

 kim hanwoong show2project

56558b2 on 23 Dec 2018

0 contributors

64 lines (52 sloc) | 2.73 KB

파이썬 클래스 응용

이번시간에는 파이썬을 이용해서 경로와 관련된 클래스 하나를 만들어보겠습니다. 일반적으로 아래 상황일 때 사용하면 좋습니다.

- 항상 회사에서 생성하는 파일규칙을 통일하고 싶을 때(네이밍 컨벤션 통일)
- 회사 규칙이 맞는지 체크하고 싶을 때

경로문제는 항상 쉬우면서 어려운 문제입니다. 회사 초기부터 프로젝트에 사용될 경로 규칙을 만들고 경로를 생성하는 습관을 들이는 것이 좋습니다. 콘텐츠를 만들면서 인원, 팀의 규모가 커지면서 다양한 규칙, 예외상황들이 발생합니다. 경로 문제는 점점 복잡해지며 체계가 필요하기 때 문에 똑똑한 사람들은 회사에서 사용가능한 경로 라이브러리를 만들어서 개발자들에게 공유합니다.

오늘은 파이썬 클래스를 다루면서 위 문제의 기초 모델을 만들어보겠습니다.

클래스는 데이터 + 메소드(클래스 내부에 정의된 함수)의 집합입니다.

pathconvention.py

```
class PathConvention:
    root = "/project"
    project = ""
    name = ""
    task = ""
    def path(self):
        return "/".join([self.root,self.project,self.name,self.task])
```

main.py

```
from pathconvention import *

p = PathConvention()
p.project = "circle"
p.name = "F00_0010"
p.task = "comp"
print p.path()
```

스크립트를 실행해 봅시다.

```
$ python main.py
```

간단한 패스 컨벤션 규칙이 만들어졌습니다. 회사의 경로를 운용하는 것은 간단합니다. 회사에서 생성되는 모든 경로에 대해서 생성, 수정, 삭제 가 해당 파이썬 클래스로만 진행되면 문제 없습니다. 하지만 자동으로 생성되는 폴더들이 있기 때문에 회사에서 사용하는 제작 소프트웨어의 이 해가 추가적으로 필요하긴 합니다.

실습

아래 항목들은 회사에서 일반적으로 일어나는 상황입니다. 아래 항목을 만들어보기 전에 먼저 토론을 해봅시다.

- 샷일 때와 에셋일 때의 데이터를 경로로 구분하기 : type 값을 추가.

- 플레이트 처리 : plate, reference, source 를 구분해보기.
- 상태처리 : 개발버전, 배포버전의 구분
- 프록시 경로 처리
- 버전처리
- 버전 자릿수 처리
- [samba서버](#) IP설정 추가
- 리눅스 경로를 윈도우즈 경로로 바꾸어주는 메소드 추가
- [UNC경로](#)를 네트워크 드라이브로 바꾸어주는 메소드 추가
- 경로를 생성하는 메소드 추가
- 경로가 존재하는 체크하는 메소드 추가