

Taiwan ID number

1 letter + 9 digits, Taiwanese national ID.

```
^[A-Z][12]\d{8}$
```

Token-by-token breakdown

Token	Meaning
<code>^</code>	start of string (or line in multiline mode)
<code>[A-Z]</code>	any of: uppercase letters
<code>[12]</code>	any of: "1", "2"
<code>\d{8}</code>	exactly 8 times: digit (0–9)
<code>\$</code>	end of string (or line in multiline mode)

Quick usage in 7 languages

JavaScript	<pre>const re = /^[A-Z][12]\d{8}\$/; re.test(value);</pre>
Python	<pre>import re re.match(r"^[A-Z][12]\d{8}\$", value)</pre>
Java	<pre>Pattern.compile("^[A-Z][12]\\d{8}\$").matcher(value).matches();</pre>
C# / .NET	<pre>Regex.IsMatch(value, @"^[A-Z][12]\d{8}\$");</pre>
Go	<pre>regexp.MustCompile(`^[A-Z][12]\d{8}\$`).MatchString(value)</pre>
Ruby	<pre>/^[A-Z][12]\d{8}\$/ .match?(value)</pre>
PHP	<pre>preg_match('~^[A-Z][12]\d{8}\$~', \$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.
- **Uppercase only.** Ranges like `[A-Z]` won't match lowercase. Add `a-z` (or the `i` flag) if lowercase input should be accepted.
- **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 taiwan id number against a source of truth (database, API, or checksum) where it matters.