CECS 174 – LAB ASSIGNMENT 6

OBJECTIVES:

- Able to use a Python 3.x IDE to build Python program(s)
- Implement a solution that requires **tuple and list.**
- Implement a solution that use **def keyword, (create functions)**
- Write Python code following an algorithm.
- Form a sophisticated expression in Python

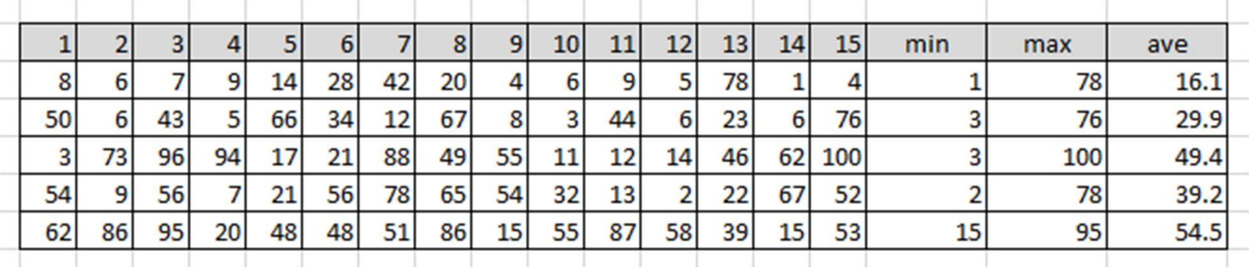
INSTRUCTIONS:

PART 1 – CREATE FUNCTION

Write a program that accept input of 10 numbers and then report the maximum value, minimum value and the average value. In this program, you will have 2 functions: main and display-stats.

Main function will ask the user to input 10 numbers. Then it will call display-stats function. The display-stats will calculate and display the maximum value, minimum value and the average value.

Use the following data to test your app

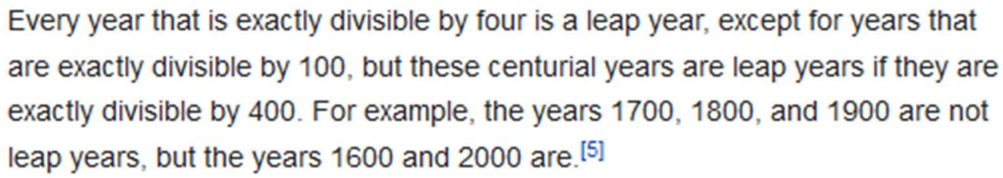


1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	min	max	ave
8	6	7	9	14	28	42	20	4	6	9	5	78	1	4	1	78	16.1
50	6	43	5	66	34	12	67	8	3	44	6	23	6	76	3	76	29.9
3	73	96	94	17	21	88	49	55	11	12	14	46	62	100	3	100	49.4
54	9	56	7	21	56	78	65	54	32	13	2	22	67	52	2	78	39.2
62	86	95	20	48	48	51	86	15	55	87	58	39	15	53	15	95	54.5

You must write your own code to find min, max and average values. You cannot use Python provided function to calculate these values. You don't have to turn in pseudocode for this part.

PART 2 DETERMINE LEAP YEAR

Our current calendar is the Gregorian calendar and it began in 1582. The leap year of Gregorian calendar is as follows



> Every year that is exactly divisible by four is a leap year, except for years that are exactly divisible by 100, but these centurial years are leap years if they are exactly divisible by 400. For example, the years 1700, 1800, and 1900 are not leap years, but the years 1600 and 2000 are.[5]

You are tasked to design an algorithm and write a python application to determine if a year is a leap year.

You can collaborate with other classmates to design an algorithm. This time, you will indicate who are part of your algorithm team.

Create at least 4 functions: main, display_output, accept_input, leap_year_verifier.

1. Main function is the first function that runs. And,
2. it will call the other three functions to help so that it can determine if a year is a leap year.

A run can be like the following run:

```
Welcome to Leap Year Verifier

Please enter a year that is 1582 or later, (0 to stop): 2018
2018 is not a leap year

Please enter a year that is 1582 or later, (0 to stop): 2020
2020 is a leap year

Please enter a year that is 1582 or later, (0 to stop): 1900
1900 is not a leap year

Please enter a year that is 1582 or later, (0 to stop): 2000
2000 is a leap year

Please enter a year that is 1582 or later, (0 to stop): 1997
1997 is not a leap year

Please enter a year that is 1582 or later, (0 to stop): 0
Thank You for using the Leap Year Verifier.
```

You will need to **turn in Pseudocode of Part 2**. Make sure you run your program with the **test year shown above**.

For Your Information

Make sure you comment your code. Also Design an algorithm with pseudocode before you write your Python code.

Turn in

- Turn in **your code and images of your test runs**, (i.e. several runs) in **1 single PDF document**.
- Your document with **have 2 parts because this assignment has 2 parts**.
- Your turn in document must be in **PDF format.** Upload to the BeachBoard account of this class.