To help us with this, we will make use of the following methods that are **not** part of the Advanced Placement Java subset:

* public int indexOf(String str, int fromIndex)  
  Returns the index within this string of the first occurrence of the substring str that starts at or beyond the character with index fromIndex. If there is no such occurrence, the value returned is -1. For example,

"abracadabra".indexOf( "a", 4 ) returns 5.

(The version of indexOf that is in the Advanced Placement Java subset takes only one input, a String, and behaves just like this version with a second input of 0.)

* public String toLowerCase()  
  Converts all of the characters in this string to lower case using the rules of the default locale (most likely, English) and returns the resulting converted string. For example,

"It went kaBOOM!".toLowerCase() returns "it went kaboom!".

* public String trim()  
  If this string has no leading or trailing whitespace, returns the string. Otherwise, returns a copy of the string with all leading and trailing whitespace removed. For example,

"    now we're talking!       ".trim() returns "now we're talking!".

The chatbot's response to an empty statement is easily handled. We can just insert an additional conditional (using the trim method) at the start of the cascading conditional in the body of the getResponse method, like this:

public static String getResponse(String statement)  
{  
    // missing code  
  
    if ( statement.trim().length() == 0 )  
    {  
        return "Please say something.";  
    }  
    else if (containsAnyOf(statement, negatives))  
    {  
        return "Why so negative?";  
    }  
    else if // more code  
}

The job of recognizing the presence of keywords is handled by the containsAnyOf method, which currently uses the simple, 1-input version of indexOf. We propose changing its definition so that it instead calls on a static method, indexOfKeyword, that you will define in the following task.