AC50002

Programming Languages for Data Engineering Python Assignment

Name : Sweta Bhagat

SUID: 2538528

This assignment consists of a Python 3 programming problem.

Code explanation with screenshots:

‘Import re’:

This line imports the regular expression module re, which is used for pattern matching in strings.

A screen shot of a computer

Description automatically generated

Function Definitions:

‘load values(filename)’:

* This function takes a filename as input and attempts to open and read the file.
* This function takes a filename as input and attempts to open and read the file.
* For each line in the file, it splits the line into letter and score using split ().
* It then populates a dictionary (values) with letter as the key and score as the corresponding value.
* It prints an error message and issues the FileNotFoundError if the file cannot be located.
* The function returns the populated values dictionary.

‘calculate\_score(letter, position)’:

* This function calculates the score for a given letter based on its position in a word.
* If the position is 0, it returns 0.
* If the position is 1, it uses the get method to retrieve the score from the values dictionary. If the letter is not found, it defaults to 0.
* If the position is 2, it adds 3 to the score.
* For any other position, it adds 6 to the score.
* The function returns the calculated score.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

‘process\_name(name)’:

* Using a given name as input, this function generates scores and abbreviations according to defined rules.
* It finds every word in the input name using regex (re.findall()), raises it, and adds it to the words list.
* It initializes an empty set (abbreviations) to store unique abbreviations.
* It then iterates through each word in the words list and, for each letter in the word (except the last one), calculates the score using the calculate\_score function.
* It uses the initial letter, the current letter, and the following letter in the word to create an abbreviation.
* The abbreviation and its score are added to the abbreviations set.
* The function returns the set of unique abbreviations.

Main Function (main):

* The main function is the entry point of the program.
* It creates the output filename using the given format after receiving user input for the input filename.
* To load letter scores from the "values.txt" file, it calls the load\_values function.
* If there is an issue loading values, it prints an error message and exits the program.
* It then attempts to open the input file, reads the names, and initializes an empty dictionary (results) to store final abbreviations and their scores.
* After calling process\_name for scores and abbreviations, looping through each name, updates the results dictionary.
* Finally, it writes the sorted results to an output file and prints a message indicating the successful completion.

A screenshot of a computer

Description automatically generated

Execution:

* After asking the user for input, the program carries out the primary task.
* It handles the case where the input file is not found by catching the FileNotFoundError and printing an error message.

Output :

The output message "Results written to bhagat\_trees\_abbrevs.txt" indicates that the program has successfully processed the input file "trees.txt" and generated a file named "bhagat\_trees\_abbrevs.txt" containing the results.

User Input:

* Here, the program prompted to enter the input filename with a .txt extension where I entered "trees.txt".
* Then it asked for the surname, and i entered "bhagat".

File Processing:

* The program read the content of the input file "trees.txt" and processed the names within it.
* For each name, it generated abbreviations according to the specified rules and calculated scores for those abbreviations.

A screenshot of a computer program

Description automatically generated

Result:

* All names were included in the program's score list for distinct abbreviations.
* It stored these results in the results dictionary, ensuring that it kept the abbreviation with the lowest score for any duplicates.

Generating Output Files:

* The software generated "bhagat\_trees\_abbrevs.txt" as the output filename by combining the input filename with your surname.
* It then wrote the sorted results to this output file, with each line containing an abbreviation and its score.

A white rectangular object with a blue border

Description automatically generated