

Expressions rationnelles

TP 05 du Module 02 – Les bases du langage PHP

Proposition de solution

- Exercice 1

```
<?php
$cb = ["1234 5678 9012 3456", "1234 5678 912 3456", "1234567890123456"];
$cbOk = "#^(\d{4}\s*){4}$#";
foreach ($cb as $num) {
    if (preg_match($cbOk, $num)) {
        echo $num . ' est un numéro de CB correct<br>';
    } else {
        echo $num . ' n\' est pas un numéro de CB correct<br>';
    }
}
```

La classe `\d` correspond à un chiffre. Le quantifieur `{4}` indique que cette classe est répétée quatre fois. Cela correspond donc à un groupe de quatre chiffres.

La classe `\s` correspond à blanc (espace, tabulation ou retour à la ligne). Le quantifieur `*` indique qu'il peut y en avoir entre zéro et l'infini.

Tout ce qui vient d'être cité est regroupé par un couple de parenthèses. Ce groupe est suivi du quantifieur `{4}` indiquant que ce groupe doit être répété quatre fois.

La fonction `preg_match()` permet de tester l'adéquation d'une chaîne de caractères avec une expression rationnelle.

Exécution

```
1234 5678 9012 3456 est un numéro de CB correct
1234 5678 912 3456 n' est pas un numéro de CB correct
1234567890123456 est un numéro de CB correct
```

- Exercice 2

```
<?php
$urls = array('www.machin.com', 'www.machintruc.uk', 'ww.machin.com', 'www.machi
n.bidule.com', 'www.machin-b.com', 'www.machin-truc-bidule.uk', 'ww.machin-
truc.com', 'www.machin.bidule.chose');

foreach ($urls as $url) {
    if (preg_match('#^w{3}\.[a-zA-Z]+([\-\\.][a-zA-Z]+)?\.[a-zA-
Z]{2,4}$#', $url)) {
        echo $url . ' est une URL valide<br>';
    } else {
        echo $url . ' <- NON VALIDE !<br>';
    }
}
```

Les caractères `^` et `$` indiquent le début et la fin de la ligne.

Le caractère `w` se représente lui-même. Le quantifieur `{3}` indique que trois `w` sont nécessaires.

Le caractère suivant doit être un point. Mais le point a une signification particulière



(n'importe quel caractère), il est donc échappé avec un caractère `\`.

L'ensemble `[a-zA-Z]` correspond à un caractère parmi l'intervalle a à z ou parmi l'intervalle A à Z, c'est-à-dire une lettre majuscule ou minuscule. Le quantifieur `+` indique qu'il doit être présent au moins une fois.

L'ensemble `[\-\.]` correspond à un tiret ou un point. Ces deux caractères ont également été échappés (le tiret est utilisé comme vu ci-dessus pour les intervalles). Ensuite il doit y avoir à nouveau une ou plusieurs lettres.

Tout ce qui vient d'être cité est regroupé par un couple de parenthèses. Ce groupe est suivi du quantifieur `?` indiquant que ce groupe est optionnel.

Enfin, il doit à nouveau y avoir un point. Et le quantifieur `{2,4}` indique que ce point doit être suivi par entre deux et quatre lettres.

Exécution

www.machin.com est une URL valide
 www.machintruc.uk est une URL valide
 ww.machin.com <- NON VALIDE !
 www.machin.bidule.com est une URL valide
 www.machin-b.com est une URL valide
 www.machin-truc-bidule.uk <- NON VALIDE !
 ww.machin-truc.com <- NON VALIDE !
 www.machin.bidule.chose <- NON VALIDE !

• Exercice 3

```
<?php
$texte='Fonctionnement\n\nPHP appartient à la grande famille des descendants '.
'du C, dont la syntaxe est très proche. En particulier, sa syntaxe et '.
'sa construction ressemblent à celles des langages Java et Perl, à '.
'ceci près que du code PHP peut facilement être mélangé avec du code '.
'HTML au sein d\'un fichier PHP.\nDans une utilisation destinée à '.
'l\'internet, l\'exécution du code PHP se déroule ainsi : lorsqu\'un '.
'visiteur demande à consulter une page de site web, son navigateur '.
'envoie une requête au serveur HTTP correspondant. Si la page est '.
'identifiée comme un script PHP (généralement grâce à l\'extension '.
'.php), le serveur appelle l\'interprète PHP qui va traiter et générer '.
'le code final de la page (constitué généralement d\'HTML ou de XHTML, '.
'mais aussi souvent de feuilles de style en cascade et de JS). Ce '.
'contenu est renvoyé au serveur HTTP, qui l\'envoie finalement au '.
'client.\n\nSource : Wikipédia (https://fr.wikipedia.org/wiki/PHP)';
echo preg_replace('#\\n#', '<br>', $texte);
```

La fonction `preg_replace()` permet de rechercher un motif et de modifier la valeur des occurrences trouvées.

Le motif à rechercher est `\n`. Tout d'abord, il faut doubler le `\` car dans une expression rationnelle, le caractère `\` est le caractère d'échappement. Mais le caractère `\` est également le caractère d'échappement des chaînes de caractères, il faut donc le tripler.

Il est à noter que la fonction `nl2br()` effectue ce travail. Il est bien évidemment préférable de ne pas recoder ce qui existe déjà !

Exécution

Fonctionnement

PHP appartient à la grande famille des descendants du C, dont la syntaxe est très proche. En particulier, sa syntaxe et sa construction ressemblent à celles des langages Java et Perl, à ceci près que du code PHP peut facilement être mélangé avec du code HTML au sein d'un fichier PHP.

Dans une utilisation destinée à l'internet, l'exécution du code PHP se déroule ainsi : lorsqu'un visiteur demande à consulter une page de site web, son navigateur envoie une requête au serveur HTTP correspondant. Si la page est identifiée comme un script PHP (généralement grâce à l'extension .php), le serveur appelle l'interprète PHP qui va traiter et générer le code final de la page (constitué généralement d'HTML ou de XHTML, mais aussi souvent de feuilles de style en cascade et de JS). Ce contenu est renvoyé au serveur HTTP, qui l'envoie finalement au client.

Source : Wikipédia (<https://fr.wikipedia.org/wiki/PHP>)

• Exercice 4

```
<?php
```

```
$texte = "=== Histoire ===
```

```
Le langage PHP fut créé en [[1994]] par [[Rasmus Lerdorf]] pour son
[[site web]]. C'était à l'origine une [[bibliothèque logicielle]] en
[[C (langage)|C]] dont il se servait pour conserver une trace des visiteurs qui
venaient consulter son [[Curriculum vitae|CV]]. Au fur et à mesure qu'il ajoutait
de nouvelles fonctionnalités, Rasmus a transformé la bibliothèque en une
implémentation capable de communiquer avec des bases de données et de créer des
applications dynamiques et simples pour le [[World Wide Web|Web]]. Rasmus décida
alors en [[1995]] de publier son code, pour que tout le monde puisse l'utiliser
et en profiter. PHP s'appelait alors PHP/FI (pour
'''P'''ersonal '''H'''ome '''P'''age Tools/'''F'''orm '''I'''nterpreter').
En [[1997]], deux étudiants, [[Andi Gutmans]] et [[Zeev Suraski]],
redéveloppèrent le cœur de PHP/FI. Ce travail aboutit un an plus tard à la
version 3 de PHP, devenu alors '''PHP: Hypertext Preprocessor'''. Peu de temps
après, Andi Gutmans et Zeev Suraski commencèrent la réécriture du moteur interne
de PHP. Ce fut ce nouveau moteur, appelé '''[[Zend Engine]]''' – le mot '''Zend'''
est la contraction de '''Ze'''ev et '''A'''nd'''i – qui servit de base à
la version 4 de PHP.
```

```
=== Utilisations ===
```

```
[[Image:Server-side websites programming languages.PNG|thumb|Répartition des "
"langages de programmation côté serveur des sites web, le 28 avril 2016.]]
En [[2002]], PHP est utilisé par plus de 8 millions de sites Web à travers le
monde, en [[2007]] par plus de 20 millions et en 2013 par plus de 244
millions.";
```

```
// conversion des gras et italiques
```

```
$texte = preg_replace('#\\'\\'\\'([\\^\\']+)\\'\\'\\'#', '<b>$1</b>', $texte);
$texte = preg_replace('#\\'\\'([\\^\\']+)\\'\\'#', '<i>$1</i>', $texte);
```

```
// conversion des titres
```

```
$texte = preg_replace_callback('#(={1,6}) (.+?) (={1,6})#', function($matches) {
    $nb = strlen($matches[1]);
    return '<h' . $nb . '>' . $matches[2] . '</h' . $nb . '>';
}, $texte);
```

```
// conversion des images
```

```
$texte = preg_replace_callback('#\\[\\[Image:(.+?)\\]\\.\\.\\.\\]#', function($matches) {
    $file = str_replace(' ', '_', $matches[1]);
    return '';
}, $texte);
```



```
// conversion des liens
$texte = preg_replace_callback('#\\[\\([\\^\\]]+?)\\|([\\^\\]]+?)\\]\\#', function($m) {
    $lien = str_replace(' ', '_', $m[1]);
    return '<a href="https://fr.wikipedia.org/wiki/' . $lien . '">' . $m[2] . '<
/a>';
}, $texte);
$texte = preg_replace_callback('#\\[\\([.\\+?)\\]\\]\\#', function($m) {
    $lien = str_replace(' ', '_', $m[1]);
    return '<a href="https://fr.wikipedia.org/wiki/' . $lien . '">' . $m[1] . '<
/a>';
}, $texte);

echo $texte;
```

Mises en forme

Les éléments en gras ou en italique sont entourés par des apostrophes. Deux pour l'italique et trois pour le gras. Il faut commencer par rechercher d'abord le texte entouré de trois apostrophes car deux apostrophes sont un cas particulier de trois apostrophes.

Pour effectuer ce premier remplacement, la fonction `preg_replace()` est utilisée avec le motif `'#\''\''\''([^\']+?)\''\''\''#'`.

Ce motif correspond à trois apostrophes, au moins un caractère qui n'est pas un apostrophe (le groupe commence par le caractère `^` correspondant à « tout sauf ») puis trois apostrophes.

Une paire de parenthèses a été positionnée autour du texte à mettre en gras. Cela permet de capturer cette valeur et de pouvoir la réutiliser dans le texte de remplacement.

Ce texte est constitué d'une balise ``, du texte capturé et d'une balise fermante ``.

Pour récupérer le texte capturé, il faut utiliser le symbole `$` suivi du numéro du groupe capturé. Dans ce cas, il n'y en a qu'un seul, c'est donc `$1`.

Pour l'italique, il faut procéder de manière similaire avec une apostrophe de moins et la balise `<i>`.

Titres

Il serait possible de procéder de manière similaire pour les titres en commençant par le titre de plus petit niveau ayant donc le plus de signes égal.

Cela se réaliserait avec l'instruction suivante :

```
$texte = preg_replace('#===== ([^=]+) =====#', '<h6>$1</h6>', $texte);
```

puis il faudrait procéder de même pour les titres de niveau 5 jusqu'au niveau 1. Ce qui produirait six instructions similaires (heureusement qu'il n'y a pas 100 niveaux de titre !).

La fonction `preg_replace_callback()` permet d'éviter cela. Cette fonction prend en paramètre une fonction qui sera appelée sur tous les textes correspondant au motif recherché.

Dans le cas des titres, le motif recherché est `'#(={1,6}) (.+?) (={1,6})#'`. Cela correspond à entre un et six signes égal puis des caractères et encore entre un et six signes égal.

Le quantifieur `+` signifie au moins un mais le moins possible (le quantifieur `+` seul prend par défaut le plus de caractères possibles). Cela permet de s'arrêter aux signes égal correspondant à la fin de ce titre.



La fonction passée en paramètre doit prendre en paramètre un tableau. Pour chaque texte correspondant au motif, deux groupes sont capturés, les signes égal avant le titre et le titre lui-même. Ces valeurs capturées sont positionnées dans le tableau à partir de la case 1 (la case 0 contient l'ensemble du texte correspondant au motif).

Dans cette situation, pour les titres, une fonction anonyme est utilisée. Dans celle-ci, l'instruction `$nb = strlen($matches[1]);` permet de compter le nombre de signes égal. La fonction retourne une chaîne de caractères avec la bonne balise HTML de titre et le titre récupéré.

Exécution

Histoire

Le langage PHP fut créé en [1994](#) par [Rasmus Lerdorf](#) pour son [site web](#). C'était à l'origine une [bibliothèque logicielle](#) en [C](#) dont il se servait pour conserver une trace des visiteurs qui venaient consulter son [CV](#). Au fur et à mesure qu'il ajoutait de nouvelles fonctionnalités, Rasmus a transformé la bibliothèque en une implémentation capable de communiquer avec des bases de données et de créer des applications dynamiques et simples pour le [Web](#). Rasmus décida alors en [1995](#) de publier son code, pour que tout le monde puisse l'utiliser et en profiter. PHP s'appelait alors PHP/FI (pour **P**ersonal **H**ome **P**age **T**ools/**F**orm **I**nterpreter). En [1997](#), deux étudiants, [Andi Gutmans](#) et [Zeev Suraski](#), redéveloppèrent le cœur de PHP/FI. Ce travail aboutit un an plus tard à la version 3 de PHP, devenu alors *PHP: Hypertext Preprocessor*. Peu de temps après, Andi Gutmans et Zeev Suraski commencèrent la réécriture du moteur interne de PHP. Ce fut ce nouveau moteur, appelé [Zend Engine](#) — le mot *Zend* est la contraction de **Z**eev et **A**ndi — qui servit de base à la version 4 de PHP.

Utilisations



En [2002](#), PHP est utilisé par plus de 8 millions de sites Web à travers le monde, en [2007](#) par plus de 20 millions et en 2013 par plus de 244 millions.