

Instagram handle

Instagram username — letters, digits, dots, underscores.

```
^@?[a-zA-Z0-9._]{1,30}$
```

Token-by-token breakdown

Token	Meaning
^	start of string (or line in multiline mode)
@?	optional (zero or one): the literal "@"
[a-zA-Z0-9._]{1,30}	between 1 and 30 times: any of: lowercase letters, uppercase letters, digits, ".", "_"
\$	end of string (or line in multiline mode)

Quick usage in 7 languages

JavaScript	<pre>const re = /^@?[a-zA-Z0-9._]{1,30}\$/; re.test(value);</pre>
Python	<pre>import re re.match(r"^@?[a-zA-Z0-9._]{1,30}\$", value)</pre>
Java	<pre>Pattern.compile("^@?[a-zA-Z0-9._]{1,30}\$").matcher(value).matches();</pre>
C# / .NET	<pre>Regex.IsMatch(value, @"^@?[a-zA-Z0-9._]{1,30}\$");</pre>
Go	<pre>regexp.MustCompile(`^@?[a-zA-Z0-9._]{1,30}\$`).MatchString(value)</pre>
Ruby	<pre>/^@?[a-zA-Z0-9._]{1,30}\$/ .match?(value)</pre>
PHP	<pre>preg_match('~^@?[a-zA-Z0-9._]{1,30}\$~', \$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.
- **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.
- **Validate beyond format.** Matching the format doesn't guarantee the value is real. Confirm the instagram handle against a source of truth (database, API, or checksum) where it matters.