

<https://mrhaki.blogspot.com/2014/06/spocklight-extra-data-variables-for.html> - grovy unroll description - auenno

[http://spockframework.org/spock/docs/1.0/data\\_driven\\_testing.html](http://spockframework.org/spock/docs/1.0/data_driven_testing.html) - more grovy unroll desc usages.

<https://e.printstacktrace.blog/groovy-regular-expressions-the-definitive-guide/> - grovy regex

Grovy RegEx:

// ~ => turns str to Pattern ==~ calls matches(), so that input.matches(pattern)

expected == (input ==~ ~pattern)

// Possible like so without ~

expected == (input ==~ pattern)

## KAEF

You need to escape your `\n` with an extra backslash..

Use: - `"a\\na"` to get `\n` in your output..

```
System.out.println("a\\na"); // Prints a\na
```

Or, you can also take a look at :- [Apache Commons](#) [StringEscapeUtils#escapeJava](#) which will automatically escape all the special `escape sequences` with an extra backslash..

```
String str = "a\nb\r\t";  
String result = StringEscapeUtils.escapeJava(str); // Takes `\\n` as `\\n`  
System.out.println(result);
```

This will print: - `a\nb\r\t`