Java findNeedles() Method

Version: 1.0

Checks if up to 5 case-sensitive "needle" strings are located within a "haystack" string. Prints out the needles that are present in the haystack and the number of times they each appear.

"*Most people stop looking when they find the proverbial needle in the haystack. I would continue looking to see if there were other needles.*" –Albert Einstein (apocryphal)

Syntax

```
findNeedles(haystack, needles)
```

Parameters (Mandatory)

Parameter	Description
`haystack`	A `String` to search through (case-sensitive)
`needles`	An `Array` of 5 or less strings to search for in haystack (case-sensitive)

Technical Details

Delimiters: \"\'\t\n\b\f\r

Example

```
String[] Needles={"universal",
  "stretch", "embroidery", "sharps",
  "quilting"};
String haystack = "hay hay
  universal hay stretch hay
  embroidery stretchstretch hay hay
  stretch quiltinghay sharps.";
findNeedles(haystack, Needles);
```

Output:
universal: 1
stretch: 2
embroidery: 1

Warning: **`findNeedles()` will not** recognize or count a needle if it included **delimiters** or is not **exactly equal** to the string stored in haystack[i]

Email to writer of code:

<sent with or as a calendar invite>

Hi <name>,

Thank you for sending findNeedles() for documentation. It's very clear, concise, and a bit whimsical :)

I've started writing out the documentation and grabbed an hour to meet and make sure that I am explaining it as well as possible (but please, feel free to change the time). I've attached my main questions below, just in case you'd prefer to respond via email:

1) Is there a reason I should give for why the limit of "needles" is 5?
2) Could you clarify why you are delimiting specifically with these characters: `"[ \"\'\t\n\b\f\r]"` Should I caution users not to include any needles with those characters?
3) Unless I'm mistaken, `findNeedles()` does not register a needle if it is inside another word or has a comma, period, etc. after it (i.e. 'cat' in 'catastrophe' or the last word of sentences). Is there a reason I should give for this?
4) I noticed that `str.compareTo()` is case-sensitive, should I caution users of this?
5) Is there a reason why it prints out the information rather than returning it in an array?
6) And forgive my ignorance, but I'm not sure why `String[] words = haystack.split("[ \"\'\t\n\b\f\r]", 0);` is in the outer loop? Could you clarify?

Thank you again,

Jonathan Cohn
Technical Writer

P.S. I might also suggest you add a 6th needle in version 2.0 #GameChanger