```
# coding: utf-8

# In[3]:

#open the file
f = open('Alice.txt', 'r')


# In[1]:

#for line in f:
 #   print line


# In[30]:

#bring cursor back to the front of the file
#f.seek(0)



# In[36]:

#Create an empty list to add words in
word_list = []

# split lines into words
for line in f:
    for words in line.split():
#lowercase all words for number counting later, and add all the words to the empty
list
        word_list = word_list + [words.lower()]

#print word_list


# In[41]:

# optional: we can sort out the unique elements in word_list
# word_set = set(word_list)
# print word_set


# In[47]:

# create empty counter
counter = []
# count the occurance of all the elements in word_list
for word in word_list:
    counter= counter + [word_list.count(word)]
#print counter


# In[50]:

# create tuples list for (word, counter)
tuples = zip(word_list, counter)
#print tuples


# In[54]:
```

```
# keep only the unique tuples
unique_tuples = set(tuples)
#print unique_tuples


# In[58]:

# create empty dictionary for later sorting based on dictionary item values
pairs = {}
# add words as the keys and their counters and values
for element in unique_tuples:
    pairs[element[0]]=element[1]
#print pairs


# In[62]:

# sort the dictionary by values in descending orders
import operator
sorted_pairs = sorted(pairs.items(), key=operator.itemgetter(1), reverse = True)
# note that the resulted sequence is a list and not dictionary any more
#print sorted_pairs


# In[64]:

# take the top 20 results for printing
s = sorted_pairs[0:20]
#print s


# In[66]:

# depackaging the tuples and print out the results in the preferred format
for element in s:
    print '%s --- %s'%(element[0],element[1])


# In[34]:

#close the file
# f.close()


# In[ ]:
```