2] No such file or directory: '나없는파일'
```

오류발생 알아보기

- 오류는 어떤 때 발생하는가?
: 0으로 나눈 경우

```
>>> a = 10/0; a  
ZeroDivisionError: division by zero
```

오류 발생 알아보기

- 오류는 어떤 때 발생하는가?
: 리스트 구조에서 index 없는 경우 발생

```
>>> fruit = ['사과','포도','배']  
>>> fruit[3]  
IndexError: list index out of range
```



try.. except..문



- try, except : 발생한 오류를 메시지로 출력

```
try:  
    ...  
except [발생 오류[as 오류 메시지 변수]]:  
    ...
```

```
try:  
    4 / 0  
except ZeroDivisionError as e:  
    print(e)
```



try, except문



- try .. except : 발생한 오류 무시. pass 문 이용

```
try:  
    ...  
except [발생 오류]:  
    pass
```

```
try:  
    4 / 0  
except ZeroDivisionError:  
    pass
```

```
try:  
    f = open("나없는파일", 'r')  
except FileNotFoundError:  
    pass
```

try .. else문

- try .. else
: 오류 발생시 except 문 실행. 아니면 else문 실행

```
try:  
    ...  
except [발생 오류[as 오류 메시지 변수]]:  
    ...  
else:  
    ...
```

```
try:  
    f = open('foo.txt', 'r')  
except FileNotFoundError as e:  
    print(str(e))  
else:  
    data = f.read()  
    f.close()
```


try .. else문

- A, b 입력 값에 따라 메시지 출력하기

```
a = int(input('a = '))  
b = int(input('b = '))  
try:  
    int(a) / int(b)  
except ZeroDivisionError:  
    print('오류 발생')  
else:  
    print(a/b)
```

a = 60
b = 0
오류 발생

a = 100
b = 50
2.0



try.. finally..문



- try .. finally... : 예외처리이후 하나의 실행문은 항상 수행

```
try:  
    ...  
finally:  
    ...
```

```
f = open('foo.txt', 'w')  
try:  
    # 무언가를 수행한다.  
finally:  
    f.close()
```

```
try:  
    10/0  
finally:  
    print('명령문 수행 완료')
```

명령문 수행 완료
ZeroDivisionError

try.. finally..문

- except, else 문과 함께 사용하는 try.. finally...

```
try:
    100/0
except ZeroDivisionError as
e:
    print(e)
finally:
    print('명령문 수행 완료')
```

division by zero
명령문 수행 완료

```
a = int(input('a = '))
b = int(input('b = '))
try:
    int(a) / int(b)
except ZeroDivisionError:
    print('오류 발생')
else:
    print(a/b)
finally:
    print('명령문 수행 완료')
```

a = 50
b = 25
2.0
명령문 수행 완료



raise문



- raise... : 사용자 정의 예외처리문. Class와 함께 사용

```
#예외처리 클래스 생성  
class MyError(Exception):  
    pass
```

```
#예외처리 함수  
def say_nick(nick):  
    if nick == '바보':  
        raise MyError()  
    print(nick)
```

```
>>> say_nick('천사')  
천사  
>>> say_nick('바보')  
MyError
```