

Biểu Thức Chính Quy Tham Khảo Nhanh

Pattern, quantifier, group, lookahead, flag

Pattern Cơ Bản

Metacharacter

- Bất kỳ ký tự nào (trừ xuống dòng)
- ^ Đầu chuỗi / dòng
- \$ Cuối chuỗi / dòng
- * 0 hoặc nhiều lần trước đó
- + 1 hoặc nhiều lần trước đó
- ? 0 hoặc 1 lần trước đó (tùy chọn)
- \ Escape metacharacter

Khớp Ký Tự

```
hello # matches "hello" exactly
a.c   # matches "abc", "alc", "a-c", etc.
.txt  # matches literal ".txt"
```

Lớp Ký Tự

Biểu Thức Ngoặc

```
[abc]   Khớp a, b hoặc c
[^abc]  Khớp bất kỳ ký tự nào trừ a, b, c
[a-z]   Chữ cái thường
[A-Z]   Chữ cái hoa
[0-9]   Chữ số
[a-zA-Z0-9]  Chữ số hoặc chữ cái
```

Lớp Viết Tắt

```
\d     Chữ số [0-9]
\D     Không phải chữ số [^0-9]
\w     Ký tự từ [a-zA-Z0-9_]
\W     Không phải ký tự từ
\s     Khoảng trắng [ \t\n\r\f]
\S     Không phải khoảng trắng
```

Quantifier

Quantifier Tham Lam

```
*     0 hoặc nhiều (tham lam)
+     1 hoặc nhiều (tham lam)
?     0 hoặc 1 (tham lam)
{n}   Chính xác n lần
{n,}  n lần trở lên
{n,m} Từ n đến m lần
```

Quantifier Lười

```
*?    0 hoặc nhiều (lười / non-greedy)
+?    1 hoặc nhiều (lười)
??    0 hoặc 1 (lười)
{n,}? Từ n đến m (lười)
```

Quantifier lười khớp ít ký tự nhất có thể

Tham Lam vs Lười

```
<.+> # greedy: "<b>bold</b>"
<.+?> # lazy: "<b>"
```

Anchor

```
^     Đầu chuỗi (hoặc dòng với flag m)
$     Cuối chuỗi (hoặc dòng với flag m)
\b    Ranh giới từ
\B    Không phải ranh giới từ
\A    Đầu chuỗi (không bị ảnh hưởng bởi m)
\Z    Cuối chuỗi (không bị ảnh hưởng bởi m)
```

Ví Dụ Anchor

```
^Hello # starts with "Hello"
world$ # ends with "world"
\bword\b # "word" as whole word
\Bword\B # "word" inside another word
```

Group & Alternation

Capturing Group

```
(abc) # capture group: match "abc"
(a|b|c) # alternation: a or b or c
(cat|dog) # match "cat" or "dog"
(\d{3})-(\d{4}) # groups: "123-4567"
```

Các Loại Group

```
(pattern) Capturing group
(?:pattern) Non-capturing group
(?P<name>pat) Named group (Python)
(?<name>pat) Named group (JS, .NET)
\1 \2 Backreference đến group 1, 2
a|b Alternation: a hoặc b
```

Lookahead & Lookbehind

```
(?=pattern) Lookahead dương
(?!pattern) Lookahead âm
(?<=pattern) Lookbehind dương
(?<!pattern) Lookbehind âm
```

Ví Dụ Lookaround

```
\d+(?= USD) # digits followed by " USD"
\d+(?! USD) # digits NOT followed by " USD"
(?<=\$)\d+ # digits preceded by "$"
(?!\$)\d+ # digits NOT preceded by "$"
```

Lookaround khớp vị trí mà không tiêu thụ ký tự

Pattern Phổ Biến

```
\d{1,3}(\.\d{1,3}){3} Địa chỉ IPv4 (cơ bản)
[\w.+-]+@[ \w.-]+\.[ \w.]+ Email (cơ bản)
https?://[\w.-/ \-?&#]=]+ URL (cơ bản)
\(?\d{3}\)?[-.\s]?\d{3}[-.\s]?\d{4} Số điện thoại Mỹ
\d{4}-\d{2}-\d{2} Ngày (YYYY-MM-DD)
#[0-9a-fA-F]{6} Mã màu hex
```

Đây là pattern đơn giản hóa; dùng thực tế có thể cần xác thực chặt hơn

Flag

```
g Global: tìm tất cả kết quả, không chỉ kết quả đầu
i Không phân biệt hoa thường
m Multiline: ^ / $ khớp ranh giới dòng
s Dotall: . khớp cả xuống dòng
x Verbose: bỏ qua khoảng trắng, cho phép comment
u Unicode: hỗ trợ Unicode đầy đủ
```

Cách Dùng Flag Theo Ngôn Ngữ

```
/pattern/gi # JavaScript
re.compile(r"pat", re.I | re.M) # Python
grep -iE "pattern" # grep (extended)
```