

2018학년도 2학기 언어와 컴퓨터

제8강 파일 처리

박수지

서울대학교 인문대학 언어학과

2018년 10월 8일 월요일

오늘의 목표

- 1 `open()` 함수를 사용하여 파일을 쓰고 읽을 수 있다.
- 2 `pickle` 모듈을 사용하여 객체를 저장하고 불러올 수 있다.

소스코드 파일 `hangu1.py`를 실행하는 두 가지 방법

■ 스크립트 모드로 실행하기

- 명령: `python hangu1.py` (코드 밖에서)
- 변수 `__name__`의 값: `'__main__'`

■ 모듈로 불러오기

- 명령: `import hangu1` (코드 안에서)
- 변수 `__name__`의 값: `'hangu1'`

조건문 헤더 `if __name__ == '__main__':`의 의미

■ 스크립트 모드에서는 스위트를 실행한다.

← 테스트 코드까지 실행한다.

■ 모듈로 불러올 때는 스위트를 실행하지 않는다.

← 함수만 가져온다.

의문

반환값을 여러 개 가질 수 있는가?

```
1 def example0():  
2     return 0  
3     return 1  
4  
5 print(example0())
```

return의 역할

실행을 멈추고 함수가 호출된 곳으로 돌아간다.

```
1 def example1():  
2     print('출력됩니다.')3     return  
4     print('출력될까요?')  
5  
6 example1()
```

```
1 def example2():  
2     for i in range(3):  
3         print(f'{i}번째 출력')  
4         return  
5  
6 example2()
```

if 절 내의 return

이후의 내용은 else 절 내에 있는 것과 같아진다.

```

1 def example3(n):
2     if n % 2:
3         print('홀수입니다.')
4         return
5     else:
6         print('짝수입니다.')
7     print('언제 출력될까
      요?')
```

```

8
9 example3(2)
10 example3(3)
```

```

1 def example4(n):
2     if n % 2:
3         return '홀'
4     return '짝'
5
6 print(example4(4))
7 print(example4(5))
```

사실 가장 간단한 홀짝 판정

```

1 def isodd(n):
2     return bool(n % 2)
```

파일 쓰기

파일 열고 닫기

```
1 f = open(file, mode='r', encoding=None)
2 ....
3 f.close()
```

`file` 파일명

`mode` 읽기/쓰기 모드

`encoding` 문자 인코딩 방식

파일 쓰기

f.writelines() 메서드

```
1 f = open('latin.txt', 'w')
2 f.writelines('a')
3 f.close()
```

```
1 f = open('hanguk.txt', 'w', encoding='utf-8')
2 f.writelines('가')
3 f.close()
```


파일 쓰기

구구단 쓰기

```
1 f = open('99.txt', 'w')
2 for i in range(1, 10):
3     f.writelines('\t'.join(str(i*j) for j in range(1, 10)))
4     f.writelines('\n')
5
6 f.close()
```

관찰

- 각 열 사이는 탭(' \t ')으로 띄운다(Tab-separated values).
비교 CSV: Comma-separated values
- 한 단이 끝나면 줄바꿈 문자(' \n ')로 줄을 바꾼다.

파일 읽기

`f.read()` 메서드: 파일 전체를 하나의 문자열로 가져오기

```
1 f = open('hangu.txt', encoding='utf-8')
2 text = f.read()
3 f.close()
4 print(text)
```

관찰

읽기 모드는 'r'이라고 명시하지 않아도 된다.

파일 읽기

`f.readlines()` 메서드: 파일 전체를 행들의 리스트로 가져오기

```
1 f = open('99.txt')
2 nines = f.readlines()
3 f.close()
4 print(nines)
```

관찰

각 행 끝에 줄바꿈 문자 '`\n`'이 달려 있다.

파일 읽기

반복문으로 파일을 한 줄씩 읽기

```
1 f = open('99.txt')
2 for line in f:
3     print(line)
4
5 f.close()
```

의문

왜 줄이 두 개씩 띄어질까?

파일 읽기

구구단 읽기

```
1 from pprint import pprint
2 f = open('99.txt')
3 nines = []
4 for line in f:
5     nines.append(line.split())
6
7 f.close()
8 pprint(nines)
```

3-5행을 한 번에 쓰기

```
[line.split() for line in f]
```

문제

값들이 문자열이다.

파일 읽기

nines의 모든 값을 정수로 변환하기

1. 어쨌든 돌아는 가는 방법

```
1 for i in range(len(nines)):
2     for j in range(len(nines[i])):
3         nines[i][j] = int(nines[i][j])
```

하지 마세요...

파일 읽기

nines의 모든 값을 정수로 변환하기

2. 아까보다 나은 방법

```
1 for i, line in enumerate(nines):  
2     for j, n in enumerate(line):  
3         nines[i][j] = int(n)
```

3. 배운 사람의 방법

```
1 nines = [[int(n) for n in line] for line in nines]
```

4. 근본적인 해결

파일을 읽는 단계에서 정수로 변환해서 가져온다.

파일 읽기

구구단 다시 읽기

```
1 f = open('99.txt')
2 nines = []
3 for line in f:
4     l = []
5     for n in line.split():
6         l.append(int(n))
7
8     nines.append(l)
9
10 f.close()
11 pprint(nines)
```

2-8행을 한 번에 쓰기

```
nines = [[int(n) for n in line.split()] for line in f]
```


파일 읽기

리스트의 항목을 한꺼번에 정수로 바꾸는 새로운 방법

```
1 f = open('99.txt')
2 nines = []
3 for line in f:
4     nines.append(list(map(int, line.split())))
5
6 f.close()
7 pprint(nines)
```

map(함수, 열): 열의 모든 항목에 같은 함수 적용하기

```
>>> tuple(map(int, '123'))
(1, 2, 3)
>>> list(map(len, ['1', '23', '456', '7890']))
[1, 2, 3, 4]
```

pickle 모듈로 객체를 파일에 저장하기

`pickle.dump(저장할 객체, 파일 객체)`

```
1 import pickle
2 f = open('99_list.pkl', 'wb')
3 pickle.dump(nines, f)
4 f.close()
```

`mode='wb'` 바이너리 파일로 쓰기

pickle 모듈로 파일에서 객체를 가져오기

`pickle.load()`(가져올 객체가 든 파일 객체)

```
1 f = open('99_list.pkl', 'rb')
2 nines_pkl = pickle.load(f)
3 f.close()
4 pprint(nines_pkl)
```

현재 작업 디렉토리에 파일이 있어야 읽을 수 있다.

현재 작업 디렉토리 (Current working directory) 확인 & 변경

```
>>> import os
>>> os.getcwd() # 현재 위치
'C:\\Users\\user'
>>> os.chdir('Downloads') # 현재 위치 아래의 Downloads 폴더로 이동
>>> os.getcwd()
'C:\\Users\\user\\Downloads'
>>> os.chdir('..') # 현재 위치에서 한 단계 상위 폴더로 이동
>>> os.getcwd()
'C:\\Users\\user'
>>> os.chdir('C:\\Users\\user\\Downloads') # 절대 경로로 이동
>>> os.getcwd()
'C:\\Users\\user\\Downloads'
```

요약

파일 열고 닫기

```
1 | f = open(file, mode, encoding)
2 | f.close()
```

텍스트 파일 처리

쓰기 f.writelines()
읽기 f.read()
f.readlines()

바이너리 파일 처리

쓰기 pickle.dump(obj, f)
읽기 pickle.load(f)

다음 시간에 배울 것

- `encoding='utf-8'`의 정체는 무엇인가?

더 생각해 볼 것

- `all()`, `any()` 함수에 대하여 알아보자.