

APLIKACJE INTERNETOWE PHP



TEMAT 5-06: Funkcje PHP wykorzystujące wyrażenia regularne.

Autor dokumentu: Wojciech Galiński

środa, 7 września 2016 r.

351203 Technik informatyk

ŹRÓDŁA WIEDZY:

<http://www.php.net/manual/pl>, <http://www.w3schools.com/php/>, <http://pl.wikibooks.org/wiki/PHP>,
<http://www.gajdaw.pl/php/wyrazenia-regularne-pcre-php-tutorial/>, <http://www.antylameriada.net/wphppreg/>.

Zagadnienia obowiązkowe

1. **Zaawansowane zagadnienia dotyczące wyrażen regularnych** – oto wybrane z nich:

→ **zmiana trybu dopasowania** – domyślnie modyfikatory działają na całe wyrażenie, ale modyfikator można zastosować także dla fragmentu wyrażenia umieszczonego w nawiasach okrągłych (w przykładach pokazano włączanie ignorowania wielkości znaków dla fragmentu wyrażenia).

PRZYKŁADY: `/N(?i:a|ie)/`, `/M(?i)ateusz/`.

→ **przechwytywanie** – to zapamiętanie (fragmentu) wyrażenia za pomocą nawiasów okrągłych w tablicy, np.

```
preg_match('/(\d)x/', 'y=3x+3', $tab); print_r($tab);  
preg_match_all('/([=+]*)(\d)([xy]*)/', '2x+4y=3', $tab); print_r($tab);
```

→ **odwołania wsteczne** – to odwołanie się do przechwyconego tekstu w następujący sposób: `\1`, `${1}` – pierwszy przechwycony tekst, `\2`, `${2}` – drugi przechwycony tekst, itd.

PRZYKŁADY: `/(\w+)=('|")\1\2/`, `/(\w+)=(?P<nawias>'|")\1(?P=nawias)/`,
`/(?P<dzien>\d{2})\.(?P<mies>\d{2})/`

2. **Funkcje wykorzystujące wyrażenia regularne** – wszystkie tego typu funkcje dostępne są pod adresem: <http://php.net/manual/pl/ref.pcre.php>. Oto najważniejsze z nich:

→ **preg_match(wzorzec, tekst)** – sprawdza, czy podany wzorzec występuje w tekście (zwraca wartość logiczną: prawda albo fałsz);

PRZYKŁAD:

```
$cyfra = 'dwa';  
if(preg_match('/^[0-9]$/', $cyfra))  
    echo '<p>To jest cyfra "'. $_POST['cyfra']. '"</p>';  
else  
    echo '<p>Nieprawidłowe dane! Wymagana jest cyfra!</p>';
```

→ **preg_match_all(wzorzec, tekst, dopasowania)** – wyszukuje wszystkie wystąpienia wzorca w tekście i wstawia tablicę wyników do trzeciego parametru (tylko wtedy, gdy został podany);

PRZYKŁAD:

```
$tekst = 'Wyszukiwarki popularne w Polsce: '  
        'http://google.pl/, http://onet.pl/, http://wp.pl/.';  
$regexp = '~(http|https|ftp)://([A-Z0-9][A-Z0-9_]*'  
        '(:?[A-Z0-9][A-Z0-9_]*)+):?(d+)?/?~i';  
echo 'Liczba wyników: '. preg_match_all($regexp, $tekst, $wynik);  
echo '<pre>'; print_r($wynik); echo '</pre>';
```

➔ **preg_replace(wzorzec, zamiana_na, zmieniany_tekst)** – zamienia tekst według podanego wyrażenia regularnego (zwraca zamieniony tekst nie modyfikując tekstu oryginalnego);

PRZYKŁAD 1:

```
echo '<p>'. preg_replace('/(a|e|i|o|ó|u|y)/ui', '<b>${1}</b>',  
                        'wykropienie u górnika'). '</p>';
```

PRZYKŁAD 2:

```
$slowo = 'w';  
$tekst = 'Nysa to piękne miasto w Polsce. W Nysie można znaleźć kościoły,'.  
        '<br />w których odczuwamy atmosferę minionych lat.';  
//      | wzorzec do zamiany | |zamiana na| |zmieniany tekst|  
echo preg_replace('/\b('.$slowo.')\b/iu', '<b>\1</b>', $tekst);
```

➔ **preg_last_error** – zwraca wynik zadziałania ostatnio wywoływanej funkcji wykorzystującej wyrażenia regularne (kod 0 oznacza brak błędu);

PRZYKŁAD:

```
preg_match('/(?:\D+|<\d+>)*[!?]/', 'foobar foobar foobar');  
if (preg_last_error())  
    echo '<b>UWAGA:</b> podczas przetwarzania wyrażeń regularnych wystąpił '.  
        'błąd o numerze <b>'. preg_last_error(). '</b>.';
```

3. **Funkcje filtrujące** – filtrują treść według predefiniowanego filtra (lista predefiniowanych filtrów wraz z flagami dostępna jest m. in. pod adresami: <http://pl1.php.net/manual/pl/filter.filters.php>, http://www.w3schools.com/php/php_ref_filter.asp).

Najważniejsza z tych funkcji to:

➔ **filter_var(zmienna, identyfikator_filtra, opcje)** – waliduje treść według wskazanego filtra (opcje nie są obowiązkowe), np.

```
$logiczna = filter_var('yes', FILTER_VALIDATE_BOOLEAN);  
$oct = filter_var('0755', FILTER_VALIDATE_INT, FILTER_FLAG_ALLOW_OCTAL);  
$opcje = array('flags'=>FILTER_FLAG_ALLOW_HEX,  
               'options'=>array('min_range'=>1, 'max_range'=>0x1a));  
$hex = filter_var('0x0f', FILTER_VALIDATE_INT, $opcje);  
$zmiennoprzecinkowa = filter_var('-1.5E-2', FILTER_VALIDATE_FLOAT);  
$email = filter_var('bob@example.com', FILTER_VALIDATE_EMAIL);  
$url = filter_var('http://example.com/', FILTER_VALIDATE_URL,  
                  FILTER_FLAG_PATH_REQUIRED);  
$ip = filter_var('200.150.100.50', FILTER_VALIDATE_IP, FILTER_FLAG_IPV4);  
var_dump($logiczna, $oct, $hex, $zmiennoprzecinkowa, $email, $url, $ip);
```

Inne funkcje filtrujące dostępne są pod adresem: <http://pl1.php.net/manual/pl/ref.filter.php>.

Zadania

1. Zapoznaj się z przedstawionymi przykładami. Wykonaj je i przeanalizuj ich działanie.

2. Ważne wyrażenia regularne:

➔ **Znajdowanie w tekście:**

- ✓ walidacja kodu pocztowego / identyfikatora PESEL / identyfikatora NIP;
- ✓ znajdowanie kolorów RGB w postaci szesnastkowej;
- ✓ walidacja nazwy użytkownika oraz hasła;
- ✓ walidacja adresu e-mail;
- ✓ walidacja adresu URL i adresu IP (IPv4, IPv6).

➔ **Zamiany w tekście:**

- ✓ zaznaczanie frazy (podciągu) w tekście, np. podczas wyszukiwania w tekście;
- ✓ zamiana emotikon tekstowych na graficzne;
- ✓ zamiana BBCode na HTML i odwrotnie.