Name of the Course : Complete Java SE8 Developer Bootcamp

Level : Difficult

Tool Stack : Java8 and Junit5

Problem Statement : Provide a code solution to calculate the symmetric difference of two given arrays and sort the same.

Symmetric Difference is the difference of A Union B and A intersection B ie.[(AUB)-(A^B)]

Union operation merges the two arrays and makes sure that common elements appear only once.

Intersection operation includes common elements from both the arrays.

Ex - A ={12,24,7,36,14} and B={11,26,7,14}

Description : Create class MainClass with static method getSymmetricDifference(Integer[] a1,Integer [] b1) which accepts the two integer array.The return type is an integer array.

2. pubic static void main method, for reading the two integer arrays and call the static method.

**Sample Input 1:**  
5  
11  
5  
14  
26  
3  
3  
5  
3  
1  
**Sample Output 1:**  
1  
11  
14  
26

Code:

**import** java.util.ArrayList;

**import** java.util.Scanner;

**import** java.util.TreeSet;

**public** **class** SymmetricDifference {

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

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

**int** n1=sc.nextInt();

**int**[] a=**new** **int**[n1];

**for**(**int** i=0;i<n1;i++)

a[i]=sc.nextInt();

**int** n2=sc.nextInt();

**int** [] b= **new** **int**[n2];

**for**(**int** i=0;i<n2;i++)

b[i]=sc.nextInt();

**int**[] res=*getSymmetricDifference*(a,b);

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

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

}

**public** **static** **int**[] getSymmetricDifference(**int** a[],**int** b[])

{

TreeSet<Integer> ts1=**new** TreeSet<Integer>();

TreeSet<Integer> ts2=**new** TreeSet<Integer>();

TreeSet<Integer> ts3=**new** TreeSet<Integer>();

ArrayList<Integer> aa=**new** ArrayList<Integer>();

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

ts1.add(a[i]);

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

ts2.add(b[i]);

ts1.addAll(ts2);

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

{

**for**(**int** j=0;j< b.length;j++)

{

**if**(a[i]==b[j])

ts3.add(a[i]);

}

}

ts1.removeAll(ts3);

aa.addAll(ts1);

**int** res[]=**new** **int**[aa.size()];

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

res[i]=aa.get(i);

**return** res;

}

}

Junit Testing

**import** java.text.ParseException;

**import** org.junit.Assert;

**import** org.junit.Test;

**import** handson.SymmetricDifference;

**public** **class** TestSymmetricDiff {

@Test

**public** **void** testSymmetric\_Success() **throws** ParseException {

**int**[] a = { 12, 24, 7, 36, 14 };

**int**[] b = { 11, 26, 7, 14 };

**int**[] c = { 11, 12, 24, 26, 36 };

Assert.*assertArrayEquals*(c, SymmetricDifference.*getSymmetricDifference*(a, b));

**int**[] a1 = { 11, 5, 14, 26, 3 };

**int**[] b1 = { 5, 3, 1 };

**int**[] c1 = { 1, 11, 14, 26 };

Assert.*assertArrayEquals*(c1, SymmetricDifference.*getSymmetricDifference*(a1, b1));

}

}

Test Data1

5  
11  
5  
14  
26  
3  
3  
5  
3  
1  
**Sample Output 1:**  
1  
11  
14  
26

Test Data2

5

12

24

7

36

14

4

11

26

7

14

Sample output :

11

12

24

26

36

Learning outcome: Participant could able to learn how to use the Arrays and Collection API.