

HTTP method

Standard HTTP method verbs.

```
^(GET|POST|PUT|PATCH|DELETE|HEAD|OPTIONS|TRACE|CONNECT)$
```

Token-by-token breakdown

Token	Meaning
^	start of string (or line in multiline mode)
(	start of a capturing group
GET	literal text "GET"
	OR — try the alternative
POST	literal text "POST"
	OR — try the alternative
PUT	literal text "PUT"
	OR — try the alternative
PATCH	literal text "PATCH"
	OR — try the alternative
DELETE	literal text "DELETE"
	OR — try the alternative
HEAD	literal text "HEAD"
	OR — try the alternative
OPTIONS	literal text "OPTIONS"
	OR — try the alternative
TRACE	literal text "TRACE"
	OR — try the alternative
CONNECT	literal text "CONNECT"
)	end of group
\$	end of string (or line in multiline mode)

Quick usage in 7 languages

JavaScript	<pre>const re = /^(GET POST PUT PATCH DELETE HEAD OPTIONS TRACE CONNECT)\$/; re.test(value);</pre>
Python	<pre>import re re.match(r"^(GET POST PUT PATCH DELETE HEAD OPTIONS TRACE CONNECT)\$", value)</pre>
Java	<pre>Pattern.compile("^(GET POST PUT PATCH DELETE HEAD OPTIONS TRACE CONNECT)\$").matcher(val ue).matches();</pre>
C# / .NET	<pre>Regex.IsMatch(value, @"^(GET POST PUT PATCH DELETE HEAD OPTIONS TRACE CONNECT)\$");</pre>

Go	<code>regexp.MustCompile(`^(GET POST PUT PATCH DELETE HEAD OPTIONS TRACE CONNECT)\$`).MatchString(value)</code>
Ruby	<code>/(GET POST PUT PATCH DELETE HEAD OPTIONS TRACE CONNECT)\$/.match?(value)</code>
PHP	<code>preg_match('~^(GET POST PUT PATCH DELETE HEAD OPTIONS TRACE CONNECT)\$~', \$value);</code>

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.
- **Validate beyond format.** Matching the format doesn't guarantee the value is real. Confirm the http method against a source of truth (database, API, or checksum) where it matters.