

파이썬(Python)

– Basic Library –

박성호 (neowizard2018@gmail.com)

zip()

glob.glob() / os.listdir()

os.getcwd(), os.mkdir(), os.chdir(), os.path.exists(), os.path.join()

zipfile – 파일 압축 및 풀기

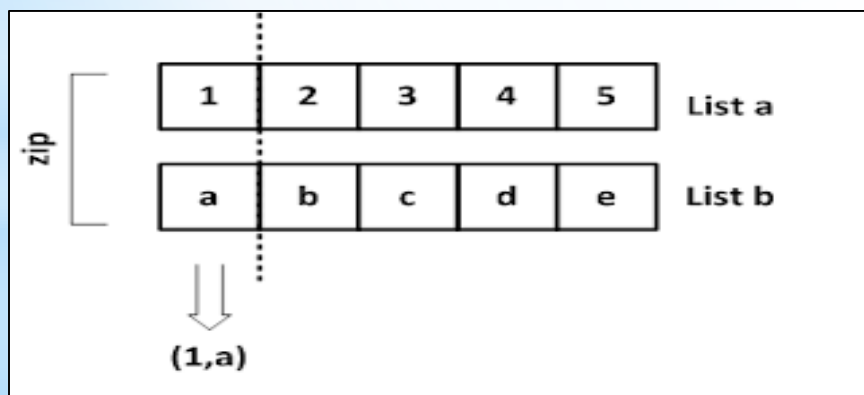
shutil.copy(), shutil.move(), shutil.rmtree(), shutil.copytree()

cv2.imread(), cv2.cvtColor(), cv2.resize(), cv2.imshow()

zip()

파이썬에서 zip()은 내장함수로 같은 길이의 리스트를 같은 인덱스끼리 잘라서 리스트로 반환을 해주는 역할을 한다. 리턴하는 객체는 zip object이므로, set, list 같은 sequence 객체로 변환해야 요소들 값 확인 가능함

for 문에서 사용할 경우에는 zip object 내부의 요소별로 리턴 (김밥을 묶어서 자르는 것과 비슷)



```
zip([1, 2, 3], [6, 7, 8])
```

```
<zip at 0x22290a48548>
```

```
list(zip([1, 2, 3], [6, 7, 8]))
```

```
[(1, 6), (2, 7), (3, 8)]
```

```
list(zip('ABC', 'XYZ'))
```

```
[('A', 'X'), ('B', 'Y'), ('C', 'Z')]
```

```
a = [1, 2, 3, 4, 5]
```

```
b = ['a', 'b', 'c', 'd', 'e']
```

```
c = list(zip(a, b))
```

```
print(c)
```

```
for x, y in zip(a,b):
```

```
    print('x = ', x, ', y = ', y)
```

```
[(1, 'a'), (2, 'b'), (3, 'c'), (4, 'd'), (5, 'e')]
```

```
x = 1 , y = a
```

```
x = 2 , y = b
```

```
x = 3 , y = c
```

```
x = 4 , y = d
```

```
x = 5 , y = e
```

glob.glob() / os.listdir() -1

- glob.glob(..), os.listdir(..) 라이브러리는 파일이나 디렉토리 목록을 추출할 때 유용한 라이브러리

다음의 출력값은 ?

```
import os
import glob
```

```
a = glob.glob('train/*')
```

```
b = glob.glob('*')
```

```
c = glob.glob '*.ipynb')
```

```
d = os.listdir('train/')
```








```
e = os.listdir('.')
```

```
f = os.listdir('*')
```







```
g = os.listdir '*.ipynb')
```

```
h = os.listdir('train/*')
```

current directory

	test_dir1
	test_dir2
	test_dir3
	train
	Python_if_for_DI
	Python_if_for_Example.ipynb
	zip_example.ipynb

train directory

이름	유형
 Agent.py	PY 파일
 cat.6	JPG 파일
 cat.9	JPG 파일
 dog.6316	JPG 파일
 dog.6331	JPG 파일
 hello	텍스트 문서

glob.glob() / os.listdir() – 2

```
import os
import glob
```

```
a = glob.glob('train/*')
print(a)
```

```
['train\\Agent.py', 'train\\cat.6.jpg', 'train\\cat.9.jpg', 'train\\dog.6316.jpg', 'train\\dog.6331.jpg', 'train\\hello.txt']
```

```
b = glob.glob('*')
print(b)
```

```
['Python_if_for_DL.pptx', 'Python_if_for_Example.ipynb', 'test_dir1', 'test_dir2', 'test_dir3', 'train', 'Untitled.ipynb', 'zip_example.ipynb', '~$Python_if_for_DL.pptx']
```

```
c = glob.glob('*.ipynb')
print(c)
```

```
['Python_if_for_Example.ipynb', 'Untitled.ipynb', 'zip_example.ipynb']
```

```
d = os.listdir('train/')
print(d)
```

```
['Agent.py', 'cat.6.jpg', 'cat.9.jpg', 'dog.6316.jpg', 'dog.6331.jpg', 'hello.txt']
```

```
e = os.listdir('.')
print(e)
```

```
['.ipynb_checkpoints', 'Python_if_for_DL.pptx', 'Python_if_for_Example.ipynb', 'test_dir1', 'test_dir2', 'test_dir3', 'train', 'Untitled.ipynb', 'zip_example.ipynb', '~$Python_if_for_DL.pptx']
```

glob.glob() / os.listdir() - 3

```
f = os.listdir('*')  
print(f)
```

OSError Traceback (most recent call last)

<ipython-input-9-79297aead07f> in <module>

```
----> 1 f = os.listdir('*')  
      2 print(f)
```

OSError: [WinError 123] 파일 이름, 디렉터리 이름 또는 볼륨 레이블 구문이 잘못되었습니다: '*'

```
g = os.listdir('*.*ipynb')  
print(g)
```

OSError Traceback (most recent call last)

<ipython-input-10-f0c337d76a6b> in <module>

```
----> 1 g = os.listdir('*.*ipynb')  
      2 print(g)
```

OSError: [WinError 123] 파일 이름, 디렉터리 이름 또는 볼륨 레이블 구문이 잘못되었습니다: '*.*ipynb'

```
h = os.listdir('train/*')  
print(h)
```

OSError Traceback (most recent call last)

<ipython-input-11-261c1df78ef5> in <module>

```
----> 1 h = os.listdir('train/*')  
      2 print(h)
```

OSError: [WinError 123] 파일 이름, 디렉터리 이름 또는 볼륨 레이블 구문이 잘못되었습니다: 'train/*'

os.getcwd(), os.mkdir(), os.chdir(), os.path.exists()

```
import os

# working directory
working_dir = 'dataset/GTSRB'

# Google Drive 에서 Colab Default Directory
colab_default_dir = '/content/gdrive/My Drive/Colab Notebooks/'

original_dir = os.getcwd()  # save original_dir dir

try:

    os.chdir(colab_default_dir)

    if not os.path.exists(working_dir):
        os.mkdir(working_dir)

    os.chdir(working_dir)  # change working dir

except Exception as err:

    # 원래의 dir 로 복귀
    os.chdir(original_dir)
    print(str(err))

print('Current dir = ', os.getcwd())
```

Current dir = /content/gdrive/My Drive/Colab Notebooks/dataset/GTSRB

zipfile – 파일 압축 및 풀기

```
import zipfile

# 압축파일안에 있는 모든 내용을 현재 디렉토리에 푼다
# close 하기 싫으면 with 구문 쓰면 된다
# with 예제는 2번째 참고사이트에 있다

zip_file = zipfile.ZipFile('./zipfile_test.zip')

zip_file.extractall()

zip_file.close()

# 압축파일안에 있는 모든 내용을 내가 원하는 디렉토리에 푼다

zip_file = zipfile.ZipFile('./zipfile_test.zip')

# my_zip_test dir 없으면 생성해서 푼다

zip_file.extractall('./my_zip_test')

zip_file.close()
```

shutil.copy(), shutil.move(), shutil.rmtree(), shutil.copytree()

cv2.imread(), cv2.cvtColor(), cv2.resize(), cv2.imshow()

소스: imread_cvtColor_resize_example.ipynb (참조사이트: <https://rueki.tistory.com/173>)

```
import numpy as np
import cv2
import matplotlib.pyplot as plt
%matplotlib inline

img = cv2.imread('데이터 경로')    # 리턴은 numpy
print(img.shape)

img = cv2.imread('asdf')    #잘못된 경로 입력해도 에러가 나질 않는다.
```

opencv에서는 cv2.imread() 를 이용해서 쉽게 읽을 수가 있으며, numpy 형태로 값이 들어오게 된다. 일반적으로 시각화전에 먼저 shape을 알아보는 습관 필요함

img = cv2.imread(..) 함수는 기본적으로 BGR 순서의 color 이미지를 읽어온다.그래서 cv2.imshow() 출력하면 제대로 나오지만, plt.imshow() 함수는 RGB 순서로 출력하기 때문에 cv2.imread() 로 읽은 이미지가 이상 하게 나온다. 따라서 cv2.imread() 이미지를 cv2.cvtColor() 로 바꾸어서 plt.imshow() 출력하면 잘 나옴

또한 plt.imshow() 에서 흑백으로 출력하려면 cmap='gray' 옵션 주어 실행하면 된다

cv2.imread(), cv2.cvtColor(), cv2.resize(), cv2.imshow()

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

# 참고 사이트: https://rueki.tistory.com/173

img_name1 = './ppoppy1.jpg'
img_name2 = './ppoppy2.jpg'

# 칼라이미지 gray 형태로 읽어온 후 정규화, 그리기
src_img = cv2.imread(img_name1, cv2.IMREAD_GRAYSCALE)
print(type(src_img), src_img.shape)
src_img = cv2.resize(src_img, dsize=(224,224))

src_img = src_img / 255.0 # 정규화

cv2.imshow("src", src_img) # opencv 는 BGR로 이미지를 보여줌
cv2.waitKey()
cv2.destroyAllWindows()

plt.figure(figsize=(8,8)) # matplotlib은 RGB로 보여줌

for idx in range(2):

    plt.subplot(1,2,idx+1)
    plt.title('ppoppy')
    plt.imshow(src_img, cmap='gray') # 정규화 해주지 않아도 출력됨

plt.show() # subplot() 에서 plt.imshow() 후에 마지막에 호출함

print(src_img.shape)
```

cv2.imread(), cv2.cvtColor(), cv2.resize(), cv2.imshow()

칼라이미지 읽어온 후 정규화, 그리기

```
src_img = cv2.imread(img_name1, cv2.IMREAD_COLOR)
print(type(src_img), src_img.shape)
src_img = cv2.resize(src_img, dsize=(224,224))
```

원본 이미지는 BGR -> RGB ??

```
dst_img = cv2.cvtColor(src_img, cv2.COLOR_BGR2RGB)
```

```
src_img = src_img / 255.0
```

```
dst_img = dst_img / 255.0
```

cs2에서는 BGR 이미지를 RGB 이미지로 자동 바꾸어 출력

```
cv2.imshow("src", src_img)
```

```
cv2.imshow("dst", dst_img)
```

```
cv2.waitKey()
```

```
cv2.destroyAllWindows()
```

```
plt.figure(figsize=(8,8))
```

```
plt.subplot(1,2,1)
```

```
plt.title('poppy')
```

```
plt.imshow(src_img)
```

```
plt.subplot(1,2,2)
```

```
plt.title('poppy')
```

```
plt.imshow(dst_img)
```

```
plt.show()
```

```
print(src_img.shape, dst_img.shape)
```

cv2.imread(), cv2.cvtColor(), cv2.resize(), cv2.imshow()

```
# 칼라이미지 읽어온 후 정규화, 그리기
src_img = cv2.imread(img_name1, cv2.IMREAD_COLOR)
print(type(src_img), src_img.shape)
src_img = cv2.resize(src_img, dsize=(224,224))

# 원본 이미지는 BGR -> GRAY ??
dst_img = cv2.cvtColor(src_img, cv2.COLOR_BGR2GRAY)

src_img = src_img / 255.0
dst_img = dst_img / 255.0

# cs2에서는 BGR 이미지를 RGB 이미지로 자동 바꾸어 출력
cv2.imshow("src", src_img)
cv2.imshow("dst", dst_img)
cv2.waitKey()
cv2.destroyAllWindows()

plt.figure(figsize=(8,8))

plt.subplot(1,2,1)
plt.title('ppoppy')
plt.imshow(src_img)

plt.subplot(1,2,2)
plt.title('ppoppy')
plt.imshow(dst_img, cmap='gray')

plt.show()

print(src_img.shape, dst_img.shape)
```