**F. Y. B. Tech Academic Year 2021-22**

**Subject:** Programming and Problem Solving **Trimester:** 2

**Name:** Shreerang Mhatre  **Division:** 11

**Roll No:** 111056  **Batch:** K3

**ASSIGNMENT NO: 9**

**PROBLEM STATEMENT:** Write a python program to accept n numbers and sort them in

ascending and descending order.

**OBJECTIVES:** To study sort and reverse function in python

**THEORY:**

**List:**

A list is a collection which is ordered and changeable. In Python lists are written with square brackets. The list is a most versatile datatype available in Python which can be written as a list of comma-separated values (items) between square brackets. Important thing about a list is that items in a list need not be of the same type.

**Sort () :**

Sort () Method. sort () is an inbuilt method in Python, it is used to sort the elements/objects of the list in Ascending and Descending Order. To sort a list in descending order, we pass reverse=True as an argument with sort () method.

**Implementation:**

**Platform: 64 –**bit Windows 10**.**

**Technology:** Anaconda Navigator ( Jupyter Notebook )

**Algorithm:**

**Step1) Take a list of n numbers from user**

**Step2) Create a sorted copy of existing list using sort ()**

**Step3) Sort the List in Place**

**Step4) print the List which is ascending order.**

**Step5) Create a sorted copy of existing list**

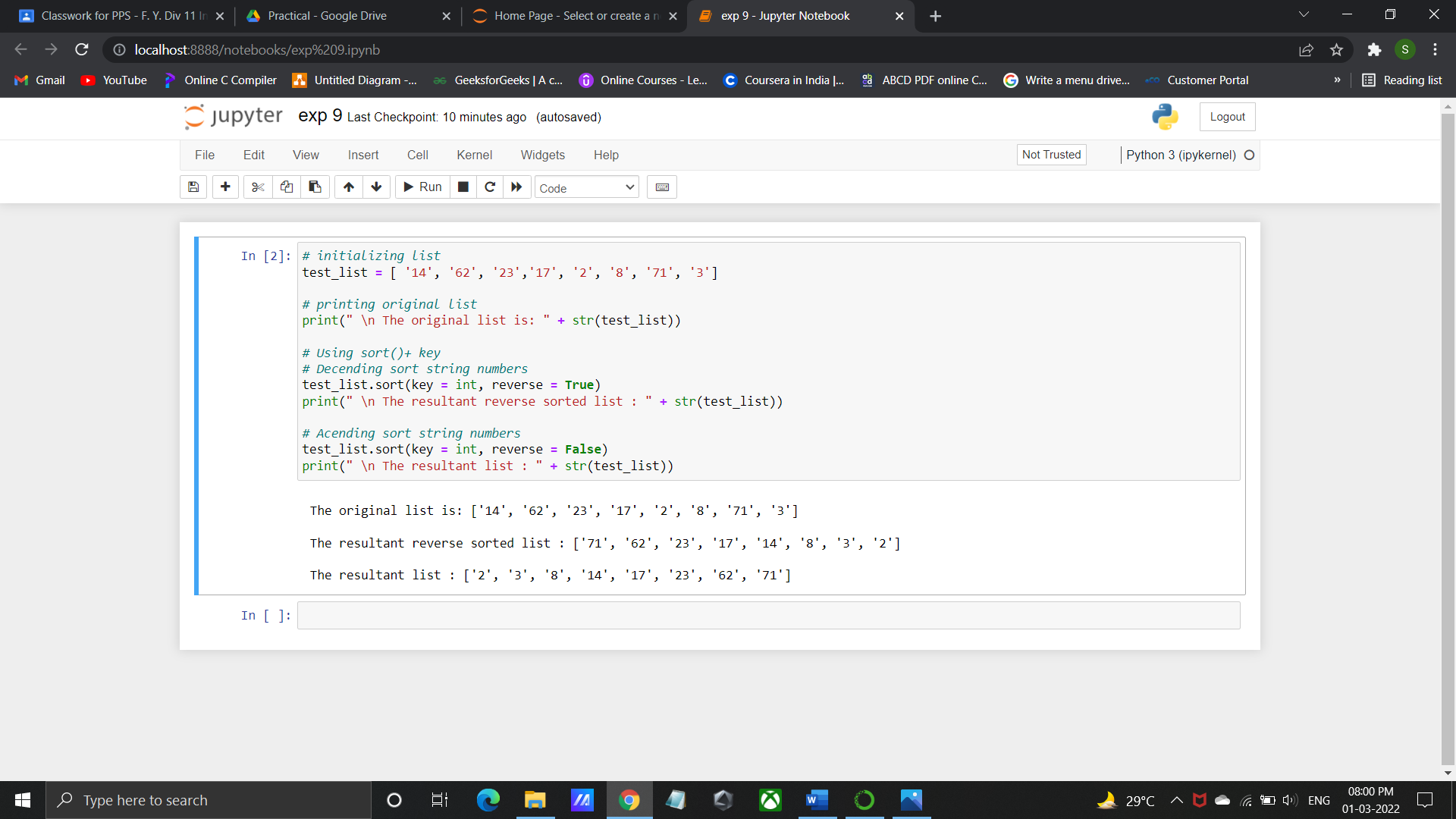
**Step6) Create sorted list in reverse order by setting reverse equal to True**

**Step7) Print the List in descending order**

**Step8) Stop**

**Anaconda Navigator:**

**Jupyter Notebook:**

****

**Program/ Code:**

# initializing list

test\_list = [ '14', '62', '23','17', '2', '8', '71', '3']

# printing original list

print(" \n The original list is: " + str(test\_list))

# Using sort()+ key

# Decending sort string numbers

test\_list.sort(key = int, reverse = True)

print(" \n The resultant reverse sorted list : " + str(test\_list))

# Acending sort string numbers

test\_list.sort(key = int, reverse = False)

print(" \n The resultant list : " + str(test\_list))

**INPUT:** [ '14', '62', '23','17', '2', '8', '71', '3']

**OUTPUT:**

The original list is: ['14', '62', '23', '17', '2', '8', '71', '3']

The resultant reverse sorted list: ['71', '62', '23', '17', '14', '8', '3', '2']

The resultant list: ['2', '3', '8', '14', '17', '23', '62', '71']

**CONCLUSION:** Thus understood the concept of inbuilt function sort.

**FAQ’s:**

1. What is a List

ANS: A list is a data structure in Python that is a mutable, or changeable, ordered sequence of elements. Each element or value that is inside of a list is called an item. Just as strings are defined as characters between quotes, lists are defined by having values between square brackets [ ]

1. Explain sort ()

ANS: sort () Method. sort () is an inbuilt method in Python, it is used to sort the elements/objects of the list in Ascending and Descending Order. To sort a list in descending order, we pass reverse=True as an argument with sort () method.

1. Explain reverse ()

ANS: Python reversed () method returns an iterator that accesses the given sequence in the reverse order.

e.g., INPUT: vowels = ['a', 'e', 'i', 'o', 'u']

print(list(reversed(vowels)))

OUTPUT: ['u', 'o', 'i', 'e', 'a']

1. What are different methods available in python?

ANS: Generally, there are three types of methods in Python:

1. Class Variable: A class variable is nothing but a variable that is defined outside the

constructor. A class variable is also called as a static variable.

2. Accessor (Getters): If you want to fetch the value from an instance variable, we call them accessors.

3. Mutator (Setters): If you want to modify the value, we call them mutators.