Card expiry (MM/YY)

Two-digit month and year separated by slash.

```
^(0[1-9]|1[0-2])/\d{2}$
```

Token-by-token breakdown

Token	Meaning
`^`	start of string (or line in multiline mode)
`(`	start of a capturing group
`0`	literal text “0”
`[1-9]`	any of: 1–9
`|`	OR — try the alternative
`1`	literal text “1”
`[0-2]`	any of: 0–2
`)`	end of group
`/`	literal text “/”
`\d{2}`	exactly 2 times: digit (0–9)
`$`	end of string (or line in multiline mode)

Quick usage in 7 languages

JavaScript

```
const re = /^(0[1-9]|1[0-2])\/\d{2}$/;
re.test(value);
```

Python

```
import re
re.match(r"^(0[1-9]|1[0-2])/\d{2}$", value)
```

Java

```
Pattern.compile("^(0[1-9]|1[0-2])/\\d{2}$").matcher(value).matches();
```

C# / .NET

```
Regex.IsMatch(value, @"^(0[1-9]|1[0-2])/\d{2}$");
```

Go

```
regexp.MustCompile(`^(0[1-9]|1[0-2])/\d{2}$`).MatchString(value)
```

Ruby

```
/^(0[1-9]|1[0-2])\/\d{2}$/.match?(value)
```

PHP

```
preg_match('~^(0[1-9]|1[0-2])/\d{2}$~', $value);
```

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 card expiry (mm/yy) against a source of truth (database, API, or checksum) where it matters.

© 2026 regexguide.com · a human-curated regex explainer & pattern library

Disclaimer: provided as-is, without warranty — test against your own data before production use.