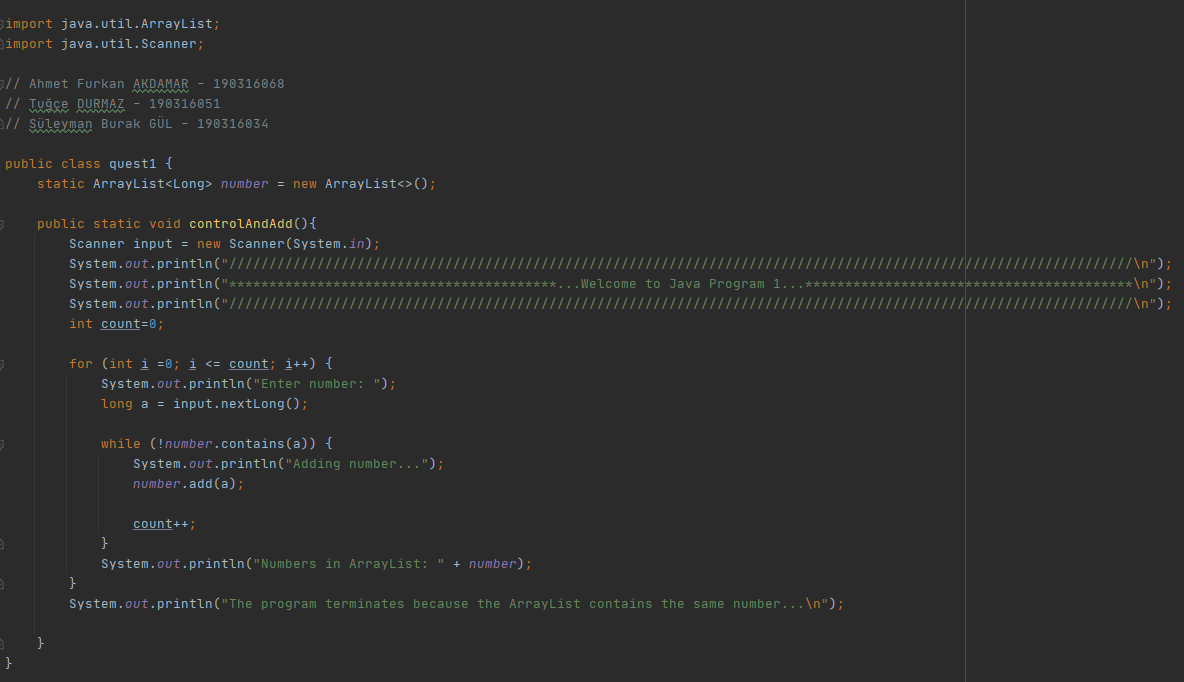
Ahmet Furkan AKDAMAR - 190316068  
Tuğçe DURMAZ - 190316051  
 Süleyman Burak GÜL - 190316034

**CSE 2105 – Data Structures**

**2021 – 2022 Fall Semester Project Report**

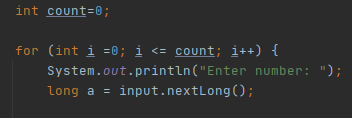
**--Question 1—**

****

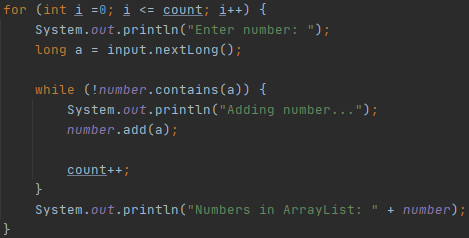
First, we imported the necessary libraries. We wrote attributes in the form of a long type arrayList that holds the numbers entered by the user.

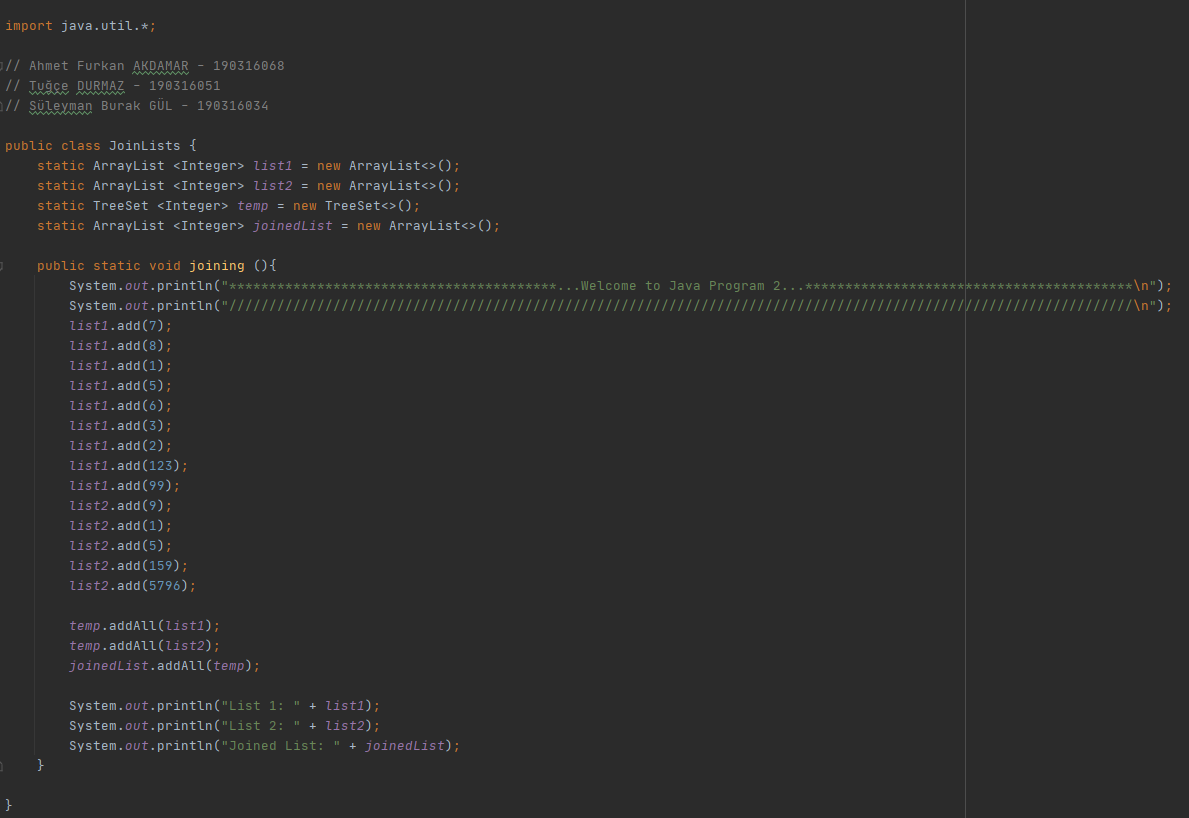


The desired operations in the program are carried out in the method called controlAndAdd(). In the controlAndAdd() method, we first created a new Scanner to get numbers from the user. We have created a for loop to get continuous numbers from the user.

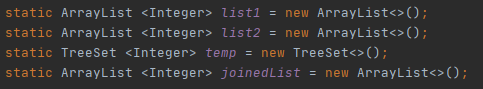


We have created a while loop inside the for loop with the contains() method, which checks whether the entered numbers are in the arrayList. In the while loop, if the numbers received from the user are not in the arrayList, the arrayList is added and the for loop is continued by increasing the count.



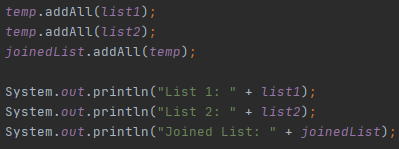
**--Question 2--** 

First, we imported java.util.\* . We wrote integer type attributes in the form of arrayList for List 1, List 2 and Joined List. When joining ArrayLists, we wrote attributes in the form of TreeSet of integer type in order not to enter the same numbers in the lists twice and to act as an intermediary to sort them.



We added random numbers to list 1 and list 2 arrayLists.

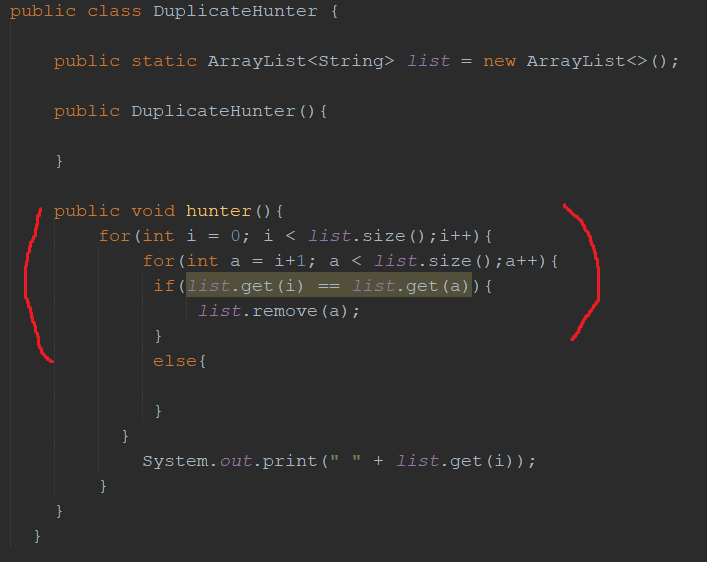


First, when joining the numbers in the lists, we added List1 and List2 to the TreeSete named temp so that only one of the same numbers from list 1 and list 2 comes and is sorted. We used the addAll() method to add all the numbers in the ArrayLists at once. Then we add the arranged numbers in the TreeSet named temp to the arrayList named joinedList.

And finally we printed the arrayLists.

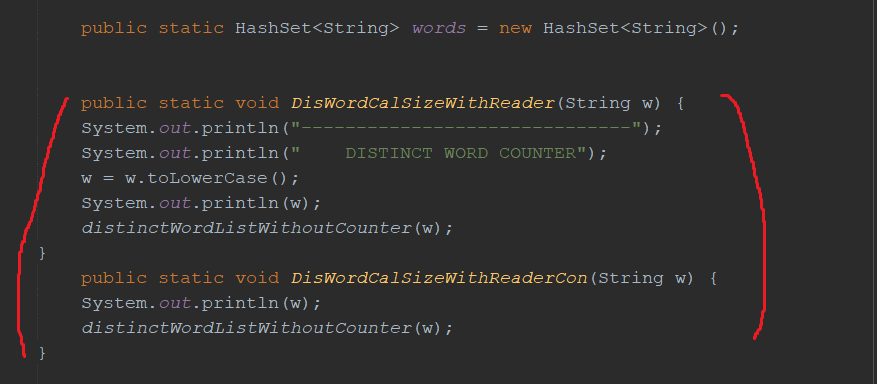
**--Question 3—**

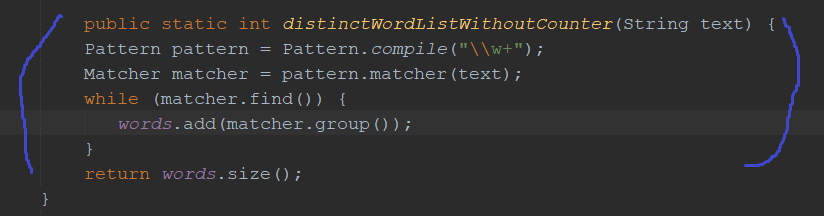
Actually, The codes infrastructure of this question is based on a very simple logic. Yhis simple logic consisst of nested for loop. We used these nested for loops using the simplest logic while writing the codes. Firstly, we created an arraylist, then we went through this arraylist with nested for loops and deleted the matching values from the list (Red Part). Finally, we finished the code by printing the list on the screen.



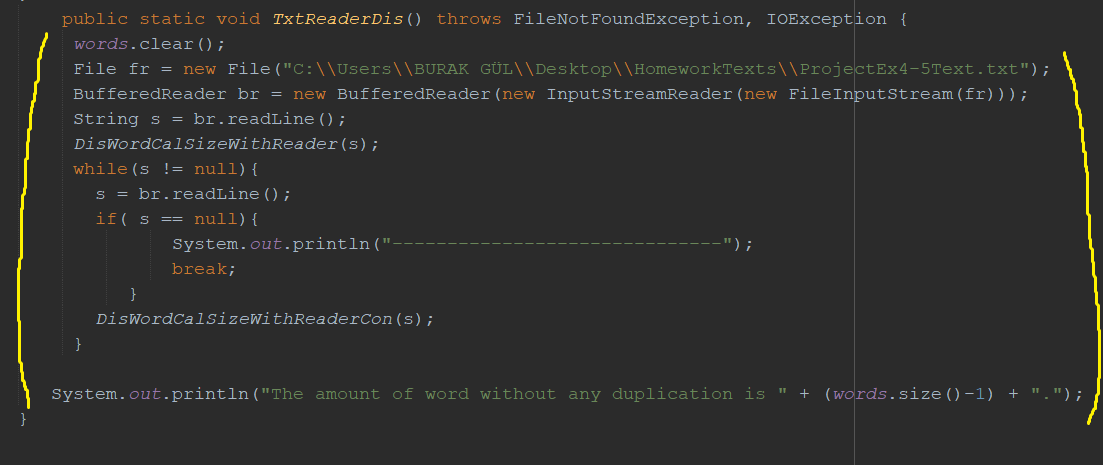
**--Question 4—**

We thought that the best way to solve this question was the HashSet list, and we tried to sıkve the question throught it. As the first thing, we created a static HashSet list. (Red Part) The reason why there are two methods that perform the same function in the next part is that we made the first method before the while loop and the other so that it Works inside the while loop. In the DisWordCalSizeWithReader() method, there is an interface that informs the user, a method that lowercase the strings taken from the text file, and codes that pass them to the distinctWordListWithoutCounter() method. The DisWordCalSizeWithReaderCon() method, is the version of the DisWordCalSizeWithReader() method, which helps to adapt it into the while loop by removing a few abilities.

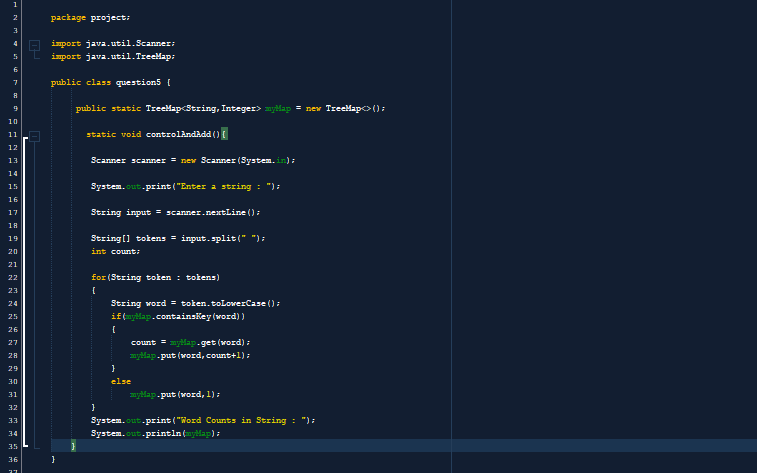


In distinctWordListWithoutCounter() method, we made the string more regular, after that we received using pattern and matcher, and we added it to our HashSet list in the while command(Blue Part). 

(Yellow Part)In the TxtReaderDis() method, we first construct our txt file. Then we apply the BufferedReader and then we pull the string from the txt file. After that, we place the drawn string into the DisWordCalSizeWithReader() method. Thanks to the while command, we continue the loop and transfer the entire txt file to the program and finally print this data on the screen.



**--Question 5—**



Firstly we imported the libraries .



In the line ; We allowed the input of the "Value" String with TreeMap and made multiple additions. We did not allow “key” values to map data. We only provided one entry.

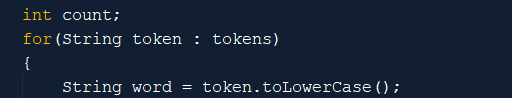
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

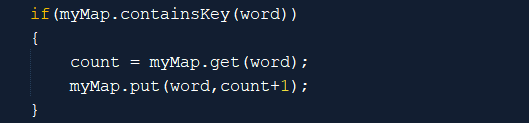
In the lines; We passed the “System.in” object to create an object of the browser class. Uses of “nextLine” to read arrays.



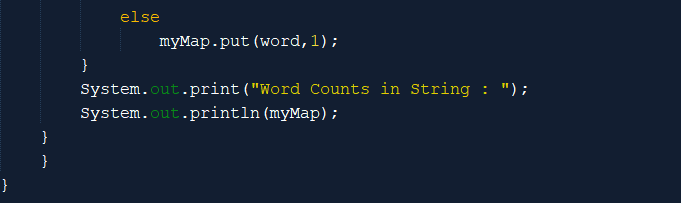
In the line; we split an array into separate tokens. We created it to detect the space after each word in the introduction.



In this code, we define count. We used " toLowerCase " in the for loop so that the initial letter does not matter whether it is uppercase or lowercase.



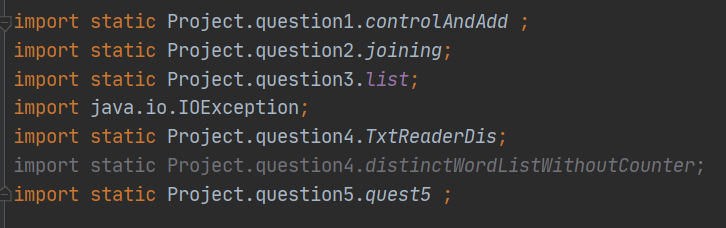
In this code, we check if the key exists with the if condition. If it exists, we call the word with "myMap.get( ) ". And when we returned the code it gave the value of the word.



If not, we returned the word " myMap.put " and only 1 value. We printed word counts on the screen with MyMap.

**Test Class**

In order to be able to call methods from other classes into this class, we imported them.



We tested the codes by calling methods from other classes in this class.

