

Solution@22-12-23

599. Minimum Index Sum of Two Lists

```
import java.util.*;
```

```
public class MinimumIndexSum {  
    public static String[] findRestaurant(String[] list1, String[] list2) {  
        Map<String, Integer> indexMap = new HashMap<>();  
        List<String> result = new ArrayList<>();  
        int minIndexSum = Integer.MAX_VALUE;  
  
        // Populate the index map with strings from the first list  
        for (int i = 0; i < list1.length; i++) {  
            indexMap.put(list1[i], i);  
        }  
  
        // Iterate through the second list to find common strings with the least index  
        // sum  
        for (int j = 0; j < list2.length; j++) {  
            if (indexMap.containsKey(list2[j])) {  
                int indexSum = j + indexMap.get(list2[j]);  
  
                // Update the result if the current string has a smaller index sum  
                if (indexSum < minIndexSum) {  
                    result.clear();  
                    result.add(list2[j]);  
                    minIndexSum = indexSum;  
                } else if (indexSum == minIndexSum) {  
                    // Add to the result if the current string has the same index sum  
                    result.add(list2[j]);  
                }  
            }  
        }  
    }  
}
```

```

    }

    // Convert the result list to an array
    return result.toArray(new String[0]);
}

public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);

    // Get input from the user for list1
    System.out.print("Enter strings for list1 (comma-separated): ");
    String[] list1 = scanner.nextLine().split(", ");

    // Get input from the user for list2
    System.out.print("Enter strings for list2 (comma-separated): ");
    String[] list2 = scanner.nextLine().split(", ");

    // Call the findRestaurant method and print the result
    String[] result = findRestaurant(list1, list2);

    // Print the result
    System.out.println("Common strings with the least index sum:");
    for (String str : result) {
        System.out.println(str);
    }

    scanner.close();
}
}

```

Output

PS C:\Users\Ajeet\Desktop\java> java MinimumIndexSum

Enter strings for list1 (comma-separated): ajeet, abhishek, yatendra, sumit, himanshu

Enter strings for list2 (comma-separated): ajeet, rajat, mohan, vivek

Common strings with the least index sum:

ajeet

605. Can Place Flowers

```
import java.util.Scanner;
```

```
public class CanPlaceFlowers {  
    public static boolean canPlaceFlowers(int[] flowerbed, int n) {  
        int count = 0;  
        int length = flowerbed.length;  
  
        for (int i = 0; i < length; i++) {  
            if (flowerbed[i] == 0) {  
                // Check if the current plot and its adjacent plots are empty  
                boolean prevEmpty = (i == 0 || flowerbed[i - 1] == 0);  
                boolean nextEmpty = (i == length - 1 || flowerbed[i + 1] == 0);  
  
                if (prevEmpty && nextEmpty) {  
                    // Plant a flower at the current plot  
                    flowerbed[i] = 1;  
                    count++;  
  
                    // Move to the next plot (skip the next plot as it cannot have a flower)  
                    i++;  
                }  
            }  
        }  
    }  
}
```

```

        return count >= n;
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Get input from the user for the flowerbed array
        System.out.print("Enter flowerbed array (comma-separated): ");
        String[] flowerbedString = scanner.nextLine().split(", ");
        int[] flowerbed = new int[flowerbedString.length];
        for (int i = 0; i < flowerbedString.length; i++) {
            flowerbed[i] = Integer.parseInt(flowerbedString[i]);
        }

        // Get input from the user for the number of flowers to plant
        System.out.print("Enter the number of flowers to plant: ");
        int n = scanner.nextInt();

        // Call the canPlaceFlowers method and print the result
        boolean result = canPlaceFlowers(flowerbed, n);
        System.out.println("Can place flowers: " + result);

        scanner.close();
    }
}

```

Output

PS C:\Users\Ajeet\Desktop\java> java CanPlaceFlowers

Enter flowerbed array (comma-separated): 1001

Enter the number of flowers to plant: 1

Can place flowers: false