

PHP - Les inclusions de fichiers : fonctions include et require

Date de dernière mise à jour : 27/06/2007 à 19:36

Source : <http://www.vulgarisation-informatique.com/include.php>.

Distribution interdite sans accord écrit d'Anthony ROSSETTO (<http://www.vulgarisation-informatique.com/contact.php>)

Présentation

Tout ce que vous venez d'apprendre sur PHP est bien beau, mais si il y a une chose essentielle en PHP, c'est bien l'inclusion de fichiers. En quoi cela consiste t-il ? Et bien imaginions que vous ayez un site qui comporte toujours le même menu à gauche ou encore la même partie haute, en (X)HTML vous étiez (à moins d'utiliser des frames dépréciées et néfastes pour le référencement) obligé de copier coller le même code sur toutes les pages de votre site, avec pour inconvénient principal des mises à jour rendues plus difficiles car vous deviez modifier toutes les pages faisant appel à votre menu en cas d'actualisation de celui-ci.

L'include, ou comment inclure des fichiers en PHP

Le code en lui-même est très simple. On utilise la structure du langage `"include"`. Elle peut donc ne pas prendre de parenthèses. Cette structure va vous permettre d'inclure un fichier PHP à interpréter. Voici un exemple permettant d'inclure la page `"haut.php"` dans le fichier courant :

```
<?php
include
'haut.php'
;
?>
```

Le principe est simple : vous allez pouvoir grâce à ce code exécuter le code PHP inclus dans le fichier `"haut.php"`. C'est à dire que si par exemple le fichier inclus contient du code HTML ou des instructions `"echo"` affichant du HTML, vous allez pouvoir afficher un code HTML bien défini sur toutes les pages de vos sites via **un seul** fichier que vous inclurez quand et où vous le voulez.

Le fichier `"haut.php"` peut contenir du code PHP mais aussi du code HTML, il est important de le souligner. Inutile donc pour afficher du code HTML d'utiliser **echo** à foison dans votre fichier inclus.

Voyons maintenant un exemple pratique qui consistera à inclure pour une page PHP, le haut du site. Toutes nos pages PHP pourront s'en resservir pour que le haut du site soit identique sur toutes les pages. Pour cet exemple, nous allons utiliser deux fichiers, l'un sera HTML (le fichier `"haut.php"`;) et sera identique sur toutes les pages pour lequel nous ferons appel à lui, et l'autre en PHP permettra d'inclure ce fichier.

Contenu du fichier `"haut.php"` :

```
&quot;://W3C//DTD XHTML 1.0 Strict//EN&quot;
&quot;http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd&quot;
>&quot;http://www.w3.org/1999/xhtml&quot;
xml:lang=&quot;fr&quot;
lang=&quot;fr&quot;
>
```

Bienvenue sur mon site web !

Bienvenue sur mon site !

Contenu du fichier `"bas.php"` qui sera le pied de page du site :

Contenu du fichier "index.php" qui sera l'accueil du site :

```
<?php
include
'haut.php'
; echo
'Ici le contenu du site'
; include
'bas.php'
;
?>
```

Ce qui donnera en HTML lorsque vous appellerez la page "index.php" :

```
&quot;http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd&quot;
&quot;http://www.w3.org/1999/xhtml&quot;
xml:lang=&quot;fr&quot;
lang=&quot;fr&quot;
>
```

Bienvenue sur mon site web !

Bienvenue sur mon site !

Ici le contenu du site

Attention à l'utilisation à foison de fichiers inclus. Plus vous utiliserez include, plus le temps de génération de la page sera ralenti (car il faut bien faire un accès disque pour lire les différents fichiers). Préférez un seul include d'un fichier un peu plus gros que de 4 fichiers très petits.

Inclusion avec require

la structure du langage require fonctionne de la même façon qu'include. Cependant, il existe une différence entre ces deux méthodes. Lorsque vous tentez d'inclure un fichier, si le fichier n'existe pas que se passe t-il ? Et bien par défaut, PHP affichera une erreur mais continuera l'exécution du script. Imaginons maintenant que vous ayez un espace membres qui, dans un fichier inclus, vérifie que l'internaute soit bien identifié comme étant membre pour afficher le reste. Voici un exemple résumé (le code n'est pas complet) :

```
<?php
include
'verification.php'
;
/* * Code qui sera exécuté si le membre est identifié */
echo
'vous êtes membre !'
;
?>
```

Et maintenant, voici le contenu du fichier "verification.php" qui est chargé de vérifier si l'internaute est bien identifié en tant que membre :

```
<?php
```

```
if(!
est_membre
()) {      exit(
'Vous devez être membre pour accéder à cette page'
); }
?>
```

Le code est bien sûr très simplifié, nous considèrerons que la fonction `est_membre()` est plus haut dans le code ;) Le but est uniquement de comprendre ce que peut entraîner l'absence d'inclusion d'un fichier sensible. Ici, nous faisons appel à la fonction `est_membre()` qui renvoie TRUE si l'internaute est identifié comme membre, et FALSE sinon. Le point d'exclamation permet d'effectuer le contraire de la condition, ici on traduit par "si l'internaute n'est pas membre"; tandis que sans le point d'exclamation on aurait dit "si l'internaute est membre". Le code exécuté arrête alors le script PHP si le visiteur n'est pas membre en utilisant la structure du langage `exit`. Tout ceci est bien, mais si le fichier "verification.php" n'existe pas, dans ce cas, PHP affichera "vous êtes membre !"; ce qui ne va pas du tout avec ce que l'on souhaite faire. Nous allons donc utiliser la structure du langage "require". Celle-ci permet d'inclure un fichier et de couper le script si le fichier n'a pu être inclus.

```
<?php
require
'verification.php'
;
//Coupera l'exécution du code si le fichier n'a pu être inclus par PHP

/* * Code qui sera exécuté si le membre est identifié */
echo
'vous êtes membre !'
;
?>
```

`require` permet à la différence d'`include` de couper l'exécution du script PHP si le fichier à inclure n'a pu être inclus.

Chapitre suivant



[Les formulaires](#)

[Présentation](#)

[La méthode POST - les textbox](#)

[Les textarea](#)

[Le select](#)

[Les cases à cocher \(checkbox\)](#)

[Les boutons radios](#)

[Les champs cachés](#)

[La méthode GET](#)

Source : <http://www.vulgarisation-informatique.com/include.php>.

Distribution interdite sans accord écrit d'Anthony ROSSETTO (<http://www.vulgarisation-informatique.com/contact.php>)