

Meta characters

\d matches any digit, equivalent to [0-9]

\D matches any character that's not a digit, equivalent to [^0-9]

\w matches any alphanumeric character (plus underscore), equivalent to [A-Za-z\_0-9]

\W matches any non-alphanumeric character, anything except [^A-Za-z\_0-9]

\s matches any whitespace character: spaces, tabs, newlines and Unicode spaces

\S matches any character that's not a whitespace

\0 matches null

\n matches a newline character

\t matches a tab character

[^] matches any character, including newline characters. It's useful on multiline strings

### Zadanie 30.

---

Która wartość tekstowa **nie pasuje** do podanego w ramce wzorca wyrażenia regularnego?

`(([A-ZłŻ][a-ząęóźżćńś]{2,})-([A-ZłŻ][a-ząęóźżćńś]{2,})?)`

- A. Kowalski
- B. Kasprowicza
- C. Jelenia Góra
- D. Nowakowska-Kowalska

### Zadanie 38.

---

Poprawne udokumentowanie wzorca weryfikacji pola nazwa w części kodu aplikacji JavaScript to

```
function validateForm(Form)
{
  reg = /^[1-9]*[A-ZŹŻĘĄĆŚÓŁŃ]{1}[a-zżżęąćśółń]{2,}$/;
  wyn = Form.nazwa.value.match(reg);
  if (wyn == null) {
    alert("Proszę podać poprawną nazwę");
    return false;
  }
  return true;
}
```

- A. /\* Pole nazwa może składać się z dowolnego ciągu cyfr (z wyłączeniem 0), małych i dużych liter. \*/
- B. /\* Pole nazwa powinno składać się w kolejności: z ciągu cyfr (z wyłączeniem 0), następnie dużej litery i ciągu małych liter. \*/
- C. /\* Pole nazwa musi składać się w kolejności: z ciągu cyfr (z wyłączeniem 0), następnie duże litery i dwóch małych liter. \*/
- D. /\* Pole nazwa może zawierać dowolny ciąg cyfr (z wyłączeniem 0), następnie musi zawiera dużą literę i ciąg minimum dwóch małych liter. \*/