정규 표현식

Update – 2018.07

Contents

- 정규표현식 개요
- 정규표현식 패턴
- 정규표현식 모듈
- 메타문자
- 정규표현식 함수

정규표현식 개요

- 정규 표현식(Regular Expressions) ?
  - 정규식, 패턴
  - 복잡한 문자열을 처리할 때 사용하는 기법
  - 특정한 규칙을 가진 문자열의 집합을 표현하는 데 사용하는 형식 언어
  - 데이터 검증, 데이터 스크래핑(일반적으로 웹 스크래핑),
  - 문법 강조 시스템 개발등에 응용

- 정규표현식 제공 언어
  1. 자바, 파이썬, POSIX C, C++
  2. Perl, 자바스크립트, 루비

정규 표현식 관련 파이썬 링크
- https://docs.python.org/3.6/howto/regex.html

정규표현식 패턴

문자 클래스 [ ]

[pattern] - 글자단위

[pattern]+ - 단어단위

[^pattern] - pattern이 아닌것

- A-Z 대문자 a-z 소문자
- 0-9 숫자
- ㄱ-힣 한글
- ^패턴 : 패턴이 아닌 것 . not

정규표현식 패턴

도트(.) : 1글자 이상

```
a.b
```

- "aab"는 가운데 문자 "a"가 모든 문자를 의미하는 `.`과 일치하므로 정규식과 매치.
- "a0b"는 가운데 문자 "0"가 모든 문자를 의미하는 `.`과 일치하므로 정규식과 매치
- "abc"는 "a"문자와 "b"문자 사이에 어떤 문자라도 하나는 있어야 하는 이 정규식과 일치하지 않으므로 매치되지 않는다

정규표현식 패턴

반복(*) : 앞글자의 반복수. 0번이상 반복

```
ca*t
```

- "ct"는 "a"가 0번 반복되어 매치
- "cat"는 "a"가 0번 이상 반복되어 매치 (1번 반복)
- "caaat"는 "a"가 0번 이상 반복되어 매치 (3번 반복)

정규표현식 패턴

반복(+) : 앞글자의 반복. 1번이상 반복

```
ca+t
```

- "ct"는 "a"가 0번 반복되어 매치되지 않음
- "cat"는 "a"가 1번 이상 반복되어 매치 (1번 반복)
- "caaat"는 "a"가 1번 이상 반복되어 매치 (3번 반복)

정규표현식 패턴

반복 ({숫자}) : 앞글자가 숫자만큼 반복

```
ca{2}t
```

- "cat"는 "a"가 1번만 반복되어 매치되지 않음
- "caat"는 "a"가 2번 반복되어 매치

정규표현식 패턴

반복 ({m,n}) : 앞글자가 m~n숫자만큼 반복

```
ca{2,5}t
```

- "cat"는 "a"가 1번만 반복되어 매치되지 않음
- "caat"는 "a"가 2번 반복되어 매치
- "caaaaat"는 "a"가 5번 반복되어 매치

정규표현식 패턴

반복 (?) : 앞글자가 0번 이상반복

```
ab?c
```

- "abc"는 "b"가 1번 사용되어 매치
- "ac"는 "b"가 0번 사용되어 매치

메타문자

- 메타문자란 특별한 기능을 하는 문자
- ₩(d/D/s/S/w/W)
  - ₩d : 10진수, [0-9]와같음
  - ₩D: 10진수외 , [^0-9]
  - ₩s :공백문자 , [₩t₩n₩r₩f₩v]
  - ₩S: 공백 문자 외, [₩t₩n₩r₩f₩v]
  - ₩w : 영문자숫자 , [a-zA-Z0-9]
  - ₩W: 영문자숫자외 , [^a-zA-Z0-9]

메타문자

- ^:문자열의 처음을 나타냄
- $:문자열의 끝
- . :임의의 한 문자
- *:바로 앞의 문자가 없거나 하나 이상 있음
- **+:**바로 앞의 문자가 하나 이상 있음
- **?:**앞의 문자가 없거나 하나임

메타문자

- ( ) (손톱묶음, 소괄호)

  : 손톱묶음 안의 문자열은 하나로 묶어서 다룸

식	뜻	맞는 문자열
(ws)p	p 앞에 ws가 붙음	s
(ws){2}p	s 앞에 ws가 두 번 붙음	wswsp
[^(web)]p	p 앞에 web이 아닌 문자나 문자열이 옴	ap, zzp, wp, wep, webpp, …

메타문자

- **| (위아랫금, 수직선)**

  : '또는', 'or'과 같은 뜻으로 선택문에 쓰임

식	뜻	맞는 문자열
word|phrase	word 또는 phrase	word, phrase
mount(ed|ing)	mounted 또는 mounting	mounted, mounting
[^(a|b|c)].+	a 또는 b 또는 c로 시작하지 않는 문자열	

메타문자

- ₩ (거꿀빗금)

  : 메타 문자의 성질을 없앨 때 붙임

식	뜻	맞는 문자열	안 맞는 문자열
₩[[^₩[₩]]+₩]	꺾쇠묶음으로 한 겹만 싸인 문자열	[a], [ab], [abc], …	[], [[abcd]], [a[]], …
[^\.]+\.(com|net)	com, net로 끝나는 바탕 도메인 이름 (서브 도메인이 아닌 것)	a.com, b.net, …	b.a.com, .a.com, .net

정규표현식 모듈

파이썬에서 정규 표현식을 지원하는 re 모듈

```
>>> import re
```

```
>>> result = re.findall('ab*','abc ab abb ann')
>>> print(result)
```

['ab', 'ab', 'abb', 'a']

정규표현식 함수

re.compile('정규표현식')

- 정규표현식을 컴파일한다.
- 컴파일 변수는 .match() 등과 함께 사용된다.

```
>>> p = re.compile('[a-z]')
result= p.match('ABC')
print(result)
```

None

```
>>> p = re.compile('[a-z]')
result= p.match('abc')
print(result)
```

<_sre.SRE_Match object; span=(0, 1), match='a'>

정규표현식 함수

정규표현식컴파일변수.match(문자열)

- Compile() 함수를 이용하여 컴파일된 정규표현식 변수와 도트(.)을 이용하여 문자열이 정규표현식에 맞는지 검사한다.
- none이나 Match object 리턴

```
>>> p = re.compile('[a-z]')
result= p.match('abc')
print(result)
```

```
<_sre.SRE_Match object; span=(0, 1), match='a'>
```

정규표현식 함수

정규표현식컴파일변수. sub(a,b)

- Compile() 함수를 이용하여 컴파일된 정규표현식 변수와 도트(.)을 이용하여 정규표현식에 맞는 글자를 특정 글자로 교체한다.

```
>>> p = re.compile('ab')
result = p.sub('kong', 'ab abcd abfg')
print(result)
```

kong kongcd kongfg

정규표현식 함수

re.findall(정규표현식,문자열)

- 문자열에서 정규표현식에 맞는 패턴을 찾아 리스트로 리턴한다.

```
>>> result = re.findall('ab*','abc ab abb ann')
print(result)
```

['ab', 'ab', 'abb', 'a']