

GitHub issue reference

#123 or org/repo#123 reference.

```
(?:[A-Za-z0-9.-]+\\[A-Za-z0-9.-]+\)?#\d+
```

Token-by-token breakdown

Token	Meaning
(?:	start of a non-capturing group
[A-Za-z0-9.-]+	one or more: any of: uppercase letters, lowercase letters, digits, ".", "-"
\[	literal "["
[A-Za-z0-9.-]+	one or more: any of: uppercase letters, lowercase letters, digits, ".", "-"
\]?	end of group, optional (zero or one)
#	literal text "#"
\d+	one or more: digit (0–9)

Quick usage in 7 languages

JavaScript	<pre>const re = /(?:[A-Za-z0-9.-]+\\[A-Za-z0-9.-]+\)?#\d+/; re.test(value);</pre>
Python	<pre>import re re.match(r"(?:[A-Za-z0-9.-]+\\[A-Za-z0-9.-]+\)?#\d+", value)</pre>
Java	<pre>Pattern.compile("(?:[A-Za-z0-9.-]+\\[A-Za-z0-9.-]+\)?#\d+").matcher(value).matches();</pre>
C# / .NET	<pre>Regex.IsMatch(value, @"(?:[A-Za-z0-9.-]+\\[A-Za-z0-9.-]+\)?#\d+");</pre>
Go	<pre>regexp.MustCompile(`(?:[A-Za-z0-9.-]+\\[A-Za-z0-9.-]+\)?#\d+`).MatchString(value)</pre>
Ruby	<pre>/(?:[A-Za-z0-9.-]+\\[A-Za-z0-9.-]+\)?#\d+/.match?(value)</pre>
PHP	<pre>preg_match('~(?:[A-Za-z0-9.-]+\\[A-Za-z0-9.-]+\)?#\d+~', \$value);</pre>

Common pitfalls

- **Not anchored.** Without `^` and `$` this can match a substring anywhere in the input — add anchors if you need the whole value to conform.
- **ASCII letters only.** `[a-zA-Z]` excludes accented and non-Latin letters (é, ü, ß, ñ, and non-Latin scripts). For international input use Unicode properties like `\p{L}` with the `u` flag.
- **Escape it correctly per language.** In Java and JavaScript strings each backslash must be doubled (`\\d`); in Python, Go, and C# use raw/verbatim strings so the backslashes survive.
- **Validate beyond format.** Matching the format doesn't guarantee the value is real. Confirm the github issue reference against a source of truth (database, API, or checksum) where it matters.