

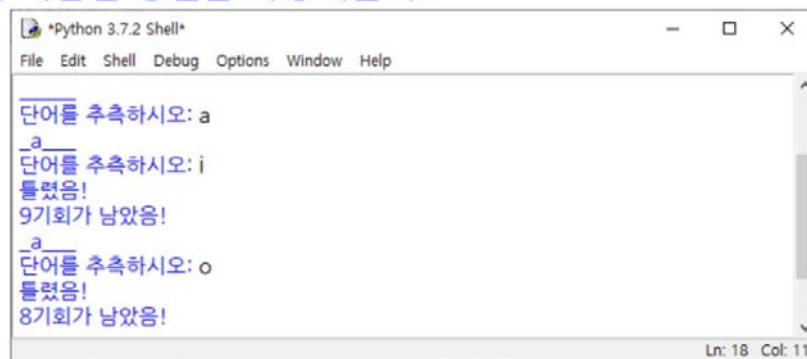
11. 파일

학습 내용

- 파일을 생성하고, 파일에 데이터를 저장하는 방법
- 파일에서 데이터를 읽는 방법
- 간단한 메모장 프로그램 작성

이번 장에서 만들 프로그램

- 간단한 단어 게임인 행맨을 작성해본다.



```
*Python 3.7.2 Shell*
File Edit Shell Debug Options Window Help

단어를 추측하시오: a
_a _ _ _
단어를 추측하시오: i
틀렸음!
9기회가 남았음!
_a _ _ _
단어를 추측하시오: o
틀렸음!
8기회가 남았음!

Ln: 18 Col: 11
```

- 윈도우에 있는 메모장과 같은 프로그램을 작성해보자.



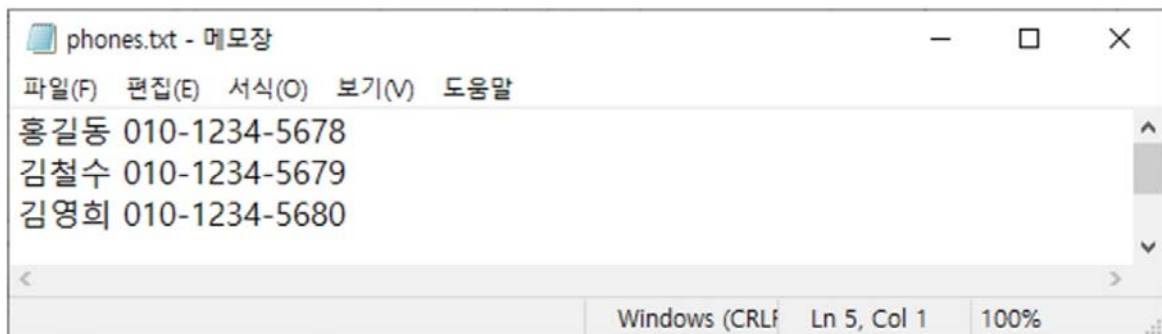
파일의 필요성



실습용 텍스트 파일 만들기

●메모장으로 다음과 같은 텍스트 파일을 작성한다.

●파일명 : d:\Temp\phones.txt



파일에서 데이터 읽기 #1

1. 파일을 연다.
2. 파일에서 데이터를 읽거나 쓸 수 있다.
3. 파일과 관련된 작업이 모두 종료되면 파일을 닫아야 한다.



파일 열고 닫기

파일 사용하기

```
infile = open( "input.txt", "r" )
...
infile .close()
```

Diagram illustrating the file opening and closing process:

- 파일 객체** (File object) points to the variable `infile`.
- 파일의 이름(name)** (File name) points to the string `"input.txt"`.
- 파일을 여는 모드(mode)** (File opening mode) points to the string `"r"`.

파일 모드

파일 모드	모드 이름	설명
"r"	읽기 모드(read mode)	파일의 처음부터 읽는다.
"w"	쓰기 모드(write mode)	파일의 처음부터 쓴다. 파일이 없으면 생성된다. 만약 파일이 존재하면 기존의 내용은 지워진다.
"a"	추가 모드(append mode)	파일의 끝에 쓴다. 파일이 없으면 생성된다.
"r+"	읽기와 쓰기 모드	파일에 읽고 쓸 수 있는 모드이다. 모드를 변경하려면 seek()가 호출되어야 한다.



"r"

파일의 처음 부터 읽는다.



"w"

파일의 처음 부터 쓴다.
만약 파일이 존재하면 기존의
내용이 지워진다.



"a"

파일의 끝에 쓴다.
파일이 없으면 생성 된다.

파일에서 읽기 (실습 - P1101)

```
infile = open("D:/Temp/phones.txt", "r")
lines = infile.read()
print(lines)
infile.close()
```

```
홍길동 010-1234-5678
김철수 010-1234-5679
김영희 010-1234-5680
```

●read() 메소드

- ◆파일 객체에서 전체 텍스트를 읽는다.
- ◆지정된 개수의 문자 읽기 → read(10)

파일에서 읽기

●파일이름 지정

●백슬래시(\) 는 특수 문자(이스케이프 코드)를 입력할 때 사용하기 때문에 2개를 사용해야 한다.

- ◆Wn → LF, Line Feed (커서를 다음 줄로 이동)
- ◆Wr → CR, Carriage Return (커서를 맨 앞쪽으로 이동)
- ◆Enter 키를 치면 : CR + LF 으로 대체되어 입력된다.

	phones.txt
1	홍길동 010-1234-5678CR LF
2	김철수 010-1234-5679CR LF
3	김영희 010-1234-5680CR LF

코드	설명
\n	개행 (줄바꿈)
\t	수평 탭
\\	문자 "\"
\'	단일 인용부호('')
\"	이중 인용부호("")
\r	캐리지 리턴
\f	폼 피드
\a	벨 소리
\b	백 스페이스
\000	널문자

파일에서 읽기

●파일 지정

●백슬래시 2개(\\)나 슬래시 (/)를 사용한다.

●절대 경로 지정하기

- ◆루트부터 지정한다. → "D:\\\\Temp\\phones.txt" 또는 "D:/Temp/phones.txt"

●상대 경로 지정하기

- ◆현재 프로그램이 있는 폴더부터 지정한다.
- ◆"." "은 부모 폴더를 의미한다.
- ◆현재 프로그램 : "D:\\Programming\\python\\file1.py"
- ◆"D:\\Temp\\phones.txt" 를 읽고자 할 경우

```
infile = open("../Temp/phones.txt", "r")
lines = infile.read()
print(lines)
infile.close()
```


파일에서 읽기 (실습 - P1102)

```
infile = open("D:/Temp/phones.txt", "r")
lines = infile.readlines()
print(lines)
infile.close()
```

```
['홍길동 010-1234-5678\n', '김철수 010-1234-5679\n', '김영희 010-1234-5680\n']
```

●readlines() 메소드

- ◆파일에 저장된 모든 데이터를 읽는다.
- ◆각각의 줄을 리스트의 한 항목으로 저장한다.
- ◆주의 사항 : 각 항목의 데이터 끝에 '\n' 이 붙어 있다.

●close()

- ◆파일을 열어서 작업하고 반드시 close() 메소드를 사용하여 닫아야 한다.
- ◆닫지 않고, 이 프로그램이 계속 수행중이라면 다른 프로그램이 phones.txt 여는데 문제가 발생

한 줄씩 읽기 (실습 - P1103)

```
infile = open("D:/Temp/phones.txt", "r")
line = infile.readline().rstrip()
while line != "":
    print(line)
    line = infile.readline().rstrip()
infile.close()
```

```
홍길동 010-1234-5678
김철수 010-1234-5679
김영희 010-1234-5680
```

●readline()

- ◆한번에 한 줄만 읽어서 문자열로 반환한다.

●rstrip()

- ◆문자열 오른쪽의 문자 이외의 것은 삭제한다.

strip() 함수 (실습 - P1104)

```
str = "    this is string example....wow!!!    "
print("원본 문자열=[" + str + "]")
print("rstrip()=[" + str.rstrip() + "]")
print("lstrip()=[" + str.lstrip() + "]")
print("strip()=[" + str.strip() + "]")
print()

str = "    this is string example....wow!!!\nWt    "
print("원본 문자열=[" + str + "]")
print("rstrip()=[" + str.rstrip() + "]")
print()

str = "88888888this is string example....wow!!!88888888"
print("원본 문자열=[" + str + "]")
print("rstrip('8')=[" + str.rstrip('8') + "]")
print("lstrip('8')=[" + str.lstrip('8') + "]")
print("strip('8')=[" + str.strip('8') + "]")
print()

txt = "banana,,,,,ssawww...."
print("원본 문자열=[" + txt + "]")
print("rstrip('.,asw')=[" + txt.rstrip('.,asw') + "]")
```

strip() 함수 (실습 - P1104)

```
str = "    this is string example....wow!!!    "
print("원본 문자열=[" + str + "]")
print("rstrip()=[" + str.rstrip() + "]")
print("lstrip()=[" + str.lstrip() + "]")
print("strip()=[" + str.strip() + "]")
print()
```

```
원본 문자열=[    this is string example....wow!!!    ]
rstrip()=[    this is string example....wow!!!]
lstrip()=[this is string example....wow!!! ]
strip()=[this is string example....wow!!!]
```

strip() 함수 (실습 - P1104)

```
str = "    this is string example....wow!!!\nWt    "  
print("원본 문자열=[" + str + "]")  
print("rstrip()=[" + str.rstrip() + "]")  
print()
```

```
원본 문자열=[    this is string example....wow!!!  
]  
rstrip()=[    this is string example....wow!!!]
```

strip() 함수 (실습 - P1104)

```
str = "88888888this is string example....wow!!!88888888"  
print("원본 문자열=[" + str + "]")  
print("rstrip('8')=[" + str.rstrip('8') + "]")  
print("lstrip('8')=[" + str.lstrip('8') + "]")  
print("strip('8')=[" + str.strip('8') + "]")  
print()
```

```
원본 문자열=[88888888this is string example....wow!!!88888888]  
rstrip('8')=[88888888this is string example....wow!!!]  
lstrip('8')=[this is string example....wow!!!88888888]  
strip('8')=[this is string example....wow!!!]
```


strip() 함수 (실습 - P1104)

```
txt = "banana,,,,,ssaaww....."  
print("원본 문자열=[" + txt + "]")  
print("rstrip(',.asw')=[" + txt.rstrip(",.asw") + "]")
```

```
원본 문자열=[banana,,,,,ssaaww.....]  
rstrip(',.asw')=[banan]
```

한 줄씩 읽기 (실습 - P1105)

- 파일 객체를 문자열의 컨테이너로 간주
 - for 루프를 이용하여 파일 객체에 대하여 반복한다.
 - 가장 많이 사용하는 방법이다.

```
infile = open("D:/Temp/phones.txt", "r")  
for line in infile:  
    line = line.rstrip()  
    print(line)  
infile.close()
```

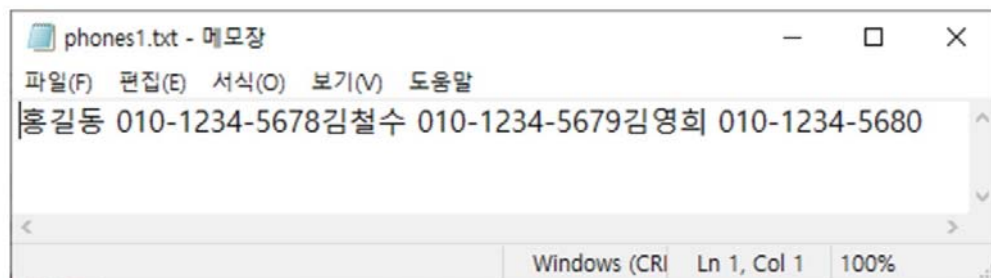
```
홍길동 010-1234-5678  
김철수 010-1234-5679  
김영희 010-1234-5680
```

파일에 데이터 쓰기 (실습 - P1106)

```
outfile = open("D:/Temp/phones1.txt", "w")

outfile.write("홍길동 010-1234-5678")
outfile.write("김철수 010-1234-5679")
outfile.write("김영희 010-1234-5680")

outfile.close()
```



파일에 데이터 쓰기 (실습 - P1107)

```
outfile = open("D:/Temp/phones1.txt", "w")

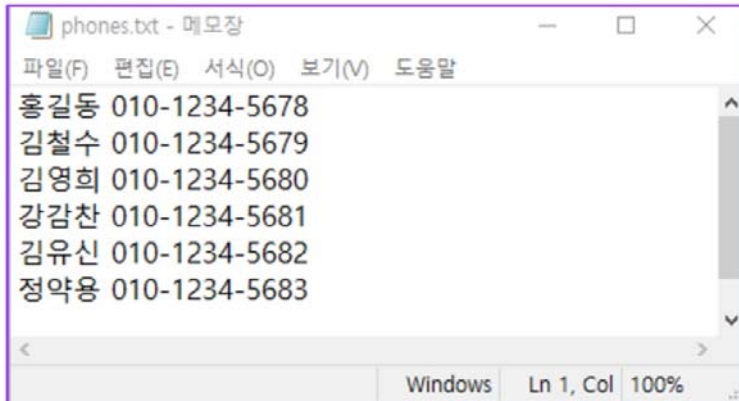
outfile.write("홍길동 010-1234-5678\\n")
outfile.write("김철수 010-1234-5679\\n")
outfile.write("김영희 010-1234-5680\\n")

outfile.close()
```



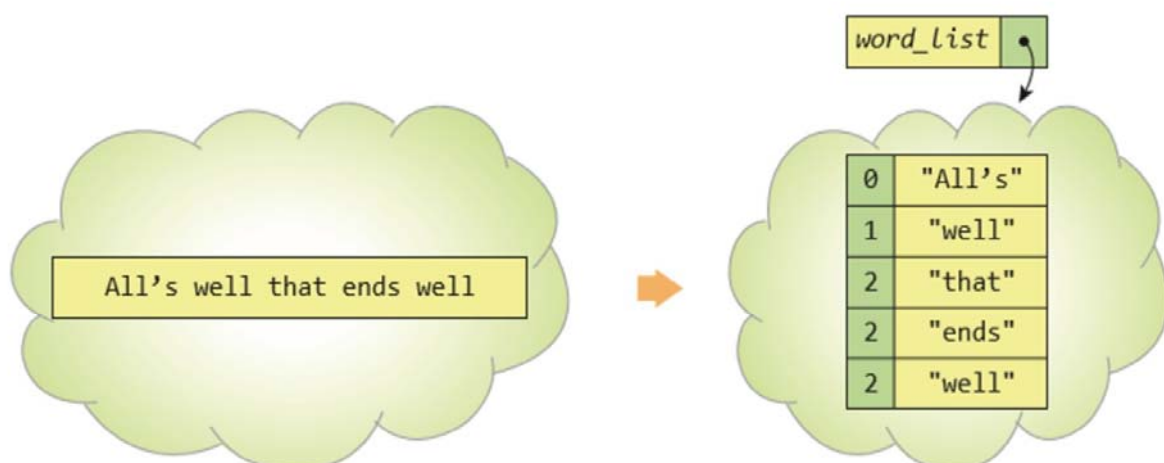
파일에 데이터 추가하기 (실습 - P1108)

```
outfile = open("D:/Temp/phones.txt", "a")  
  
outfile.write("강감찬 010-1234-5681\\n")  
outfile.write("김유신 010-1234-5682\\n")  
outfile.write("정약용 010-1234-5683\\n")  
  
outfile.close()
```



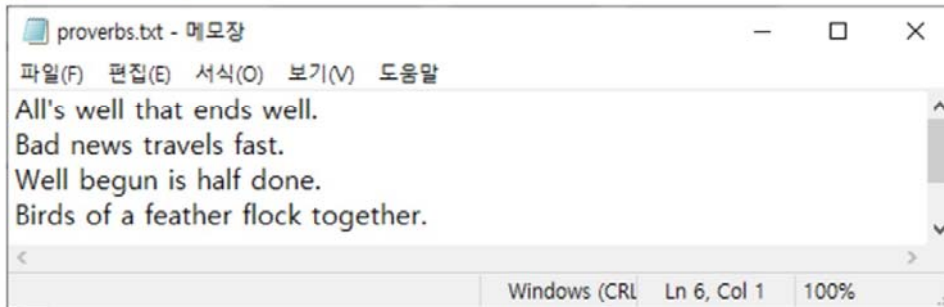
파일에서 단어 읽기

●split() 함수



파일에 데이터 추가하기 (실습 - P1109)

```
infile = open("proverbs.txt", "r")
for line in infile:
    line = line.rstrip()
    word_list = line.split()
    for word in word_list:
        print(word)
infile.close()
```



All's
well
that
ends
well.
Bad
news
travels
fast.
Well
begun
is
half
done.
Birds
of
a
feather
flock
together.

파일 복사하기 (실습 - P1110)

- 파일을 복사하는 프로그램을 작성해보자.
- 파일의 이름은 사용자가 입력하도록 하자.



파일 복사하기 (실습 - P1110)

```
# 입력 파일 이름과 출력 파일 이름을 받는다.
infilename = input("입력 파일 이름: ");
outfilename = input("출력 파일 이름: ");

# 입력과 출력을 위한 파일을 연다.
infile = open(infilename, "r")
outfile = open(outfilename, "w")

# 전체 파일을 읽는다.
s = infile.read()

# 전체 파일을 쓴다.
outfile.write(s)

# 파일을 닫는다.
infile.close()
outfile.close()
```

파일을 읽는 함수

●readline()

- 파일의 한 줄을 읽는다.

●readlines()

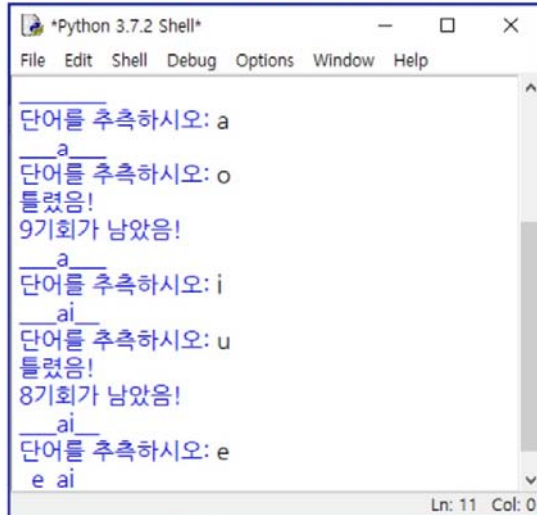
- 파일의 모든 라인을 읽어서, 각각의 줄을 요소로 갖는 리스트를 리턴한다.

●read()

- 파일의 내용 전체를 문자열로 리턴한다.

행맨 (실습 - P1111)

- 단어 게임으로 유명한 것이 행맨(hangman)이다.
- 행맨은 컴퓨터가 생각하는 단어를 맞춰가는 게임이다.
- 사용자는 한번에 하나의 글자만을 입력할 수 있으며 맞으면 글자가 보이고 아니면 시도 횟수만 하나 증가한다.
- 사용할 단어가 들어 있는 파일(words.txt) 는 제공한다.



```
*Python 3.7.2 Shell*
File Edit Shell Debug Options Window Help

단어를 추측하시오: a
a _ _ _ _ _
단어를 추측하시오: o
틀렸음!
9기회가 남았음!
a _ _ _ _ _
단어를 추측하시오: i
a i _ _ _ _ _
단어를 추측하시오: u
틀렸음!
8기회가 남았음!
a i _ _ _ _ _
단어를 추측하시오: e
e a i _ _ _ _ _
Ln: 11 Col: 0
```

행맨 (실습 - P1111)

```
import random

guesses = ""
turns = 10

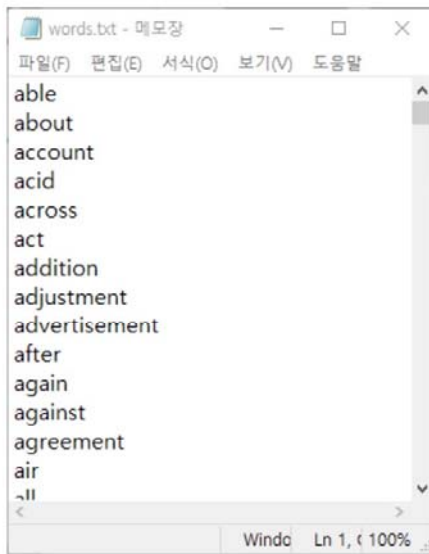
infile = open("words.txt", "r")
lines = infile.readlines()
word = random.choice(lines)

while turns > 0:
    failed = 0
    for char in word:
        if char in guesses:
            print(char, end=" ")
        else:
            print("_", end=" ")
            failed += 1
    if failed == 0:
        print("사용자 승리")
        break
```

```
print("")
guess = input("단어를 추측하시오: ")
guesses += guess
if guess not in word:
    turns -= 1
    print("틀렸음!")
    print(str(turns) + '기회가 남았음!')
if turns == 0:
    print("사용자 패배 정답은 " + word)

infile.close()
```

●words.txt



과제

●리스트와 딕셔너리의 과제인 SalesDict.py 문제 수정

●판매 품목인 items 를 파일에 저장하는 방법으로 변경하세요.

- ◆프로그램 실행시 items 를 파일로부터 읽어 들인다.
- ◆“저장” 명령을 입력하면 파일에 items 의 변경된 내용을 저장한다.
- ◆“추가” 명령을 입력하면 items 에 판매 물품을 추가한다.
 - ❖판매하는 물품이름, 단가, 판매수량, 재고 정보를 입력받아 물품을 추가한다.
- ◆“입고” 명령을 입력하면 items 의 재고 수량을 늘릴 수 있도록 변경한다.
- ◆“변경” 명령을 입력하면 판매 품목의 단가를 변경할 수 있도록 한다.

●변경 사항에서 사용하는 함수이름은 임의로 정해서 사용할 것

- ◆가능한 기능별로 함수를 별도로 만들어 필요할 때 호출할 수 있도록 한다.

●이 과제는 클래스에서도 계속하여 변경하므로 수강생은 반드시 프로그램을 작성하여 제출하여야 한다.

●완성된 코드는 제공되지 않음.

과제

●제출파일명 : 판매관리2-학번-이름.zip

●제출 파일

●소스 코드 : *.py 를 반드시 제출할 것

◆소스 코드를 캡처하여 제출할 경우, 제출하지 않은 것으로 간주함

●실행화면 캡처

◆각 기능별로 실행화면을 캡처하여 제출한다.

◆프로그램으로 작성한 기능이라도 캡처가 되어 있지 않은 경우, 구현하지 않은 것으로 간주됨

●설명 파일

◆동작등을 설명하고 싶은 경우, 별도의 설명파일(한컴, MS Word, PowerPoint)을 작성하여 제출하여도 된다.

◆모든 캡처한 실행화면을 설명파일에 넣어 제출하여도 된다.

●주의 사항

●본 과제는 프로그래밍 기초과목의 과제로 인정되며, LAB의 과제가 아니므로 제출시 주의할 것

이번 장에서 배운 것

●파일은 컴퓨터 전원이 꺼져도 없어지지 않는다.

●변수에 들어 있는 값들은 컴퓨터 전원이 꺼지면 없어진다.

●파일을 읽을 때는 파일을 열고, 데이터를 읽은 후에, 파일을 닫는 절차가 필요하다.

●파일 모드에서 “r”, “w”, “a”가 있다.

●각각 읽기모드, 쓰기모드, 추가모드를 의미한다.