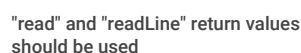


**Quick Fix** (42)

Search by name...



## Analyze your code

Code Smell Critical ? regex performance

The underlying implementation of `String::replaceAll` calls the `java.util.regex.Pattern.compile()` method each time it is called even if the first argument is not a regular expression. This has a significant performance cost and therefore should be used with care.

When `String::replaceAll` is used, the first argument should be a real regular expression. If it's not the case, `String::replace` does exactly the same thing as `String::replaceAll` without the performance drawback of the regex.

This rule raises an issue for each `String::replaceAll` used with a `String` as first parameter which doesn't contains special regex character or pattern.

### Noncompliant Code Example

```
String init = "Bob is a Bird... Bob is a Plane... Bob is Sup
String changed = init.replaceAll("Bob is", "It's"); // Nonco
changed = changed.replaceAll("\\.\\.\\.\\.\\.\"", ";"); // Noncompli
```

### Compliant Solution

```
String init = "Bob is a Bird... Bob is a Plane... Bob is Sup  
String changed = init.replace("Bob is", "It's");  
changed = changed.replace("...", ";");
```

Or, with a regex:

```
String init = "Bob is a Bird... Bob is a Plane... Bob is Sup  
String changed = init.replaceAll("\\w*\\sis", "It's");  
changed = changed.replaceAll("\\.\\{3}", "");
```






**See**

- {rule:java:S4248} - Regex patterns should not be created needlessly

Available In:

sonarlint  | sonarcloud  | sonarqube 

© 2008-2022 SonarSource S.A., Switzerland. All content is copyright protected. SONAR, SONARSOURCE, SONARLINT, SONARQUBE and SONARCLOUD are trademarks of SonarSource S.A. All other trademarks and copyrights are the property of their respective owners. All rights are expressly reserved.  
Privacy Policy

 Bug
<div>Inappropriate regular expressions should not be used</div> <div> Bug</div>
<div>Conditionally executed code should be reachable</div> <div> Bug</div>
<div>"notifyAll" should be used</div> <div> Bug</div>
<div>Blocks should be synchronized on "private final" fields</div> <div> Bug</div>