PLTL Exercise: My Word! (The String class)

Part (a).

Practice using the `String` methods by playing the following game.

- Each team thinks of a short word, 3 to 5 letters long. Whoever guesses the other team's word first wins!
- Each team takes three cards from a pile.
- Alternate turns. When it's your turn, play one of your three cards, then pick a new card. You can pick an `equals` card any time.
- When you play a card, the other team tells you what value their word, `s`, would return if the method call on your card were made. For example, if the other team's word is "hat" and you play `s.substring(1, 2)`, they must say "a"; if you play `s.substring(2, 4)`, they must say "error" (because "hat" has only 3 letters).
- If the card you play has a blank (e.g., `s.indexOf(__)`), you get to fill in the blank with whatever you want.
- You win if you play a `compareTo` card and the other team says 0, or if you play an `equals` card and the other team says "true" (i.e., you guessed their word).

To avoid mistakes, write down your word, and number the letters starting from zero.

Part (b).

A *palindrome* is a string that is the same forwards and backwards.

Assume that you have a `String` variable called `word`. Write an algorithm that uses the `String` methods to determine whether the sequence of characters in `word` is a palindrome. Assume that all punctuation has been removed and all letters are lower case.

Challenge: If you still have time, change your algorithm so that the word can include spaces (and punctuation if you want), but those aren't included in the palindrome test. So for example, your code should say that the word "a man a plan a canal panama!" is a palindrome.