

AI Programming

Lecture 15

Assignment 11 Solution

```
def exponential(x, N):  
    if N == 0:  
        return 1  
    else:  
        return x**N/factorial(N) + exponential(x, N-1)
```

```
def sine(x, N):  
    if N == 0:  
        return x  
    else:  
        return ((-1)**N)*(x**(2*N+1))/factorial(2*N+1) + sine(x, N-1)
```

```
def cosine(x, N):  
    if N == 0:  
        return 1  
    else:  
        return ((-1)**N)*(x**(2*N))/factorial(2*N) + cosine(x, N-1)
```

Review

- **File I/O**

- `f=open(filename, mode)~f.close()`
- `readline()`, `readlines()`, `read()`
- `writelines()`

11.5 파일 입출력의 심화 내용

11.5.1 파일 및 디렉터리 다루기

File and Directory

- **shutil module**

```
import shutil  
dir(shutil)
```

출력 결과

... 생략 ...

```
'chown', 'collections', 'copy', 'copy2', 'copyfile', 'copyfileobj', 'copymode',  
'copystat', 'copytree', 'disk_usage', 'errno', 'fnmatch', 'get_archive_formats',  
'get_terminal_size', 'get_unpack_formats', 'getgrnam', 'getpwnam', 'ignore_patterns',  
'make_archive', 'move', 'nt', 'os',
```

... 생략 ...

File and Directory

- **os module**

```
import os  
dir(os)
```

출력 결과

... 생략 ...

```
'chdir', 'chmod', 'close', 'closerange', 'cpu_count', 'curdir', 'defpath',  
'device_encoding', 'devnull', 'dup', 'dup2', 'environ', 'errno', 'error', 'execl',  
'execle', 'execlp', 'execlpe', 'execv', 'execve', 'execvp', 'execvpe', 'extsep', 'fdopen',  
'fsdecode', 'fsencode', 'fspath', 'fstat', 'fsync', 'ftruncate', 'get_exec_path',
```

... 생략 ...

File and Directory

- `os.path` **module**

```
import os.path  
dir(os.path)
```

출력 결과

... 생략 ...

```
'dirname', 'exists', 'expanduser', 'expandvars', 'extsep', 'genericpath', 'getatime',  
'getctime', 'getmtime', 'getsize', 'isabs', 'isdir', 'isfile', 'islink', 'ismount',  
'join', 'lexists', 'normcase', 'normpath', 'os', 'pardir', 'pathsep', 'realpath',  
'relpath', 'samefile', 'sameopenfile', 'samestat', 'sep', 'split',
```

... 생략 ...

File and Directory

• Copy

- `shutil.copy(source, target)`
- `shutil.copytree(source, target)`

```
1 inFp, outFp = None, None
2 inStr = ""
3
4 inFp = open("C:/Windows/notepad.exe", "rb")
5 outFp = open("C:/Temp/notepad.exe", "wb")
6
7 while True :
8     inStr = inFp.read(1)
9     if not inStr :
10         break
11     outFp.write(inStr)
12
13 inFp.close()
14 outFp.close()
15 print("---- 이진 파일이 정상적으로 복사되었음 ----")
```

```
import shutil
shutil.copy('C:/Windows/notepad.exe', 'C:/Temp/myNote.exe')
```

출력 결과

```
'C:/Temp/myNote.exe'
```

```
shutil.copytree('C:/CookPython/GIF/', 'C:/Temp/GIF')
```


File and Directory

- **Create/delete directory**

- `os.mkdir('FolderName')`
- `shutil.rmtree('FolderName')`

```
1 import os
2 import shutil
3 os.mkdir('C:/myDir/')
4 os.mkdir('C:/myDir/dir1/')
5 shutil.rmtree('C:/myDir/')

```

출력 결과

정상적으로 처리되면 아무런 메시지도 표시하지 않음

File and Directory

- List of files

- `os.walk('FolderName')`

```
1 import os
2 for dirName, subDirList, fnames in os.walk('C:\\Windows\\debug') :
3     for fname in fnames :
4         os.path.join(dirName, fname)
```

출력 결과

```
'C:\\Windows\\debug\\ESE.TXT'
'C:\\Windows\\debug\\mrt.log'
'C:\\Windows\\debug\\netlogon.log'
'C:\\Windows\\debug\\NetSetup.LOG'
'C:\\Windows\\debug\\PASSWD.LOG'
'C:\\Windows\\debug\\WIA\\wiatrace.bak.log'
'C:\\Windows\\debug\\WIA\\wiatrace.log'
```

File and Directory

- Check file/folder

```
import os.path  
os.path.exists('C:/Windows/notepad.exe')  
os.path.isfile('C:/Windows/notepad.exe')  
os.path.isdir('C:/Windows')
```

출력 결과

모두 True

File and Directory

- **Delete**

- `os.remove('FileName')`

```
import os
os.remove('C:/Temp/myNote.exe')
os.remove('C:/Temp/myNote.exe')
```

출력 결과

FileNotFoundError: [WinError 2] 지정된 파일을 찾을 수 없습니다: 'C:/Temp/myNote.exe'

File and Directory

- **Size of file**

- `os.path.getsize('FileName')`

```
import os.path  
os.path.getsize('C:/CookPython/GIF/dog.gif')
```

출력 결과

22332

File and Directory

- Zip

- `zipfile.write()`

```
1 import zipfile
2 newZip = zipfile.ZipFile('C:/Temp/new.zip', 'w')
3 newZip.write('C:/Windows/notepad.exe', compress_type = zipfile.ZIP_DEFLATED)
4 newZip.close()
```

출력 결과

정상적으로 처리되면 아무런 메시지도 표시하지 않음

File and Directory

- Unzip

- `zipfile.extractall()`

```
1 extZip = zipfile.ZipFile('C:/Temp/new.zip', 'r')
2 extZip.extractall('C:/Temp/')
3 extZip.close()
```

출력 결과

정상적으로 처리되면 아무런 메시지도 표시하지 않음

11.5 파일 입출력의 심화 내용

11.5.2 예외 처리

Exception Handling

- **Exception (예외)**

- Syntax error: Python detects an incorrect statement.

```
>>> print( 0 / 0 )  
SyntaxError: invalid syntax
```

- Exception error

- syntactically correct, but the code is terminated due to an error.

```
>>> print( 0 / 0 )  
Traceback (most recent call last):  
  File "<pysHELL#28>", line 1, in <module>  
    print( 0 / 0 )  
ZeroDivisionError: division by zero
```

Exception Handling

Syntax: 예외 처리

형식

`try:`

예외가 발생할 수 있는 문장

`except(오류):`

예외를 처리하는 문장

예

`try:`

`z = x/y`

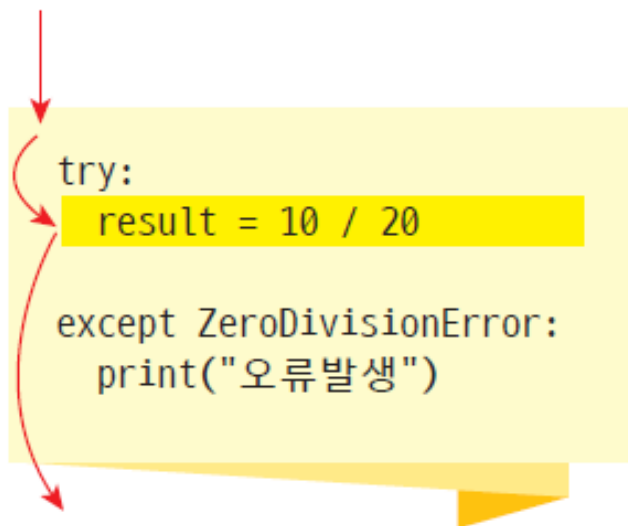
예외가 발생할 수 있는 문장

`except ZeroDivisionError:`

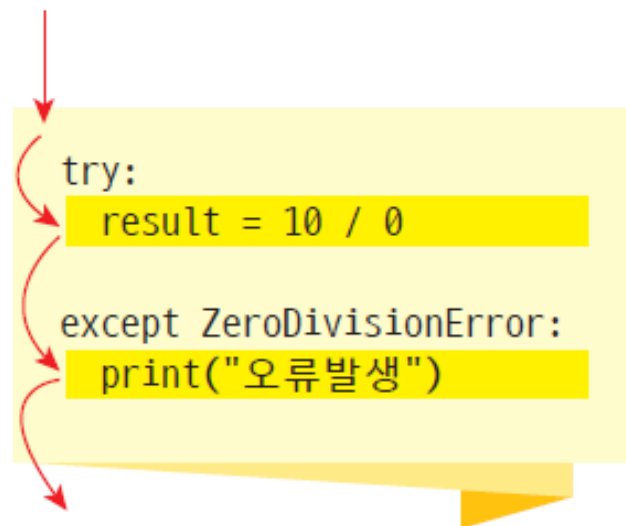
예외

`print ("0으로 나누는 예외")`

Exception Handling



예외가 발생하지 않은 경우



예외가 발생한 경우

Exception Handling

- Exercise

```
x, y = 2, 0
try:
    z = x/y
except ZeroDivisionError:
    print ("0으로 나누는 예외")
```

```
x, y = 2, 0
try:
    z = x/y
except ZeroDivisionError as e:
    print (e)
```

Exception Handling

- Exercise

```
while True:
    try:
        n = input("숫자를 입력하시오 : ")
        n = int(n)
        break
    except ValueError:
        print("정수가 아닙니다. 다시 입력하시오. ")
print("정수 입력이 성공하였습니다!")
```

숫자를 입력하시오 : 23.5
정수가 아닙니다. 다시 입력하시오.
숫자를 입력하시오 : 10
정수 입력이 성공하였습니다!

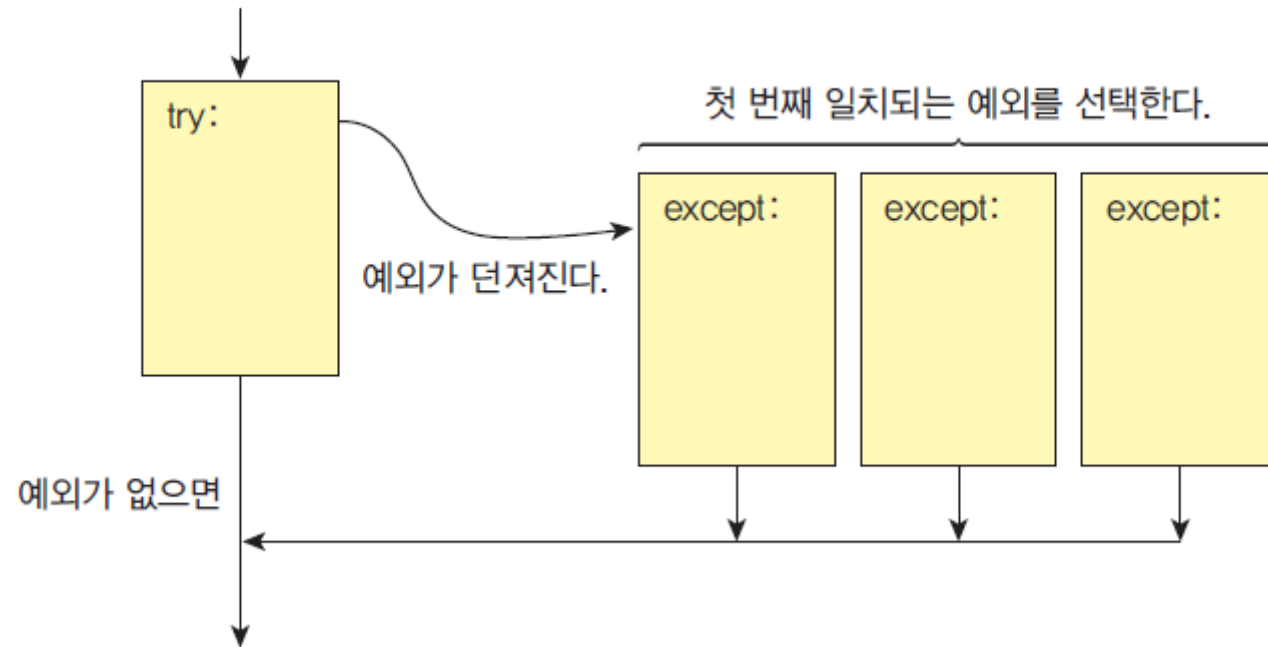
Exception Handling

- Multiple exceptions

```
try :  
    실행할 문장들  
except 예외_종류 1 :  
    오류일 때 실행할 문장들  
except 예외_종류 2 :  
    오류일 때 실행할 문장들
```

종류	설명
ImportError	import 문에서 오류가 발생할 때
IndexError	리스트 등 첨자의 범위를 벗어날 때
KeyError	딕셔너리에서 키가 없을 때
KeyboardInterrupt	프로그램 실행 중 Ctrl + C 를 누를 때
NameError	변수명이 없는 것에 접근할 때
RecursionError	재귀 호출의 횟수가 시스템에서 설정한 것보다 넘칠 때(1000번)
RuntimeError	실행이 발생할 때
SyntaxError	exec()나 eval()에서 문법상 오류가 발생할 때
TypeError	변수형의 오류가 발생할 때. 예 '문자열-문자열' 연산
ValueError	함수의 매개변수에 잘못된 값을 넘길 때. 예 int('파이썬')
ZeroDivisionError	0으로 나눌 때
IOError	파일 처리 등 오류일 때

Exception Handling



Exception Handling

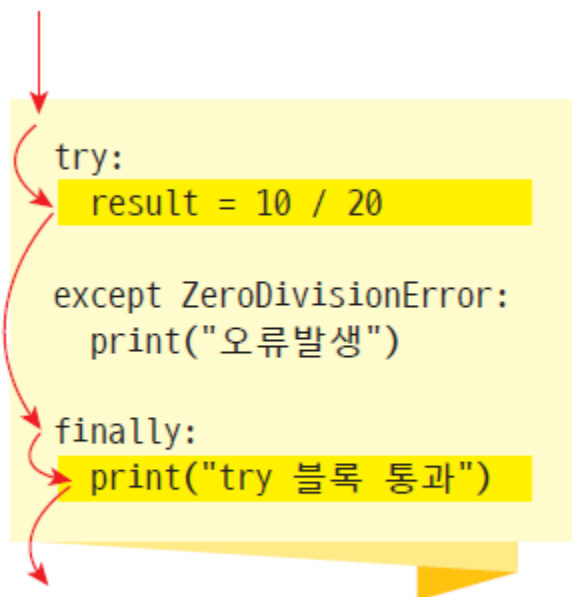
- Handling multiple exceptions

```
1 num1 = input('숫자1 -->')
2 num2 = input('숫자2 -->')
3
4 try :
5     num1 = int(num1)
6     num2 = int(num2)
7     while True :
8         res = num1 / num2
9
10 except ValueError :
11     print('문자열은 숫자로 변환할 수 없습니다.')
12
13 except ZeroDivisionError :
14     print('0으로 나눌 수 없습니다.')
15
16 except KeyboardInterrupt :
17     print('Ctrl+C를 눌렀군요.')
```

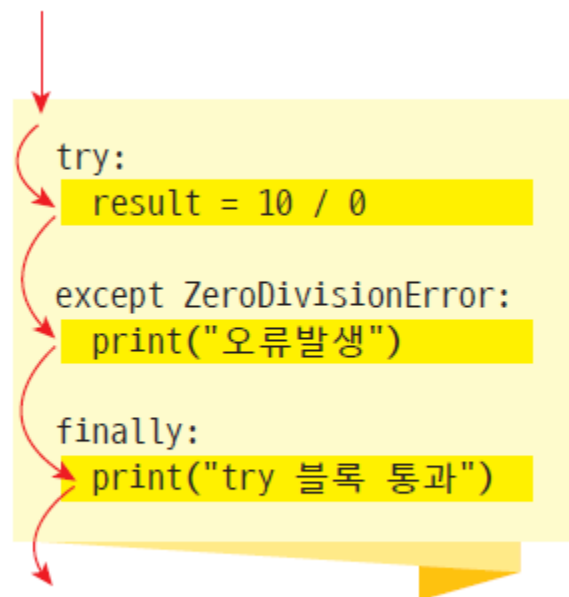
출력 결과

숫자1 -->이건 뭐죠?
숫자2 -->10
문자열은 숫자로 변환할 수 없습니다.
숫자1 -->10
숫자2 -->0
0으로 나눌 수 없습니다.
숫자1 -->10
숫자2 -->5
Ctrl+C를 눌렀군요. ← 무한 반복에서 Ctrl+C를 눌렀을 때

Exception Handling



예외가 발생하지 않은 경우



예외가 발생한 경우

Exception Handling

- try~except~else~finally

```
1 num1 = input('숫자1 -->')
2 num2 = input('숫자2 -->')
3
4 try :
5     num1 = int(num1)
6     num2 = int(num2)
7
8 except :
9     print('오류가 발생했습니다.')
10
11 else :
12     print(num1, '/', num2, '=', num1 / num2)
13
14 finally :
15     print('이 부분은 무조건 나옵니다.')
```

출력 결과

숫자1 -->100
숫자2 -->50
100 / 50 = 2.0
이 부분은 무조건 나옵니다.
숫자1 -->100
숫자2 -->모름
오류가 발생했습니다.
이 부분은 무조건 나옵니다.

Exception Handling

- 예외 발생 (raise)

```
try:
    x = int(input("3의 배수 입력: "))

    if x % 3 != 0:
        raise ValueError('3의 배수 아님')

except ValueError as e:
    print("ValueError:", e)
```

3의 배수 입력: 7
ValueError: 3의 배수 아님

Summary

• Exception handling

- try~except~else~finally

```
try :  
    실행할 문장들  
except 예외_종류 1 :  
    오류일 때 실행할 문장들  
except 예외_종류 2 :  
    오류일 때 실행할 문장들
```

종류	설명
ImportError	import 문에서 오류가 발생할 때
IndexError	리스트 등 첨자의 범위를 벗어날 때
KeyError	딕셔너리에서 키가 없을 때
KeyboardInterrupt	프로그램 실행 중 Ctrl + C 를 누를 때
NameError	변수명이 없는 것에 접근할 때
RecursionError	재귀 호출의 횟수가 시스템에서 설정한 것보다 넘칠 때(1000번)
RuntimeError	실행이 발생할 때
SyntaxError	exec()나 eval()에서 문법상 오류가 발생할 때
TypeError	변수형의 오류가 발생할 때. 예 '문자열-문자열' 연산
ValueError	함수의 매개변수에 잘못된 값을 넘길 때. 예 int('파이썬')
ZeroDivisionError	0으로 나눌 때
IOError	파일 처리 등 오류일 때

Assignment 13

• 예외 처리 함수 구현

- 사용자에게 정수 입력을 받아 리스트 원소 출력하기

- 예외 오류 발생 상황

- 리스트의 길이보다 큰 입력 정수

- 정수가 아닌 입력

- Ctrl + C

```
def func(a):  
    index = int(input("정수 입력: "))  
    print(a[index])  
  
func([1, 2, 3, 4, 5])
```

Assignment 13

• 동작 예시

- 범위 내의 정수를 입력받는 경우
 - 리스트 원소를 출력하고 함수 종료 (return)
- 잘못된 입력을 받는 경우
 - 각 예외에 대한 오류 메시지 출력 후 **함수 재동작**
 - 예외가 발생하지 않을 때 까지 무한루프
 - **while, for 사용 금지**
- Ctrl + C
 - 종료 문구를 출력하고 함수 호출 종료

```
정수 입력 : 10
리스트의 길이를 초과함
정수 입력 : □
지원되지 않는 데이터형
정수 입력 : JJ
지원되지 않는 데이터형
정수 입력 : 2
3
정수 입력 :
프로그램 종료
```