Gita Leoni Dwi Fripta

Kelas A

Interable dan map data struktur

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

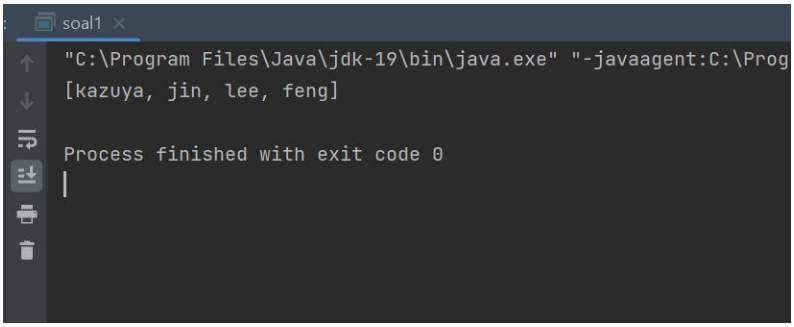
   Input: ['kazuya', 'jin', 'lee'], ['kazuya', 'feng']

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

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

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

   Program tersebut yaitu menggabungkan 2 array menjadi satu, dengan hasil output menggabungkan array 1 dan array 2.



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

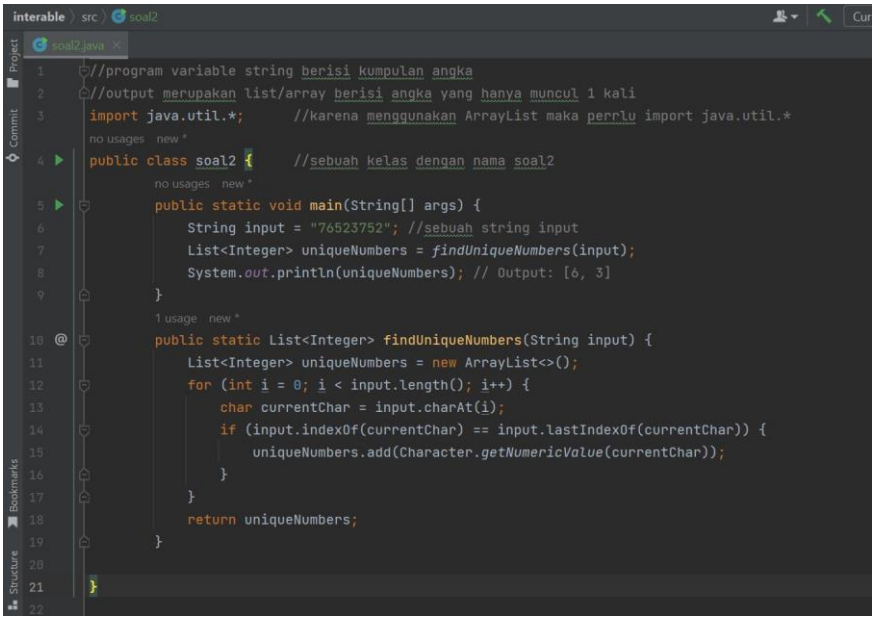
   Input: “76523752”

   Output: [6, 3]

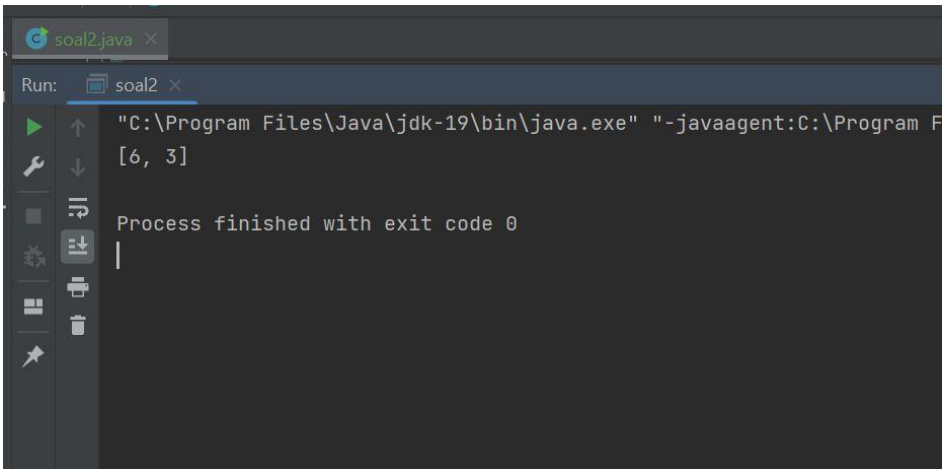
   Input: “1122”

   Output: []

   code sample test case 1
   variable string berisi kumpulan angka, merupakan array list



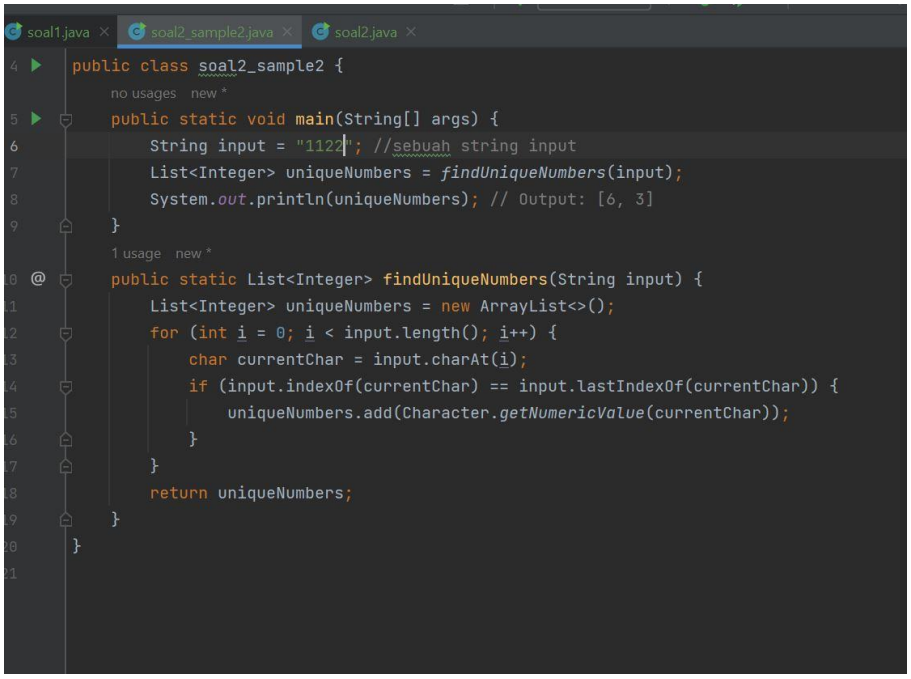
```java
//program variable string berisi kumpulan angka
//output merupakan list/array berisi angka yang hanya muncul 1 kali
import java.util.*;      //karena menggunakan ArrayList maka perrlu import java.util.*
public class soal2 {     //sebuah kelas dengan nama soal2
    public static void main(String[] args) {
        String input = "76523752"; //sebuah string input
        List<Integer> uniqueNumbers = findUniqueNumbers(input);
        System.out.println(uniqueNumbers); // Output: [6, 3]
    }
    public static List<Integer> findUniqueNumbers(String input) {
        List<Integer> uniqueNumbers = new ArrayList<>();
        for (int i = 0; i < input.length(); i++) {
            char currentChar = input.charAt(i);
            if (input.indexOf(currentChar) == input.lastIndexOf(currentChar)) {
                uniqueNumbers.add(Character.getNumericValue(currentChar));
            }
        }
        return uniqueNumbers;
    }
}
```

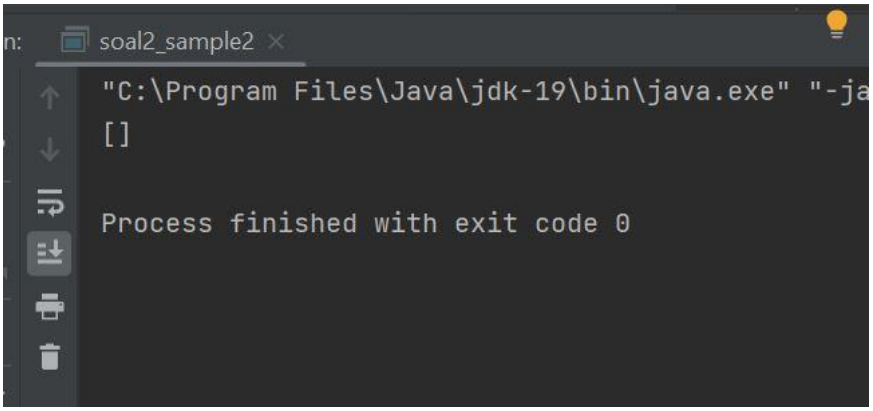


```
"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program F
[6, 3]

Process finished with exit code 0
```

Sample test case2





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]

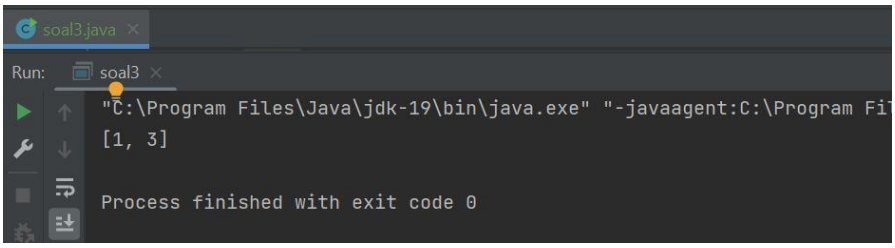
   Explanation: The numbers at index 0 and 2 add up to 11: 2+9=11

   Sample test case 1

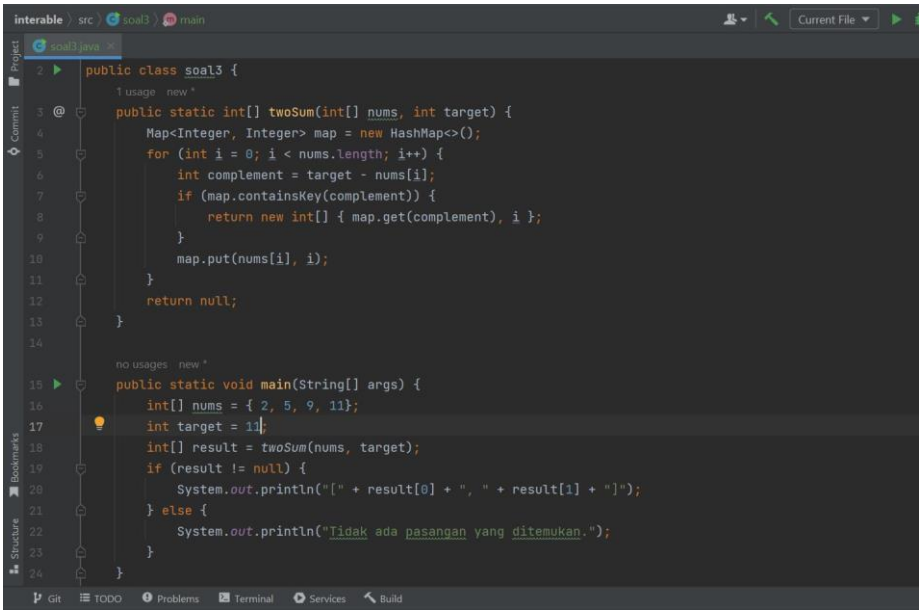
soal3.java

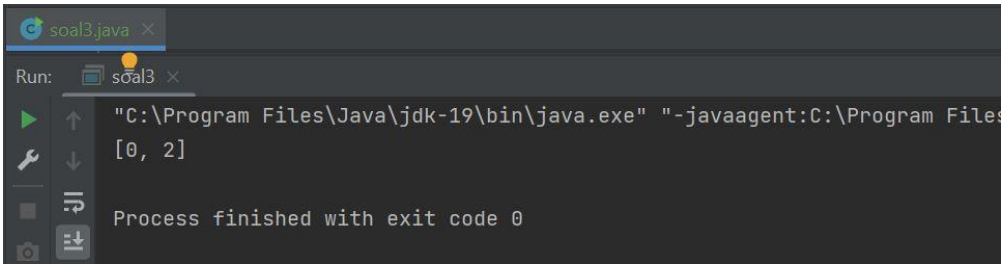
```java
import java.util.*;
public class soal3 {
    public static int[] twoSum(int[] nums, int target) {
        Map<Integer, Integer> map = new HashMap<>();
        for (int i = 0; i < nums.length; i++) {
            int complement = target - nums[i];
            if (map.containsKey(complement)) {
                return new int[] { map.get(complement), i };
            }
            map.put(nums[i], i);
        }
        return null;
    }

    public static void main(String[] args) {
        int[] nums = { 1, 2, 3, 4, 6 };
        int target = 6;
        int[] result = twoSum(nums, target);
        if (result != null) {
            System.out.println("[" + result[0] + ", " + result[1] + "]");
        } else {
            System.out.println("Tidak ada pasangan yang ditemukan.");
```



Sample test case 2





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]

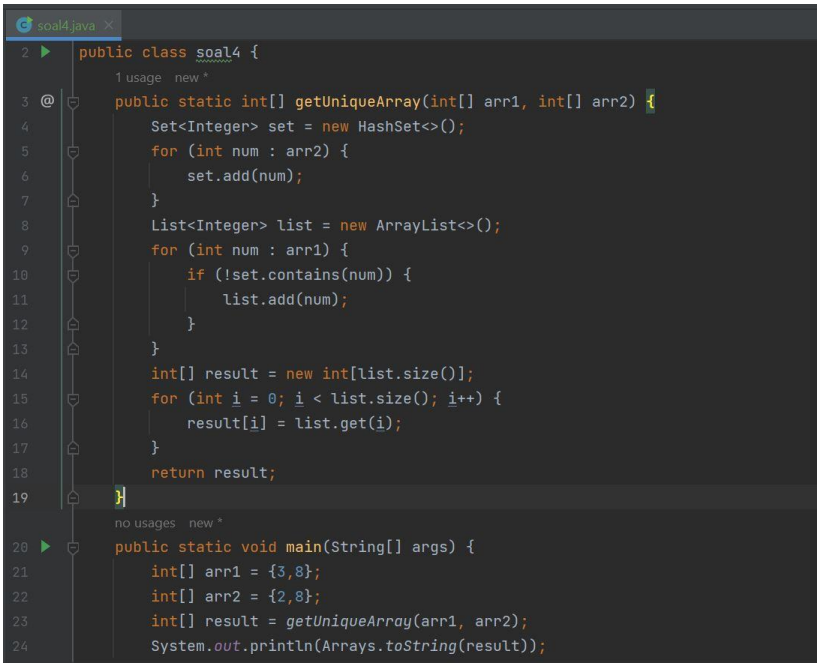
   Sample test cae 1

```java
public class soal4 {
    1 usage  new *
    public static int[] getUniqueArray(int[] arr1, int[] arr2) {
        Set<Integer> set = new HashSet<>();
        for (int num : arr2) {
            set.add(num);
        }
        List<Integer> list = new ArrayList<>();
        for (int num : arr1) {
            if (!set.contains(num)) {
                list.add(num);
            }
        }
        int[] result = new int[list.size()];
        for (int i = 0; i < list.size(); i++) {
            result[i] = list.get(i);
        }
        return result;
    }
    no usages  new *
    public static void main(String[] args) {
        int[] arr1 = {1, 2, 3, 4};
        int[] arr2 = {1, 3, 5, 10, 16};
        int[] result = getUniqueArray(arr1, arr2);
        System.out.println(Arrays.toString(result));
    }
}
```

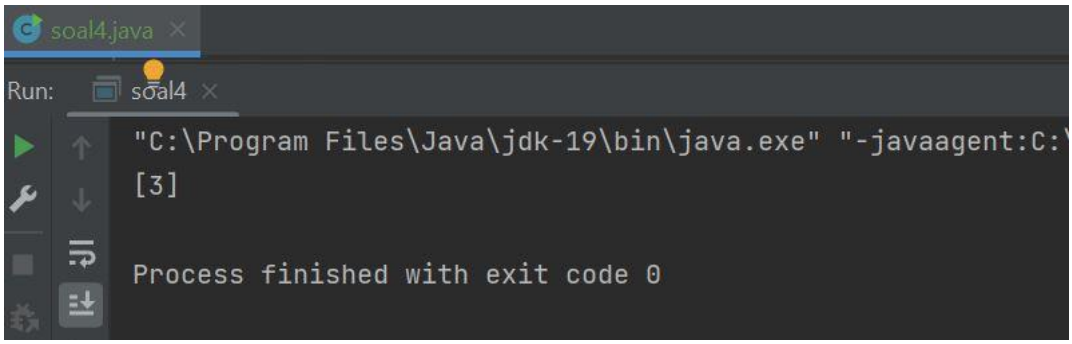
```
soal4.java
Run:  soal4
"C:\Program Files\Java\jdk-19\bin\java.exe"
[2, 4]

Process finished with exit code 0
```

Sample test case 2



```java
public class soal4 {
    1 usage  new *
    public static int[] getUniqueArray(int[] arr1, int[] arr2) {
        Set<Integer> set = new HashSet<>();
        for (int num : arr2) {
            set.add(num);
        }
        List<Integer> list = new ArrayList<>();
        for (int num : arr1) {
            if (!set.contains(num)) {
                list.add(num);
            }
        }
        int[] result = new int[list.size()];
        for (int i = 0; i < list.size(); i++) {
            result[i] = list.get(i);
        }
        return result;
    }
    no usages  new *
    public static void main(String[] args) {
        int[] arr1 = {3,8};
        int[] arr2 = {2,8};
        int[] result = getUniqueArray(arr1, arr2);
        System.out.println(Arrays.toString(result));
```



```
soal4.java
Run:  soal4
"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:
[3]

Process finished with exit code 0
```

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

```java
import java.util.*;
public class soal5 {
    public static int removeDuplicates(int[] nums) {
        int n = nums.length;
        if (n == 0 || n == 1) {
            return n;
        }
        int j = 0;
        for (int i = 0; i < n - 1; i++) {
            if (nums[i] != nums[i + 1]) {
                nums[j++] = nums[i];
            }
        }
        nums[j++] = nums[n - 1];
        return j;
    }
    public static void main(String[] args) {
        int[] nums1 = {2, 3, 3, 3, 6, 9, 9};
        int[] nums2 = {2, 2, 2, 11};
        System.out.println("Panjang subarray tanpa duplikat dari nums1 adalah " + removeDuplicates(nums1));
        System.out.println("Panjang subarray tanpa duplikat dari nums2 adalah " + removeDuplicates(nums2));
    }
}
```

```
soal5.java
Run: soal5
"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C
Panjang subarray tanpa duplikat dari nums1 adalah 4
Panjang subarray tanpa duplikat dari nums2 adalah 2

Process finished with exit code 0
```

6. [Opsional / Nilai Tambah] Given an array of positive numbers and a positive number ‘k’, find the maximum sum of any contiguous subarray of size ‘k’.

   Sample Test Case

   Input: [2, 1, 5, 1, 3, 2], k=3

   Output: 9

   Explanation: Subarray with maximum sum is [5, 1, 3].

   Sample Test Case

   Input: [2, 3, 4, 1, 5], k=2

   Output: 7

   Explanation: Subarray with maximum sum is [3, 4].

   2, 1, 5, 1, 3, 2