

 kim hanwoong 내용 추가하기.

7f265dc 22 days ago

1 contributor

59 lines (49 sloc) | 2.16 KB

Python & ImageMagick

파이썬을 활용한 첫번째 실습입니다.

git 명령어를 이용하여 ~/examples 파일을 다운로드 합니다.

```
$ cd ~
$ git clone http://github.com/cgiseminar/examples
```

프로그래밍이 처음인 분을 위해서 코드가 쉽게 읽히도록 os.system 함수를 이용해서 코드를 작성해 보겠습니다. os.system 함수는 간단합니다. 터미널 창에서 입력한 명령어 그대로 입력하면 명령어가 그대로 실행됩니다.

```
import os
p = os.path.expanduser("~/examples/F00_0010")
for f in os.listdir(p):
    cmd = "convert %s/%s %s/%s" % (p, f, p, f.replace(".jpg", ".png"))
    print cmd
    os.system(cmd)
```

실습

- 각 코드마다 일어날 수 있는 버그를 이야기 해봅시다.
- 각 폴더마다 대표하는 이미지 썸네일을 한장만 생성해 봅시다.
- 썸네일을 생성할 때 중앙 프레임에 해당하는 이미지 썸네일을 생성하도록 코드를 리팩토링 해보세요.
- 몇 퍼센트 진행중인지 출력해봅시다.
- 폴더에 이상한 파일이 들어있어도 작동되는지 체크해봅시다.
- 인터넷을 검색하여 os.system 함수 대신 subprocess 모듈을 사용해보세요.
- 실습에 생성된 .jpg 시퀀스를 일괄로 제거하는 방법.

```
$ find . -name "*.jpg" -type f -delete
```

제안

기존에는 리눅스 명령어 형태에 맞추어 이해를 돕기 위해서 os.system 함수를 사용했습니다. 하지만 앞으로 프로그래밍을 할 때는 os.system 함수 보다는 subprocess 를 사용하는 것을 추천합니다. 명령어를 실행할 때 프로세스를 생성해서 실행할 수 있으며, 환경변수 및 STDIN, STDOUT, STDERR 처리를 할 수 있습니다.

```
#!/usr/bin/env python
#coding:utf8
import subprocess

p = subprocess.Popen(["echo", "hello world"], stdout=subprocess.PIPE, stderr=subprocess.PIPE)
print p.communicate()
```

응용 : 환경변수와 같이 사용하기

```
import subprocess
import os
env = os.environ.copy()
env["PATH"] = "/usr/sbin:/sbin:" + env["PATH"]
subprocess.Popen(["echo","hello world"], env=env)
print p.communicate()
```