**Python Data Type: String**

Name: Aleksandra Jaworska

Album no. : 162589

Group: WAW\_2023\_L\_N\_I\_INF7\_C4

Student email: [wwx19038@student.warszawa.merito.pl](mailto:wwx19038@student.warszawa.merito.pl)

GitHub link:

Structure of task presentation:

* The content of the task
* Theoretical issues
* Code
* Code result

1. Write a Python program to calculate the length of a string.

Def - Define a function named string\_length that takes one argument, str.

String – a data type used to store a string of characters (string variables)

str = input("type something: ")

def string\_length(str):

count = 0

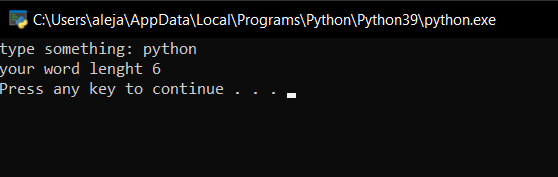
for char in str:

count += 1

return count

print("your word lenght", (string\_length(str)))

OUTCOME:



1. Write a Python program to count the number of characters (character frequency) in a string.

str = input("type someting: ")

def char\_frequency(str):

dict = {}

for n in str:

keys = dict.keys()

if n in keys:

dict[n] += 1

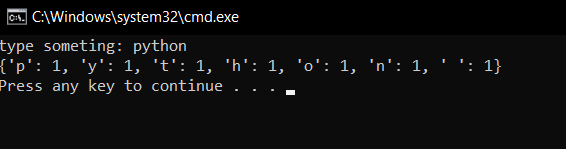
else:

dict[n] = 1

return dict

print(char\_frequency(str))

OUTCOME:



1. Write a Python program to get a string from a given string where all occurrences of its first char have been changed to '$', except the first char itself.

str = "restart"

def change\_char(str):

char = str[0]

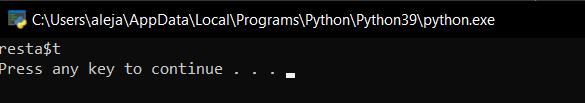
str = str.replace(char, '$')

str = char + str[1:]

return str

print(change\_char(str))

OUTCOME:



1. Write a Python program to get a single string from two given strings, separated by a space and swap the first two characters of each string

words = input("Please enter two words separated by a space: ")

def chars\_mix\_up():

a, b = words.split()

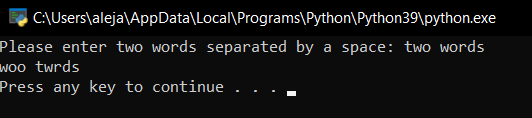
new\_a = b[:2] + a[2:]

new\_b = a[:2] + b[2:]

return new\_a + ' ' + new\_b

print(chars\_mix\_up())

OUTCOME:



1. Write a Python program to add 'ing' at the end of a given string (length should be at least 3). If the given string already ends with 'ing', add 'ly' instead. If the string length of the given string is less than 3, leave it unchanged.

str1 = input("type str: ")

def add\_string(str1):

length = len(str1)

if length > 2:

if str1[-3:] == 'ing':

str1 += 'ly'

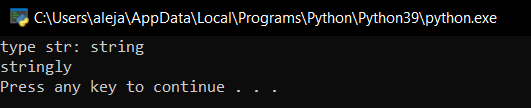
else:

str1 += 'ing'

return str1

print(add\_string(str1))

OUTCOME:



1. Write a Python function that takes a list of words and return the longest word and the length of the longest one.

words = input("Please enter a list of words, separated by spaces: ")

def find\_longest\_word():

words\_list = words.split()

word\_len = []

for n in words\_list:

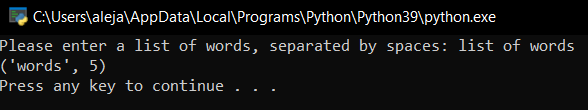
word\_len.append((len(n), n))

word\_len.sort()

return word\_len[-1][1], word\_len[-1][0]

print(find\_longest\_word())

OUTCOME:



1. Write a Python program to change a given string to a newly string where the first and last chars have been exchanged.

str = input("type someting: ")

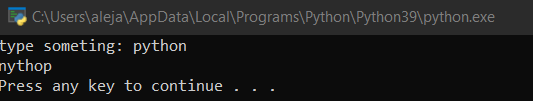
def change\_str():

new\_str = str[-1] + str[1:-1] + str[0]

return new\_str

print(change\_str())

OUTCOME:



1. Write a Python program to remove characters that have odd index values in a given string.

str = input("type someting: ")

def remove\_odd\_index\_chars():

result = ""

for i in range(len(str)):

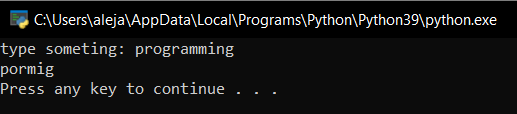
if i % 2 == 0:

result = result + str[i]

return result

print(remove\_odd\_index\_chars())

OUTCOME:



1. Write a Python program to count the occurrences of each word in a given sentence.

sentence = input("Please enter a list of words, separated by spaces: ")

def word\_count():

counts = dict()

words = sentence.split()

for word in words:

if word in counts:

counts[word] += 1

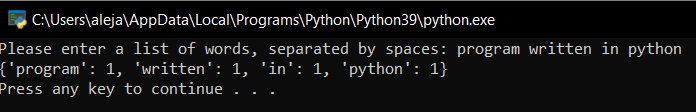
else:

counts[word] = 1

return counts

print(word\_count())

OUTCOME:



1. Write a Python script that takes input from the user and displays that input back in upper and lower cases.

str = input("write something: ")

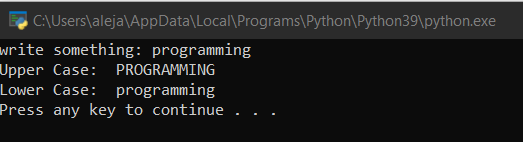
def change\_case():

print("Upper Case: ", str.upper())

print("Lower Case: ", str.lower())

change\_case()

OUTCOME:



1. Write a Python function to get a string made of 4 copies of the last two characters of a specified string (length must be at least 2).

str = input("write something: ")

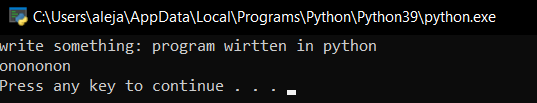
def insert\_end():

sub\_str = str[-2:]

return sub\_str \* 4

print(insert\_end())

OUTCOME:



1. Write a Python function to get a string made of the first three characters of a specified string. If the length of the string is less than 3, return the original string.

str = input("write something: ")

def get\_first\_three\_chars():

if len(str) > 2:

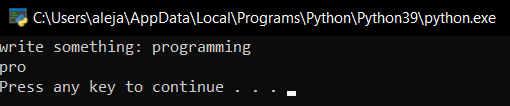
return str[:3]

else:

return str

print(get\_first\_three\_chars())

OUTCOME:



1. Write a Python function to reverse a string if its length is a multiple of 4.

str = input("write something: ")

def reverse\_string():

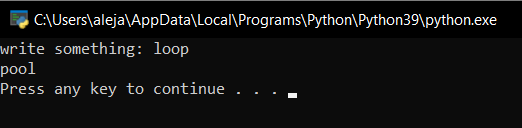
if len(str) % 4 == 0:

return str[::-1]

return str

print(reverse\_string())

OUTCOME:



1. Write a Python function to convert a given string to all uppercase if it contains at least 2 uppercase characters in the first 4 characters.

str = input("write something: ")

def to\_uppercase():

uppercase\_chars = [char for char in str[:4] if char.isupper()]

if len(uppercase\_chars) >= 2:

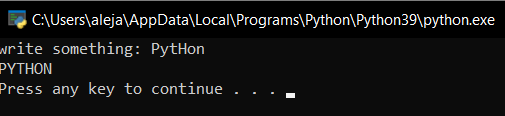
return str.upper()

else:

return str

print(to\_uppercase())

OUTCOME:



1. Write a Python program to sort a string lexicographically.

str = input("write something: ")

def to\_uppercase():

uppercase\_chars = [char for char in str[:4] if char.isupper()]

if len(uppercase\_chars) >= 2:

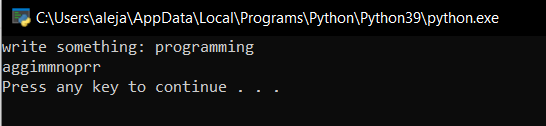
return str.upper()

else:

return str

print(to\_uppercase())

OUTCOME:



1. Write a Python program to print the following numbers up to 2 decimal places.

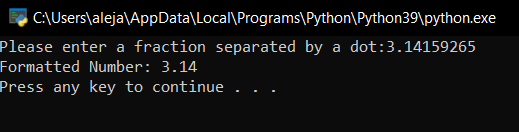
num = float(input('Please enter a fraction separated by a dot:'))

def print\_number():

print("Formatted Number: {:.2f}".format(num))

print\_number()

OUTCOME:



1. Write a Python program to print the following positive and negative numbers with no decimal places.

numbers = input("Please enter a list of numbers, separated by spaces: ")

def print\_numbers():

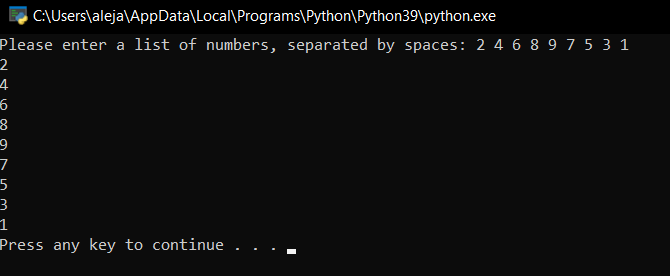
numbers\_list = numbers.split()

for num in numbers\_list:

print(int(float(num)))

print\_numbers()

OUTCOME:



1. Write a Python program to format a number with a percentage.

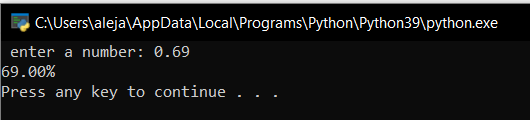
number = float(input(" enter a number: "))

def format\_percentage():

return "{:.2%}".format(number)

print(format\_percentage())

OUTCOME:



1. Write a Python program to reverse a string.

str = input("enter a string: ")

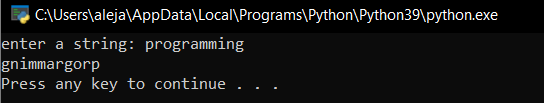
def reverse\_string():

reversed\_s = str[::-1]

return reversed\_s

print(reverse\_string())

OUTCOME:



1. Write a Python program to lowercase the first n characters in a string.

s = input("Please enter a string: ")

n = int(input("Please enter the number of characters to lowercase: "))

def lowercase\_first\_n\_chars():

new\_s = s[:n].lower() + s[n:]

return new\_s

print(lowercase\_first\_n\_chars())

OUTCOME:

