

**Name: Akanksha Tiwari**

**Student ID: 2506868**

**Course: Programming Languages for Data Engineering**

**Course ID: AC50002**

# REPORT

## Link of GitHub:

[https://github.com/BeginnerPythonEnthusiast/Python\\_Assignment\\_Akanksha](https://github.com/BeginnerPythonEnthusiast/Python_Assignment_Akanksha)

1. We have taken the input as the file trees.txt and the values.txt file which contain the names and the score associated with the letters.
2. Then the data has been cleaned as we removed any unwanted characters like "-", " " etc.
3. The score for different letters and the other rules has been implemented in below code.

```
def cost(abb, name_list):  
    first, last, score = "", "", 0  
    for i in name_list:  
        if len(i) > 1:  
            last += i[-1]
```

4. Then we create the abbreviation of three words where all the abbreviations should have first letter in them and if not then we ignore those abbreviations.

```
# Creating the abbreviations from the names  
def abbrev(name_list):  
    name = "".join(name_list)  
    n = len(name)  
    abb = dict()  
    for i in range(1,n):  
        for j in range(i+1,n):  
            temp = name[0]+name[i]+name[j]
```

5. Since the abbreviation should be such that it should be unique and shall not be existing in any of the other words' abbreviations, we will make sure of that through the code.
6. In input we will provide the file trees.txt so that I can be evaluated, and all the functions created can be implemented on that.
7. The result gets are stored in the **Final\_Result.txt** file

```
# Saving our final result in different file named as Final_Result.txt
for i in range(len(mylist)):
    for j in range(i + 1, len(mylist)):
        temp = mylist[i].intersection(mylist[j])
        if len(temp)>0:
            common+=temp
out = open("Final_Result.txt","w")
```

8. Result we got on Console:

Thus, output matches our expected result as it follows all the rules mentioned

```
Success
{'Alder': '', 'Crab Apple': 'CAA ', 'Common Ash': 'CAS ', 'Silver Birch': 'SVB SRH ', 'Downy Birch': 'DYB DBH ', 'European
Beech': 'EBH ', 'Box': 'BOX ', 'Wild Cherry': 'WCY ', 'Bird Cherry': 'BDC BCY ', 'Blackthorn': 'BKN ', 'Wych Elm': 'WEM ',
'Smooth leaved Elm': 'SHL SHE ', 'Common Hawthorn': 'CHH ', 'Midland Hawthorn': 'MHH ', 'Common Hazel': 'CHZ ', 'European H
ornbeam': 'EEM EHN EHM ', 'European Holly': 'EHY ', 'Common Juniper': 'CNJ CJN CJR ', 'Small leaved Lime': 'SDI ', 'Large l
eaved Lime': 'LLL ', 'Field Maple': 'FDM ', 'Pedunculate Oak': 'POK ', 'Sessile Oak': 'SSS SSO ', 'Scots Pine': 'SSP ', 'As
pen': 'APN ', 'Black Poplar': 'BPP ', 'European Rowan': 'EWN ', 'Common Whitebeam': 'CWM ', 'Service Tree': 'SVT ', 'Wild S
ervice Tree': 'WST ', 'Strawberry Tree': 'STT ', 'Bay Willow': 'BWW ', 'Crack Willow': 'CCW CWw ', 'White Willow': 'WwW ',
'Almond leaved Willow': 'ALL ALW AAW AAL AWW ', 'European Yew': 'EYW ', 'Alder Buckthorn': 'ARB ABR ABN ', 'Purging Bucktho
rn': 'PGB PNB PBN ', 'Elder': 'EDE ', 'Common Dogwood': 'CDD ', 'Rock Whitebeam': 'RKW RWM ', 'Sea buckthorn': 'SBN ', 'Spi
ndle': 'SPD ', 'Sallow': 'SLW ', 'Grey Willow': 'GWW ', 'Purple Willow': 'PPW PWW ', 'Common Osier': 'CSR ', 'Eared Willow'
: 'EWW ', 'Guelder Rose': 'GRR ', 'Wayfaring tree': 'WGT ', 'Common Privet': 'CPT ', 'Plot s Elm': 'PSE '}
```