**Assignment - AC50002 - Programming Languages for Data Engineering**

***Python Assignment 1***

Student ID: 2499976

1. First of all, I read the assignment file to understand the problem.
2. Then I identified that the following actions must be done in order to solve the problem and then created functions for the relevant action required to perform each task:
   1. Run the code, request for input and return the result - main()
   2. Create abbreviations - abbreviations(newstr)
   3. Remove duplicate abbreviations - removeduplicates()
   4. Remove unwanted characters from the tree names list - normalisedata(a1)
   5. Assign values to the letter in alphabet - assignlettervalues(filename)
   6. Calculate values of abbreviations - calculateabbvalues(abbval)
3. I used PyCharm as the development IDE.
4. The code is uploaded in Github:

Github URL: <https://github.com/MoonlightO2/Python/blob/main/Peiris%20-%20Assignment01.py>

| import re  import os  #Main function  def main():  tree = input("Enter a name of a tree: ")  print("Assign letter values:")  x = assignlettervalues("values.txt")  print(x)  print("Remove unwanted characters:")  with open('trees.txt', 'r') as treenames:  normalisedata(treenames)  print("Creating abbreviations:")  with open('file1.txt', 'r') as abbs:  for abbvs in abbs:  abbreviations(abbvs)  # Write abbreviations (without duplicates) into new file  fileop1 = open('Peiris\_trees\_abbrevs.txt', 'a')  fileop1.write(abbval)  fileop2 = open('Peiris\_trees\_abbrevs.txt', 'r')  print(fileop2.read())  #Creating abbreviations  def abbreviations(newstr):  print(newstr)  """Print all 3-element subsequences of a string."""  for i in range(len(newstr)):  for j in range(i + 1, len(newstr)):  for k in range(j + 1, len(newstr)):  m = newstr[i] + newstr[j] + newstr[k]  mm = m.strip()  if len(mm) == 3:  if mm[0] == newstr[0]:  print(mm)  print()  #Remove duplicate abbreviations  def removeduplicates():  mylist = ["a", "b", "a", "c", "c"]  mylist = list(dict.fromkeys(mylist))  print(mylist)  #Remove unwanted characters ("-", "'", " ")  def normalisedata(a1):  os.remove("file1.txt")  for a2 in a1:  #Convert to upper case  a3 = a2.upper()  #print(a3)  #Remove unwanted character "-"  a4 = a3.replace("-", " ")  # Remove unwanted character "'"  a5 = a4.replace("'", " ")  # Remove unwanted character " "  a6 = a5.replace(" ", "")  a6 = a6.strip()  #print(a6)  with open("file1.txt", "a") as a7:  a7.write(a6)  #Assign values for letters in value.txt file  def assignlettervalues(filename):  with open(filename, 'r') as modules:  return dict(map(lambda x: x.rstrip().split(maxsplit=1), modules))  #Assign values for abbreviations  def calculateabbvalues(abbval):  print("Assign values to letters")  with open('file1.txt', 'r') as file1:  lettervalue = assignlettervalues("values.txt")  print(lettervalue)  for i in j:  sss |
| --- |