제8강 File Read and Write

학습 목차

- 파일 경로와 이름
- Current Working Directory
- 절대 경로와 상대 경로
- 경로 탐색
- 파일 읽기 쓰기

File Name과 Path

- **File Name**
  - 파일의 이름과 확장자로 구성

report.txt

- **Path**
  - 컴퓨터에서 파일의 위치를 가리킴.

```
C:\Documents\dustinlee\ScriptLanguage
Windows: "C:\\Documents\\dustinlee\\ScriptLanguage"
Linux, Mac OS: "C:/Documents/dustinlee/ScriptLanguage"
```

폴더 이름 구분 ₩ \ /

- Microsoft Windows : ₩
- Linux, Mac OS : /

```
>>> import os
>>> os.path.join('usr', 'bin', 'spam')
'usr\\bin\\spam'
```

Working Directory (folder)

```
>>> import os
>>> os.getcwd()
'C:\\Python34'
>>> os.chdir('C:/Windows/System32')
>>> os.getcwd()
'C:\\Windows\\System32'
```

절대경로와 상대경로

- Absolute path
  - Root 폴더로 시작

- Relative path
  - 특정 위치를 기준으로 한 경로
  - 일반적으로 cwd 가 기준

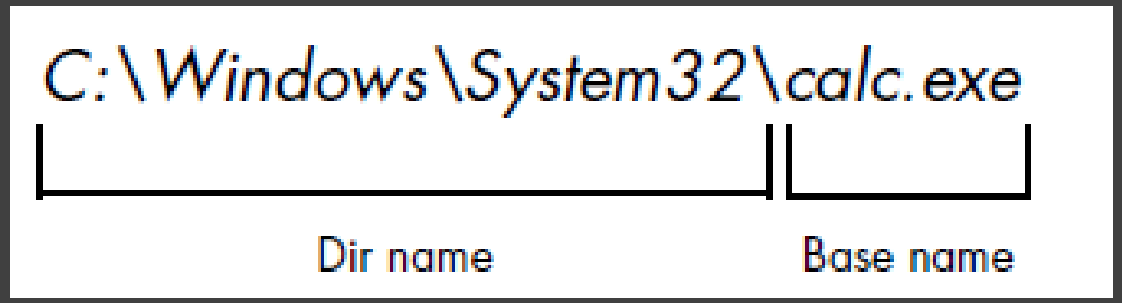
- . 와 ..
  - . 현재 폴더
  - .. 상위 폴더

```
>>> os.path.abspath('.')
'C:\\Users\\dustinlee\\PycharmProjects\\temp'
>>> os.listdir()
['.idea', 'filereadwrite.py', 'sample.py']
>>> os.mkdir('sub')
>>> os.path.abspath('./sub')
'C:\\Users\\dustinlee\\PycharmProjects\\temp\\sub'
>>> os.path.isabs('.')
False
>>> os.path.isabs('c:/users')
True
>>> os.path.relpath('c:/users', 'c:/users/dustinlee')
'..'
>>> os.path.relpath('c:/users/dustinlee', 'c:/users')
'dustinlee'
>>> os.path.relpath('c:/users/dustinlee/temp/sub', 'c:/users')
'dustinlee\\temp\\sub'
>>> os.path.relpath('c:/users', 'c:/users/dustinlee/pycharmprojects/temp/sub')
'..\\..\\..\\..'
```

새 폴더 만들기 makedirs

```
>>> import os
>>> os.makedirs('c:/test')
>>> os.makedirs('c:/Windows/test')
Traceback (most recent call last):
  File "<pyshell#2>", line 1, in <module>
    os.makedirs('c:/Windows/test')
  File "C:₩Python₩lib₩os.py", line 225, in makedirs
    mkdir(name, mode)
PermissionError: [WinError 5] 액세스가 거부되었습니다: 'c:/Windows/test'
>>> os.makedirs('c:/test2/test')
>>>
```

Dir name, base name




```
>>> abs_path = os.path.abspath('sample.py')
>>> abs_path
'C:\\Users\\dustinlee\\PycharmProjects\\temp\\sample.py'
>>> os.path.basename(abs_path)
'sample.py'
>>> os.path.dirname(abs_path)
'C:\\Users\\dustinlee\\PycharmProjects\\temp'
>>> os.path.split(abs_path)
('C:\\Users\\dustinlee\\PycharmProjects\\temp', 'sample.py')
```

File size

```
>>> os.chdir('c:/windows/system32')
>>> os.path.getsize('calc.exe')
27648
>>> total_size = 0
>>> for fn in os.listdir():
...     total_size += os.path.getsize(fn)
...
>>> total_size
21444465108
```

Path 유효성 검증

```
>>> os.path.exists('C:\\Windows')
True
>>> os.path.exists('C:\\some_made_up_folder')
False
>>> os.path.isdir('C:\\Windows\\System32')
True
>>> os.path.isfile('C:\\Windows\\System32')
False
>>> os.path.isdir('C:\\Windows\\System32\\calc.exe')
False
>>> os.path.isfile('C:\\Windows\\System32\\calc.exe')
True
```

폴더 탐색 os.walk()

```
for root, subfolders, filenames in os.walk('.'):
    print((' ROOT: ' + root + ' ').center(80, '='))
    print('Subfolder: '.ljust(15), subfolders)
    print('files: '.ljust(15), filenames)
    print('\n')
```

File open / read / seek / readline / readlines

```
>>> f = open('sample.py')
>>> f.read()
"import random\n\n\ndef calculate_avg(values):\n    sum = 0\n    for v in values:\n        sum += v\n    retu
>>> f.read()
''
>>> f.seek(0)
0
>>> f.read()
"import random\n\n\ndef calculate_avg(values):\n    sum = 0\n    for v in values:\n        sum += v\n    retu
>>> f.close()
>>> f = open('sample.py')
>>> f.read()
"import random\n\n\ndef calculate_avg(values):\n    sum = 0\n    for v in values:\n        sum += v\n    retu
>>> f.read()
''
>>> f.seek(0)
0
>>> f.readline()
'import random\n'
>>> f.readlines()
['\n', '\n', 'def calculate_avg(values):\n', '    sum = 0\n', '    for v in values:\n', '        sum += v\n',
>>> f.close()
```

File write / append

```
>>> baconFile = open('bacon.txt', 'w')
>>> baconFile.write('Hello world!\n')
13
>>> baconFile.close()
>>> baconFile = open('bacon.txt', 'a')
>>> baconFile.write('Bacon is not a vegetable.')
25
>>> baconFile.close()
>>> baconFile = open('bacon.txt')
>>> content = baconFile.read()
>>> baconFile.close()
>>> print(content)
Hello world!
Bacon is not a vegetable.
```

Shelve 모듈

- 파이썬의 모든 변수들을 파일로 기록하고(serialization), 읽을 수 있음(de-serialization)
- 딕셔너리 형태로 액세스

```
>>> import shelve
>>> import random
>>> data = [random.randint(0, 100) for i in range(100)]
>>> data
[19, 8, 59, 65, 11, 47, 2, 40, 31, 26, 31, 72, 92, 26, 51, 60, 12, 48, 28, 33, 9, 66, 48,
>>> data_file = shelve.open('mydata')
>>> data_file['data'] = data
>>> data_file.close()
```

```
>>> os.listdir()
['.idea', 'filereadwrite.py', 'mydata.bak', 'mydata.dat', 'mydata.dir', 'sample.py', 'sub']
>>> read_file = shelve.open('mydata')
>>> read_data = read_file['data']
>>> read_data
[19, 8, 59, 65, 11, 47, 2, 40, 31, 26, 31, 72, 92, 26, 51, 60, 12, 48, 28, 33, 9, 66, 48, 22, 8
>>> read_file.close()
```

Dictionary sort

```
def get_key(x):
    return x[1]

data = {'jisu':100, 'momo':20, 'rose':200, 'iu': 3}
result = sorted(data.items(), key=get_key, reverse=True)
```