MODULE 12

Compute unique words and their frequency from file

In the previous module (module 11), we compute the unique words and their frequencies for the given sentence. This module is extension for the module 11. In this module, we read the sentence from the file instead of taking from the user through command line.

To solve the above problem we need to use the dictionaries which you already learnt in the previous module. Let's recap the dictionary concepts to solve the given problem.

Dictionary - Important Points

- ✓ Hashes or dictionaries are (key, value) pairs
- ✓ Appropriate for quick lookup/search operations
- ✓ Keys of a dictionary are immutable
- ✓ Duplicate keys are not possible in dictionaries
- ✓ Assigning value to existing key will wipe out the old value

Examples of using Dictionaries

```
>>> tel = {'jack': 4098, 'sape': 4139}
>>> tel['guido'] = 4127
>>> tel
{'sape': 4139, 'guido': 4127, 'jack': 4098}
>>> tel['jack']
4098
>>> del tel['sape']
>>> tel['irv'] = 4127
>>> tel
{'guido': 4127, 'irv': 4127, 'jack': 4098}
>>> tel.keys()
['guido', 'irv', 'jack']
>>> 'guido' in tel
True
```

Looping Techniques in Dictionaries

When looping through dictionaries, the key and corresponding value can be retrieved at the same time using the **`iteritems()`** method.

```
>>> knights = {'gallahad': 'the pure', 'robin': 'the brave'}
>>> for k, v in knights.iteritems():
...     print k, v
...
gallahad the pure
robin the brave
```

When looping through a sequence, the position index and corresponding value can be retrieved at the same time using the **`enumerate()`** function.

```
>>> for i, v in enumerate(['tic', 'tac', 'toe']):
...     print i, v
...
0 tic
1 tac
2 toe
```

To loop over a sequence in reverse, first specify the sequence in a forward direction and then call the **`reversed()`** function.

```
>>> for i in reversed(xrange(1,10,2)):
...     print i
...
9
7
5
3
1
```

Problem

Read a file "list.txt". compute unique words and their frequencies.

Steps to solve the above problem

1. Create an empty dictionary
2. Read word by word (which you already learnt in the previous modules) from the given file "list.txt"
    a. If that word is not there in Dictionary,
        i. Put that word as a key in dictionary and set value as 1
    b. If that word is already there in Dictionary,
        i. Increase the value of that word by 1
3. Repeat the above step until you reach the end of the file
4. Now print all keys and corresponding values of the dictionary