**Department :** Computer Engineering

**Class :** SE

**Subject :** Data Structure Lab

**Name:** Kartiki Uday Khare.

**Roll no:** 21494

**Batch:** H4

**Assignment No : 02**

* **Problem Statement:**

Write a Python program to compute following operations on String:

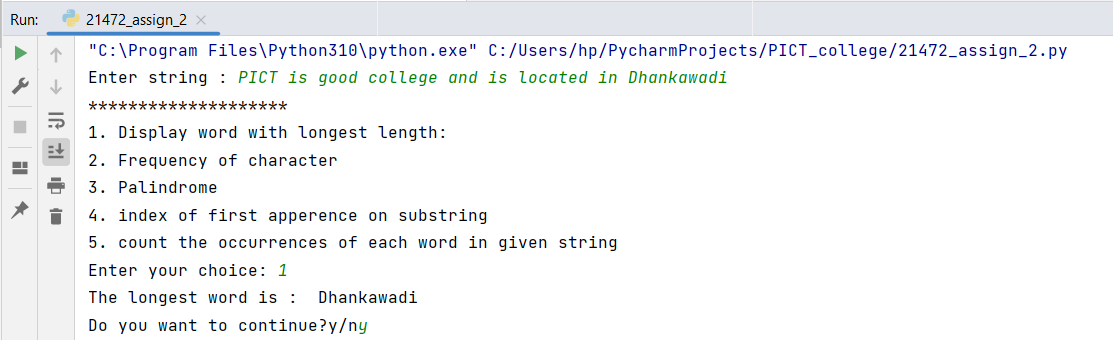
1. To display word with the longest length
2. To determines the frequency of occurrence of particular character in the string
3. To check whether given string is palindrome or not
4. To display index of first appearance of the substring
5. To count the occurrences of each word in a given string

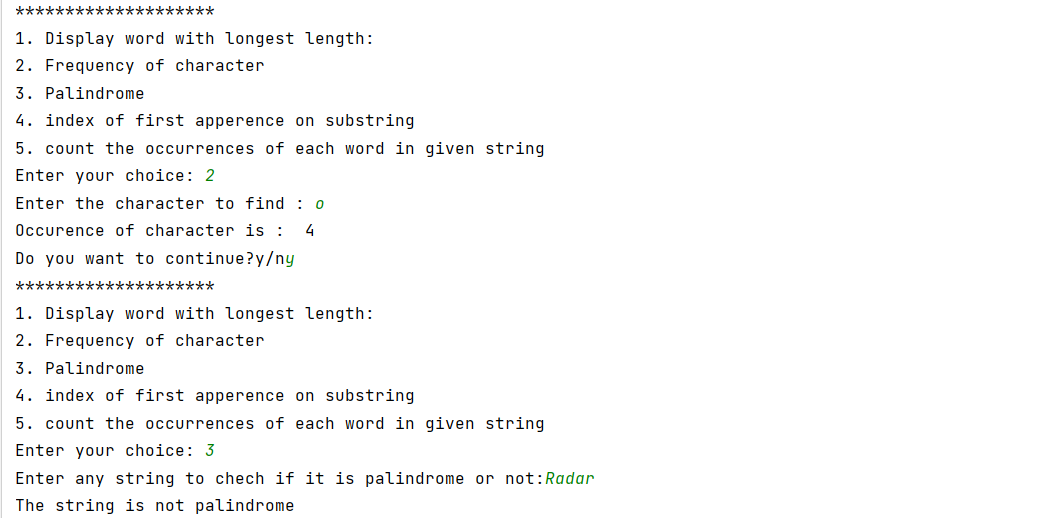
(Does not use string built-in functions)

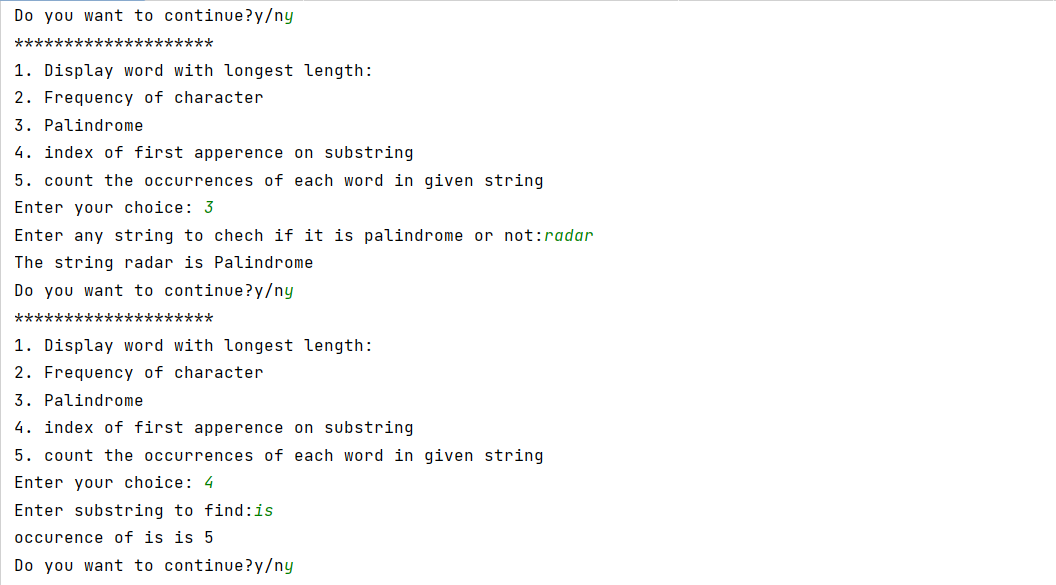
* **Code:**

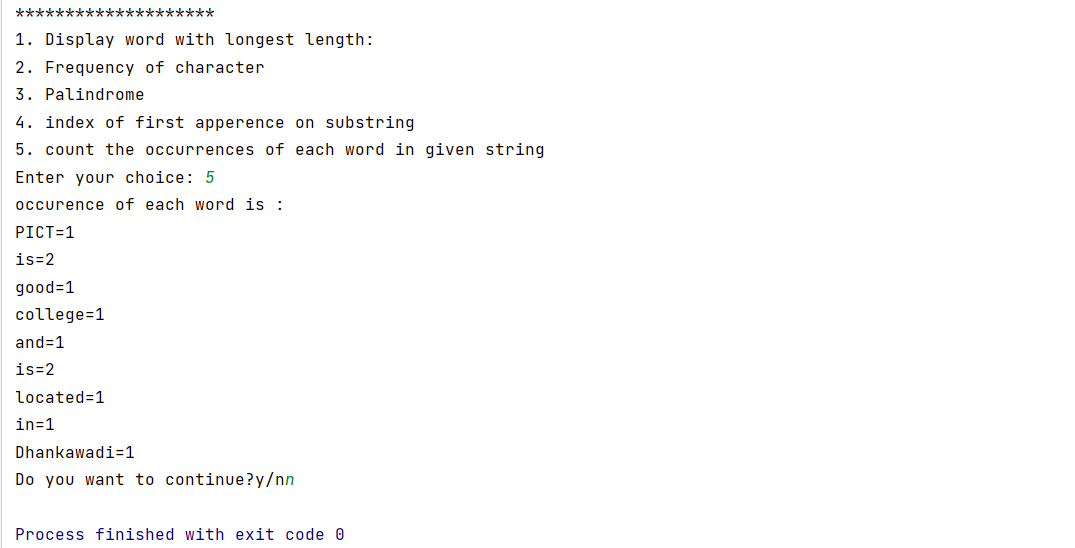
|  |
| --- |
| class assignment\_2: def split(self,a,delimeter=' \t\n'): split\_value = []  temp = '' *# splitting string and storing it in form of list* for c in a: if c == ' ': *# checking space between words in string to split them(delimiter used id space)* split\_value.append(temp)  temp = '' else:  temp += c if temp: split\_value.append(temp) return split\_value  def findlength(self,a):  c = 0 for iin a:  c += 1 return c  def longest\_word(self,a): split\_list= [] split\_list = self.split(a)  length = 0 word = 0 for iin split\_list:  n = self.findlength(i) if(n>length):  length = n  word = i return word  def frequency(self,a,b):  c = 0 for iin a: if i == b:  c += 1 return c   def palindrome(self):  c = 0 flag = 0 str2 = input("Enter any string to chech if it is palindrome or not:") for iin str2:  c += 1 for j in range(c): if (str2[j] != str2[c - j - 1]):  flag = 1 break  if (flag == 0): print("The string",str2, "is","Palindrome") else: print("The string is not palindrome")  def substring1(self,a,sub):  c = 0 e = 0 j = 0 index1 = 0 for l in a:  c += 1 for k in sub:  e += 1 for iin range(c): if (a[i] != sub[0]): continue  else:  d = i for j in sub: if (a[d] == j):  d = d + 1 else: break  if (d - e == i): return i else: continue    def each\_word\_occur(self,a): c = 0 v = 0 size = 0 result = []list1 = self.split(a) for j in list1: for kin list1: if (j == k):  v = v + 1 p = "=" q = str(v)  r = j + p + q result.append(r)  v = 0 for iin result: print(i)  def assignment2(self):  a = input("Enter string : ") while True: print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*") print("1. Display word with longest length:") print("2. Frequency of character") print("3. Palindrome") print("4. index of first apperence on substring") print("5. count the occurrences of each word in given string")  n = int(input("Enter your choice: ")) if (n == 1): print("The longest word is : ", self.longest\_word(a)) if (n == 2):  b = input("Enter the character to find : ") print("Occurence of character is : ", self.frequency(a, b)) if (n == 3): self.palindrome() if (n == 4):  sub = input("Enter substring to find:") print("occurence of", sub, "is", self.substring1(a, sub)) if (n == 5): print("occurence of each word is :") self.each\_word\_occur(a) user\_input = input("Do you want to continue?y/n") if user\_inputin 'Yy': continue  else: break object = assignment\_2() object.assignment2() |

* **Output:**

****

****

****

****