Lab Assignment - 2

**GITHUB REPOSITORY**: <https://github.com/daksharma3163/Assignment-2>

# Q1:

names = ['Daksh Sharma','Saurabh Kumar','Aditya Singh','Prajjwal Arya','Prateekasd','AdityaRana','HarshwardhanSingh','AyushRaj','Junaid ','Ashok Kumar']

# part a

print("PrintingFirstnames=>") for name in names:

first\_name=name.split('')[0] print(first\_name,end=', ')

# part b

print("\nPrintingLastnames=>") for name in names:

last\_name=name.split('')[-1] print(last\_name,end=', ')

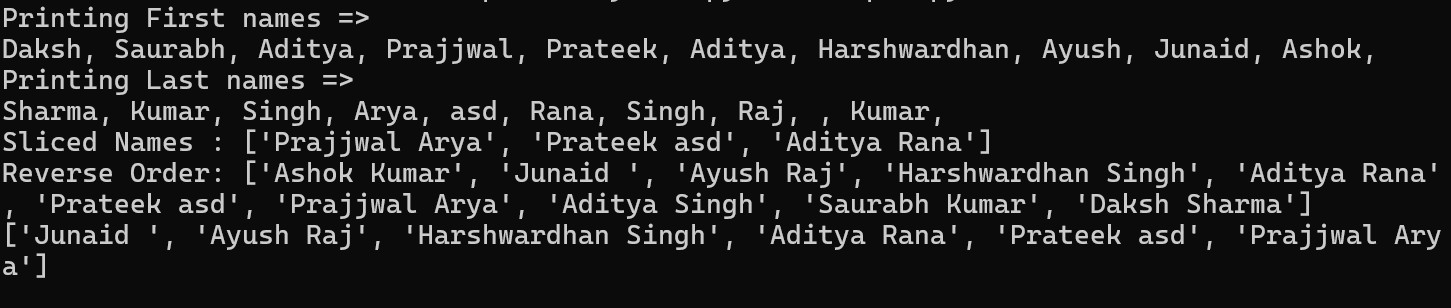
# part c

sliced\_names = names[3:6] print(f"\nSlicedNames:{sliced\_names}")

# part d

reverse\_order = names[::-1] print(f"ReverseOrder:{reverse\_order}")

# part e

sliced\_reverse=names[8:2:-1] print(sliced\_reverse)

# Q2:

names=('DakshSharma','SaurabhKumar','AdityaSingh','Prajjwal Arya','Prateek')

# part a

for name in names: print(name,end=',')

# part b

name=input("\nEnteranewname:") names = names + (name, ) print(names)

# part c

name\_list = list(names) name\_list.remove('Prateek') names = tuple(name\_list) print(names)

# part d

print(names[1:4])

# part e

new\_list = list(names)

name = input("Enter a name : ") print("Aftermodifyingsecondindex=>") new\_list[2] = name

names=tuple(new\_list) print(names)

# Q3:

students = { "Alice": 19,

"Bob": 22,

"Charlie": 21,

"David": 18

}

# part a

print("Students older than 20:") forname,ageinstudents.items():

if age > 20: print(f"{name}:{age}")

# part b students["Eve"]=30

print("\nAdded Eve with age 30.")

# part c

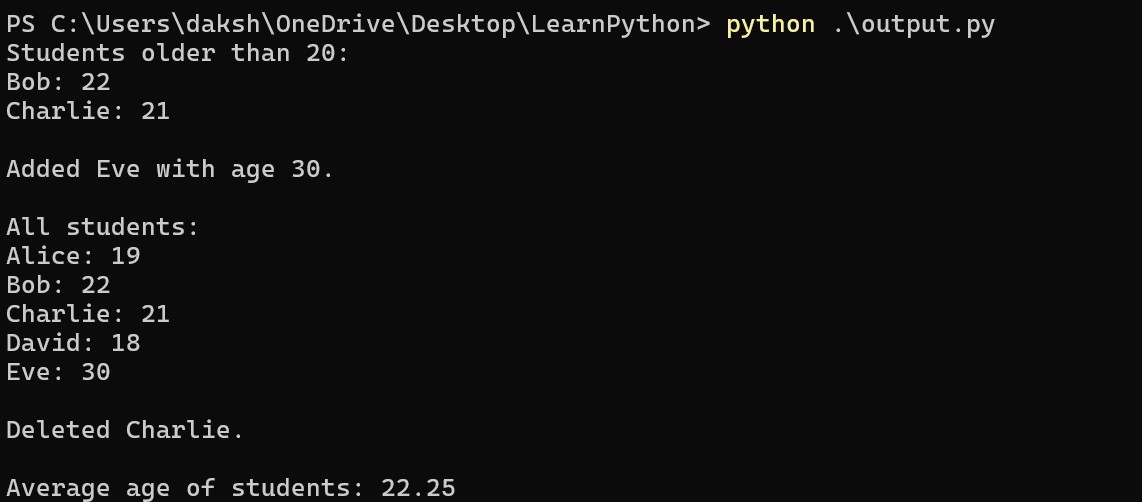
print("\nAll students:")

forname,ageinstudents.items(): print(f"{name}: {age}")

# part d

del students["Charlie"] print("\nDeletedCharlie.")

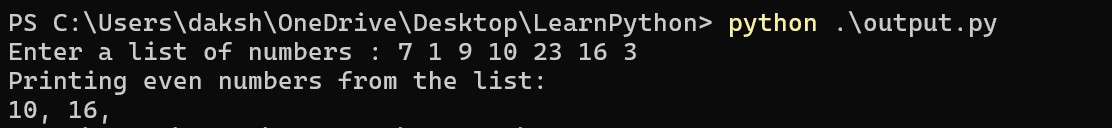
# part e

average\_age=sum(students.values())/len(students) print(f"\nAverage age of students: {average\_age:.2f}")

# Q4:

user\_list=input("Enteralistofnumbers:") numbers = list(map(int, user\_list.split())) print("Printingevennumbersfromthelist:")

foriinnumbers: if i % 2 == 0:

print(i,end=', ')

# Q5:

user\_list=input("Enteralistofnumbers:") numbers = list(map(int, user\_list.split()))

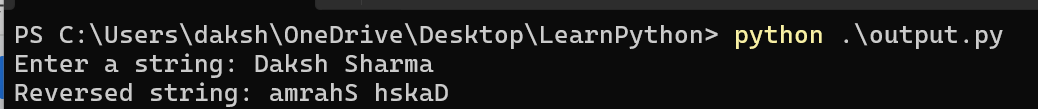
duplicates = []

for num in numbers:

ifnumbers.count(num)>1andnumnotinduplicates: duplicates.append(num)

print(duplicates)

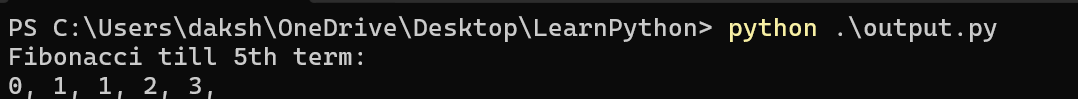
# Q6:

user\_input = input("Enter a string: ") reversed\_string = user\_input[::-1] print("Reversedstring:",reversed\_string)

# Q7:

a,b = 0,1

print("Fibonaccitill5thterm:") for i in range(500):

print(a,end=',') a,b = b, a + b

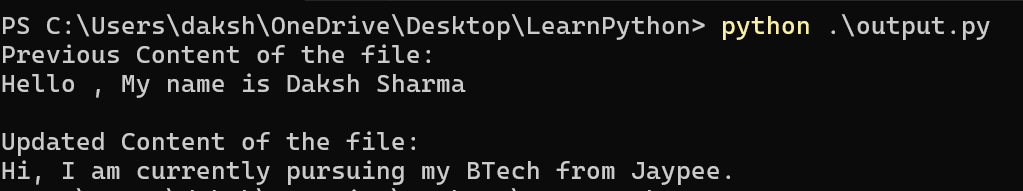
# Q8:

withopen("content.txt","r+")asfile: content = file.read()

print(f"PreviousContentofthefile:\n{content}") file.seek(0)

file.write("Hi,IamcurrentlypursuingmyBTechfromJaypee.") file.truncate()

withopen("content.txt","r")asfile: updated\_content = file.read()

print(f"\nUpdated Content of the file:\n{updated\_content}")

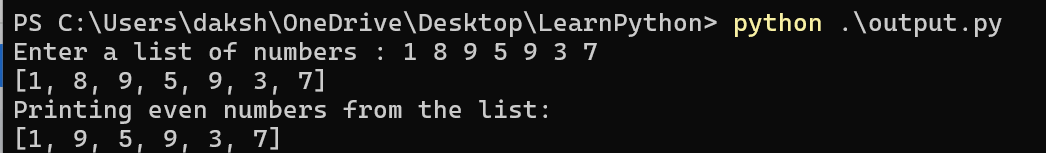
# Q9:

with open('content.txt','r') as file: content = file.readlines() print(f"PreviousContent\n{content}") for line in reversed(content):

print(line.rstrip())

# Q10:

user\_list=input("Enteralistofnumbers:") numbers = list(map(int, user\_list.split())) print(numbers)

print("Printingevennumbersfromthelist:") odd\_list = [x for x in numbers if x % 2 != 0] print(odd\_list)

# Q11:

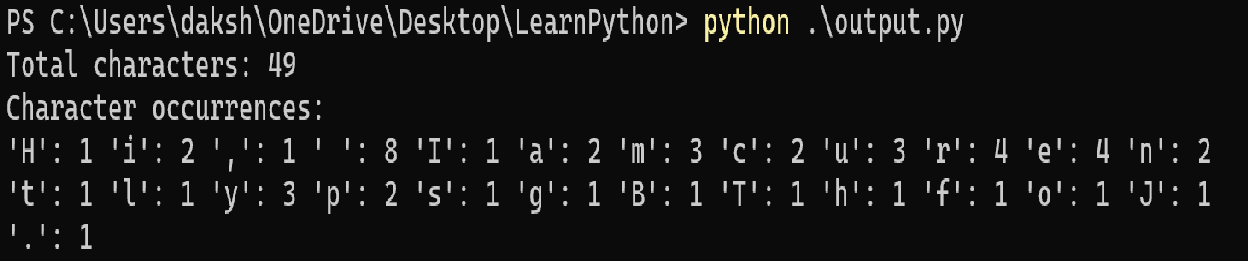
withopen("content.txt",'r')asfile: content = file.read() char\_count = {}

for char in content:

char\_count[char] = char\_count.get(char, 0) + 1

print(f"Totalcharacters:{len(content)}") print("Character occurrences:")

forchar,countinchar\_count.items(): print(f"'{char}': {count}",end='')



# Q12:

fromcollectionsimportdefaultdict def group\_anagrams(words):

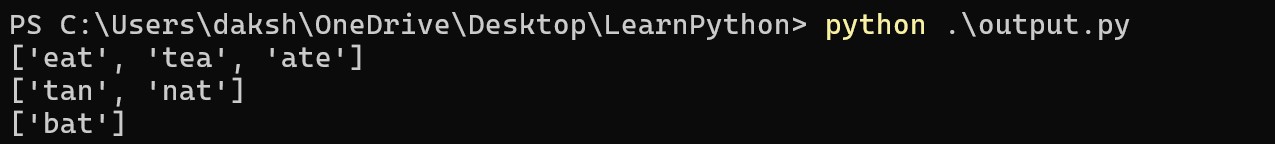
anagram\_map = defaultdict(list)

for word in words:

sorted\_word = ''.join(sorted(word)) anagram\_map[sorted\_word].append(word)

return list(anagram\_map.values()) word\_list=['eat','tea','tan','ate','nat','bat']

anagram\_groups = group\_anagrams(word\_list)

forgroupinanagram\_groups: print(group)

**Extra Assignment – 2**

# Q1:

def group(lst, size):

result = []

for i in range(0, len(lst),size):

part = lst[i:i+size]

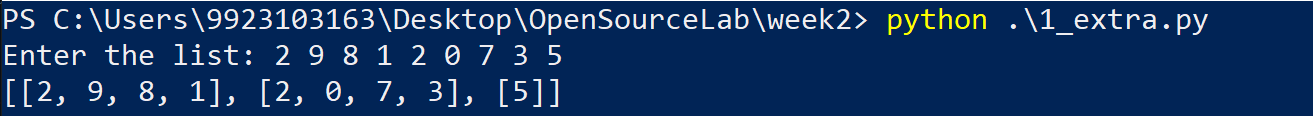
result.append(part)

return result

user\_list = input("Enter the list: ")

user\_list = list(map(int, user\_list.split()))

print(group(user\_list,4))



# Q2:

# def lensort(lst):

# return sorted(lst,key=len)

# def extsort(file\_list):

# return sorted(file\_list, key=lambda x: x.split('.')[-1] if '.' in x else '')

# user\_list = input("Enter the string list: ")

# user\_list = list(map(str,user\_list.split()))

# print(lensort(user\_list))

# files = ["report.docx", "data.csv", "image.png", "notes.txt", "archive.zip"]

# print(extsort(files))

# 

# Q3:

# with open('example.txt', 'w') as file:

# file.write("This is the first line.\n")

# file.writelines(["This is the second line.\n", "This is the third line.\n"])

# with open('example.txt', 'r') as file:

# content = file.read()

# print("Content using read():")

# print(content)

# with open('example.txt', 'r') as file:

# print("Content using readline():")

# line = file.readline()

# while line:

# print(line, end='')

# line = file.readline()

# 

# Q4:

def file\_stats(filename):

with open(filename, 'r', encoding='utf-8') as f:

lines = f.readlines()

num\_lines = len(lines)

num\_words = sum(len(line.split()) for line in lines)

num\_chars = sum(len(line) for line in lines)

return num\_chars, num\_words, num\_lines

filename = "example.txt"

chars, words, lines = file\_stats(filename)

print(f"Characters: {chars}, Words: {words}, Lines: {lines}")

