– KINDLY DON’T USE BUILT-IN METHODS FOR 1-10 Task –

1. **Store Data:**

Use a list or array to store the data sample: 15, -9, 0, 45, 87, 98, 23, 56.

1. **Print Method:**

Create a method called PrintData which prints the data sample to the console.

1. **Sort Method:**

Create a method called SortData which takes the data, sorts it, and returns the sorted data. Take an extra optional param of type char whose value could be A or D. If the value is A then sort it in ascending order; if D descending, Its by default value would be D

1. **IsItemFound Method:**

Create a method IsItemFound which takes the data and an integer entry, returning true if the entry is found and false otherwise.

1. **Search Method:**

Create a method SearchData which takes the data and an integer parameter. It uses the IsItemFound method to check if the item exists. If it does, print the index where the item is found; otherwise, print a message indicating that the item was not found.

1. **Reverse Data Method:**

Create a method ReverseData which takes the data and returns it in reverse order.

1. **Find Maximum Method:**

Create a method FindMax which finds and returns the maximum value in the data without using built-in methods.

1. **Find Minimum Method:**

Create a method FindMin which finds and returns the minimum value in the data without using built-in methods.

1. **Compute Average Method:**

Create a method ComputeAverage which computes and returns the average of the data values.

1. **Count Occurrences Method:**

Create a method CountOccurrences which takes the data and an integer entry and returns the number of times the entry appears in the data.

– BUILT-IN METHODS ARE ALLOWED FOR THESE 5 TASKS –

1. **Find Median Method:**

Create a method FindMedian that sorts the list and finds the median value, which is the middle element in an odd-length list, or the average of the two middle elements in an even-length list.

1. **Standard Deviation Method:**

Create a method ComputeStandardDeviation that calculates the standard deviation of the list, giving you an idea of how spread out the numbers are from the average.

1. **Find Unique Values Method:**

Create a method FindUniqueValues that returns a list of unique values in the data, removing any duplicates.

1. **Mode Method:**

Create a method FindMode that finds the most frequently occurring number in the list.

1. **Sort Two Lists Method:**

Create a method Sort that takes two lists, sorts them, and returns a sorted list

1. **Merge Two Lists Method:**

Create a method Merge that takes two lists, merges them into one, and returns the sorted merged list.

**Additional Instructions**

* Create a new GitHub repository and push the code there.
* Add comments for each method
* Give names properly