Programming Languages for Data Engineering

Python Assignment

Name: Inam Abdul Ghafar

ID: 2495895

About Python:

In terms of syntax, Python has always been known as a simple programming language. It prioritises readability and provides clutter - free, easy-to-learn syntax. Furthermore, the Python style guide, PEP 8, includes a set of rules to help with code formatting. Python can easily meet this requirement. Because it is a general programming language, it can generate CSV output for easy data interpretation in a spreadsheet. Python is not only multifunctional, but it is also lightweight and efficient when it comes to code execution. It can be used anywhere because it supports object-oriented, structural, and functional programming styles.

Therefore it provides very easy solutions for completing the assignment.

Files:

As per asignment we have two files first one consisting of tree names and the second one was values.

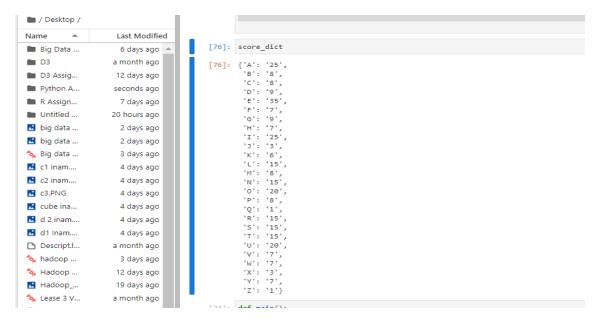
Sor first of all I have opened the files in a python3 tool .the code is given bellow.

```
with open("trees (1).txt","r") as name_f:
    name_file =name_f.readlines()
with open('values (1).txt', 'r') as val_f:
    file_ofValue = val_f.readlines()
```

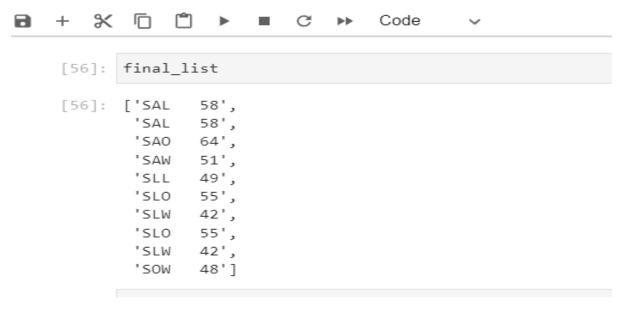
```
ALD ALE ALR ADE ADR AER
Last Modified
                                                         [88]: name_file
      6 days ago
   a month ago
                                                         [88]: ['Alder\n',
'Crab Apple\n',
'Common Ash\n',
'Silver Birch\n',
'Downy Birch\n',
    12 days ago
   seconds ago
       aconds ago
7 days ago
a day ago
2 days ago
2 days ago
3 days ago
4 days ago
      7 days ago
                                                                              'European Beech\n',
'Box\n',
'Wild Cherry\n',
'Bird Cherry\n',
      2 days ago
      2 days ago
                                                                              'Blackthorn\n',
'Wych Elm\n',
'Smooth-leaved Elm\n',
'Common Hawthorn\n',
      3 days ago
      4 days ago
      4 days ago
                                                                               'Midland Hawthorn\n',
                                                                             'Midland Hawthorn\n',
'Common Hazel\n',
'European Hornbeam\n',
'European Holly\n',
'Common Juniper\n',
'Small-leawed Lime\n',
'Field Maple\n',
'Pedunculate Gak\n',
'Sessile Gak\n',
'Scots Pine\n',
'Aspen\n',
'Black Poplar\n',
'European Rowan\n',
'Common Whitebeam\n',
      4 days ago
      4 days ago
      4 days ago
      4 days ago
  a month ago
      3 days ago
    12 days ago
    19 days ago
   a month ago
                                                                             'European Rowan\n',
'Common Whitebeam\n',
'Service Tree\n',
'Wild Service Tree\n',
'Strawberry Tree\n',
'Bay Willow\n',
'Crack Willow\n',
'White Willow\n',
'Almond-leaved Willow\n',
'European Yew\n',
'Alder Buckthorn\n',
'Purging Buckthorn\n',
      2 days ago
     2 days ago
 a month ago
  a month ago
  a month ago
     4 days ago
  a month ago
                                                                              'Purging Buckthorn\n',
'Elder\n',
'Common Dogwood\n',
'Rock Whitebeam\n',
                                                                              "Rock Whitebeam\n',
"Sea-buckthorn\n',
"Spindle\n',
"Sallow\n',
"Grey Willow\n',
"Purple Willow\n',
"Common Osier\n',
"Eared Willow\n',
"Guelder Rose\n',
"Wayfaring tree\n',
"Common Privet\n',
"Plot's Elm"]
```

Score:

Actually when I was starting to work assignment I use jupyter node book when I finished assignment then I use VS Code for python so here is the screen shot of the sore out put which is in dictionry.



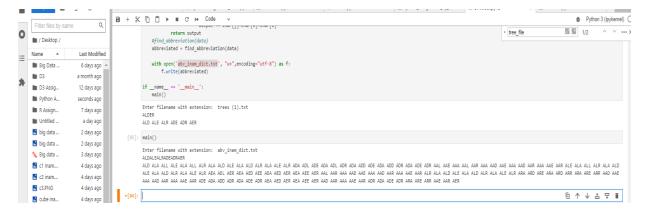
Here is the abbreviation of the out put bellow.



Here is the out put of just one word Alder which is the first word in tree text file you can see bellow more than one abbreviation has been created.similarly I can apply so on in the text file. Each abbreviation is given a score which indicates how good it is.

Enter the your file then it will the out of abbreviation like as bellow.

```
🗞 Hadoop ...
                    12 days ago
                                         [86]: main()
Hadoop_...
                   19 days ago
🍫 Lease 3 V...
                   a month ago
                                               Enter filename with extension: trees (1).txt
                                               ALDER
Mitigatin...
                    2 days ago
                                               ALD ALE ALR ADE ADR AER
Mitigatin...
                    2 days ago
MongoD...
                   a month ago
  4.4
```



Code of Main file:

I have created many functions and also I have created two main function this code is for taken the file then it will create the Upper level of the abbreviation. This code I use in VS Code as well as in Jupiter notebook .I have attached the python code file you can see easily.

```
def main():

#prompt user to input file
filename = input('Enter filename with extension: ')

try:

| try: | with open(filename, 'r', encoding="utf-8") as f:
| data = f.read() |
| data = data.replace("-", "") |
| data = data.replace("-", "") |
| data = data.strip() |
| with open("trees (1).txt", 'r') as scores:
| score = scores.readlines() |
| # print("Remove Punc: ', filename) |
| except FileNotFoundError: | print("File Not Found") |
| data = data.replace("-") |
| data = data.strip() |
| with open("trees (1).txt", 'r') as scores:
| score = scores.readlines() |
| # print("Remove Punc: ', filename) |
| except FileNotFoundError: | print("File Not Found") |
| data = data.strip() |
| # print("File Not Found") |
| data = data.strip() |
| # print("Remove Punc: ', filename) |
| except FileNotFoundError: | print("File Not Found") |
| data = data.strip() |
| # print("Remove Punc: ', filename) |
| # print("File Not Found") |
| data = data.strip() |
| # print("File Not Found") |
| data = data.strip() |
| # print("Remove Punc: ', filename) |
| # print("File Not Found") |
| # print("Remove Punc: ', filename) |
| # print("Remove Punc: ', filename
```

```
return dicts
ညို
                         def find_abbreviation(string):
                               Generatd three accronym from string by iterating over the string
                               for count, char in enumerate(string.splitlines()):
if count == 1:
continue
                                     print(char)
                                     for i in char:
    if i != None:
        continue
print(i, end='')
                                     for j in range(0,len(char)):
    if j != 0:
        continue
                                           continue
for k in range(j+1,len(char)):
    for l in range(k+1,len(char)):
        print(char[j]+char[k]+char[1], end=' ')
        output += char[j]+char[k]+char[1]
                         return output
#find_abbreviation(dat
                         abbreviated = find_abbreviation(data)
                         with open('abv inam_dict.txt', "w+",encoding="utf-8") as f:
    f.write(abbreviated)
                   if <u>name</u>
main()
```

I have more output but it will mad long file so I have attached code you can check all the output yourself.

Thank you