**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **02-07-2020** | | | | **Name:** | | | **RAVALI P** | |
| **Sem & Sec** | **6th -Sem ‘B’ Sec** | | | | **USN:** | | | **4AL17CS076** | |
| **Online Test Summary** | | | | | | | | | |
| **Subject** | | **-** | | | | | | | |
| **Max. Marks** | | **-** | | | | | **Score** | **-** | |
| **Pre-Placement Training Summary** | | | | | | | | | |
| **Course** | **-** | | | | | | | | |
| **Faculty** | | | **-** | | | **Duration** | | | **-** |
| **Coding Challenges** | | | | | | | | | |
| **Problem Statement:** 1. Write a program that will read a sequence of positive real numbers entered by the user and will print the same numbers in sorted order from smallest to largest. The user will input a zero to mark the end of the input. Assume that at most 100 positive numbers will be entered. | | | | | | | | | |
| **Status: Executed** | | | | | | | | | |
| **Uploaded the report in Github** | | | | **Yes** | | | | | |
| **If yes Repository name** | | | | <https://github.com/Ravalipr/Daily-Status.git> | | | | | |
| **Uploaded the report in slack** | | | | **Yes** | | | | | |

**Coding Challenge:**

1. Write a program that will read a sequence of positive real numbers entered by the user and will print the same numbers in sorted order from smallest to largest. The user will input a zero to mark the end of the input. Assume that at most 100 positive numbers will be entered.

import java.util.\*;

public class Main{

public static void main(String[] args) {

Scanner sc= new Scanner(System.in);

double[] numbers;

int numCt;

double num;

numbers = new double[100];

numCt = 0;

System. out. println("Enter up to 100 positive numbers; Enter 0 to end");

while (true) {

System. out. println("? ");

num = sc.nextInt();;

if (num <= 0)

break;

numbers[numCt] = num;

numCt++;

}

selectionSort(numbers, numCt);

System. out. println("\nYour numbers in sorted order are:\n");

for (int i = 0; i < numCt; i++) {

System. out. println( numbers[i] );

}

}

static void selectionSort(double[] A, int count) {

for ( int lastPlace = count - 1; lastPlace > 0; lastPlace-- ) {

int maxLoc = 0;

for (int j = 1; j <= lastPlace; j++) {

if (A[j] > A[maxLoc]) {

maxLoc = j;

}

}

double temp = A[maxLoc];

A[maxLoc] = A[lastPlace];

A[lastPlace] = temp;

}

}

}

**Output:**

