

A word is a palindrome if it reads the same forwards and backwards. For example, the word level is a palindrome.

Create a `PalindromeNonRecursive` class that will help a user (via a Driver class) determine whether or not a given phrase (see below for examples) is a palindrome.

The driver that I used is shown below:

```
/**
 * Dominguez
 */
public class PalindromeNonRecursiveTest
{
    public static void main( String args[] )
    {
        String phrase;
        char again;
        System.out.print("Welcome to the palindrome program!");
        do
        {
            System.out.print("Enter a phrase...");
            phrase = SavitchIn.readLine();
            if( PalindromeNonRecursive.isIt( phrase ) )
                System.out.print( "\n" + "The phrase \"" + phrase +
                    "\" is a palidrome.");
            else
                System.out.print( "\n" + "The phrase " + phrase +
                    " is not a palidrome.");
            System.out.print("\n" +
                "Would you like to try another phrase? y/n...");
            again = SavitchIn.readLineNonwhiteChar();
        }while( again == 'y' || again == 'Y' );
    }
}
```

Notice that class `PalindromeNonRecursive` has only one public `static` method (i.e. `isIt`) which receives a `String` object and returns a `boolean` value of true if the `String` is a palindrome and false if it is not.

Why is it that the driver, `PalindromeNonRecursiveTest`, does not create a `PalindromeNonRecursive` object?

Remember that static methods work in a similar manner to those in the `Math` and `SavitchIn` classes. This means that since `PalindromeNonRecursive` only has one public method and it is static you don't have to create a `PalindromeNonRecursive` object to call this method.

The idea of a palindrome can be extended to phrases or sentences if we ignore details like punctuation. Here are two familiar examples:

Madam, I'm Adam

A man, a plan, a canal: Panama

We can recognize these more elaborate examples as palindromes by considering the text that is obtained by removing all spaces and punctuation marks and converting all letters to their lower-case form.

Madam, I'm Adam ==> madamimadam
A man, a plan, a canal: Panama ==> amanaplanacanalpanama

If the "word" obtained from a phrase in this manner is a palindrome, then the phrase is a palindrome. Your method should ignore the case of the letters. A palindrome is determined by considering only alphabetic characters (a – z, A – Z) and numbers (0 – 9) as valid text.

Note: The World's Longest Palindrome, created at 8:02 PM on the 20th of February (a palindromic time/date - 20:02 02/20 2002) is reported at <http://www.norvig.com/palindrome.html>

Use these sample phrases as inputs for your run outputs:

radar
Lewd did I live, & evil I did dwel.
I like Java
Straw? No, too stupid a fad, I put soot on warts.

SAMPLE OUTPUT

Welcome to the palindrome program!
Enter a phrase...radar

The phrase "radar" is a palindrome.

Would you like to try another phrase? y/n...y
Enter a phrase...I like java

The phrase "I like java" is not a palindrome.

Would you like to try another phrase? y/n...y
Enter a phrase...Straw? No, too stupid a fad. I put soot on warts.

The phrase "Straw? No, too stupid a fad. I put soot on warts." is a palindrome.

Would you like to try another phrase? y/n...n

Goodbye!

The output below shows the strategy that I used to write this program. Basically, the **public static** method **isIt** calls a **private static** method **compress** that removes all non-important characters (' ', '\', '\n', '.', ',', '!', '?', ';', ':') until you are left with a continuous alphanumeric String. Then, I have another **private static boolean** method which trims the String by removing characters from both ends. If at any time the first and last characters are different then a value of **false** is returned to method **isIt** to indicate that the String in question is not a palindrome. Keep in mind that what I'm telling you here is just a rough outline; there is still plenty of thinking left for you to do. Enjoy.

Welcome to the palindrome program!

Enter a phrase...radar

Trimmed string: "ada"

Trimmed string: "d"

The phrase "radar" is a palindrome.

Would you like to try another phrase? y/n...y

Enter a phrase...y

The phrase "y" is a palindrome.

Would you like to try another phrase? y/n...y

Enter a phrase...Straw? No, too stupid a fad. I put soot on warts.

Trimmed string: "trawnotoostupidafadiputsootonwart"

Trimmed string: "rawnotoostupidafadiputsootonwar"

Trimmed string: "awnotoostupidafadiputsootonwa"

Trimmed string: "wnotoostupidafadiputsootonw"

Trimmed string: "notoostupidafadiputsooton"

Trimmed string: "otoostupidafadiputsooto"

Trimmed string: "toostupidafadiputsoot"

Trimmed string: "oostupidafadiputsoo"

Trimmed string: "ostupidafadiputso"

Trimmed string: "stupidafadiputs"

Trimmed string: "tupidafadiput"

Trimmed string: "upidafadipu"

Trimmed string: "pidafadip"

Trimmed string: "idafadi"

Trimmed string: "dafad"

Trimmed string: "afa"

Trimmed string: "f"

The phrase "Straw? No, too stupid a fad. I put soot on warts." is a palindrome.

Would you like to try another phrase? y/n...n

Goodbye!