

Cập nhật tháng 8 năm 2024

[Bài đọc] Quy tắc viết biểu thức chính quy

1. Thư viện hỗ trợ
- Java cung cấp hỗ trợ biểu thức chính quy thông qua gói `java.util.regex`, gồm hai lớp chính:
 - Pattern**: biên dịch biểu thức chính quy
 - Matcher**: áp dụng biểu thức với chuỗi cụ thể

2. Cú pháp biểu thức chính quy cơ bản

Ký hiệu	Ý nghĩa
.	Bất kỳ ký tự nào (trừ dòng mới)
\d	Một chữ số (0-9)
\D	Một ký tự không phải là chữ số
\w	Một ký tự từ (a-z, A-Z, 0-9, _)
\W	Ký tự không phải từ
\s	Ký tự trắng (space, tab,...)
\S	Ký tự không phải trắng
^	Bắt đầu chuỗi
\$	Kết thúc chuỗi
[...]	Một tập hợp ký tự
[^...]	Tập hợp phủ định
a?	Có hoặc không có a
a*	0 hoặc nhiều a
a+	1 hoặc nhiều a
a{n}	Chính xác n lần
a{n,}	Ít nhất n lần

a{n,m}	Từ n đến m lần
--------	----------------

- **Lưu ý:** Trong Java, nhiều ký tự đặc biệt cần được **thoát bằng dấu **. Ví dụ: để viết \d bạn phải dùng "\\d" trong chuỗi Java

3. Ví dụ mã Code Java

```

RegexExample.java

import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class RegexExample {
    public static void main(String[] args) {
        String input = "Email: user123@gmail.com";
        Pattern pattern = Pattern.compile("\\w+@\\w+\\.com");
        Matcher matcher = pattern.matcher(input);

        if (matcher.find()) {
            System.out.println("Tìm thấy email: " + matcher.group());
        } else {
            System.out.println("Không tìm thấy email.");
        }
    }
}

```

4. Một số biểu thức thường dùng

Mục đích	Biểu thức chính quy (Java String)
Số nguyên	"^-\?\d+\$"
Email	"^[\\w.-]+@[\\w.-]+\\.\\w{2,}\$"
Mật khẩu mạnh	"^(?=.*[A-Z])(?=.*\\d)(?=.*[@#\$%^&+=]).{8,}\$"

5. Một số lưu ý

- Luôn kiểm tra kỹ chuỗi biểu thức để đảm bảo thoát đúng ký tự
- Dùng **Pattern.compile()** một lần để tăng hiệu suất
- Có thể kết hợp với **replaceAll()**, **split()**, v.v

Tài nguyên đọc thêm: https://www.w3schools.com/java/java_regex.asp

