

23. Python 예외처리

1.1 예외처리

1) 파이썬 예외처리

- 이클립스 등의 통합개발환경 도구에서는 자동으로 실행 전에 구문 에러를 체크 해 줌
- 파이썬은 상대적으로 언어적 문법이 간단하기 때문에 구문 자체의 에러 발생 비율이 낮거나 다른 도구를 사용하여 완벽하게 제거할 수 있음
- 예외 (Exception) : 구문 에러는 없으나 프로그램 실행 중 더 이상 진행 할 수 없는 상황
- 예외가 발생하면 프로그램은 바로 종료
- 예외 클래스의 계층 구조(<https://docs.python.org/2/library/exceptions.html> 참조)

BaseException

+-- SystemExit

+-- KeyboardInterrupt

+-- GeneratorExit

+-- Exception

+-- StopIteration

+-- StandardError

| +-- BufferError

| +-- ArithmeticError

|| +-- FloatingPointError

|| +-- OverflowError

|| +-- ZeroDivisionError

| +-- AssertionError

| +-- AttributeError

.....

1.2 예외처리

1] 예시

```
def division():  
    for n in range(0, 5):  
        try:  
            print("n->" , n)  
            print (" 10.0 / n ->", 10.0 / n)  
        except ZeroDivisionError:  
            msg = "ZeroDivisionError"  
            print ("msg->",msg)
```

division()

결과

```
-----  
n-> 0  
msg-> ZeroDivisionError  
n-> 1  
  10.0 / n -> 10.0  
n-> 2  
  10.0 / n -> 5.0  
n-> 3  
  10.0 / n -> 3.3333333333333335  
n-> 4  
  10.0 / n -> 2.5
```

1.3 예외처리

예시 2 (파일에서 숫자를 읽어와서 읽은 숫자로 나누기를 하는 예제)

```
import os
print ("os.getcwd()->", os.getcwd())
filename = 't.txt'

try:
    f = open(filename, 'r')
except IOError:
    msg = "IOError"
    print ("msg->",msg)
else:
    a = float(f.readline())
    print("a->" , a)
    try:
        answer = 1.0 / a
    except ZeroDivisionError:
        print('ZeroDivisionError3 !!!')
        print ("msg->",msg)
    else:
        print ("answer->", answer)
    finally:
        print ("Finally!!!")
        f.close()
```

결과

```
-----
os.getcwd()-> C:\PyCharmProject\Sources\Wkk1\Wch23
a-> 0.0
ZeroDivisionError3 !!!
msg-> ZeroDivisionError
Finally!!!
```