IT 166 Lab 6

Python text files and classes

Objectives

- Be able to create, read and write text files.
- Be able to create and apply Python classes to solve problems.

Preparation

- Launch the Jupyter notebook.
- Rename the notebook page as “lab6”.
- Solution to one problem should occupy one cell.

Please provide solutions to the problems below.

Problem 1

Download the file, “song.txt”, and define a Python class, named “Reverse”. The constructor method of the class takes a file object, created based on the file, as input and assigns the object to self.f. The reverse method will reverse the information read from the file line by line and write the reversed information to a new file, “reversed_song.txt”.

Create an object named, “reverser”, using the class, then use the object to call the reverse method to create the new file.

Hint

- The Reverse class has only two methods, the constructor method and the reverse method.
- Save the downloaded file under your notebook’s directory.

Problem 2

Download the file, “information.txt”, and define a Python class, named “FileParser”. The constructor method of the class takes a file object, created based on the file, as input and assigns the object to self.f. The method also assigns an empty list to self.list and an empty dictionary to self.dict. The parse method uses a loop to read the file line by line and save all the numbers to the list and save all the words into the dictionary. As for each entry in the dictionary, the key is of a word, the value is the number occurrence of the word in the file.

Create an object named, “parser”, using the class, then use the object to call the parse method to fill in the list and the dictionary.

Expect outcome:

```
The list of numbers is: [2006, 4783, 6799]
The dictionary is: {'Since': 1, 'reviews': 2, 'of': 3, 'much': 1, 'work': 2, 'on': 3,
'sentiment': 3, 'analysis': 1, 'have': 1, 'already': 1, 'been': 1, 'included': 1, 'in
': 3, 'this': 1, 'section': 1, 'we': 1, 'will': 1, 'only': 1, 'review': 1, 'some': 2,
'previous': 1, 'upon': 1, 'which': 1, 'our': 1, 'research': 1, 'is': 4, 'essentially':
 2, 'based': 3, 'Hu': 1, 'and': 4, 'Liu': 1, 'summarized': 1, 'a': 7, 'list': 4, 'posi
tive': 3, 'words': 5, 'negative': 3, 'respectively': 1, 'customer': 1, 'The': 1, 'cont
ains': 1, 'the': 2, 'has': 1, 'Both': 1, 'lists': 1, 'also': 1, 'include': 1, 'misspel
led': 1, 'that': 3, 'are': 1, 'frequently': 1, 'present': 1, 'social': 1, 'media': 1,
'content': 2, 'Sentiment': 2, 'categorization': 1, 'classification': 2, 'problem': 1,
'where': 2, 'features': 1, 'contain': 1, 'opinions': 1, 'or': 2, 'information': 1, 'sh
ould': 1, 'be': 1, 'identified': 1, 'before': 1, 'For': 1, 'feature': 1, 'selection':
1, 'Pang': 1, 'Lee': 1, 'suggested': 1, 'to': 2, 'remove': 1, 'objective': 1, 'sentenc
es': 1, 'by': 1, 'extracting': 1, 'subjective': 2, 'ones': 1, 'They': 1, 'proposed':
1, 'text-categorization': 1, 'technique': 1, 'able': 1, 'identify': 1, 'using': 1, 'mi
nimum': 1, 'cut': 1, 'Gann': 1, 'et': 1, 'al.': 1, 'selected': 1, 'tokens': 1, 'Twitte
r': 1, 'data': 1, 'each': 1, 'token': 3, 'assigned': 1, 'score': 1, 'namely': 1, 'Tota
l': 1, 'Index': 1, 'featuring': 1, 'itself': 1, 'as': 1}
```