

# 언어별 정규표현식, 조사할 내용

작성일: 2020.03.05

작성자: 도원진

## 1. 패턴

### php

종류	정규식	비고
E-Mail	<code>^[a-zA-Z0-9]+@[a-zA-Z0-9]+\$</code>	
URL	<code>/(http:\\V)?([a-zA-Z0-9-]+.[a-zA-Z0-9-]+)/</code>	
ID	<code>/^[a-z0-9_]\w{5,20}\$/</code>	시작은 영문으로만, "를 제외한 특수문자 안되며 영문, 숫자, "_"으로만 이루어진 5 ~ 12자 이하
주민번호	<code>\d{6} - [1-4]\d{6}</code>	
정수	<code>\-?\d{0, }</code>	
숫자(음이 아닌 정수)	<code>/^[0-9]*\$/</code>	
알파벳	<code>^[a-zA-Z]*\$</code>	
대,소문자 구분		
IP주소	<code>/([0-9]{1,3}) . ([0-9]{1,3}) . ([0-9]{1,3}) . ([0-9]{1,3})/</code>	

한글	시[가-힣]*\$
휴대폰	<code>^01(?:0 1 [6-9]) - (?:\d{3} \d{4}) - \d{4}\$</code>
일반전화	<code>^\d{2,3} - \d{3,4} - \d{4}\$</code>

### MySQL

- Email

```
SELECT *
FROM [DB_name].[Table_name]
WHERE e_mail REGEXP "^[a-zA-Z0-9._-]+@[a-zA-Z0-9-]+\.[a-zA-Z.]{2,10}$"
```

- URL

```
SELECT field FROM table
WHERE field REGEXP "^(https?://|www\\.\\.)([\\.A-Za-z0-9\\-]+\\.[a-zA-Z]{2,4})"
```

- ID

- 

```
SELECT * FROM BOARD_CONTENT WHERE CONTENT RLIKE '[0-9]{6}-[0-9]{7}';
```

## JavaScript

[영문 대문자 또는 소문자로 시작하는 아이디, 끝날때 영문 대문자 또는 소문자]

```
var idReg = /^[A-Za-z]$/g;
```

[영문 대문자 또는 소문자로 시작하는 아이디, 끝날때 제한 없음]

```
var idReg = /^[A-Za-z]/g;
```

[영문 대문자 또는 소문자 또는 숫자로 시작하는 아이디, 끝날때 영문 대문자 또는 소문자 또는 숫자]

```
var idReg = /^[A-Za-z0-9]$/g;
```

[영문 대문자 또는 소문자 또는 숫자로 시작하는 아이디, 끝날때 제한 없음]

```
var idReg = /^[A-Za-z0-9]/g;
```

```
##
```

## java

아래 링크 참고

<https://github.com/wonjin-do/BizSpring/tree/master/php%EC%8B%A4%EC%8A%B5/reg>

## 2. 브라우저 에이전트 판별방식

- 판별방법
  - 브라우저에서 보내온 사용자 에이전트(User Agent) 문자열을 분석/판별
- PHP
  - 다른 언어 request 에 해당하는것이 php 에서는 대부분 \$\_SERVER[] 에 있습니다.

```
$_SERVER["REMOTE_ADDR"] // 사용자 IP
$_SERVER["HTTP_REFERER"] // 이전 페이지
$_SERVER["HTTP_USER_AGENT"] // 브라우저 정보
```

```
01. $userAgent = $_SERVER["HTTP_USER_AGENT"];
02.
```

```

03. if ( preg_match("/MSIE*/", $userAgent) ) {
04.     // 익스플로러
05.
06.     if ( preg_match("/MSIE 6.0[0-9]*/", $userAgent) ) {
07.         $browser = "Explorer 6";
08.     }elseif ( preg_match("/MSIE 7.0*/", $userAgent) ) {
09.         $browser = "Explorer 7";
10.     }elseif ( preg_match("/MSIE 8.0*/", $userAgent) ) {
11.         $browser = "Explorer 8";
12.     }elseif ( preg_match("/MSIE 9.0*/", $userAgent) ) {
13.         $browser = "Explorer 9";
14.     }elseif ( preg_match("/MSIE 10.0*/", $userAgent) ) {
15.         $browser = "Explorer 10";
16.     }else{
17.         // 익스플로러 기타
18.         $browser = "Explorer ETC";
19.     }
20.
21. } elseif ( preg_match("/Trident*/", $userAgent && preg_match("/rv:11.0*/",
$userAgent &&
preg_match("/Gecko*/", $userAgent) ) ) ) {
22.     $browser = "Explorer 11";
23.
24. } elseif ( preg_match("/(Mozilla)*/", $userAgent) ) {
25.     // 모질라 (파이어폭스)
26.     $browser = "mozilla";
27.
28. } elseif ( preg_match("/(Nav|Gold|X11|Mozila|Nav|Netscape)*/", $userAgent) )
{
29.     // 네스케이프, 모질라(파이어폭스)
30.     $browser = "Netscape/mozilla";
31.
32. } elseif ( preg_match("/Opera*/", $userAgent) ) {
33.     // 오페라
34.     $browser = "Opera";
35.
36. } else {
37.     $browser = "Other";
38. }
39.
40. echo $browser;

```

## 3. 조사할 내용

### 3-1. URL, URI

- URI 가 더 넓은 범위
  - 자원에 접근할 때, identifier (식별자)를 통해 접근
- URL
  - 자원에 접근할 때, 위치를 기반으로 접근
- 두 url을 비교해보자
  - <http://test.com/test.pdf?docid=111> vs <http://test.com/test.pdf?docid=112>
  - 결론

- 동일 URL : <http://test.com/test.pdf>  
다른 URI : 뒤의 쿼리(식별자)가 다름

## 3-2. PATH

```
$file_server_path = realpath(__FILE__);
// PHP 파일 이름이 들어간 절대 서버 경로

$php_filename = basename(__FILE__);
// PHP 파일 이름

$server_path = str_replace(basename(__FILE__), "", $file_server_path);
// PHP 파일 이름을 뺀 절대 서버 경로

$server_root_path = $_SERVER['DOCUMENT_ROOT'];
// 서버의 웹 뿌리(루트) 경로(절대 경로)

$relative_path = eregi_replace("\\[/\\^/]*\\.php$", "/", $_SERVER['PHP_SELF']);
$relative_path = preg_replace("\\[/\\^/]*\\.php`i", "/", $_SERVER['PHP_SELF']);
// 웹 문서의 뿌리 경로를 뺀 상대 경로

$relative_file_server_path = $relative_path.$php_filename;
// PHP 파일 이름이 들어간 상대 경로

$base_URL = ($_SERVER['HTTPS'] == 'on') ? 'https://' : 'http://';
$base_URL .= ($_SERVER['SERVER_PORT'] != '80') ?
$_SERVER['HTTP_HOST'].':'.$_SERVER['SERVER_PORT'] : $_SERVER['HTTP_HOST'];
// 바탕 URL

$web_path = $base_URL.$relative_path;
// PHP 파일이 있는 곳의 웹 경로

$full_URL = $web_path.$php_filename;
$full_URI = $base_URL.$_SERVER['REQUEST_URI'];

echo "문서가 있는 곳의 절대 경로 : ".$server_path;
echo "\n";
echo "문서가 있는 곳의 전체 웹 경로 : ".$web_path;
echo "\n";
echo "문서가 있는 곳의 상대 경로 : ".$relative_path;
echo "\n";
echo "문서의 절대 경로 : ".$file_server_path;
echo "\n";
echo "문서의 상대 경로 : ".$relative_file_server_path;
echo "\n";
echo "문서의 전체 URL : ".$full_URL;
echo "\n";
echo "문서의 전체 URI : ".$full_URI;
```

## 3-3. REFERER

- 요청 헤더의 한 종류
- 기능
  - 웹페이지 방문시 흔적을 남긴다.

- 분석, 로깅, 혹은 캐싱 최적화에 사용될 수도 있다
- 문법

```
Referer: https://www.naver.com
```

## 3-4. Browser Agent

- 상황
  - 소프트웨어 에이전트가 네트워크 프로토콜 안에서 동작할 때
- 동작방식
  - 문자적 식별 [문자열](#)을 피어(peer)에 제출함으로써 종종 자기 자신과 애플리케이션 유형, [운영 체제](#), 소프트웨어 업체, 소프트웨어 리비전을 식별한다.
  - 식별 정보는 *User-Agent*라는 헤더 필드를 통해 전달된다.

## 3-5. 1사 쿠키, 3사 쿠키

- 제 1 사 쿠키
  - 호스트 도메인과 관련된 쿠키입니다.
- 제 3 사 쿠키
  - 다른 도메인의 쿠키입니다.
  - 예를 들어, 주소 표시줄에 URL을 입력하여 [www.example1.com](http://www.example1.com)을 방문하고  
sample.example1.com, [www.example2.com](http://www.example2.com) 및 [www.example1.net](http://www.example1.net)의 배너 광고가 이 사이트에 있다고 가정
  - 이 사이트가 모두 쿠키를 설정한 경우
    - [www.example1.com](http://www.example1.com)과 sample.example1.com 쿠키
      - 제 1 사 컨텍스트에 존재
    - [www.example2.com](http://www.example2.com) 및 [www.example1.net](http://www.example1.net) 쿠키
      - 제 3 사 컨텍스트에 존재