rules.sonarsource.com

Kotlin static code analysis: Character classes in regular expressions should not contain the same character twice

2 minutes

Character classes in regular expressions are a convenient way to match one of several possible characters by listing the allowed characters or ranges of characters. If the same character is listed twice in the same character class or if the character class contains overlapping ranges, this has no effect.

Thus duplicate characters in a character class are either a simple oversight or a sign that a range in the character class matches more than is intended or that the author misunderstood how character classes work and wanted to match more than one character. A common example of the latter mistake is trying to use a range like `[0-99]` to match numbers of up to two digits, when in fact it is equivalent to `[0-9]`. Another common cause is forgetting to escape the – character, creating an unintended range that overlaps with other characters in the character class.

Noncompliant Code Example

```
Regex("[0-99]") // Noncompliant, this won't actually match strings with two digits
Regex("[0-9.-_]") // Noncompliant, .-_ is a range that already contains 0-9 (as well as various other characters such as capital letters)
```

Compliant Solution

```
Regex("[0-9]{1,2}")
Regex("[0-9.\\-_]")
```