

AIP- HW03

202255595 임영훈

-HW03_A

```
1  import os.path
2
3  fileName = "Names.txt"          #파일이름을 Names.txt로 설정한다
4
5  def main():
6      mySet = readSetFromFile()
7      name = inputName()
8      modifiedSet = insertSet(mySet, name)
9      writeToFile(modifiedSet)
10
11 def readSetFromFile(): # implement functions
12     if(os.path.isfile(fileName)):      #Names.txt라는 파일이 있다면
13         infile = open(fileName, 'r')   #파일을 read모드로 열고
14
15         #파일에 들어 있는 이름들을 끝에 있는 개행문자를 제거하여 set에 넣어준다
16         mySet = {name.rstrip() for name in infile}
17
18         infile.close()                 #파일을 닫는다
19
20         return mySet                  #앞에서 만든 set을 return 한다
21
22     else:                              #만약 Names.txt라는 파일이 없다면
23         print("{} does not exist.".format(fileName)) #파일이 없다는 메시지를 출력하고
24         print("Terminate program.")
25         exit()                        #프로그램을 종료한다.
26
27 def inputName():
28     name = input("Enter a first name to be included: ") #이름을 입력받고
29     return name                                         #입력받은 이름을 return한다
30
31 def insertSet(mySet, name):
32
33     if name in mySet:      #이름이 set에 있다면
34         print("{} is already in {}".format(name, fileName)) #이미 있다는 메시지를 출력한다
35     else:                  #이름이 set에 없다면
36         print("{} is added in {}".format(name, fileName))  #이름을 추가하였다는 메시지를 출력한다.
37         mySet.add(name)   #set에 이름을 추가한다
38
39     return mySet
40
41 def writeToFile(modifiedSet):
42     modifiedList = list(modifiedSet) #입력받은 set을 list로 바꿔준다
43     modifiedList.sort()              #list를 정렬한다.
44     f = open(fileName, 'w')          #파일을 write모드로 연다
45     f.write('\n'.join(modifiedList)) #list에 있는 name들에 개행문자를 붙여 전체 한 문자열로 만들고 파일에 쓴다
46     f.close()                       #파일을 닫는다
47
48 main()
```

- 출력1

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORT
Enter a first name to be included: Grape
Grape is added in Names.txt
PS C:\ATP>
```

-before

```
Names.txt - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V)
Apple
Orange|
```

-after

```
Names.txt - Windows 메모장
파일(F) 편집(E) 서식(O) .
Apple
Grape
Orange
```

- 출력2

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PC
Enter a first name to be included: Grape
Grape is already in Names.txt
PS C:\ATP>
```

-before

```
Names.txt - Windows 메모장
파일(F) 편집(E) 서식(O) .
Apple
Grape
Orange|
```

-after

```
Names.txt - Windows 메모장
파일(F) 편집(E) 서식(O) .
Apple
Grape
Orange
```

- 출력3

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Names.txt does not exist.
Terminate program.
PS C:\ATP>
```

-HW03_B

```
1 filename = "Units.txt"
2
3 def main():
4     feet = populateDictionary()
5     orig, dest, length = getInput()
6     ans = length * feet[orig] / feet[dest]
7     print("Length in {0}: {1:,.4f}".format(dest, ans))
8
9 def populateDictionary():
10    f = open(filename, 'r')    #파일을 read 모드로 연다.
11    feet = {}                 #dictionary하나를 선언한다.
12    for line in f:            #파일을 한줄씩 읽는다
13        data = line.split(',') #콤마로 구분하여 데이터 리스트로 만든다
14        feet[data[0]] = eval(data[1]) #dictionary에 key-value 쌍을 추가해준다.
15
16    f.close()                 #파일을 닫는다.
17    return feet               #새로 만든 dictionary를 return 한다.
18
19 def getInput():
20    orig = input("Unit to convert from: ")    #original unit을 입력받는다
21    dest = input("Unit to convert to: ")      #destination unit을 입력받는다
22    length = eval(input("Enter length in {}: ".format(orig))) #original unit으로 길이를 입력받는다
23
24    return (orig, dest, length)               #입력받은 값들을 return 한다.
25
main()
```

PROBLEMS	OUTPUT	DEBUG CONSOLE	TERMINAL
			Unit to convert from: yard Unit to convert to: mile Enter length in yard: 555 Length in mile: 0.3153 PS C:\AIP>