17. Files 입출력

open(file 명) 을 호출하면 해당 file 을 open 하고 file 객체를 반환한다.

f = open('file_name', 'r') : file read 를 위한 file 객체를 변수 f 로 저장

f = open('file_name', 'w') : file write 를 위한 file 객체를 변수 f 로 저장

f.close() : file close

주요 methods

- f.read()
- f.write()
- f.tell()

새로운 file 생성

In [9]:

```
f = open('test.txt', 'w')
f.write('This is file write test\n')
f.write('This is 2nd line\n')
f.write('이것은 3번째 줄입니다.')
f.close()
```

이미 존재하는 file open

In [30]:

```
f = open('test.txt', 'br')     # binary mode

print(f.tell())

text = f.read()

print(text)
print(len(text))

print(f.tell())
f.close()
```

```
0
b'This is file write test\r\nThis is 2nd line\r\n\xc0\xcc\xb0\xcd\xc0\xba 3\xb9\xf8\x
c2\xb0 \xc1\xd9\xc0\xd4\xb4\xcf\xb4\xd9.'
65
65
```

In [31]:

```
f = open('test.txt', 'r')      # text mode

lines = f.readlines()

for line in lines:
    print(line, end="")

f.close()
```

```
This is file write test
This is 2nd line
이것은 3번째 줄입니다.
```

try / except

In [24]:

```
import sys
file_name = 'test2.txt'

try:
    f = open(file_name, 'r')
    text = f.read()
    f.close()
except IOError:
    sys.stderr.write('file reading error : ' + file_name)
except OSError as err:
    print("OS error: {0}".format(err))
except ValueError:
    print("Could not convert data to an integer.")
except:
    print("Unexpected error:")
```

```
file reading error : test2.txt
```

file path 지정

In [25]:

```
import os

os.listdir()
```

Out[25]:

```
['.ipynb_checkpoints',
 '00.TableOfContets.ipynb',
 '01.print and help.ipynb',
 '02.arithmetic calculation.ipynb',
 '03.data-types.ipynb',
 '04.variable and practice.ipynb',
 '05.functions-variable scope-builtin function.ipynb',
 '06.list-indexing-slicing.ipynb',
 '07.for-loop-range.ipynb',
 '08.range.ipynb',
 '09.dictionary.ipynb',
 '10.string-escape-.ipynb',
 '11.string-methods and formatting.ipynb',
 '12.if-comparison-logical operation.ipynb',
 '13.import-module.ipynb',
 '14.object-oriented-programming.ipynb',
 '15.set.ipynb',
 '16-1.euclidGCD.pptx',
 '16.while-loop.ipynb',
 '17.File system .ipynb',
 '18-1.Regular Expression.pptx',
 '18.Regular Expression.ipynb',
 '19-1_factorialImage.pptx',
 '19.Functional Programming.ipynb',
 '20.List Comprehension.ipynb',
 '21.Binary Search Algoritym.ipynb',
 '22.Big-O notation and Algorithmic Analysis with Python.ipynb',
 '23-1.이항계수DP.pptx',
 '23.Memoization and Dynamic Programming.ipynb',
 '24-1.sorting.pptx',
 '24.Sort Algorithm.ipynb',
 '25.zipfile using Python.ipynb',
 '26.GraphDataStructure.ipynb',
 '27.Various_Python_Libraries.ipynb',
 'big_o_chart.jpg',
 'bubblesort.PNG',
 'Command_shortcuts.docx',
 'compressed.zip',
 'DPCOINLIST.PNG',
 'DPCOINTree.PNG',
 'DPCONLIST1.PNG',
 'graph_diagram.png',
 'images',
 'matrix.PNG',
 'matrix_path.PNG',
 'memoization.PNG',
 'MergeSort.png',
 'module_test.py',
 'new_compressed_file.zip',
 'new_file.txt',
 'new_file2.txt',
 'new_file_1.txt',
 'new_file_2.txt',
 'quicksort.PNG',
 'selectionsort.png',
 'sierpinski.PNG',
 'test.ipnb',
 'test.txt',
 'test_text.txt',
 'unzipped_content',
 'unzipped_content_2',
 'unzipped_content_3',
 'zipped_content3.zip',
 '__pycache__',
 '사전 설문.docx']
```

In [26]:

```
os.path.abspath('test.txt')
```

Out[26]:

```
'C:\\Users\\trimu\\Desktop\\AlgorithmPython\\jupytorNotebooks\\test.txt'
```

In [27]:

```
os.path.join('C:\\Users\\trimu\\Desktop\\Python-Teaching\\jupytorNotebooks', 'test.txt')
```

Out[27]:

```
'C:\\Users\\trimu\\Desktop\\Python-Teaching\\jupytorNotebooks\\test.txt'
```

In [28]:

```
os.path.dirname(os.path.abspath('test.txt'))
```

Out[28]:

```
'C:\\Users\\trimu\\Desktop\\AlgorithmPython\\jupytorNotebooks'
```

In [29]:

```
os.path.exists('test.txt')
```

Out[29]:

```
True
```

with open 문으로 file 읽기/쓰기

with 문을 이용하면 with 블록을 벗어나는 순간 열린 파일 객체가 자동으로 close되어 편리함.

In [11]:

```
with open('test.txt', "r") as file2:
    data = file2.read()

print(data)
```

```
This is file write test
This is 2nd line
이것은 3번째 줄입니다.
```

In [12]:

```
with open('test.txt', "w") as file3:
    data = file3.write("This is 3rd line")
```

requests

requests module 을 이용하여 url 을 fetch 할 수 있다.

In [13]:

```
import requests
```

In [14]:

```
source = requests.get('http://www.google.com')
```

In [15]:

```
html = source.text
```

In [16]:

```
html[:1000]
```

Out[16]:

```
'<!doctype html><html itemscope="" itemtype="http://schema.org/WebPage" lan
g="ko"><head><meta content="text/html; charset=UTF-8" http-equiv="Content-
Type"><meta content="/images/branding/googleg/1x/googleg_standard_color_12
8dp.png" itemprop="image"><title>Google</title><script nonce="xEP6SYNqIxWZ
3xv69r5yCw==">(function(){window.google={kEI:\'wEQyXqH9NcynoATs_LD4DA
\',kEXPI:\'0,18168,1335579,5662,730,32,192,5105,206,2955,249,10,50,1001,17
5,364,1366,69,4,60,690,52,75,383,849,252,1128270,1197730,294,125,39,32907
9,1294,12383,4855,32691,15248,867,6056,13341,9287,363,3320,5505,8384,485
9,1361,4323,4967,3028,2816,1924,3118,7915,1808,1976,2044,8909,5072,225,2
054,920,873,1217,1710,1,1264,2784,2607,315,724,11306,2884,20,317,1981,253
7,1396,1381,520,399,2277,8,2796,1593,1279,390,1822,202,328,149,1103,840,5
20,1471,48,158,662,3438,260,52,1137,2,2063,606,1839,184,545,1232,520,1947,
245,502,429,44,1009,93,328,1284,16,84,417,1708,718,1425,214,608,473,1339,7
48,209,830,2197,1030,773,2072,7,1320,3488,791,469,311'
```

연습문제

다음 문장을 file 에 write 하고 다시 읽어서 print 한다.

"한국소프트웨어 산업협회 훈련과정 :
알고리즘으로 배우는 Python"