**JAVA PROGRAMS**

**//Find the duplicates in an arary**

**import** java.util.ArrayList;

**import** java.util.HashSet;

**import** java.util.List;

**import** java.util.Set;

**public** **class** Numberdulpicate {

**public** **static** **void** main(String[] args){

**int**[] arr= {4,3,2,7,8,2,3,1};

**for**(**int** i=0;i<arr.length;i++){

System.***out***.print(" "+arr[i]);

}

System.***out***.println();

Set<Integer> hash= **new** HashSet<Integer>();

List<Integer>list=**new** ArrayList<Integer>();

**for**(**int** i=0;i<arr.length;i++){

**if**(! hash.add(arr[i])){

list.add(arr[i]);

}

}

System.***out***.println(list);

}

}

Output

4 3 2 7 8 2 3 1

[2, 3]

//Merge and Sort without Duplicates

**import** java.util.HashSet;

**import** java.util.TreeSet;

//Merge and Sort without Duplicates

**public** **class** MergeAndSort {

**public** **static** **void** main( String[] args){

**int** [] arr1={22,3,6,1,8,4,9};

**int**[] arr2={3,5,1,7,5,8,9};

TreeSet<Integer> tree=**new** TreeSet<Integer>();

**for**(**int** i=0;i<arr1.length;i++){

tree.add(arr1[i]);

}

**for**(**int** i=0;i<arr2.length;i++){

tree.add(arr2[i]);

}

System.***out***.println(tree);

Output

[1, 3, 4, 5, 6, 7, 8, 9, 22]

**//String Anagram**

**import** java.util.Arrays;

**import** java.util.Scanner;

//Anagram examples silent and listen,read and dear

**public** **class** StringAnagramShuffle {

**public** **static** **void** main(String[] args){

System.***out***.println("enter first string to be checked");

Scanner sc= **new** Scanner(System.***in***);

String a=sc.nextLine();

System.***out***.println("enter second string to be checked");

String b=sc.nextLine();

sc.close();

**char**[] a1= a.toCharArray();

**char**[] b1=b.toCharArray();

*sort1*(a1);

*sort1*(b1);

//Arrays.sort(a1);

//Arrays.sort(b1);

**if**(Arrays.*equals*(a1, b1)){

System.***out***.println("Strings are Anagrams");

}

**else**{

System.***out***.println("Strings are not Anagrams");

}

}

**public** **static** **void** sort1(**char** a[]){

**for**(**int** i=0;i<(a.length-1);i++){

**for**(**int** j=0;j<(a.length-1);j++){

**if**((a[j])>(a[j+1])){

**char** temp=a[j];

a[j]=a[j+1];

a[j+1]=temp;

}

}

}

}

}

Output

enter first string to be checked

silent

enter second string to be checked

listen

Strings are Anagrams

//Sorting Negative nos without taking the signs and putting back once sorting is done

**public** **class** SortingNegativeNos {

**public** **static** **void** main(String[] args){

**int** arr[]={2,10,-5};

//output = {-1,1,2,4,-6,8}

**int** l=arr.length;

System.***out***.println("UnSorted array");

**for**(**int** i=0;i<l;i++){

System.***out***.println(arr[i]);

}

**for**(**int** i=0;i<l;i++){

**for**(**int** j=i+1;j<l;j++){

**if** (Math.*abs*(arr[i])>Math.*abs*(arr[j])){

**int** temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

}

}

}

System.***out***.println("Sorted array");

**for**(**int** i=0;i<l;i++){

System.***out***.println(arr[i]);

}

}

}

UnSorted array

2

10

-5

Sorted array

2

-5

10