

## 7 - Iterable & Map Data Structure

Gieldy Akhyat Affandi Utama, QE-A

1. Buatlah sebuah program menggabungkan 2 array yang diberikan, dan jangan sampai terdapat nama yang sama di data yang sudah tergabung tadi.

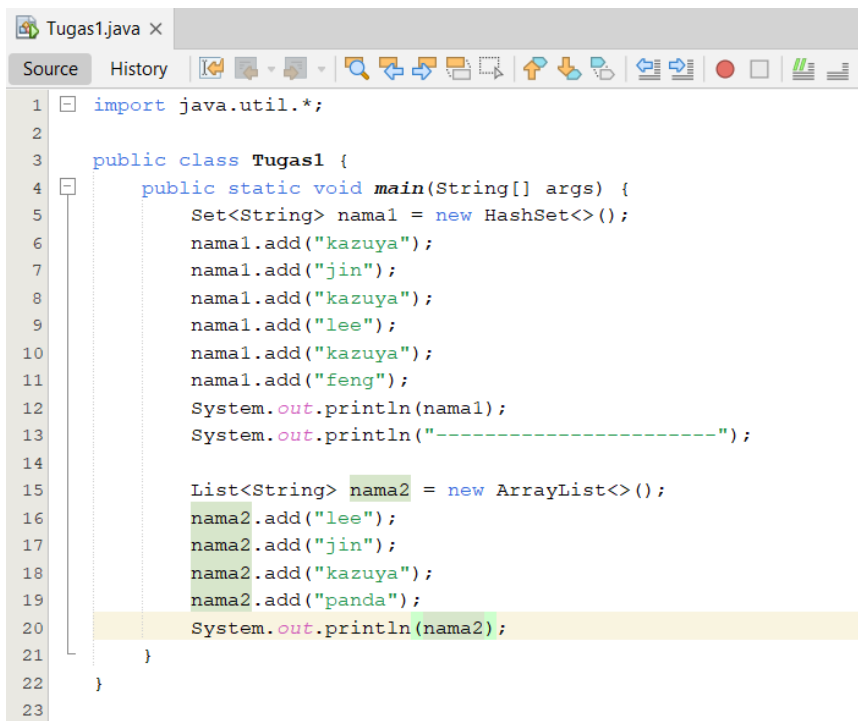
### Sample Test Cases

Input: ['kazuya', 'jin', 'lee'], ['kazuya', 'feng']

Output: ['kazuya', 'jin', 'lee', 'feng']

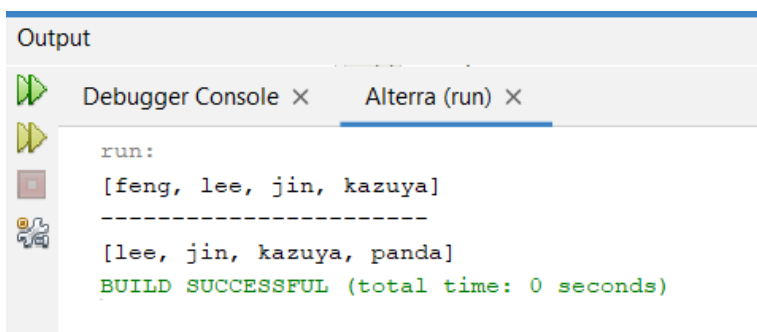
Input: ['lee', 'jin'], ['kazuya', 'panda']

Output: ['lee', 'jin', 'kazuya', 'panda']



```
1 import java.util.*;
2
3 public class Tugas1 {
4     public static void main(String[] args) {
5         Set<String> nama1 = new HashSet<>();
6         nama1.add("kazuya");
7         nama1.add("jin");
8         nama1.add("kazuya");
9         nama1.add("lee");
10        nama1.add("kazuya");
11        nama1.add("feng");
12        System.out.println(nama1);
13        System.out.println("-----");
14
15        List<String> nama2 = new ArrayList<>();
16        nama2.add("lee");
17        nama2.add("jin");
18        nama2.add("kazuya");
19        nama2.add("panda");
20        System.out.println(nama2);
21    }
22 }
23
```

Ini merupakan Source Code menggunakan Set sehingga kata yang sama tidak akan terpenggil atau tidak akan keluar dua kali.



```
Output
run:
[feng, lee, jin, kazuya]
-----
[lee, jin, kazuya, panda]
BUILD SUCCESSFUL (total time: 0 seconds)
```

Diatas adalah hasilnya ketika di run. Sehingga sesuai dengan permintaan pada soal

2. Buat program sesuai dengan deskripsi di bawah. Input merupakan variable string berisi kumpulan angka. Output merupakan list / array berisi angka yang hanya muncul 1 kali pada input.

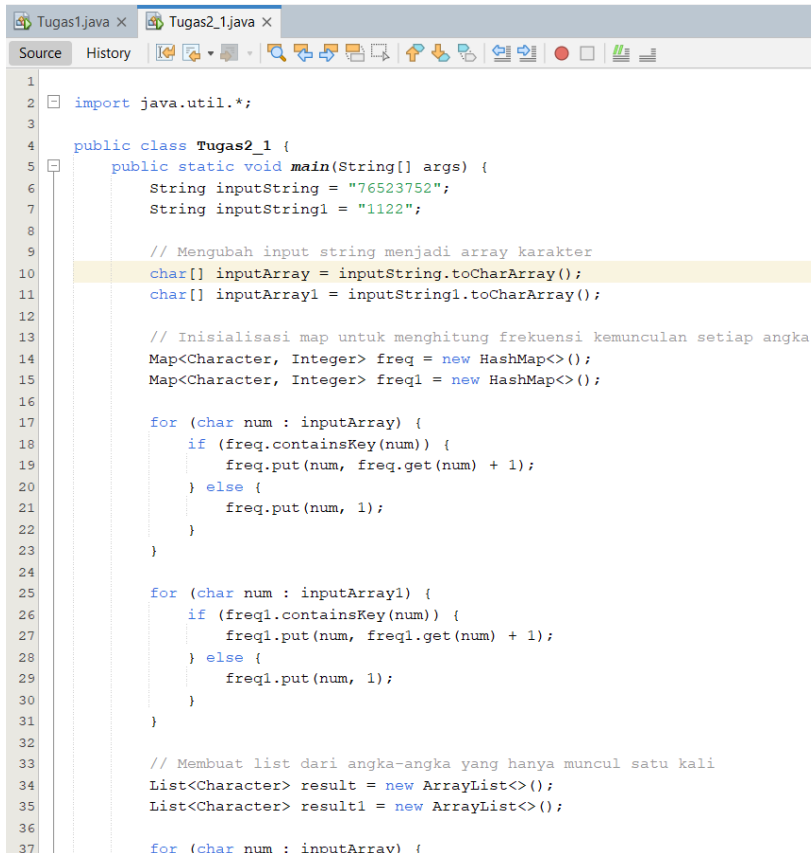
### Sample Test Case

Input: "76523752"

Output: [6, 3]

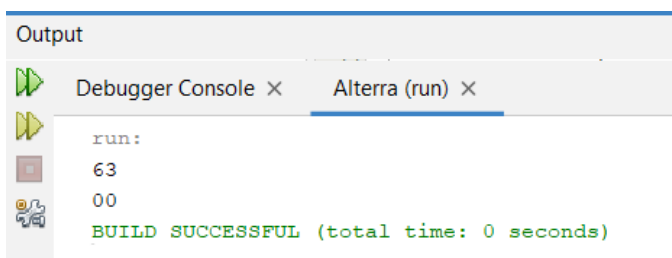
Input: "1122"

Output: []



```
1
2 import java.util.*;
3
4 public class Tugas2_1 {
5     public static void main(String[] args) {
6         String inputString = "76523752";
7         String inputString1 = "1122";
8
9         // Mengubah input string menjadi array karakter
10        char[] inputArray = inputString.toCharArray();
11        char[] inputArray1 = inputString1.toCharArray();
12
13        // Inisialisasi map untuk menghitung frekuensi kemunculan setiap angka
14        Map<Character, Integer> freq = new HashMap<>();
15        Map<Character, Integer> freq1 = new HashMap<>();
16
17        for (char num : inputArray) {
18            if (freq.containsKey(num)) {
19                freq.put(num, freq.get(num) + 1);
20            } else {
21                freq.put(num, 1);
22            }
23        }
24
25        for (char num : inputArray1) {
26            if (freq1.containsKey(num)) {
27                freq1.put(num, freq1.get(num) + 1);
28            } else {
29                freq1.put(num, 1);
30            }
31        }
32
33        // Membuat list dari angka-angka yang hanya muncul satu kali
34        List<Character> result = new ArrayList<>();
35        List<Character> result1 = new ArrayList<>();
36
37        for (char num : inputArray) {
```

Pada gambar diatas merupakan source code program, input berupa string namun kumpulan angka. Sehingga menggunakan Map untuk menghitung angka yang muncul lebih dari 1, lalu membuat list untuk angka yang hanya muncul hanya sebanyak 1 kali saja.



```
Output
Debugger Console X Alterra (run) X
run:
63
00
BUILD SUCCESSFUL (total time: 0 seconds)
```

Setelah di run keluarlah hasilnya dimana angka yang keluar adalah angka yang tidak memiliki duplikasi.

3. Given an array of sorted numbers and a target sum, find a pair in the array whose sum is equal to the given target. Write a function to return the indices of the two numbers (i.e. the pair) such that they add up to the given target.

### Challenges:

Solve with linear complexity  $O(n)$ , **not**  $O(n^2)$  if you can!

### Sample Test Cases

Input: [1, 2, 3, 4, 6], target=6

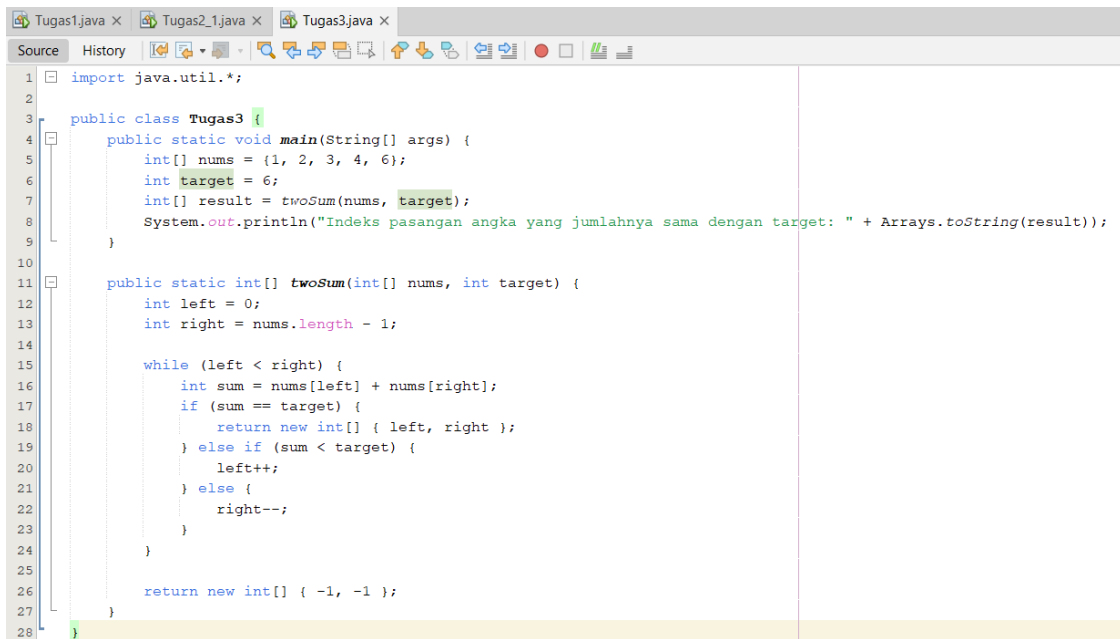
Output: [1, 3]

Explanation: The numbers at index 1 and 3 add up to 6:  $2+4=6$

Input: [2, 5, 9, 11], target=11

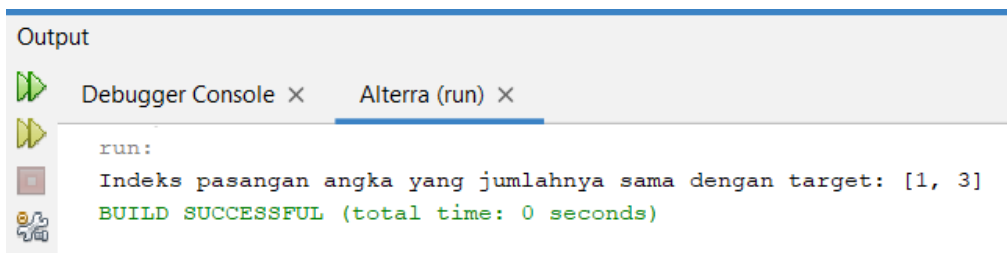
Output: [0, 2]

Explanation: The numbers at index 0 and 2 add up to 11:  $2+9=11$



```
1 import java.util.*;
2
3 public class Tugas3 {
4     public static void main(String[] args) {
5         int[] nums = {1, 2, 3, 4, 6};
6         int target = 6;
7         int[] result = twoSum(nums, target);
8         System.out.println("Indeks pasangan angka yang jumlahnya sama dengan target: " + Arrays.toString(result));
9     }
10
11     public static int[] twoSum(int[] nums, int target) {
12         int left = 0;
13         int right = nums.length - 1;
14
15         while (left < right) {
16             int sum = nums[left] + nums[right];
17             if (sum == target) {
18                 return new int[] { left, right };
19             } else if (sum < target) {
20                 left++;
21             } else {
22                 right--;
23             }
24         }
25
26         return new int[] { -1, -1 };
27     }
28 }
```

Berikut program untuk menemukan pasangan array yang apabila dijumlahkan akan senilai dengan target, dimana terlihat pada indeks/array 1 bernilai 2, dan indeks/array 3 bernilai 4

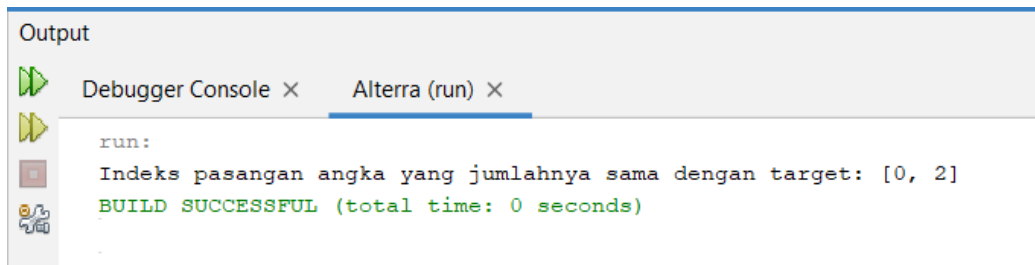


```
Output
Debugger Console x Alterra (run) x
run:
Indeks pasangan angka yang jumlahnya sama dengan target: [1, 3]
BUILD SUCCESSFUL (total time: 0 seconds)
```

Ketika program dijalankan maka indeks yang keluar pula adalah indeks 1 dan 3 sesuai dengan nilai target yang ingin dicapai.

```
int[] nums = {2, 5, 9, 11};  
int target = 11;
```

Pada sample test 2 dengan source code yang sama namun hanya variabelnya saja yang diganti, terlihat pada gambar target yang ingin dicapai ialah 11, maka nilai yang perlu dijumlahkan adalah 2 yaitu indeks/array ke-0, dan 9 yaitu array ke-2



Maka hasil yang didapatkan pula sesuai dengan tujuan dari target yang ingin dicapai.

4. Buatlah sebuah program **ArrayUnique** yang menerima 2 parameter berupa array angka. Output adalah program adalah satu array berupa kumpulan angka di array pertama tetapi tidak memiliki duplikasi di di array kedua.

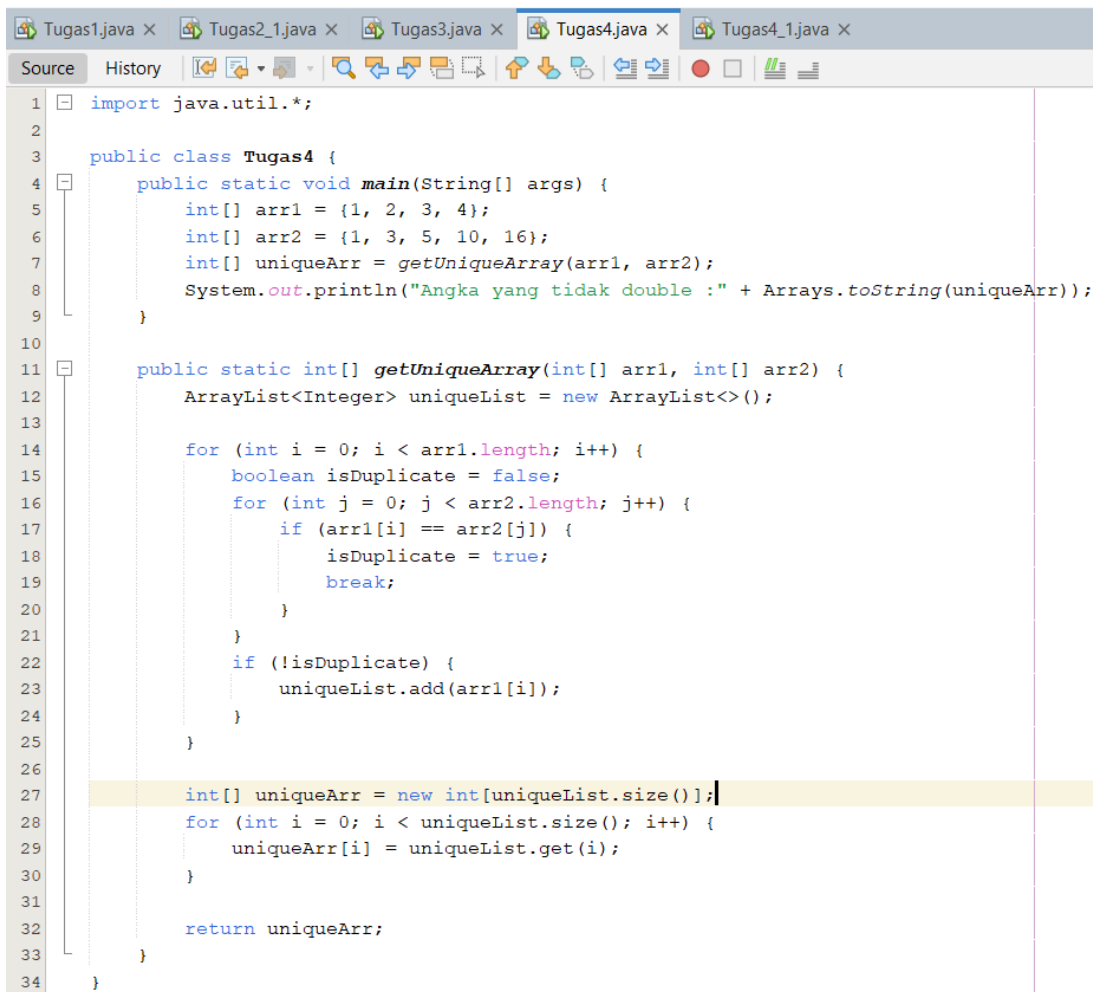
### Sample Test Case

input: [1, 2, 3, 4] dan [1, 3, 5, 10, 16]

Output: [2, 4]

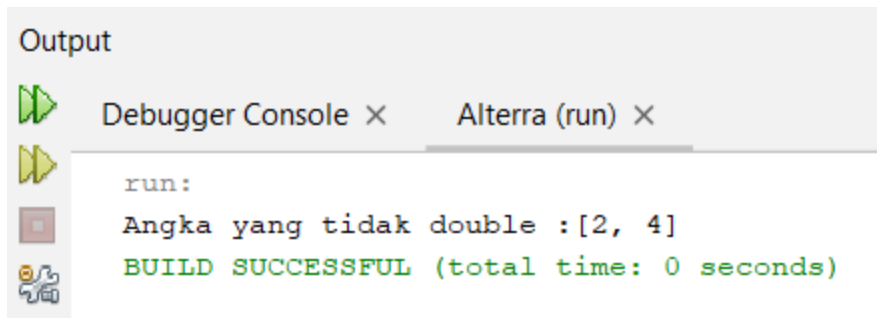
input: [3, 8] dan [2, 8]

Output: [3]



```
1 import java.util.*;
2
3 public class Tugas4 {
4     public static void main(String[] args) {
5         int[] arr1 = {1, 2, 3, 4};
6         int[] arr2 = {1, 3, 5, 10, 16};
7         int[] uniqueArr = getUniqueArray(arr1, arr2);
8         System.out.println("Angka yang tidak double :" + Arrays.toString(uniqueArr));
9     }
10
11     public static int[] getUniqueArray(int[] arr1, int[] arr2) {
12         ArrayList<Integer> uniqueList = new ArrayList<>();
13
14         for (int i = 0; i < arr1.length; i++) {
15             boolean isDuplicate = false;
16             for (int j = 0; j < arr2.length; j++) {
17                 if (arr1[i] == arr2[j]) {
18                     isDuplicate = true;
19                     break;
20                 }
21             }
22             if (!isDuplicate) {
23                 uniqueList.add(arr1[i]);
24             }
25         }
26
27         int[] uniqueArr = new int[uniqueList.size()];
28         for (int i = 0; i < uniqueList.size(); i++) {
29             uniqueArr[i] = uniqueList.get(i);
30         }
31
32         return uniqueArr;
33     }
34 }
```

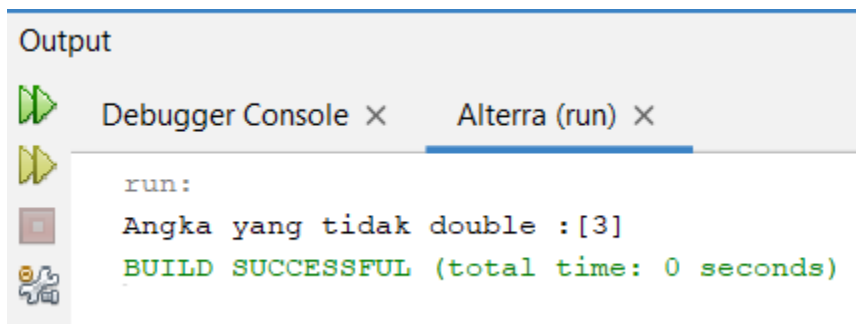
Diatas adalah program untuk mengumpulkan angka yang tidak duplikasi pada array pertama terhadap array kedua. Pada array pertama 1,2,3,4 dan array kedua 1,3,5,10,16. Maka yang seharusnya keluar adalah 2 dan 4 karena angka 2 dan 4 tidak ada pada array kedua



Output yang keluar adalah 2 dan 4 sesuai dengan sample test yang diberikan

```
int[] arr1 = {3, 8};  
int[] arr2 = {2, 8};
```

Selanjutnya pada sample test kedua hanya mengganti arraynya saja dimana pada array pertama yaitu 3 dan 8, pada array kedua 2 dan 8 sehingga output yang akan keluar adalah angka 3



Output yang keluar adalah 3 sesuai dengan sample test yang diberikan, mengapa angka 2 tidak keluar, ya karena pada array kedua tidak akan muncul walaupun tidak memiliki duplikasi karena program yang diminta adalah angka yang tidak terduplikasi pada array pertama saja

5. Given an array of sorted numbers, remove all duplicates from it. You should not use any extra space; after removing the duplicates in-place return the length of the subarray that has no duplicate in it.

### Sample Test Case

Input: [2, 3, 3, 3, 6, 9, 9]

Output: 4

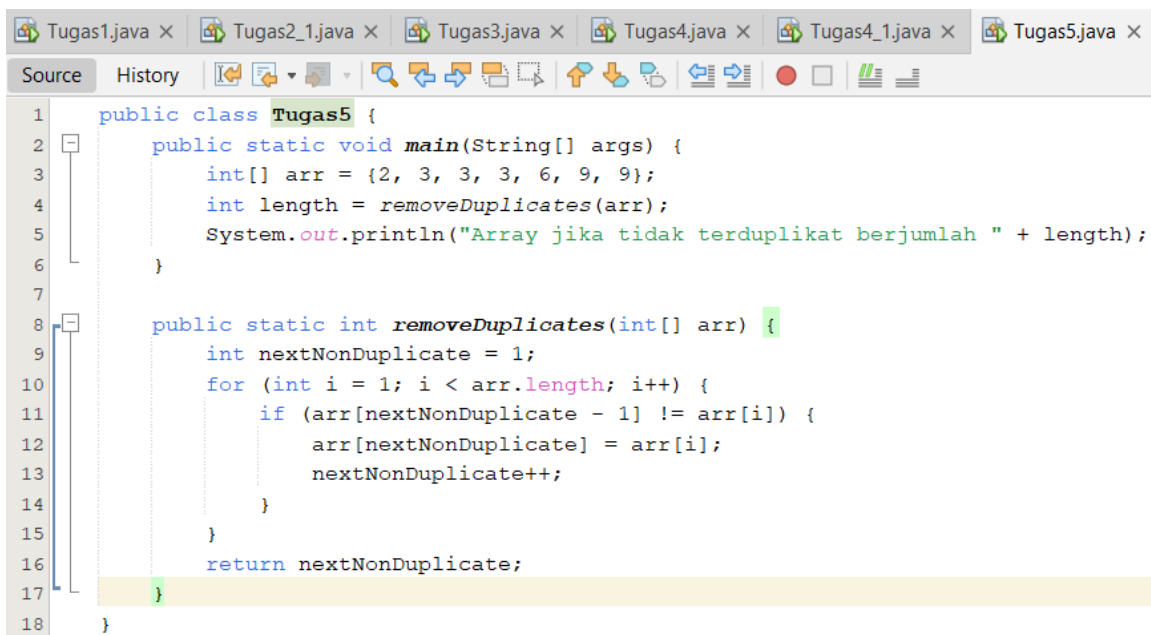
Explanation: The first four elements after removing the duplicates will be [2, 3, 6, 9].

### Sample Test Case

Input: [2, 2, 2, 11]

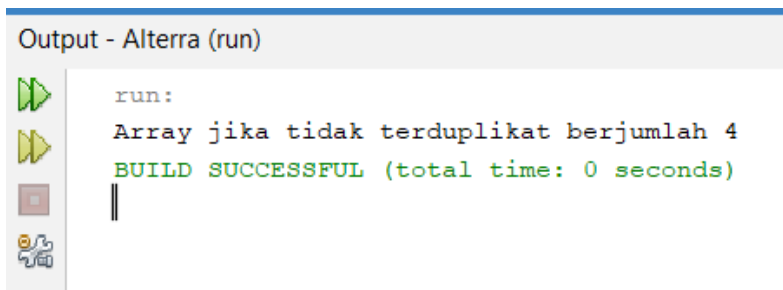
Output: 2

Explanation: The first two elements after removing the duplicates will be [2, 11].



```
1 public class Tugas5 {
2     public static void main(String[] args) {
3         int[] arr = {2, 3, 3, 3, 6, 9, 9};
4         int length = removeDuplicates(arr);
5         System.out.println("Array jika tidak terduplikat berjumlah " + length);
6     }
7
8     public static int removeDuplicates(int[] arr) {
9         int nextNonDuplicate = 1;
10        for (int i = 1; i < arr.length; i++) {
11            if (arr[nextNonDuplicate - 1] != arr[i]) {
12                arr[nextNonDuplicate] = arr[i];
13                nextNonDuplicate++;
14            }
15        }
16        return nextNonDuplicate;
17    }
18 }
```

Pada program ini adalah menghitung angka yang tidak terduplikat dalam array, sample case pertama arraynya adalah 2,3,3,3,6,9,9 maka nilai yang seharusnya ada adalah berjumlah 4 karena 2,3,6,9



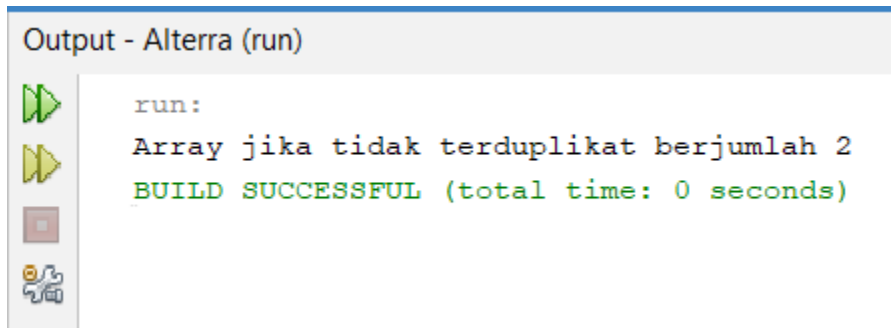
```
Output - Alterra (run)

run:
Array jika tidak terduplikat berjumlah 4
BUILD SUCCESSFUL (total time: 0 seconds)
```

Output yang keluar adalah 4 sesuai dengan sample test yang diberikan. Karena nilai yang dihitung adalah nilai yang sudah tidak ada duplikatnya yaitu 2,3,6,9

```
int[] arr = {2, 2, 2, 11};
```

Pada sample case yang kedua masih menggunakan source code yang sama namun diganti nilai arraynya adalah 2,2,2,11



Maka hasil yang keluar adalah 2 karena nilai yang keluar adalah nilai yang sudah tidak terduplikat yaitu 2,11