

Fastcampus Web Programming SCHOOL

Python

File I/O

File I/O

```
f = open(filename, mode)
f.close()
```

- mode(b: binary mode(default: text))
 - r(b) - 읽기모드
 - w(b) - 쓰기모드
 - a(b) - 추가모드(파일의 마지막에 새로운 내용을 추가)

Create New File

```
f = open("Newfile.txt", 'w')  
f.close()
```

Write text

```
f = open("Newfile.txt", 'a')
for i in range(1,11):
    text = "line %d. \n" % i
    f.write(text)
f.close()
```

Read text

```
f = open("Newfile.txt", 'r')
text = f.readline()
print(text)
f.close()
```

Read All text

```
f = open("Newfile.txt", 'r')
while True:
    text = f.readline()
    if not text: break
    print(text)
f.close()
```

Read All text using readlines

```
f = open("Newfile.txt", 'r')
texts = f.readlines()
for text in texts:
    print(texts)
f.close()
```

Add text

```
f = open("Newfile.txt", 'a')
for i in range(11, 20):
    text = "New line %d \n" % i
    f.write(text)
f.close()
```


Get rid of f.close()

```
with open("foo.txt", 'w') as f:  
    f.write("foo is text dummy")
```

Do these with CSV format

1. install `pandas`

or

Do these with CSV format

```
import csv
```

file I/O with json

```
import json
```

dump(dictionary to json)

```
with open('', 'w') as f:
    user_dict = {}
    user_dict.setdefault('users', []).append({
        'name': 'KD Hong',
        'locale': 'Seoul, KR',
    })
    json.dump(user_dict, f)
```

load(json to dictionary)

```
with open('','r') as f:  
    json.load(f)
```

do it yourself!

`best_boxing_movies.csv` 파일의 모든 텍스트를 읽어 리스트로 구성한 뒤,
`best_boxing_movies.json` 과 같은 형태가 되도록 json 포맷의 파일을 생성하세요.

File I/O with xlsx

<https://openpyxl.readthedocs.io/en/stable/>

```
$ pip install openpyxl
```

```
from openpyxl import Workbook( )
```


Error Handle - Try Except

Error Handle

by using `try, except`

필요한 만큼만 적절히 사용하셔야 합니다 by PEP 8

Error Handle - Syntax

```
try:
    실행문
except:
    실행문
```

Error Handle - ValueError

```
try:
    some_input = int(input("type some number: "))
except ValueError:
    print("I said type some NUMBER!!!!")
```

Error Handle - ValueError

```
try:
    some_input = int(input("type some number: "))
except ValueError as e:
    print("I said type some NUMBER!!!!")
    print(e)
```

Error Handle - FileNotFoundError

```
try:
    f = open('error_example.txt', 'r')
except FileNotFoundError as e:
    print(e)
else:
    text = f.read()
    f.close()
```

Error Handle - Multiple Error

```
try:  
    ...  
except error type 1:  
    ...  
except error type 2:  
    ...
```

Error Handle - Pass Error

```
try:
    f = open('error_example.txt', 'r')
except FileNotFoundError as e:
    pass
else:
    text = f.read()
    f.close()
```


Finally!

```
try:
    f = open('error_example.txt', 'r')
except FileNotFoundError as e:
    print("Oops!")
    pass
else:
    text = f.read()
    f.close()
finally:
    print("어쨌거나 끝났습니다")
```

Do it yourself!

임의의 숫자(1~1000 사이의 정수) `두 개` 로 이루어진 100개의 `tuple` 을 csv파일로 저장한 뒤,

이를 불러와 `곱셈 연산을 수행` 하여 새로운 파일에 두 수와 곱셈 결과를 다시 csv파일로 작성하는 파일을 작성하세요.

(단, 파일을 불러올 때 `try except`를 적용하여, `FileNotFoundError`가 발생했을시 에러메시지만 출력한 뒤, `pass`하세요.)