

08

정규표현식

PHP는 정규표현에 관련된 기능과 함수를 제공합니다. 정규표현식은 PHP 언어 이외에도 다른 언어에서 일반적으로 사용하는 패턴을 처리하는 개별 언어의 일종입니다.

PHP는 정규표현에 대해서 perl(PCRE) 방식을 지원합니다. 기존 POSIX 스타일의 정규표현은 PHP 5.3.x 이후에서 삭제되었습니다.

정규표현은 특수한 문자들의 패턴입니다.

08.1 정규 패턴

내장 함수 `preg_match()`는 패턴 정규표현식을 입력된 문자열 `$subject`에서 찾습니다. `preg_match()` 함수는 `subject` 문자열에서 패턴을 매칭하여 결과값을 `$matches` 변수에 배열로 반환합니다.

| 내장 함수 |

```
int preg_match ( string $pattern , string $subject [, array &$matches [, int $flags = 0 [,  
int $offset = 0 ]]] )
```

이 전에 `ereg()` 함수를 사용했으나 PHP 7.x에서 예전 함수는 삭제되었습니다.

- `matches`: 세 번째 인자 `matches`가 있을 경우에는 검색 결과를 배열로 반환합니다.
- `flags`: `PREG_OFFSET_CAPTURE` 값은 매치된 모든 문자열의 시작 위치를 함께 반환합니다.
- `offset`: 문자열 검색의 시작 포인트를 지정할 수 있습니다. 기본값은 0입니다.

예제 파일 | `preg-01.php`

```
1  <?php
2
3      $string = "2017-06-33";
4      $pattern = "/([0-9]{4})-([0-9]{1,2})-([0-9]{1,2})/";
5
6      if (preg_match($pattern, $string,$matches) ) {
7          echo"날짜 형식이 맞습니다.<br>";
8
9          if (checkdate($match[2],$match[3],$match[1])) {
10             echo "유효한 날짜입니다.<br>";
11         } else {
12             echo "날짜 형식은 맞지만, 유효한 날짜는 아닙니다.<br>";
13         }
14     } else {
15         echo"날짜 형식이 다릅니다.<br>";
16     }
17
18  ?>
```

화면 출력

날짜 형식이 맞습니다.

날짜 형식은 맞지만, 유효한 날짜는 아닙니다.

08.2 패턴 변환

내장 함수 `preg_replace()`는 패턴을 찾아서 다음 문자열로 변환합니다. 문자열 `subject`에서 패턴 배열의 값과 같은 값들을 검사하여 `replacement` 문자열로 변환합니다.

| 내장 함수 |

```
mixed preg_replace ( mixed $pattern , mixed $replacement , mixed $subject [, int $limit [, int &$count ]])
```

예제 파일 | `preg-02.php`

```
1  <?php
2      $string = 'jun 21, 2017';
3      $pattern = '/(\w+) (\d+), (\d+)/i';
4      $replacement = '${1} ${2}, ${3}';
5      echo preg_replace($pattern, $replacement, $string);
6      echo "<br>";
7
8      $string = 'hello world! my name is jiny.';
9
10     $patterns[0] = '/hello world/';
11     $patterns[1] = '/my name is/';
12     $patterns[2] = '/jiny/';
13
14     $replacements[2] = '안녕하세요';
15     $replacements[1] = '제 이름은 ';
16     $replacements[0] = '지니';
17
18     echo preg_replace($patterns, $replacements, $string);
19
20  ?>
```

화면 출력

```
jun 21, 2017
안녕하세요! 제 이름은 지니.
```

08.3 패턴 분리

내장 함수 `preg_split()`는 정규표현 패턴에 따라서 문자열을 분리하여 배열로 반환합니다.

| 내장 함수 |

```
array preg_split ( string $pattern , string $subject [, int $limit [, int $flags ] ] )
```

예제 파일 | `preg-03.php`

```
1  <?php
2      // " ", \r, \t, \n, \f를 포함하여
3      $keywords = preg_split("/[\\s,]+/", "php language, programming");
4      echo "임의 개수의 콤마와 스페이스로 구문을 나눕니다. ";
5      print_r($keywords);
6      echo "<br>";
7
8
9      $str = 'string';
10     echo "문자 단위로 분리합니다. ";
11     $chars = preg_split('//', $str, -1, PREG_SPLIT_NO_EMPTY);
12     print_r($chars);
13     echo "<br>";
14
15  ?>
```

화면 출력

임의 개수의 콤마와 스페이스로 구문을 나눕니다. Array ([0] => php [1] => language [2] => programming)

문자 단위로 분리합니다. Array ([0] => s [1] => t [2] => r [3] => i [4] => n [5] => g)

| 내장 함수 |

```
string preg_quote ( string $str [, string $delimiter ] )
```

내장 함수 `preg_quote()`는 정규표현식의 문자를 인용합니다. `preg_quote()` 함수는 입력된 문자열에 대해서 모든 문자 앞에 백슬래시를 추가합니다.

이러한 표현은 정규표현식을 처리하는 문자열 처리할 때 유용합니다.

정규표현식 특수 문자: `. \w + * ? [^] $ () { } = ! < > | :`

예제 파일 | `preg-04.php`

```
1  <?php
2      $keywords = '$123 for a hojin/jiny';
3      $keywords = preg_quote($keywords, '/');
4      echo $keywords;
5      echo "<br>";
6
7      // preg_quote($word)는 정규표현식에서 이스터라이크(*) 처리
8      $body = "안녕하세요 이 책은 매우 *쉽게* 작성을 하였습니다.";
9      $word = "*매우*";
10     $body = preg_replace("/".preg_quote($word)."/", "<i>" . $word . "</i>", $body);
11     echo $body;
12
13  ?>
```

화면 출력

`\$123 for a hojin\jiny`

안녕하세요 이 책은 매우 *쉽게* 작성을 하였습니다.

