Percentages in Eclipse method suggestions

From: http://stackoverflow.com/questions/11438307/eclipse-autocomplete-percent-sign-in-juno

"The percentage represents how likely the Eclipse Code Recommenders think it is that you are looking for a certain completion *based on the context and maybe prior usage* and other variables (there are "5 Intelligent Code Completion Engines"). It is not only the bare usage statistics. So *a value might change* from 13% to 95% *between some lines, depending what you did in between*."

From: http://www.eclipse.org/recommenders/manual/#intelligent-code-completion

"The Call Completion engine, for example, provides you with recommendations of likely methods to call ... The call completion engine bases its recommendations on *what other developers in a **similar situation** have called on an object of the given type*.

Split method

Example:

```
public static void main(String[] args) {

	String test = "Testing the split method\nStill testing it\nYet another test.";
	String[] words = test.split("[ |\\n]");
	System.out.println("Number of words: " + words.length);
	for (String w : words) {
		System.out.println(w);
	}

	System.out.println("\n-----------------------------------\n");

	String text = "Sentence 1. Sentence 2.   Sentence 3! Sentence 4? Sentence 5.";
	String[] sentences = text.split("[\\.|?|!]");
	for (String s : sentences) {
		System.out.println(s.trim());
	}
}
```

To learn more about the split method (nice explanations and examples):
http://javadevnotes.com/java-string-split-tutorial-and-examples

What are regular expressions? (from: http://regex.bastardsbook.com/)

The easiest way to describe them is: "Find-and-Replace on steroids".

A longer description**:** Regular expressions – **regexes** for short – are just a way to describe **patterns in text**, either to find or to replace.