

Expresiile regulate în Java

Tabel – Simboluri utilizate în expresii regulate Java

Simbol	Semnificație	Context/Detalii	Exemplu	Ce face expresia
^	Începutul șirului	în afara []	^abc	șirul începe cu abc
^	Negare	în interiorul []	[^0-9]	orice caracter în afară de cifră
\$	Sfârșitul șirului	-	abc\$	șirul se termină cu abc
.	Orice caracter	în afara [], nu include newline	a.b	potrivire aXb, X fiind orice caracter
*	De zero sau mai multe ori	repetă caracterul anterior	ab*	a, ab, abb, abbb, etc.
+	De una sau mai multe ori	repetă caracterul anterior	ab+	ab, abb, abbb, nu doar a
?	Opțional	caracterul anterior apare o dată sau deloc	ab?	a sau ab
[]	Set de caractere	definește o clasă de caractere	[abc]	oricare dintre a, b, c
[^...]	Set negat	exclue caracterele	[^abc]	orice caracter diferit de a, b, c
-	Interval	în interiorul []	[a-z]	orice literă mică
()	Grupare	grupează expresii	(ab)+	ab, abab, ababab etc.
	Sau logic	alternativă între expresii	abc def	abc sau def
\	Escape	pentru caractere speciale	\\.	căutare caracter .
{n}	Exact n apariții	-	a{3}	aaa
{n,}	Cel puțin n apariții	-	a{2,}	aa, aaa, aaaa...
{n,m}	Între n și m apariții	-	a{2,4}	aa, aaa, aaaa

Exemple suplimentare utile

[A-Za-z0-9] – literă sau cifră

\d – echivalent cu [0-9] (digit)

\D – orice caracter non-cifră

\w – echivalent cu [A-Za-z0-9_] (word)

\W – orice caracter non-word ((spațiu), !, ,, ,, ?, @)

\s – spațiu alb (spațiu, tab, newline)

\S – orice caracter non-spațiu

Exemple în Java

"abc123".matches("^[a-z]+\d+\$") // true – litere mici urmate de cifre

"123-45-6789".matches("\d{3}-\d{2}-\d{4}") // true – format SSN

1. Pentru a valida **un număr pozitiv cu opțional două cifre după virgulă** în Java, poți folosi următorul regex. Acest regex va valida numere întregi și numere cu una sau două zecimale. Dacă nu sunt specificate zecimalele, acestea sunt opționale.

`^[0-9]+(?:\.[0-9]{1,2})?$`

Sau:

`^\d+(?:\.\d{1,2})?$`

Explicație:

- `^` - începutul șirului
- `[0-9]+` - unul sau mai multe cifre între 0 și 9
- `(?:\.[0-9]{1,2})?` - un punct și exact una sau două cifre după punct; `(?: ...)` este o grupare non-capturatoare
- `$` - sfârșitul șirului

2. Pentru a valida denumirea sistemului de operare și versiunile acestuia, poți folosi următorul regex. Acest regex va valida denumiri de sistem de operare precum "Windows 11", "Windows Home", "macOS 10", "Linux Ubuntu", etc.

`^[A-Za-z]+(?:[0-9]+)?(?:[A-Za-z]+)?$`

Explicație:

- `^` - începutul șirului
- `[A-Za-z]+` - o serie de una sau mai multe litere (pentru denumirea sistemului de operare)
- `(?:[0-9]+)?` - un spațiu urmat de una sau mai multe cifre, opțional (pentru versiunea sistemului de operare)
- `(?:[A-Za-z]+)?` - un spațiu urmat de o serie de una sau mai multe litere, opțional
- `$` - sfârșitul șirului

3. Pentru validarea numelui autorului, poți folosi următorul regex. Acest regex va valida numele autorului, permițând nume cu un singur cuvânt sau cu mai multe cuvinte separate prin spații.

`^[A-Za-z]+(?:[A-Za-z]+)*$`

Sau:

`^[A-Za-z]+(?:\s+[A-Za-z]+)*$`

Explicație:

- `^` - începutul șirului
- `[A-Za-z]+` - o serie de una sau mai multe litere pentru primul nume
- `(?:[A-Za-z]+)*` - zero sau mai multe grupuri de litere urmate de spațiu, pentru nume suplimentare
- `$` - sfârșitul șirului

4. Pentru validarea denumirii cărții, poți folosi următorul regex. Acest regex va valida denumirile cărților care conțin litere, cifre și o serie de caractere speciale, precum și spații.

`^[A-Za-z0-9 !@#$%^&*()_+ -= , . ; : ' " \ \ /] + $`

Explicație:

- `^` - începutul șirului
- `[A-Za-z0-9 !@#$%^&* ()_+==, . ; : ' " \ \ /] +` - o serie de una sau mai multe litere, cifre și caractere speciale acceptate pentru denumirea cărții
- `$` - sfârșitul șirului

5. Pentru validarea unui număr de telefon format doar din cifre și poate să conțină orice număr de cifre, fără alte caractere speciale, poți folosi următorul regex. Această expresie regulată va valida un număr de telefon care conține doar cifre și nu permite alte caractere, cum ar fi spații sau caractere speciale.

`^[0-9]+$`

Explicație:

- `^` – începutul șirului
- `[0-9]+` – una sau mai multe cifre
- `$` – sfârșitul șirului

6. Pentru a valida un număr de telefon cu formatul specific **022-xx-xx-xx**, unde **x** reprezintă orice cifră, poți folosi următorul regex:

`^022-(?:\d{2}-){2}\d{2}$`

Explicație:

- `^` – începutul șirului
- `022-` – specifică prefixul specificat pentru numărul de telefon
- `(?:\d{2}-){2}` – grup non-capturabil care corespunde exact două grupuri de două cifre urmate de cratimă, repetate de două ori (aceasta asigură că avem exact trei grupuri de două cifre separate prin cratime)
- `\d{2}` – două cifre pentru ultimele două grupuri
- `$` – sfârșitul șirului

7. Pentru a valida un număr de telefon cu formatul specific **+373 - xxx - xx - xxx**, unde **x** reprezintă orice cifră, poți folosi următorul regex:

`^\+373 - \d{3} - \d{2} - \d{3}$`

Sau:

`^\+373\s-\s\d{3}\s-\s\d{2}\s-\s\d{3}$`

Explicație:

- `^` - începutul șirului
- `\+373 -` - specifică prefixul specificat pentru numărul de telefon, inclusiv spațiile și cratimele
- `\d{3}` - trei cifre pentru primul grup de cifre
- `\d{2}` - două cifre pentru al doilea grup de cifre
- `\d{3}` - trei cifre pentru al treilea grup de cifre
- `$` - sfârșitul șirului

Exerciții practice

1. Scrie o expresie regulată care validează un șir ce conține doar cifre (una sau mai multe).
2. Scrie o expresie regulată care validează un cod poștal format din exact 6 cifre.
3. Scrie o expresie regulată care verifică dacă un șir începe cu litera A și urmează orice caractere.
4. Scrie o expresie regulată care verifică dacă un șir NU începe cu o cifră.
5. Scrie o expresie regulată care validează un număr de telefon în formatul 07xx-xxx-xxx.
6. Scrie o expresie regulată care acceptă doar șiruri ce conțin litere mici și mari și cifre (dar fără simboluri speciale).
7. Scrie o expresie regulată care validează un email simplu (ex: ceva@domeniu.md).
8. Găsește șiruri care se potrivesc expresiei `^[0-9].*` și explică de ce.
9. Scrie o expresie regulată care acceptă doar cuvinte de exact 5 caractere.
10. Scrie o expresie regulată care permite litere și spații, dar exclude cifre.