

파일 입출력

1.1 open()

1.2 파일 형식

1.3 with문

• Open()

◆ 변수 = open(파일이름,오픈형식)

- ex) f = open('test.txt', 'w')
- open함수는 파일을 원하는 형식으로 열어줍니다. 이 때 변수는 열어준 파일을 가리킵니다.

◆ 파일변수.close()

- close함수는 파일변수가 가리키는 파일을 닫아줍니다.
- 파일을 닫아주지 않으면 프로그램이 종료되더라도 파일이 열려있는 채로 남아있을 수 있습니다.

파일 입출력

• 파일 입출력 형식

문자	의미
r	읽기모드, 파일 없으면 Error
r+	읽기/쓰기 모드, 파일 없으면 Error (기존 내용 삭제 X, 덮어쓰기)
w	쓰기 모드, 파일 없으면 생성
w+	읽기/쓰기 모드, 파일 없으면 생성 (기존 내용 삭제, 새로 쓰기)
a	파일 추가(FP가 파일 끝으로 이동)로 쓰기모드, 파일 없으면 새로 생성
a+	읽기/파일 추가 모드, 파일 없으면 생성
t	텍스트 모드로 열기 ex) rt, w+t
b	바이너리 모드로 열기 ex)rb, w+b

- with open() as param:

- ◆ with open(file, mode, encoding) as file:

- *ex)* with open('test.txt', 'w', encoding='UTF-8') as f:
- with 로 파일을 열 경우 `file.close()` 필요 X
- with 문은 파일을 반드시 닫아주기 때문에 더 좋습니다. 심지어 with 블록 안에서 예외(exception)가 발생해도 그렇습니다.
- 메모리 누수 없음

파일 저장하기

2.1 CSV

2.2 OpenPyXL

1.3 with문

- CSV

- ◆ Comma separated value

- ‘,’ 로 구분되어 있는 파일
- Ex) 박정현,꿈에,임창정,소주한잔

- ◆ `f.write(param + ‘,’ + parmam)`

- param의 값들을 ‘,’ 로 이어주어 저장.

2.2

OpenPyXL

• import openpyxl as xl

◆ 워크북(엑셀파일) 생성하기

- `wb = xl.Workbook()`

◆ 시트 선택, 데이터 쓰기

- `sheet = wb.active`
- `sheet['셀'] = 'hello'` # 해당 셀 데이터 삽입
- `sheet.cell(row=m, column=n).value = "BYE!!"` # m행 n열 데이터 삽입
- `sheet.append(LIST)` # FP가 가르키고 있는 다음 첫 줄에 데이터 입력

◆ 워크북 저장하기

- `wb.save('파일.xlsx')`

2.2

OpenPyXL

- import openpyxl as xl

◆ 기존 파일 불러오기

- wb = xl.load_workbook('파일이름.xlsx')
- 파일이 존재하지 않을 경우, Error

예외처리 (Exception Handling)

3.1 try, except

3.2 else문

3.3 finally문

• 예외처리 기본 문법

◆ try: ~ except:

- 에러가 발생할 것 같은 코드를 try안에 넣고 except 뒤에 발생할 수 있는 에러의 이름을 적어두면, 에러 발생시 프로그램이 멈추지 않고 별도 처리가 가능하다.

- ex) try:

```
list1 = []  
print(list[1])
```

except:

```
print( 'Error raised' )
```

- 특정 오류에 관해서만 예외처리

- ◆ except 예외상황:

- 특정 에러에 관해서만 처리하기 위해 except 부분에 예외 종류 제시

- ex) try:

```
with open('test.txt', 'r+') as file:  
    file.write('hello')  
except FileNotFoundError:  
    print('파일이 존재하지 않습니다.')
```

3.1

예외처리

- 발생 오류와 오류 메시지 변수까지 포함한 except문

- ◆ except 예외상황 as 변수:

- 에러에 관한 내용을 출력하기 위한 방법

- ex) try:

```
with open('test.txt', 'r+') as file:
```

```
    file.write('hello')
```

```
except Exception/FileNotFoundError as e:
```

```
    print('Error message', e)
```

• else 문

◆ try: ~ except: ~ else: ~

- try문에서 정상적으로 작동하였을 시, else문 코드 실행
- ex) try:

```
    with open( 'test.txt' , 'r+' ) as file:  
        file.write( 'hello' )  
except Exception/FileNotFoundError as e:  
    print( 'Error message' , e)  
else:  
    print( 'file exist!' )
```

• finally 문

◆ try: ~ except: ~ else: ~ finally: ~

- try 문에 연관된 모든 코드가 끝난 뒤 실행
- ex) try:

```
    with open('test.txt', 'r+') as file:  
        file.write('hello')  
except Exception/FileNotFoundError as e:  
    print('Error message', e)  
else:  
    print('file exist!')  
finally:  
    print('Exception Handling code Finished')
```

Q & A