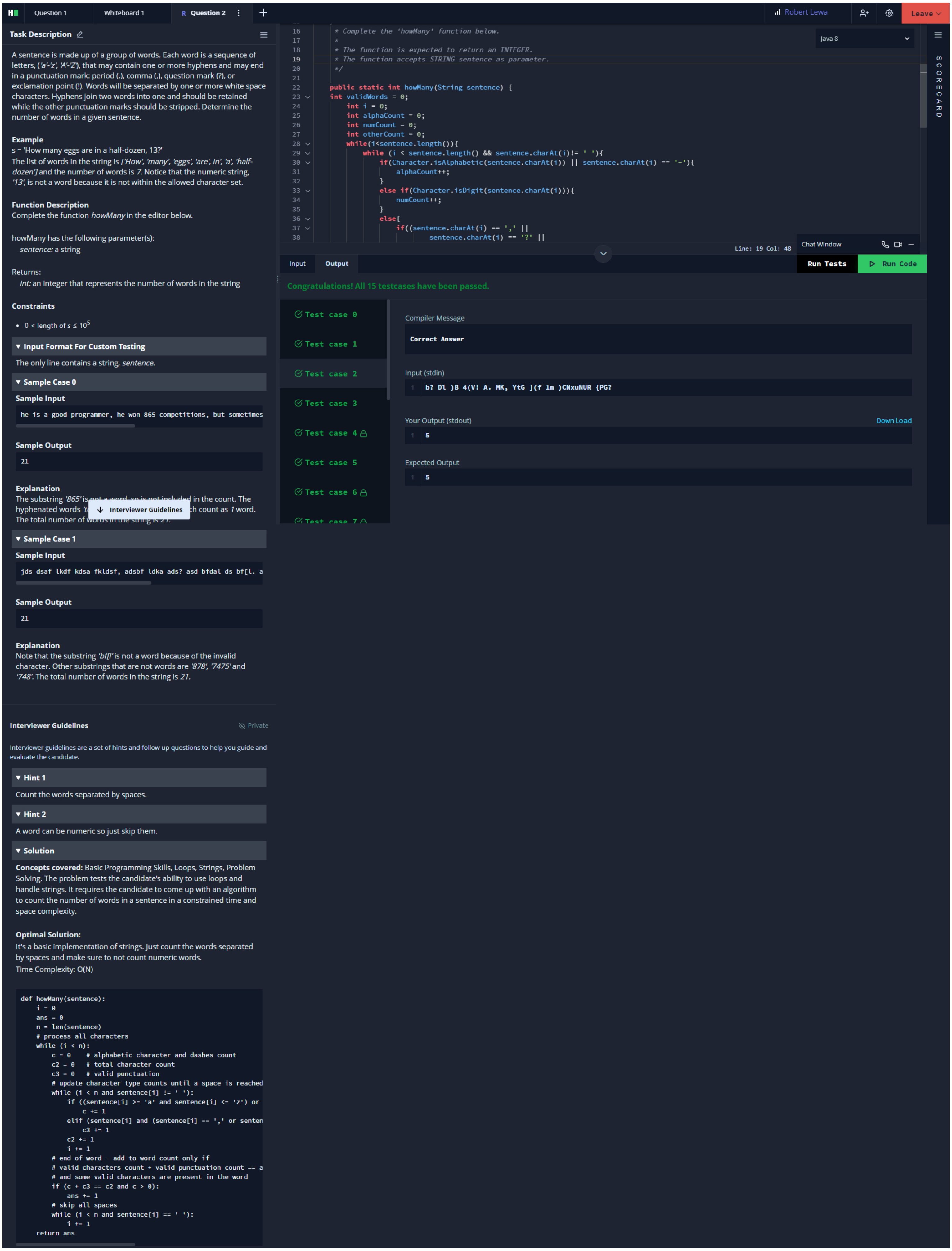

Question 1 | Whiteboard 1 | Question 2

Robert Lewa | Leave

Task Description

A sentence is made up of a group of words. Each word is a sequence of letters, ('a'-'z', 'A'-'Z'), that may contain one or more hyphens and may end in a punctuation mark: period (.), comma (,), question mark (?), or exclamation point (!). Words will be separated by one or more white space characters. Hyphens join two words into one and should be retained while the other punctuation marks should be stripped. Determine the number of words in a given sentence.

Example
s = 'How many eggs are in a half-dozen, 13?'
The list of words in the string is *['How', 'many', 'eggs', 'are', 'in', 'a', 'half-dozen']* and the number of words is *7*. Notice that the numeric string, *'13'*, is not a word because it is not within the allowed character set.

Function Description
Complete the function *howMany* in the editor below.

howMany has the following parameter(s):
  sentence: a string

Returns:
  int: an integer that represents the number of words in the string

Constraints

- 0 < length of *s* ≤ 10^5

▼ Input Format For Custom Testing

The only line contains a string, *sentence*.

▼ Sample Case 0

Sample Input

```
he is a good programmer, he won 865 competitions, but sometimes
```

Sample Output

```
21
```

Explanation
The substring *'865'* is not a word, so is not included in the count. The hyphenated words *'t* Interviewer Guidelines ch count as *1* word. The total number of words in the string is *21*.

▼ Sample Case 1

Sample Input

```
jds dsaf lkdf kdsa fkldsf, adsbf ldka ads? asd bfdal ds bf[l. a
```

Sample Output

```
21
```

Explanation
Note that the substring *'bf[l'* is not a word because of the invalid character. Other substrings that are not words are *'878'*, *'7475'* and *'748'*. The total number of words in the string is *21*.

Interviewer Guidelines Private

Interviewer guidelines are a set of hints and follow up questions to help you guide and evaluate the candidate.

▼ Hint 1

Count the words separated by spaces.

▼ Hint 2

A word can be numeric so just skip them.

▼ Solution

Concepts covered: Basic Programming Skills, Loops, Strings, Problem Solving. The problem tests the candidate's ability to use loops and handle strings. It requires the candidate to come up with an algorithm to count the number of words in a sentence in a constrained time and space complexity.

Optimal Solution:
It's a basic implementation of strings. Just count the words separated by spaces and make sure to not count numeric words.
Time Complexity: O(N)

```
def howMany(sentence):
    i = 0
    ans = 0
    n = len(sentence)
    # process all characters
    while (i < n):
        c = 0    # alphabetic character and dashes count
        c2 = 0   # total character count
        c3 = 0   # valid punctuation
        # update character type counts until a space is reached
        while (i < n and sentence[i] != ' '):
            if ((sentence[i] >= 'a' and sentence[i] <= 'z') or
                c += 1
            elif (sentence[i] and (sentence[i] == ',' or senten
                c3 += 1
            c2 += 1
            i += 1
        # end of word - add to word count only if
        # valid characters count + valid punctuation count == a
        # and some valid characters are present in the word
        if (c + c3 == c2 and c > 0):
            ans += 1
        # skip all spaces
        while (i < n and sentence[i] == ' '):
            i += 1
    return ans
```

Java 8

```
     * Complete the 'howMany' function below.
     *
     * The function is expected to return an INTEGER.
     * The function accepts STRING sentence as parameter.
     */

    public static int howMany(String sentence) {
    int validWords = 0;
        int i = 0;
        int alphaCount = 0;
        int numCount = 0;
        int otherCount = 0;
        while(i<sentence.length()){
            while (i < sentence.length() && sentence.charAt(i)!= ' '){
                if(Character.isAlphabetic(sentence.charAt(i)) || sentence.charAt(i) == '-'){
                    alphaCount++;
                }
                else if(Character.isDigit(sentence.charAt(i))){
                    numCount++;
                }
                else{
                    if((sentence.charAt(i) == ',' ||
                            sentence.charAt(i) == '?' ||
```

Line: 19 Col: 48

Chat Window

Input | Output | Run Tests | Run Code

Congratulations! All 15 testcases have been passed.

✓ Test case 0
✓ Test case 1
✓ Test case 2
✓ Test case 3
✓ Test case 4
✓ Test case 5
✓ Test case 6
✓ Test case 7

Compiler Message

```
Correct Answer
```

Input (stdin)

```
b? Dl )B 4(V! A. MK, YtG ](f 1m )CNxuNUR {PG?
```

Your Output (stdout) Download

```
5
```

Expected Output

```
5
```

Error Handling:

1. The words are separated by spaces and there might be multiple spaces in a sentence.
2. Numeric words(i.e. those words which consists of numbers) should not be counted as a valid word.

▼ Complexity Analysis

Time Complexity - O(n).
We make linear time operations on the input sentence.

Space Complexity - O(1) - No extra space is required.