

Signed integer

Integer with optional minus sign.

```
^-?(0|[1-9]\d*)$
```

Token-by-token breakdown

Token	Meaning
^	start of string (or line in multiline mode)
-?	optional (zero or one): the literal "-"
(	start of a capturing group
0	literal text "0"
	OR — try the alternative
[1-9]	any of: 1–9
\d*	zero or more: digit (0–9)
)	end of group
\$	end of string (or line in multiline mode)

Quick usage in 7 languages

JavaScript	<pre>const re = /^-?(0 [1-9]\d*)\$/; re.test(value);</pre>
Python	<pre>import re re.match(r"^-?(0 [1-9]\d*)\$", value)</pre>
Java	<pre>Pattern.compile("^-?(0 [1-9]\d*)\$").matcher(value).matches();</pre>
C# / .NET	<pre>Regex.IsMatch(value, @"^-?(0 [1-9]\d*)\$");</pre>
Go	<pre>regexp.MustCompile(`^-?(0 [1-9]\d*)\$`).MatchString(value)</pre>
Ruby	<pre>/^-?(0 [1-9]\d*)\$/.match?(value)</pre>
PHP	<pre>preg_match('~^-?(0 [1-9]\d*)\$~', \$value);</pre>

Common pitfalls

- **Anchored to the whole string.** This pattern uses ^ and \$, so it requires the entire input to match. To find it inside a longer text, drop the anchors and use the global (g) 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 signed integer against a source of truth (database, API, or checksum) where it matters.