

05

파일 읽고 쓰기

05 파일 읽고 쓰기

■ 파일 생성하기

- 입출력 방법이 꼭 input()과 print()만 있는 것은 아님
- 파일을 통한 입출력도 가능

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

- 소스코드를 실행하면 프로그램을 실행한 디렉터리에 새로운 파일이 하나 생성됨

05 파일 읽고 쓰기

■ 파일 생성하기

- 파일을 생성하기 위해 파이썬 내장 함수 open 사용
- open() 함수
 - 파일을 생성하기 위해 사용
 - '파일 이름'과 '파일 열기 모드'를 입력값으로 받고 결과값으로 파일 객체 반환

파일 객체 = open(파일 이름, 파일 열기 모드)

05 파일 읽고 쓰기

■ 파일 생성하기

■ 파일 열기 모드

| 파일 열기 모드 | 설명 |
|----------|-----------------------------------|
| r | 읽기 모드 - 파일을 읽기만 할 때 사용 |
| w | 쓰기 모드 - 파일에 내용을 쓸 때 사용 |
| a | 추가 모드 - 파일의 마지막에 새로운 내용을 추가할 때 사용 |

- 파일을 쓰기 모드(w)로 열면 해당 파일이 이미 존재할 경우 원래 있던 내용이 모두 사라지고, 해당 파일이 존재하지 않으면 새로운 파일이 생성됨

05 파일 읽고 쓰기

■ 파일 쓰기 모드로 열어서 출력값 적기

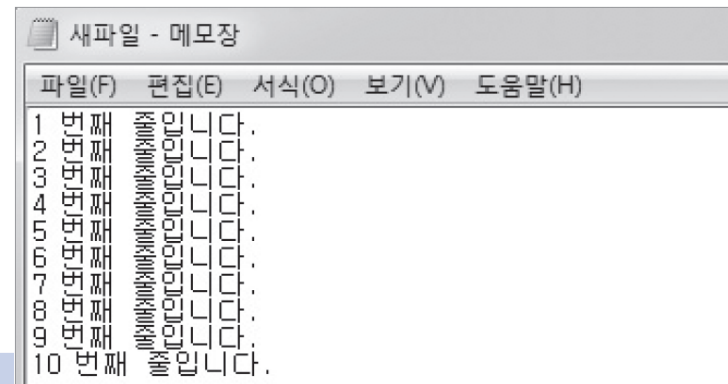
■ 모니터 출력(콘솔 출력)

```
for i in range(1, 11):  
    data = "%d번째 줄입니다.\n" % i  
    print(data)
```

■ 파일 쓰기 모드

```
f = open("새파일.txt", 'w')  
for i in range(1, 11):  
    data = "%d번째 줄입니다.\n" % i  
    f.write(data)  
f.close()
```

< 실행 결과 >



05 파일 읽고 쓰기

■ 파일 읽는 방법

1. readline 함수 사용하기

- f.open("새파일.txt", 'r')로 파일을 읽기 모드로 연 후
readline()을 사용해서 파일의 첫 번째 줄을 읽어 출력하는 코드

```
# readline.py  
f = open("새파일.txt", 'r')  
line = f.readline()  
print(line)  
f.close()
```

1번째 줄입니다.

05 파일 읽고 쓰기

■ 파일 읽는 방법

1. readline 함수 사용하기

- 모든 줄을 읽어서 화면에 출력하기
 - 무한루프 안에서 f.readline()을 사용해 파일을 계속해서 한 줄씩 읽어 들임
- 더 이상 읽을 줄이 없으면 break 수행
- readline()은 더 이상 읽을 줄이 없을 경우 **None**을 출력

```
# readline_all.py
f = open("새파일.txt", 'r')
while True:
    line = f.readline()
    if not line: break
    print(line)
f.close()
```

05 파일 읽고 쓰기

■ 파일 읽는 방법

2. readlines 함수 사용하기

- readlines() 함수는
파일의 모든 줄을 읽어서 각각의 줄을 요소로 갖는 리스트로 반환
 - ["1 번째 줄입니다.", "2 번째 줄입니다.", ..., "10 번째 줄입니다."] 를 반환

```
f = open("새파일.txt", 'r')
lines = f.readlines()
for line in lines:
    print(line)
f.close()
```


05 파일 읽고 쓰기

■ 파일 읽는 방법

3. read 함수 사용하기

- f.read()는 파일의 내용 전체를 문자열로 반환
- data는 파일의 전체 내용

```
f = open("C:/doit/새파일.txt", 'r')  
data = f.read()  
print(data)  
f.close()
```

05 파일 읽고 쓰기

■ 파일 추가 방법

```
f = open("새파일.txt", 'a')
for i in range(11, 20):
    data = "%d번째 줄입니다.\n" % i
    f.write(data)
f.close()
```

05 파일 읽고 쓰기

■ with문 사용법

- 지금까지 파일을 열고 닫은 방법

```
f = open("foo.txt", 'w')  
f.write("Life is too short, you need python")  
f.close()
```

- f.close()는 열려 있는 파일 객체를 닫아 주는 역할
- 쓰기 모드로 열었던 파일을 닫지 않고 다시 사용하면 오류가 발생하기 때문에, close()를 사용해서 열려 있는 파일을 직접 닫아 주는 것이 좋음

05 파일 읽고 쓰기

■ with문 사용법

- with문은 파일을 열고 닫는 것을 자동으로 처리해주는 문법
- 앞선 예제를 with문을 사용하여 수정한 코드

```
with open("foo.txt", "w") as f:  
    f.write("Life is too short, you need python")
```

- with문을 사용하면 with 블록을 벗어나는 순간 열린 파일 객체 f가 자동으로 close 됨

05 파일 읽고 쓰기

■ 실습문제

- 다음과 같은 내용을 지닌 파일 test.txt가 있다.
- 이 파일의 내용 중 "java"라는 문자열을 "python"으로 바꾸어서 저장해 보자.

```
Life is too short  
You need python
```

05 파일 읽고 쓰기

■ 실습문제

- 10명 학생의 점수가 [70, 60, 55, 75, 95, 90, 80, 80, 85, 100]이다. 이를 파일(score.txt)에 저장하시오.
- score.txt 파일에 저장한 점수를 불러와서 총점과 평균을 구하시오.

감사합니다

**“Life is too short,
You need Python!”**

인생은 너무 짧으니,
파이썬이 필요해!