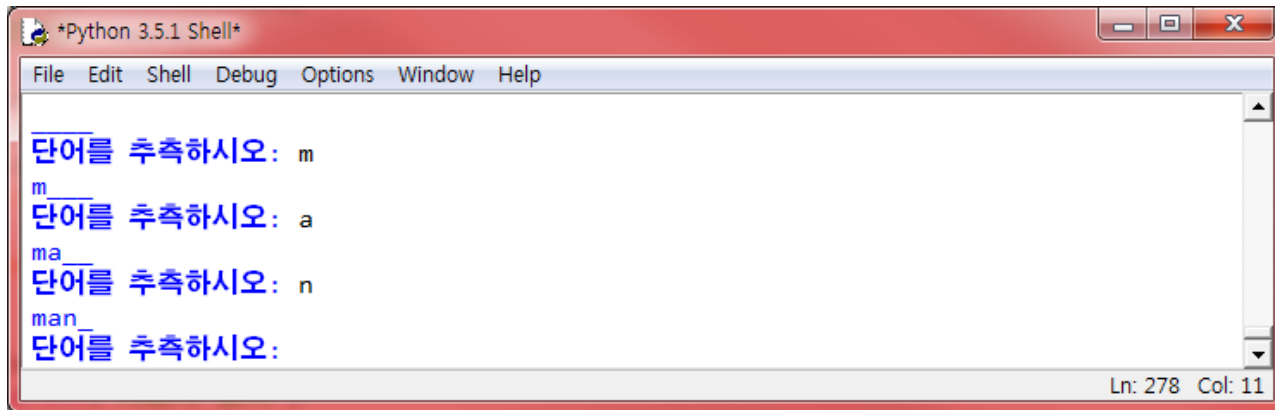




11장 파일을 사용해봅시다.

# 이번 장에서 만들 프로그램

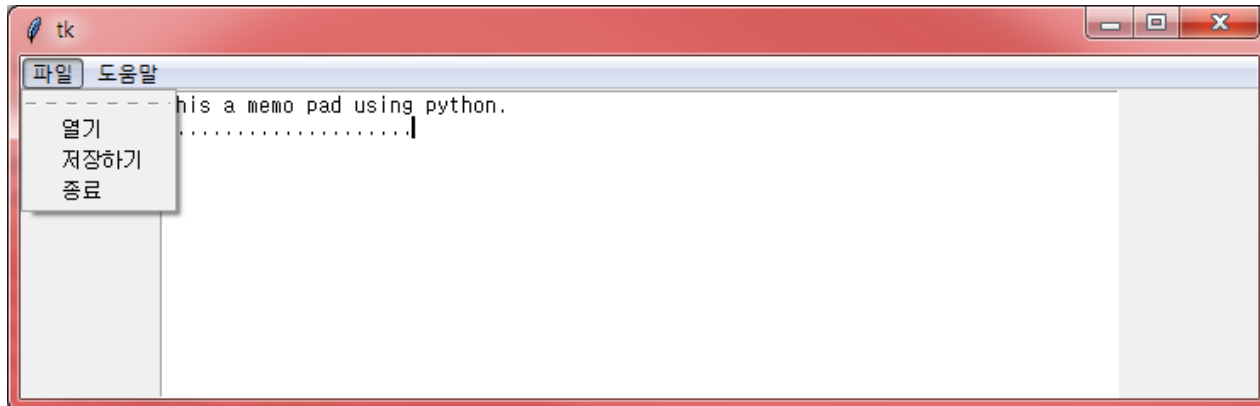
(1) 간단한 단어 게임인 행맨을 작성해본다.



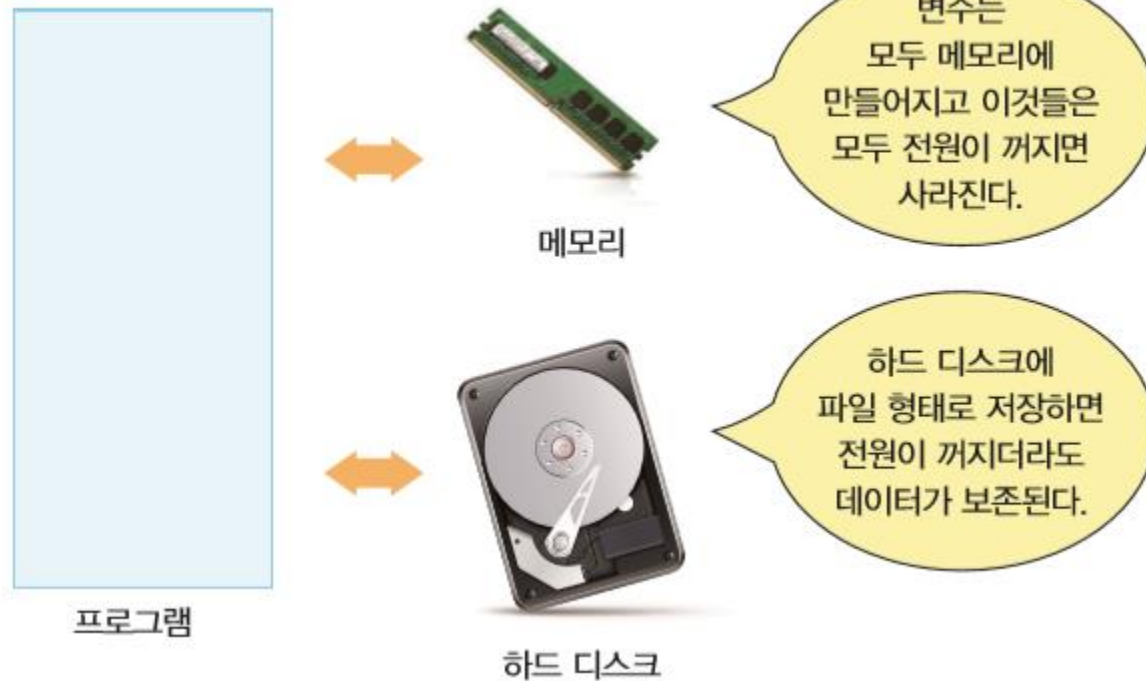
A screenshot of a Python 3.5.1 Shell window. The window has a menu bar with 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The main text area shows a Hangman game in progress. The prompt '단어를 추측하시오:' is followed by the input 'm', then 'a', then 'n', and finally 'man'. The status bar at the bottom right shows 'Ln: 278 Col: 11'.

```
*Python 3.5.1 Shell*
File Edit Shell Debug Options Window Help
단어를 추측하시오: m
m
단어를 추측하시오: a
ma
단어를 추측하시오: n
man
단어를 추측하시오:
Ln: 278 Col: 11
```

(2) 윈도우에 있는 메모장과 같은 프로그램을 작성해보자.



# 파일의 필요성

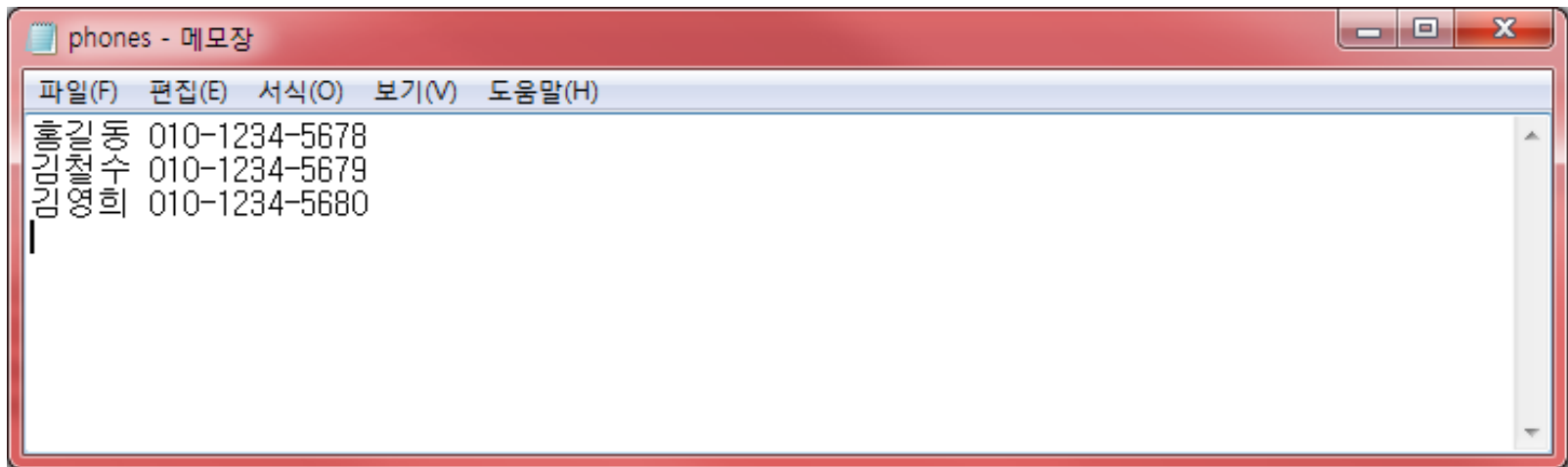


데이터를 영구  
보관하고 싶나요?  
그렇다면 파일에 저장  
해야 해요.



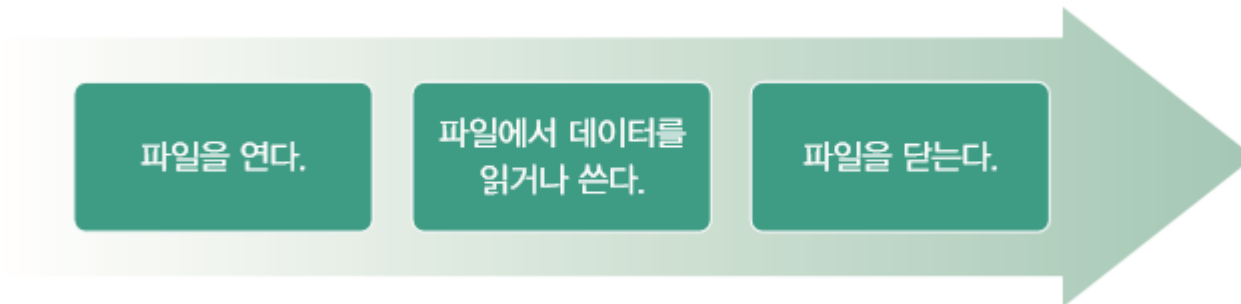
# 실습용 텍스트 파일 만들기

- 메모장으로 다음과 같은 텍스트 파일을 작성한다.



# 파일에서 데이터 읽기

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()
```

The diagram illustrates the process of opening and closing a file in Python. It shows the code `infile = open( "input.txt", "r" )`, followed by an ellipsis `...`, and then `infile .close()`. Annotations with arrows point to specific parts of the code: a green box labeled "파일 객체" (File object) points to `infile`; a green box labeled "파일의 이름(name)" (File name) points to the string `"input.txt"`; a green box labeled "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

```
infile = open("d:\\phones.txt", "r")  
lines = infile.read()  
print(lines)
```

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



# 파일에서 읽기

```
infile = open("d:\\phones.txt", "r")  
lines = infile.readlines()  
print(lines)
```

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

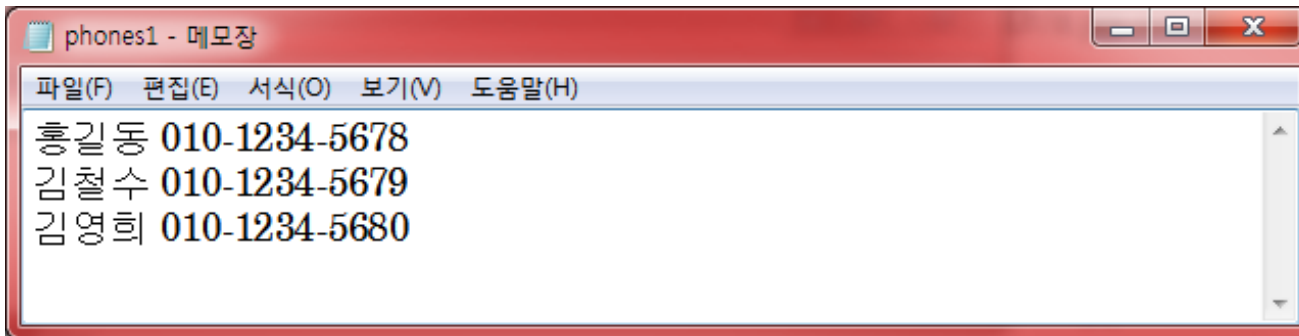
# 한 줄씩 읽기

```
infile = open("d:\\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
```

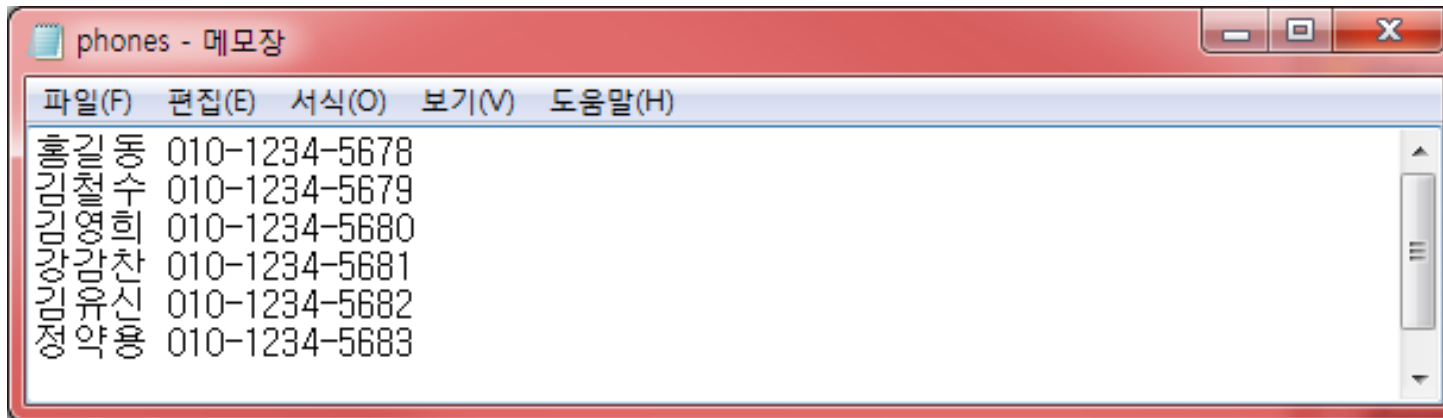
# 파일에 데이터 쓰기

```
outfile = open("d:\\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()
```



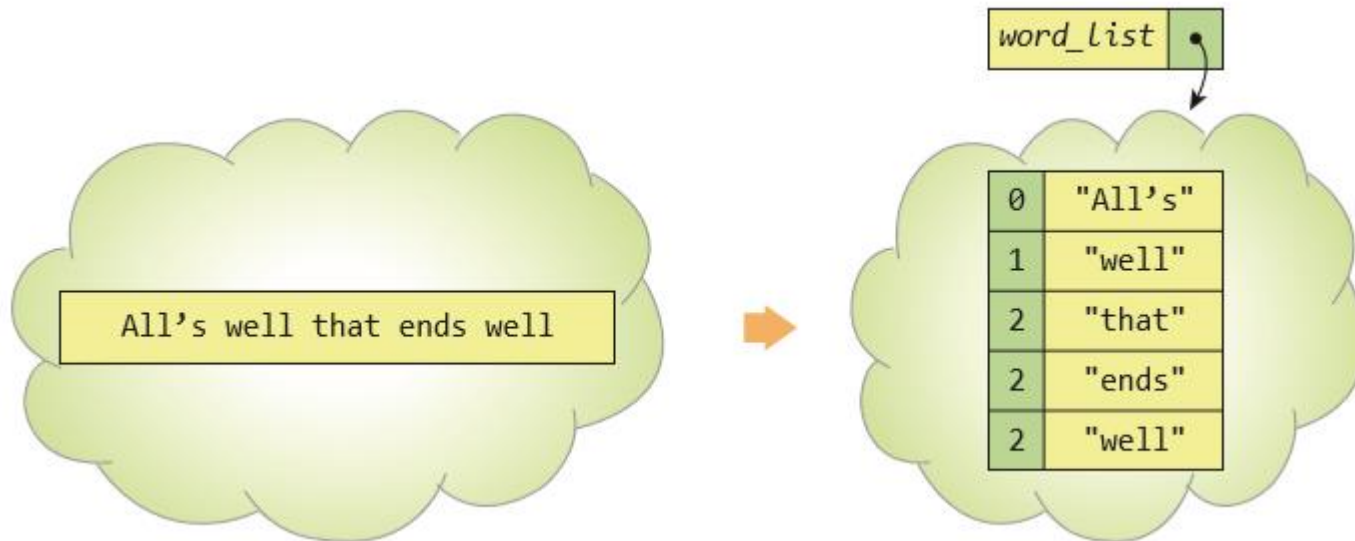
# 파일에 데이터 추가하기

```
outfile = open("d:\\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()
```

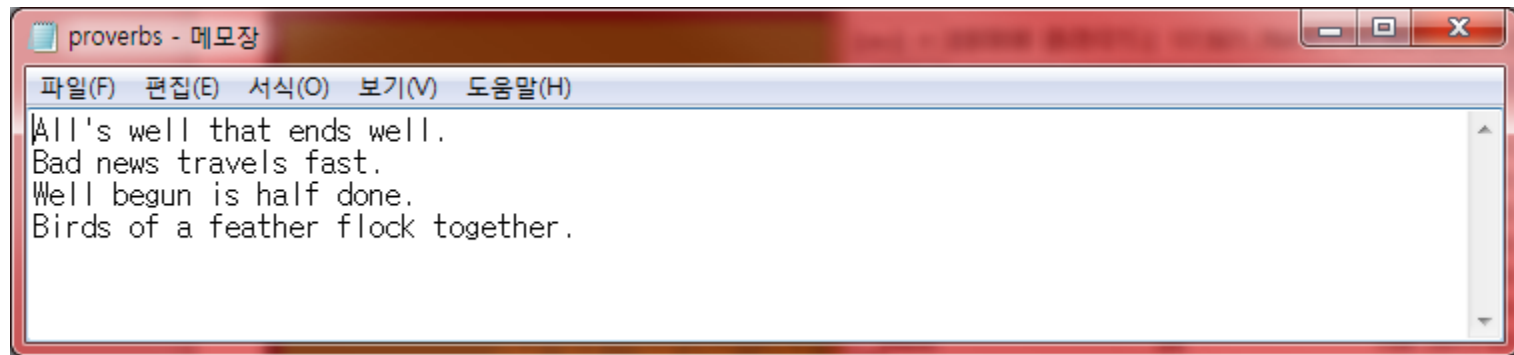


# 파일에서 단어 읽기

## □ split() 함수



# 속담을 저장한 파일



# 파일에 데이터 추가하기

```
infile = open("d:\\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
...
flock
together.
```

# Lab: 파일 복사하기



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





# Solution

```
# 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()
```

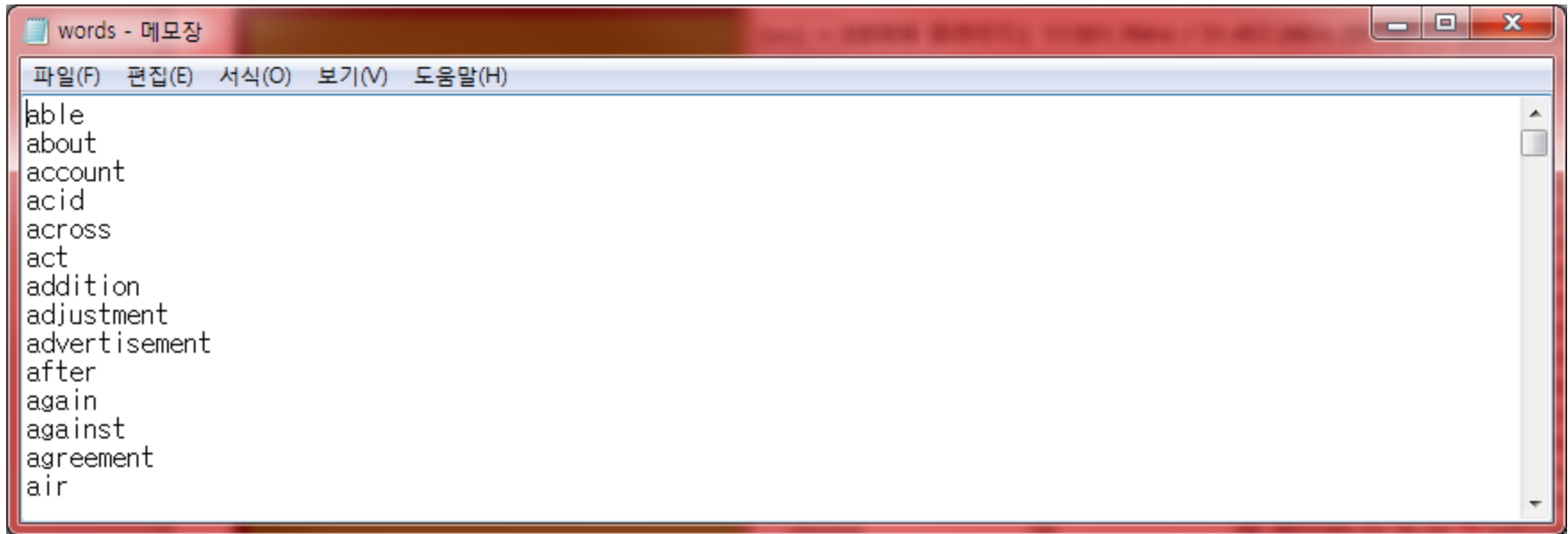
# Lab: 행맨



- 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.

A screenshot of a Python 3.5.1 Shell window. The window has a red title bar with the text '\*Python 3.5.1 Shell\*'. Below the title bar is a menu bar with 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The main area of the window is white and contains the following text in Korean: '단어를 추측하시오: m', 'm', '단어를 추측하시오: a', 'ma', '단어를 추측하시오: n', 'man', and '단어를 추측하시오:'. The text is in blue. At the bottom right of the window, there is a status bar that says 'Ln: 278 Col: 11'.

# 단어들이 저장된 파일



# Solution

```
import random

guesses = ''
turns = 10

infile = open("d:\\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
```

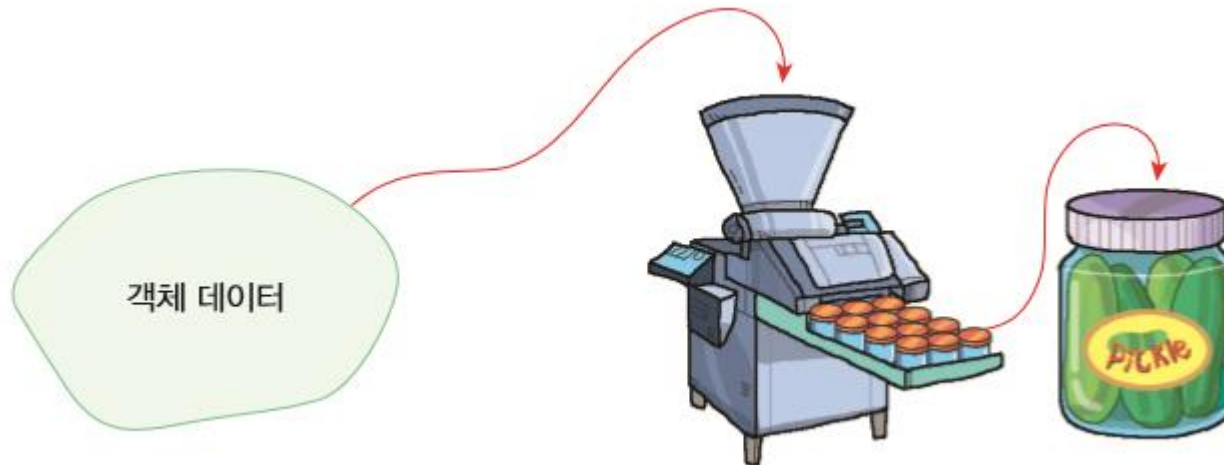
# Solution

```
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()
```

# 객체 입출력

- pickle 모듈의 dump()와 load() 메소드를 사용하면 객체를 쓰고 읽을 수 있다.



# 객체 쓰기

```
import pickle
# 게임에서 사용되는 딕셔너리
gameOption = {
    "Sound": 8,
    "VideoQuality": "HIGH",
    "Money": 100000,
    "WeaponList": ["gun", "missile", "knife" ]
}

# 이진 파일 오픈
file = open( "d:\\save.p", "wb" )
# 딕셔너리를 피클 파일에 저장
pickle.dump( gameOption, file )
# 파일을 닫는다.
file.close()
```

# 객체 읽기

```
import pickle
# 이진 파일 오픈
file = open( "d:\\save.p", "rb" )
# 피클 파일에 딕셔너리를 로딩
obj = pickle.load( open( "save.p", "rb" ) )
print(obj)
```

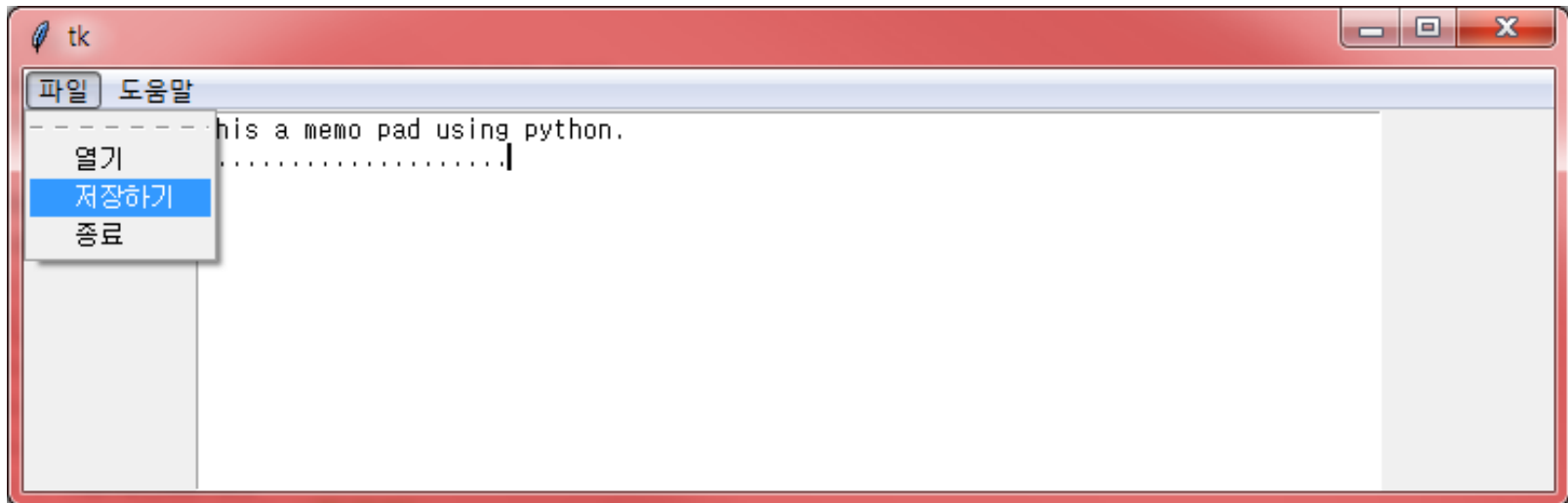
```
{'WeaponList': ['gun', 'missile', 'knife'], 'Money': 100000,
'VideoQuality': 'HIGH', 'Sound': 8}
```



# Lab: 메모장



- 메모장의 기능을 수행하는 애플리케이션을 작성해본다.



```
from tkinter import *

def open():
    file = filedialog.askopenfile(parent=window, mode='r')
    if file != None:
        lines = file.read()
        text.insert('1.0', lines)
        file.close()

def save():
    file = filedialog.asksaveasfile(parent=window, mode='w')
    if file != None:
        lines = text.get('1.0', END+'-1c')
        file.write(lines)
        file.close()

def exit():
    if messagebox.askokcancel("Quit", "종료하시겠습니까?"):
        window.destroy()

def about():
    label = messagebox.showinfo("About", "메모장 프로그램")
```

# Solution

```
window = Tk()
text = Text(window, height=30, width=80)
text.pack()

menu = Menu(window)
window.config(menu=menu)
filemenu = Menu(menu)
menu.add_cascade(label="파일", menu=filemenu)
filemenu.add_command(label="열기", command=open)
filemenu.add_command(label="저장하기", command=save)
filemenu.add_command(label="종료", command=exit)
helpmenu = Menu(menu)
menu.add_cascade(label="도움말", menu=helpmenu)
helpmenu.add_command(label="프로그램 정보", command=about)

window.mainloop()
```

# 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.



# Q & A

