

정규 표현식 빠른 참조

패턴, 수량자, 그룹, 전방탐색, 플래그

기본 패턴

메타 문자

- 임의의 문자 (개행 제외)
- ^ 문자열 / 줄의 시작
- \$ 문자열 / 줄의 끝
- * 앞의 것이 0회 이상
- + 앞의 것이 1회 이상
- ? 앞의 것이 0회 또는 1회 (선택)
- \ 메타 문자 이스케이프

리터럴 매칭

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

문자 클래스

대괄호 표현식

```
[abc] a, b, 또는 c 매칭
[^abc] a, b, c를 제외한 모든 것 매칭
[a-z] 소문자
[A-Z] 대문자
[0-9] 숫자
[a-zA-Z0-9] 영숫자
```

단축 클래스

```
\d 숫자 [0-9]
\D 비숫자 [^0-9]
\w 단어 문자 [a-zA-Z0-9_]
\W 비단어 문자
\s 공백 [ \t\n\r\f]
\S 비공백
```

수량자

탐욕적 수량자

```
* 0회 이상 (탐욕적)
+ 1회 이상 (탐욕적)
? 0회 또는 1회 (탐욕적)
{n} 정확히 n회
{n,} n회 이상
{n,m} n회에서 m회 사이
```

게으른 수량자

```
*? 0회 이상 (게으른 / 비탐욕적)
+? 1회 이상 (게으른)
?? 0회 또는 1회 (게으른)
{n,m}? n회에서 m회 사이 (게으른)
```

게으른 수량자는 가능한 적은 문자를 매칭

탐욕적 vs 게으른

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

앵커

```
^ 문자열 시작 (m 플래그로 줄 시작)
$ 문자열 끝 (m 플래그로 줄 끝)
\b 단어 경계
\B 비단어 경계
\A 문자열 시작 (m 영향 없음)
\Z 문자열 끝 (m 영향 없음)
```

앵커 예시

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

그룹 및 교대

캡처 그룹

```
(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"
```

그룹 유형

```
(pattern) 캡처 그룹
(?:pattern) 비캡처 그룹
(?P<name>pat) 이름 있는 그룹 (Python)
(?<name>pat) 이름 있는 그룹 (JS, .NET)
\1 \2 그룹 1, 2 역참조
a|b 교대: a 또는 b
```

전방탐색 및 후방탐색

```
(?=pattern) 긍정 전방탐색
(?:pattern) 부정 전방탐색
(?<=pattern) 긍정 후방탐색
(?!pattern) 부정 후방탐색
```

탐색 예시

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

탐색은 문자를 소비하지 않고 위치를 매칭

일반 패턴

```
\d{1,3}(\.\d{1,3}){3} IPv4 주소 (기본)
[\w.+-]+@[ \w-]+\.[ \w.]+ 이메일 (기본)
https?://[\w. /-?&#]=]+ URL (기본)
\((?\d{3}\)?[-.\s]?(\d{3})[-.\s]?(\d{4}) 미국 전화번호
\d{4}-\d{2}-\d{2} 날짜 (YYYY-MM-DD)
#?[0-9a-fA-F]{6} 16진수 색상 코드
```

이는 간소화된 패턴; 실제 사용 시 더 엄격한 검증 필요

플래그

```
g 전역: 첫 번째가 아닌 모든 매치 찾기
i 대소문자 무시 매칭
m 여러 줄: ^ / $가 줄 경계 매칭
s Dotall: .이 개행도 매칭
x 상세: 공백 무시, 주석 허용
u 유니코드: 전체 유니코드 지원
```

언어별 플래그 사용

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