

Week1-4

1. 학습정리

1. Python Object Oriented Programming

1. __ 의미

- ## 1. 특수한 예약 함수

2. __str__

- ## 1. print 할 때 실행됨

3. `__add__` , `__eq__`

2. oop 특징

- ## 1. 상속 : inheritance

- ## 2. 다형성 : polymorphism

- ### 3. visibility (캡슐화, 정보은닉)

1. 변수 앞에 언더바 두개 쓰면 외부에서 접근 불가

- ## 2. ex : __변수

3. decorate

- ## 1. 일급객체 : First-class object

- ## 1. 변수나 데이터 구조에 할당이 가능한 객체

- ## 2. 파라미터나 리턴값으로 사용 가능

- ### 3. 파이썬의 함수는 일급 함수

2. 내재함수 : Inner function

- ## 1. 함수안에 함수

- ## 2. 클로저 : closure

- ## 1. 내재함수를 리턴값으로 반환

3. decorator function

- ## 1. @함수명

2. 아래에 정의된 함수를 호출하면 @함수의 아규먼트로 전달됨

decorator function

decorate



- 복잡한 클로저 함수를 간단하게!

```
def star(func):
    def inner(*args, **kwargs):
        print("*" * 30)
        func(*args, **kwargs)
        print("*" * 30)
    return inner
```

```
def percent(func):
    def inner(*args, **kwargs):
        print("%" * 30)
        func(*args, **kwargs)
        print("%" * 30)
    return inner
```

```
@star
@percent
def printer(msg):
    print(msg)
printer("Hello")
```

[illegible]

decorator function

decorate

```
def generate_power(exponent):
    def wrapper(f):
        def inner(*args):
            result = f(*args)
            return exponent**result
        return inner
    return wrapper

@generate_power(2)
def raise_two(n):
    return n**2
```

```
print(raise_two(7))
```

562949953421312

x

2. Module and Project

1. 모듈 사용법

1. 같은 폴더내에 있는 py 파일을 import

2. namespace

1. 모듈을 호출할 때 범위를 정함

namespace example

Alias 설정하기 - 모듈명을 별칭으로 써서

변경

```
import fah_converter as fah    fah_converter를 fah라는 이름으로
print(fah.covert_c_to_f(41.6))    그 안에 covert_c_to_f 함수를 쓴다
```

모듈에서 특정 함수 또는 클래스만 호출하기

```
from fah_converter import (covert_c_to_f
print(covert_c_to_f(41.6))    covert_c_to_f 함수만 호출함
```

모듈에서 모든 함수 또는 클래스를 호출하기

```
from fah_converter import *
print(covert_c_to_f(41.6))    전체 호출
```

3. 패키지

1. 다양한 모듈들의 합, 폴더로 연결됨
2. __init__, __main__ 등 키워드 파일명이 사용됨
3. 폴더 안에 있는 것을 from을 사용해서 불러올 수 있음
4. __init__.py 파일
 1. 현재 폴더가 패키지임을 알리는 초기화 스크립트
 2. 없을 경우 패키지로 간주하지 않음 (지금은 아님)

```
echo.py  __init__.py ×  
__init__.py > {} stage  
1  __all__ = ["image", "sound", "stage"]  
2  
3  from . import image  
4  from . import sound  
5  from . import stage  
6
```

- 3. 이런 식으로 안에 있는 것들을 다 선언해 줌
- 4. __main__
 - 1. 이 파일이있으면 폴더 자체를 실행 가능
- 5. 패키지 내에 모듈을 부를 땐 상대참조로 호출함
- 2. 피어세션
 - 1. 오늘 강의 리뷰
 - 2. 조교님 소개
 - 3. 자기소개
 - 4. 인터뷰 연습 일정 잡기
 - 5. 조교님에 관한 질답
- 3. 과제
 - 1. 당일 과제 완료
 - 2. join 을 써야하는데 공백 문자 때문에 쓰기가 애매해져서 사용하지 못했음