**3**

**[Based on Python Data Types (String, List and Tuple]**

**Q1: Write a Python program to sum all the items in a list.**

def sumOflist(list1):

sum=0

for i in list1:

sum=sum+i

return sum

list=[1,3,5,7]

sumOflist(list)

o/p: 16

**Q2: Write a Python program to get the largest number from a list.**

def largestlist(list1):

largest=list1[0]

for i in range(len(list1)):

if (largest<list1[i]):

largest=list1[i]

return largest

list=[-5,-9,-5,-7,0]

largestlist(list)

o/p: 0

**Q3: Write a Python program to get the smallest number from a list.**

def smallestlist(list1):

smallest=list1[0]

for i in range(len(list1)):

if (smallest>list1[i]):

smallest=list1[i]

return smallest

list=[-5,-9,-5,-7,0]

smallestlist(list)

o/p: -9

**Q4: Write a Python program to display the first and last colors from the following list.**

color\_list = ["Red","Green","White" ,"Black"]

def first\_last(list):

print(list[0],list[len(list)-1])

color\_list = ["Red","Green","White" ,"Black"]

first\_last(color\_list)

o/p: Red Black

**Q5: Write a Python program to check if the input year is a leap year or not.**

**#year should be divisible by 400 and 100**

**#if year divisible by 4 and not divisible by 100**

def leapyear(year):

if(year%400==0 and year%100==0 ):

print(f"{year} is a leap year")

elif(year%4==0 and year%100!=0):

print(f"{year} is a leap year")

else:

print(f"{year} is not a leap year")

y=int(input("Enter year: "))

leapyear(y)

Enter year: 2021

2021 is not leap year

**Q6: write a Program to check if a string is palindrome or not**

def palindrome(s):

reverse=s[: :-1]

print(f"reverse of entered string {s} is {reverse}")

if(s==reverse):

print(f"so {s} is palindrome")

else:

print(f"so {s} is not palindrome")

string=input("Enter string: ")

palindrome(string)

Enter string: radar

reverse of entered string radar is radar

so radar is palindrome

**Q7: Given a nested list. Write a python program to extend it with adding sub list ["h", "i", "j"] in a such a way that it will look like the following list**

Given List:

list1 = ["a", "b", ["c", ["d", "e", ["f", "g"], "k"], "l"], "m", "n"]

Sub List to be added = ["h", "i", "j"]

Expected output:

['a', 'b', ['c', ['d', 'e', ['f', 'g', 'h', 'i', 'j'], 'k'], 'l'], 'm', 'n']

list1 = ["a", "b", ["c", ["d", "e", ["f", "g"], "k"], "l"], "m", "n"]

Sublist= ["h", "i", "j"]

**list1[2][1][2].extend(Sublist)**

list1

o/p: ['a', 'b', ['c', ['d', 'e', ['f', 'g', 'h', 'i', 'j'], 'k'], 'l'], 'm', 'n']

**Q8: Write a python program for Given a Python list, to find value 20 in the list, and if it is present, replace it with 200. Only update the first occurrence of a value**

list1 = [5, 10, 15, 20, 25, 50, 20]

Expected output:

list1 = [5, 10, 15, 200, 25, 50, 20]

list1[list1.index(20)]=200

list1

[5, 10, 15, 200, 25, 50, 20]

**Q9: Rotate a List Right by N Positions. Given a list L1= [10, 20, 30, 40, 50] and an integer n=2, rotate L1 to the right by n positions.**

Expected output: L1= [40, 50, 10, 20, 30]

def rightRotate(list,n):

difference=len(list)-n

list=list[difference:len(list)]+list[0:difference]

print(list)

L1= [10, 20, 30, 40, 50]

rightRotate(L1,2)

o/p: [40, 50, 10, 20, 30]