**Assignment**

**3.       We would prefer to see three implementations (one that should take into consideration #4 below) and an explanation of what use-cases are suitable to each implementation**

1. Remove Duplicates using List

This use case is for the first implementation which uses list to remove duplicates

1. Remove Duplicates using Set

This use case is for the second implementation which uses set to remove duplicates

1. Remove Duplicates using Arrays

This use case is for the third implementation which uses arrays to remove duplicates

**5.       What are the positives and negatives of your solution?**

**a.       Can you implement it another way so as to avoid the negatives?**

1. Method 1 – Removing duplicates using list

Positives – It adds only the unique values from original list and returns the array back to the main method

Negatives – Each every value is traversed and checked whether it’s available in the List already

1. Method 2 – Remove duplicates using set

Positives – Set is faster than Arraylist. It does not allow duplicates to be added to the list

Negatives – No negatives. Hence prefer this way of implementing for this scenario.

1. Method 3 – Remove duplicates using arrays

Positives - No Positives

Negatives – Traversal and swap each and every element to sort the array elements. After sorting the duplicate elements has to be removed. It involves more code and becomes complex with less readability.