

20190 - 2020

Php

HyperText Preprocessor

THIOUNE Seydina
2e Année BTS SIO
20190 - 2020



Sommaire

Fiche technique	2
Introduction	3
Fonctionnement	4
Balise de délimitation du code Php	5
Traitement des formulaires avec \$_GET et \$_POST	5
Méthode \$_GET	6
Méthode \$_POST	6
Programmation orienté objet	7
La sécurité avec Php	7
Avantages et inconvénients du langage Php	9
Avantages	9
Inconvénients	10
Conclusion	11
Sources	12



Fiche technique

Date de première version : 1994

Paradigme : Impératif, orienté objet, fonctionnel, procédural, réflexif et interprété

Auteur : Rasmus Lerdorf

Développeurs: The Php Group

Dernière version: 7.3.10 - 24 Sept 2019

Typage : Dynamique, faible

Influencé par C, C++, Java et Perl

Écrit en C

Système d'exploitation multiplateforme

Licence libre

Site web : <https://www.php.net>

Extension de fichier : php, phtml, php4, php3, php5, phps et phar

Introduction

Pour avoir une trace de ce qui ont consulté son site web Rasmus Lerdorf a créé le langage Php. A l'origine une bibliothèque en C, son auteur a ajouté de nouvelles fonctionnalités ce qui le transformant en une implémentation capable d'une communication avec des bases de données et de création de site web simple et dynamique. Ainsi en 1995, il décide de publier son code afin que tout le monde puisse l'utiliser et en bénéficier.

En 1997, deux étudiants, Andi Gutmans et Zeev Suraski, ont développé le cœur de PHP/FI (***Personal Home Page Tools/Form Interpreter*** - le nom d'origine du Php). Ce travail a abouti un an plus tard à la version 3 de PHP. Peu de temps après, Andi Gutmans et Zeev Suraski ont commencé la réécriture du moteur interne de PHP. C'est ce nouveau moteur, appelé Zend Engine (le mot Zend est la contraction de Zeev et Andi) qui a servi de base à la version 4 de PHP.

Comme on l'a défini dans notre précédent dossier sur le HTML, PHP: Hypertext Preprocessor, plus connu sous son sigle PHP, est un langage de programmation libre, principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale. PHP est un langage impératif orienté objet.

PHP a permis de créer un grand nombre de sites web célèbres, comme Facebook, Wikipédia, etc. Il est considéré comme une des bases de la création de sites web dits dynamiques mais également des applications web.

Fonctionnement

Comparer aux langages qui l'ont inspiré dont C ou Perl, le Php contient des fragments de html au lieu d'écrire des multitudes de lignes. Il se distingue aussi des autres langages de script tel que JavaScript car exécutant son code sur le serveur donnant ainsi un résultat sous format html et reçu par le client. Ce dernier pourra faire aucune différence entre le site web statique ou dynamique.

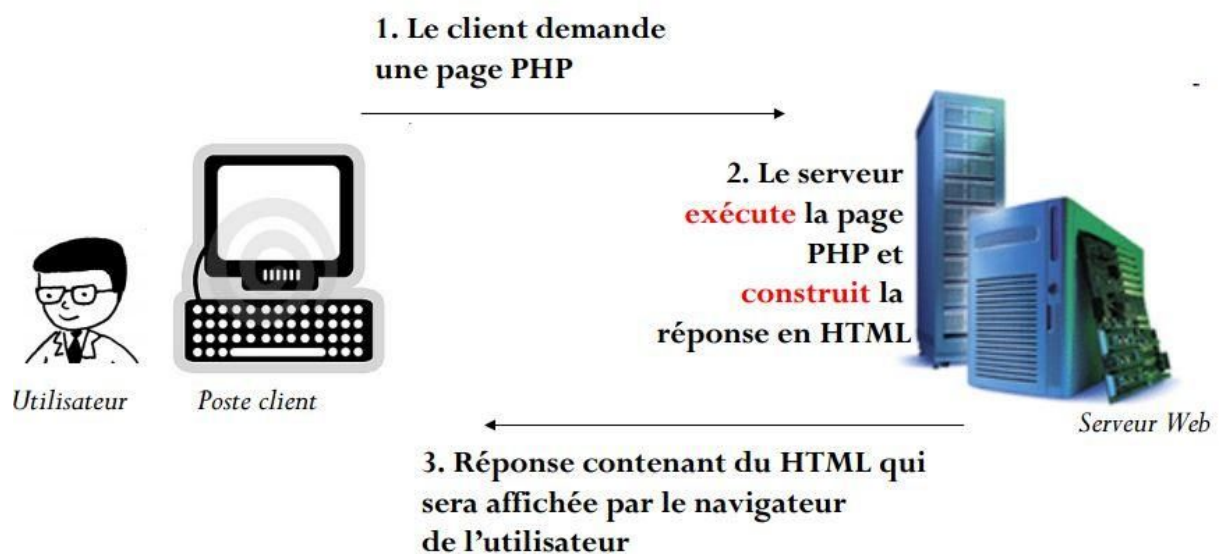
-Exemple de code Php et html donnant le même le résultat:

```
<?php
$nom = 'Albert';
echo '<div>Bonjour ' . $nom . ' !</div>';
```

```
1
2 <p> Bonjour Albert <p>
```

Ces deux codes affichent la même chose à savoir: Bonjour Albert

-Fonctionnement de Php



Très souvent utilisé pour sa fonction de dialogue avec des bases de données, PHP ouvre classiquement une connexion au serveur de SGBD voulu, lui transmet des requêtes et en récupère le résultat, avant de fermer la connexion.

Php permet de générer aussi des fichiers médias, des documents PDF, des images, des animations flash avec l'aide de bibliothèques. Avec Php, on peut aussi générer du code XML, XHTML ou même du texte au lieu de croire à une limitation de produire du code html.

Balise de délimitation du code Php

On note beaucoup de balise permettant de délimiter notre code php :

- `<?php et ?>`
- `<? et ?>`
- `<?= et ?>`
- `<% et %>`
- `<script language="php"> et </script?>`

Cependant le plus utilisé reste `< ?php ?>`. Cela s'explique par son adaptation à tous les versions de Php

Traitement des formulaires avec `$_GET` et `$_POST`

Quand on parle de sites web dynamiques avec Php, on va surement parler de formulaire. Et peut-on parler de formulaire sans lever le sujet des méthodes `$_GET` et `$_POST` ? Bien évidemment que c'est inconcevable.

Pour un traitement de formulaire, le langage Php est surtout mis au-devant de la scène par rapport aux autres comme JavaScript. Cela peut se justifier par son exécution coté serveur, entraînant l'agissement seul du programme ainsi une bonne sécurité. Mais

surtout de sa série de fonctions natives capables de manipuler les variables et de les contrôler.

Méthode \$_GET

La méthode GET renvoie les informations utilisateur codées ajoutées à la demande de page. La page et les informations codées sont séparées par le caractère “?”.

Exemple: `http://www.test.com/index.htm?name1=value1&name2=value2`

- La méthode GET génère une longue chaîne qui apparaît dans les journaux de votre serveur, dans la zone Emplacement du navigateur.
- La méthode GET est limitée à l'envoi de 1024 caractères maximum.
- N'utilisez jamais la méthode GET si vous avez un mot de passe ou d'autres informations sensibles à envoyer au serveur.
- GET ne peut pas être utilisé pour envoyer des données binaires, telles que des images ou des documents Word, au serveur.
- Les données envoyées par la méthode GET sont accessibles à l'aide de la variable d'environnement QUERY_STRING.
- PHP fournit un tableau associatif \$_GET pour accéder à toutes les informations envoyées à l'aide de la méthode GET.

Méthode \$_POST

La méthode POST transfère des informations via des en-têtes HTTP. Les informations sont codées comme décrit dans le cas de la méthode GET et placées dans un en-tête appelé QUERY_STRING.

- La méthode POST n'a aucune restriction quant à la taille des données à envoyer.
- La méthode POST peut être utilisée pour envoyer des fichiers ASCII ainsi que des données binaires.

- Les données envoyées par la méthode POST passent par l'en-tête HTTP, de sorte que la sécurité dépend du protocole HTTP. En utilisant HTTP sécurisé, vous pouvez vous assurer que vos informations sont sécurisées.
- PHP fournit un tableau associatif `$_POST` pour accéder à toutes les informations envoyées à l'aide de la méthode POST.

Programmation orienté objet

Comme en Pascal, PHP permet de programmer en orienté objet, en créant des classes contenant des attributs et des méthodes, qui peuvent être instanciées ou utilisées en statique.

Toutefois, PHP est un langage à héritage simple, c'est-à-dire qu'une classe ne peut hériter que d'au plus une seule autre classe (sinon il faut utiliser un trait pour simuler l'héritage multiple par composition). Cependant les interfaces peuvent en étendre plusieurs autres.

La sécurité avec Php

Comme partout sur les plateformes web, Php aussi n'est pas à l'abri d'attaques malveillants. Ainsi cela pousse aux développeurs et administrateurs de base de données à devoir trouver des solutions pour une bonne protection contre les malveillants.

Le premier conseil de sécurité qu'on note est d'être à jour. Mais aussi le développeur doit activer et/ou utiliser les paramètres de configuration appropriés surtout concernant les fichiers de la base de données et des sessions qui sont les plus souvent attaqués.

L'obscurité est parfois aussi un des moyens de se prévenir ou de retarder des attaques.

Exemple #1 Masquer PHP avec un autre langage

```
# Faire que le code PHP ressemble à un autre langage
AddType application/x-httpd-php .asp .py .pl
```

Exemple #2 Masquer PHP avec des types inconnus

```
# Faire que le code PHP ressemble à un autre langage qui n'existe pas
AddType application/x-httpd-php .bop .foo .133t
```

Exemple #3 Utiliser le type HTML pour les extensions PHP

```
# Faire que le code PHP ressemble à du html
AddType application/x-httpd-php .htm .html
```

Parfois aussi les pirates utilisent des injections sur les requêtes SQL afin d'avoir accès à la base de données. Mais dans cette partie il ne faut pas intervenir sur le Php mais plus tôt sur le SQL. Il y a aussi le fait de sécuriser sa base de données avec des mots de passes même si parfois peut être sauté.

De plus en plus, on note de techniques novices de hacking mais aussi un travail sans relâche des développeurs afin de les contrer.

Avantages et inconvénients du langage Php

Comme nombreux autres langages, Php a des avantages mais aussi des inconvénients qu'on va énumérer ci-dessous.

Avantages

- PHP est Open Source et gratuit. Il peut être téléchargé n'importe où et facilement disponible pour le développement d'applications Web
- PHP est extensible.
- Un grand nombre de bases de données sont prises en charge.
- PHP est indépendant de la plateforme. Il fonctionnera sur la plupart des plates-formes. Les applications basées sur PHP peuvent fonctionner sur tous les systèmes d'exploitation tels que UNIX, Linux et Windows , etc.
- Il dispose d'un puissant support de bibliothèque permettant d'utiliser les différents modules de fonction pour la représentation des données.
- Compatible avec les serveurs comme IIS et APACHE.
- Faible coût de développement et de maintenance avec des performances et une fiabilité très élevées.
- C'est simple et facile à utiliser.
- Il est efficace et capable de connecter la base de données et de charger l'application plus rapidement.
- Il fournit plus de sécurité à l'application Web grâce à l'utilisation de frameworks PHP .
- Il est plus familier pour les développeurs et un support en ligne est fourni aux débutants.

- Il fait preuve de flexibilité et de capacité à s'intégrer à un autre langage de programmation.
- C'est léger et nombre de cadres disponibles à utiliser pour développer l'application Web.
- Cela aide à réaliser rapidement le développement d'une application Web.

Inconvénients

- Ce n'est pas si sûr à cause de son code source ouvert, car le code source peut être facilement disponible.
- Il n'est pas adapté aux applications Web à contenu volumineux.
- Son type est faible, ce qui peut entraîner des données et des informations incorrectes pour l'utilisateur.
- Les frameworks PHP doivent apprendre à utiliser les fonctionnalités intégrées de PHP pour éviter d'écrire du code supplémentaire.
- L'utilisation de plus de fonctionnalités du framework PHP et des outils conduit à une performance médiocre des applications Web.
- PHP n'autorise pas le changement ou la modification du comportement principal des applications Web.
- Les frameworks PHP n'ont pas le même comportement que leurs performances et fonctionnalités.

Conclusion

PHP est le langage de programmation côté serveur et de programmation le plus utilisé pour le développement d'applications Web. L'utilisation de Php en tant que générateur de pages Web dynamiques est la plus répandue, mais il peut aussi être utilisé comme langage de programmation ou de script en ligne de commande sans utiliser de serveur HTTP ni de navigateur. Il permet alors d'utiliser de nombreuses fonctions du langage C et plusieurs autres sans nécessiter de compilation à chaque changement du code source.

Introduction à PHP est devenu un langage populaire avec un excellent support en ligne et une communauté pour aider les débutants. Il est purement basé sur les tendances ou les exigences pour lesquelles un langage de programmation doit s'appliquer pour le développement de toute application.

Sources

<https://fr.wikipedia.org/wiki/PHP>

<https://www.php.net/manual/fr/intro-what-is.php>

<https://www.isquaretechnologies.com/benefits-php-web-development/>

<https://apprendre-php.com/tutoriels/tutoriel-12-traitement-des-formulaires-avec-get-et-post.html>

<https://www.grafikart.fr/tutoriels/php-langage-merde-1001>

<http://glossaire.infowebmaster.fr/php/>