

# EXPRESIONES REGULARES EN JAVA — GUÍA VISUAL

## 1. CLASES DE CARACTERES (Character Classes)

Patrón	Significado
[abc]	Cualquier carácter 'a', 'b' o 'c'
[^abc]	Cualquier carácter excepto 'a', 'b', 'c'
[a-z]	Cualquier letra minúscula de la 'a' a la 'z'
[A-Z]	Cualquier letra mayúscula
[0-9]	Cualquier dígito
[a-zA-Z0-9_]	Cualquier carácter alfanumérico típico
[a-z&&[aeiou]]	Intersección: solo vocales minúsculas
[a-z&&[^aeiou]]	Sustracción: consonantes minúsculas

Ejemplos en Java:

`Pattern.compile("[A-Z]")` — una letra mayúscula

`Pattern.compile("[a-zA-Z0-9_]")` — carácter alfanumérico

## 2. CLASES PREDEFINIDAS

Patrón	Significado
\d	Dígito (equivale a [0-9])
\w	Alfanumérico (letra, dígito, guion bajo)
\s	Espacio en blanco (espacio, tab, salto de línea)
\D	No dígito
\W	No alfanumérico
\S	No espacio en blanco

IMPORTANTE en Java: en los literales de cadena debes usar doble barra invertida.

`Pattern.compile("\\d+")` — uno o más dígitos

## 3. CUANTIFICADORES

Patrón	Significado
x*	0 o más
x+	1 o más

<code>x?</code>	0 o 1
<code>x{n}</code>	Exactamente n repeticiones
<code>x{n,}</code>	Al menos n repeticiones
<code>x{n,m}</code>	Entre n y m repeticiones

Ejemplos en Java:

`Pattern.compile("[A-Z]{2,4}")` — entre 2 y 4 mayúsculas

`Pattern.compile("\\d{3}-\\d{2}-\\d{4}")` — formato tipo 000-00-0000

## 4. GRUPOS Y CAPTURA

- `(abc)` → Grupo de captura
- `(a|b|c)` → Alternancia dentro de un grupo
- `(\\d{2})-(\\d{2})` → dos grupos que capturan pares de dígitos

Ejemplo:

`Pattern.compile("(\\d{2})/(\\d{2})/(\\d{4})")`

Coincide con fechas tipo 25/12/2025 y permite extraer día, mes y año.

## 5. ANCLAS (ANCHORS)

Patrón	Significado
<code>^</code>	Inicio de la cadena
<code>\$</code>	Fin de la cadena
<code>\\b</code>	Límite de palabra
<code>\\A</code>	Inicio absoluto
<code>\\Z</code>	Fin absoluto

Ejemplos:

`Pattern.compile("^Hola")` — debe empezar con 'Hola'

`Pattern.compile("fin$")` — debe terminar con 'fin'

`Pattern.compile("\\bcat\\b")` — coincide con la palabra 'cat' completa