



# 05장 파이썬 날개달기

---

클래스, 모듈, 예외 처리



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

클래스는 도대체 왜 필요한가?

```
result = 0

def adder(num):
    global result
    result += num
    return result
```



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## 두 개의 계산기

```
result1 = 0
result2 = 0

def adder1(num):
    global result1
    result1 += num
    return result1

def adder2(num):
    global result2
    result2 += num
    return result2
```



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## 클래스를 이용한 계산기

```
class Calculator:
    def __init__(self):
        self.result = 0

    def adder(self, num):
        self.result += num
        return self.result

cal1 = Calculator()
cal2 = Calculator()
```



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## 클래스의 구조

```
class 클래스이름[(상속 클래스명)]:  
    <클래스 변수 1>  
    <클래스 변수 2>  
    ...  
    def 클래스함수1(self[, 인수1, 인수2,,,]):  
        <수행할 문장 1>  
        <수행할 문장 2>  
        ...  
    def 클래스함수2(self[, 인수1, 인수2,,,]):  
        <수행할 문장1>  
        <수행할 문장2>  
        ...
```



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## 사칙연산 클래스 1

```
>>> class FourCal:  
...     pass  
...  
>>>
```

```
>>> a = FourCal()  
>>> type(a)  
<class '__main__.FourCal'>
```



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## 사칙연산 클래스 2

```
>>> a.setdata(4, 2)
```

```
>>> class FourCal:
...     def setdata(self, first, second):
...         self.first = first
...         self.second = second
...
>>>
```



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## 사칙연산 클래스 3

```
>>> a = FourCal()  
>>> a.setdata(4, 2)  
>>> print(a.sum())  
6
```





# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## 사칙연산 클래스 4

```
>>> class FourCal:
...     def setdata(self, first, second):
...         self.first = first
...         self.second = second
...     def sum(self):
...         result = self.first + self.second
...         return result
...
>>>
```



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## \_\_init\_\_ 메서드로 초기값을 설정 1

```
>>> class HousePark:
...     lastname = "박"
...     def __init__(self, name):
...         self.fullname = self.lastname + name
...     def travel(self, where):
...         print("%s, %s여행을 가다." % (self.fullname,
where))
...
>>>
```



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## \_\_init\_\_ 메서드로 초기값을 설정 2

```
>>> pey = HousePark()  
TypeError: __init__() takes exactly 2 arguments (1 given)
```

```
>>> pey = HousePark("응용")
```



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## 클래스의 상속

```
>>> class HouseKim(HousePark):  
...     lastname = "김"  
...  
>>>
```

```
>>> juliet = HouseKim("줄리엣")  
>>> juliet.travel("독도")  
김줄리엣, 독도여행을 가다.
```



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## 메서드 오버라이딩

```
>>> class HouseKim(HousePark):  
...     lastname = "김"  
...     def travel(self, where, day):  
...         print("%s, %s여행 %d일 가네." % (self.fullname,  
where, day))  
...  
>>>
```

```
>>> juliet = HouseKim("줄리엣")  
>>> juliet.travel("독도", 3)  
김줄리엣, 독도여행 3일 가네.
```



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## 연산자 오버로딩 1

```
>>> pey = HousePark("응용")
>>> juliet = HouseKim("줄리엣")
>>> pey + juliet
박응용, 김줄리엣 결혼했네
>>>
```



# 05-1 파이썬 프로그래밍의 핵심, 클래스

---

## 연산자 오버로딩 2

```
class HousePark:
    lastname = "박"
    def __init__(self, name):
        self.fullname = self.lastname + name
    def travel(self, where):
        print("%s, %s여행을 가다." % (self.fullname, where))
    def __add__(self, other):
        print("%s, %s 결혼했네" % (self.fullname,
other.fullname))
```



## 05-2 모듈

---

### 모듈 만들고 불러 보기

```
# C:\Python\Mymodules\mod1.py
def sum(a, b):
    return a + b
```

```
cd C:\Python\Mymodules
```

```
>>> import mod1
>>> print(mod1.sum(3,4))
7
```





## 05-2 모듈

---

from 모듈이름 import 모듈함수

```
>>> from mod1 import sum  
>>> sum(3, 4)  
7
```



## 05-2 모듈

---

sys.path.append

```
>>> import sys
>>> sys.path
['', 'C:\\Windows\\SYSTEM32\\python35.zip',
'c:\\Python35\\DLLs',
'c:\\Python35\\lib', 'c:\\Python35',
'c:\\Python35\\lib\\site-packages']
```

```
sys.path.append("C:/Python/Mymodules")
```



## 05-2 모듈

---

### PYTHONPATH 환경 변수 사용하기

```
C:\Users\home>set PYTHONPATH=C:\Python\Mymodules
C:\Users\home>python
Python 3.5.1 (v3.5.1:37a07cee5969, Dec 6 2015, 01:54:25) [MSC
v.1900 64 bit (AM...
Type "help", "copyright", "credits" or "license" for more
information.
>>> import mod1
```



## 05-3 패키지

---

### 가상의 game 패키지 예

```
game/  
  __init__.py  
  sound/  
    __init__.py  
    echo.py  
    wav.py  
  graphic/  
    __init__.py  
    screen.py  
    render.py  
  play/  
    __init__.py  
    run.py  
    test.py
```



## 05-3 패키지

---

### 테스트를 위해 패키지 만들기

```
C:/Python/game/__init__.py  
C:/Python/game/sound/__init__.py  
C:/Python/game/sound/echo.py  
C:/Python/game/graphic/__init__.py  
C:/Python/game/graphic/render.py
```

```
# echo.py  
def echo_test():  
    print ("echo")
```

```
# render.py  
def render_test():  
    print ("render")
```



## 05-3 패키지

---

### 패키지 안의 함수 실행하기 1

```
C:\> set PYTHONPATH=C:/Python
```

```
C:\> python
```

```
Python 3.5.1 (v3.5.1:37a07cee5969, Dec 6 2015, 01:54:25) [MSC  
v.1900 64 bit (AM...
```

```
Type "help", "copyright", "credits" or "license" for more  
information.
```

```
>>>
```



## 05-3 패키지

---

### 패키지 안의 함수 실행하기 2

```
>>> import game.sound.echo
>>> game.sound.echo.echo_test()
echo
```

```
>>> from game.sound import echo
>>> echo.echo_test()
echo
```

```
>>> from game.sound.echo import echo_test
>>> echo_test()
echo
```



## 05-3 패키지

---

`__all__`

```
>>> from game.sound import *
>>> echo.echo_test()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'echo' is not defined
```

```
# C:/Python/game/sound/__init__.py
__all__ = ['echo']
```





## 05-3 패키지

---

### relative 패키지

```
# render.py
from ..sound.echo import echo_test

def render_test():
    print ("render")
    echo_test()
```



## 05-4 예외처리

---

오류는 어떤 때 발생하는가?

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



## 05-4 예외처리

---

try, except문

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

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



## 05-4 예외처리

---

try .. else

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



## 05-4 예외처리

---

try .. finally

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



## 05-4 예외처리

---

### 오류 회피하기

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



## 05-4 예외처리

---

### 오류 일부러 발생시키기

```
class Bird:
    def fly(self):
        raise NotImplementedError
```

```
class Eagle(Bird):
    def fly(self):
        print("very fast")
```

```
eagle = Eagle()
eagle.fly()
```



## 05-5 내장함수

---

abs

```
>>> abs(3)
3
>>> abs(-3)
3
>>> abs(-1.2)
1.2
```





## 05-5 내장함수

---

all

```
>>> all([1, 2, 3])  
True  
>>> all([1, 2, 3, 0])  
False
```



## 05-5 내장함수

---

any

```
>>> any([1, 2, 3, 0])
```

```
True
```

```
>>> any([0, ""])
```

```
False
```



## 05-5 내장함수

---

chr

```
>>> chr(97)
```

```
'a'
```

```
>>> chr(48)
```

```
'0'
```



## 05-5 내장함수

---

dir

```
>>> dir([1, 2, 3])  
['append', 'count', 'extend', 'index', 'insert', 'pop',...]  
>>> dir({'1':'a'})  
['clear', 'copy', 'get', 'has_key', 'items', 'keys',...]
```



## 05-5 내장함수

---

divmod

```
>>> divmod(7, 3)
(2, 1)
>>> divmod(1.3, 0.2)
(6.0, 0.099999999999999978)
```



## 05-5 내장함수

---

### enumerate

```
>>> for i, name in enumerate(['body', 'foo', 'bar']):  
...     print(i, name)  
...  
0 body  
1 foo  
2 bar
```



## 05-5 내장함수

---

eval

```
>>> eval('1+2')
3
>>> eval("'hi' + 'a'")
'hia'
>>> eval('divmod(4, 3)')
(1, 1)
```



## 05-5 내장함수

---

filter

```
def positive(x):  
    return x > 0
```

```
print(list(filter(positive, [1, -3, 2, 0, -5, 6])))
```

```
>>> print(list(filter(lambda x: x > 0, [1, -3, 2, 0, -5,  
6])))
```





## 05-5 내장함수

---

hex

```
>>> hex(234)
```

```
'0xea'
```

```
>>> hex(3)
```

```
'0x3'
```



## 05-5 내장함수

---

id

```
>>> a = 3
>>> id(3)
135072304
>>> id(a)
135072304
>>> b = a
>>> id(b)
135072304
```



## 05-5 내장함수

---

input

```
>>> a = input()
hi
>>> a
'hi'
>>> b = input("Enter: ")
Enter: hi
```

```
>>> b
'hi'
```



## 05-5 내장함수

---

int

```
>>> int('3')
```

```
3
```

```
>>> int(3.4)
```

```
3
```

```
>>> int('11', 2)
```

```
3
```

```
>>> int('1A', 16)
```

```
26
```



## 05-5 내장함수

---

### isinstance

```
>>> class Person: pass
...
>>> a = Person()
>>> isinstance(a, Person)
True
```

```
>>> b = 3
>>> isinstance(b, Person)
False
```



## 05-5 내장함수

---

lambda

```
>>> sum = lambda a, b: a+b  
>>> sum(3,4)  
7
```

```
>>> myList = [lambda a,b:a+b, lambda a,b:a*b]  
>>> myList  
[at 0x811eb2c>, at 0x811eb64>]
```



## 05-5 내장함수

---

len

```
>>> len("python")
```

```
6
```

```
>>> len([1,2,3])
```

```
3
```

```
>>> len((1, 'a'))
```

```
2
```



## 05-5 내장함수

---

list

```
>>> list("python")  
['p', 'y', 't', 'h', 'o', 'n']  
>>> list((1,2,3))  
[1, 2, 3]
```

```
>>> a = [1, 2, 3]  
>>> b = list(a)  
>>> b  
[1, 2, 3]
```





## 05-5 내장함수

---

map

```
>>> def two_times(x): return x*2
```

```
>>> list(map(two_times, [1, 2, 3, 4]))  
[2, 4, 6, 8]
```

```
>>> list(map(lambda a: a*2, [1, 2, 3, 4]))  
[2, 4, 6, 8]
```



## 05-5 내장함수

---

max

```
>>> max([1, 2, 3])
3
>>> max("python")
'y'
```

min

```
>>> min([1, 2, 3])
1
>>> min("python")
'h'
```



## 05-5 내장함수

---

oct

```
>>> oct(34)
'0o42'
>>> oct(12345)
'0o30071'
```



## 05-5 내장함수

---

open

mode	설명
w	쓰기 모드로 파일 열기
r	읽기 모드로 파일 열기
a	추가 모드로 파일 열기
b	바이너리 모드로 파일 열기

```
>>> f = open("binary_file", "rb")
```



## 05-5 내장함수

---

ord

```
>>> ord('a')
```

```
97
```

```
>>> ord('0')
```

```
48
```



## 05-5 내장함수

---

pow

```
>>> pow(2, 4)
```

```
16
```

```
>>> pow(3, 3)
```

```
27
```



## 05-5 내장함수

---

range

```
>>> list(range(5))  
[0, 1, 2, 3, 4]
```

```
>>> list(range(5, 10))  
[5, 6, 7, 8, 9]
```

```
>>> list(range(1, 10, 2))  
[1, 3, 5, 7, 9]  
>>> list(range(0, -10, -1))  
[0, -1, -2, -3, -4, -5, -6, -7, -8, -9]
```



## 05-5 내장함수

---

sorted

```
>>> sorted([3, 1, 2])  
[1, 2, 3]  
>>> sorted(['a', 'c', 'b'])  
['a', 'b', 'c']  
>>> sorted("zero")  
['e', 'o', 'r', 'z']  
>>> sorted((3, 2, 1))  
[1, 2, 3]
```





## 05-5 내장함수

---

str

```
>>> str(3)
'3'
>>> str('hi')
'hi'
>>> str('hi'.upper())
'HI'
```



## 05-5 내장함수

---

tuple

```
>>> tuple("abc")  
('a', 'b', 'c')  
>>> tuple([1, 2, 3])  
(1, 2, 3)  
>>> tuple((1, 2, 3))  
(1, 2, 3)
```



## 05-5 내장함수

---

type

```
>>> type("abc")
<class 'str'>
>>> type([ ])
<class 'list'>
>>> type(open("test", 'w'))
<class '_io.TextIOWrapper'>
```



## 05-5 내장함수

---

zip

```
>>> list(zip([1, 2, 3], [4, 5, 6]))
[(1, 4), (2, 5), (3, 6)]
>>> list(zip([1, 2, 3], [4, 5, 6], [7, 8, 9]))
[(1, 4, 7), (2, 5, 8), (3, 6, 9)]
>>> list(zip("abc", "def"))
[('a', 'd'), ('b', 'e'), ('c', 'f')]
```



## 05-6 외장함수

---

sys.argv

```
# argv_test.py  
import sys  
print(sys.argv)
```

```
C:/Python/Mymodules>python argv_test.py you need python  
['argv_test.py', 'you', 'need', 'python']
```



## 05-6 외장함수

---

### pickle

```
>>> import pickle
>>> f = open("test.txt", 'wb')
>>> data = {1: 'python', 2: 'you need'}
>>> pickle.dump(data, f)
>>> f.close()
```

```
>>> import pickle
>>> f = open("test.txt", 'rb')
>>> data = pickle.load(f)
>>> print(data)
{2: 'you need', 1: 'python'}
```



## 05-6 외장함수

---

OS

```
>>> import os
>>> os.environ
environ({'PROGRAMFILES': 'C:\\Program Files', 'APPDATA': ...
생략 ...})
>>> os.chdir("C:\\WINDOWS")
>>> os.getcwd()
'C:\\WINDOWS'
>>> os.system("dir")
>>> f = os.popen("dir")
>>> print(f.read())
```



## 05-6 외장함수

---

shutil

```
>>> import shutil  
>>> shutil.copy("src.txt", "dst.txt")
```





## 05-6 외장함수

---

glob

```
>>> import glob
>>> glob.glob("C:/Python/q*")
['C:\\Python\\quiz.py', 'C:\\Python\\quiz.py.bak']
>>>
```



## 05-6 외장함수

---

### tempfile

```
>>> import tempfile  
>>> filename = tempfile.mktemp()  
>>> filename  
'C:\WINDOWS\TEMP\~-275151-0'
```

```
>>> import tempfile  
>>> f = tempfile.TemporaryFile()  
>>> f.close()
```



## 05-6 외장함수

---

### time 1

```
>>> import time
>>> time.time()
988458015.73417199
```

```
>>> time.localtime(time.time())
time.struct_time(tm_year=2013, tm_mon=5, tm_mday=21,
tm_hour=16,
tm_min=48, tm_sec=42, tm_wday=1, tm_yday=141, tm_isdst=0)
```



## 05-6 외장함수

---

### time 2

```
>>> time.asctime(time.localtime(time.time()))  
'Sat Apr 28 20:50:20 2001'
```

```
>>> time.ctime()  
'Sat Apr 28 20:56:31 2001'
```

```
>>> import time  
>>> time.strftime('%x', time.localtime(time.time()))  
'05/01/01'  
>>> time.strftime('%c', time.localtime(time.time()))  
'05/01/01 17:22:21'
```



## 05-6 외장함수

---

calendar

```
>>> calendar.weekday(2015, 12, 31)  
3
```

```
>>> calendar.monthrange(2015,12)  
(1, 31)
```



## 05-6 외장함수

---

random

```
>>> import random  
>>> random.random()  
0.53840103305098674
```

```
>>> random.randint(1, 10)  
6
```

```
>>> data = [1, 2, 3, 4, 5]  
>>> random.shuffle(data)  
>>> data  
[5, 1, 3, 4, 2]
```



## 05-6 외장함수

---

webbrowser

```
>>> import webbrowser  
>>> webbrowser.open("http://google.com")
```

```
>>> webbrowser.open_new("http://google.com")
```



## 05-6 외장함수

---

### threading

```
import threading
import time

def say(msg):
    while True:
        time.sleep(1)
        print(msg)

for msg in ['you', 'need', 'python']:
    t = threading.Thread(target=say, args=(msg,))
    t.daemon = True
    t.start()

for i in range(100):
    time.sleep(0.1)
    print(i)
```