

PHP - Introduction

Les bases de la programmation web serveur avec PHP

Pierre Blarre - Florian Rodriguez

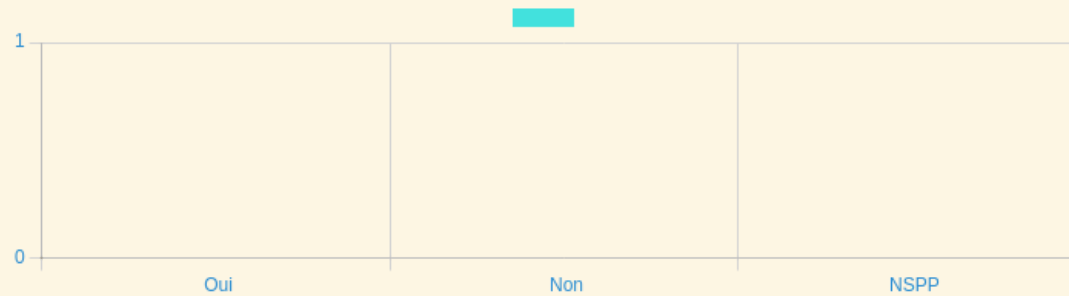
Avez-vous déjà des bases en PHP ?

Oui

Non

NSPP

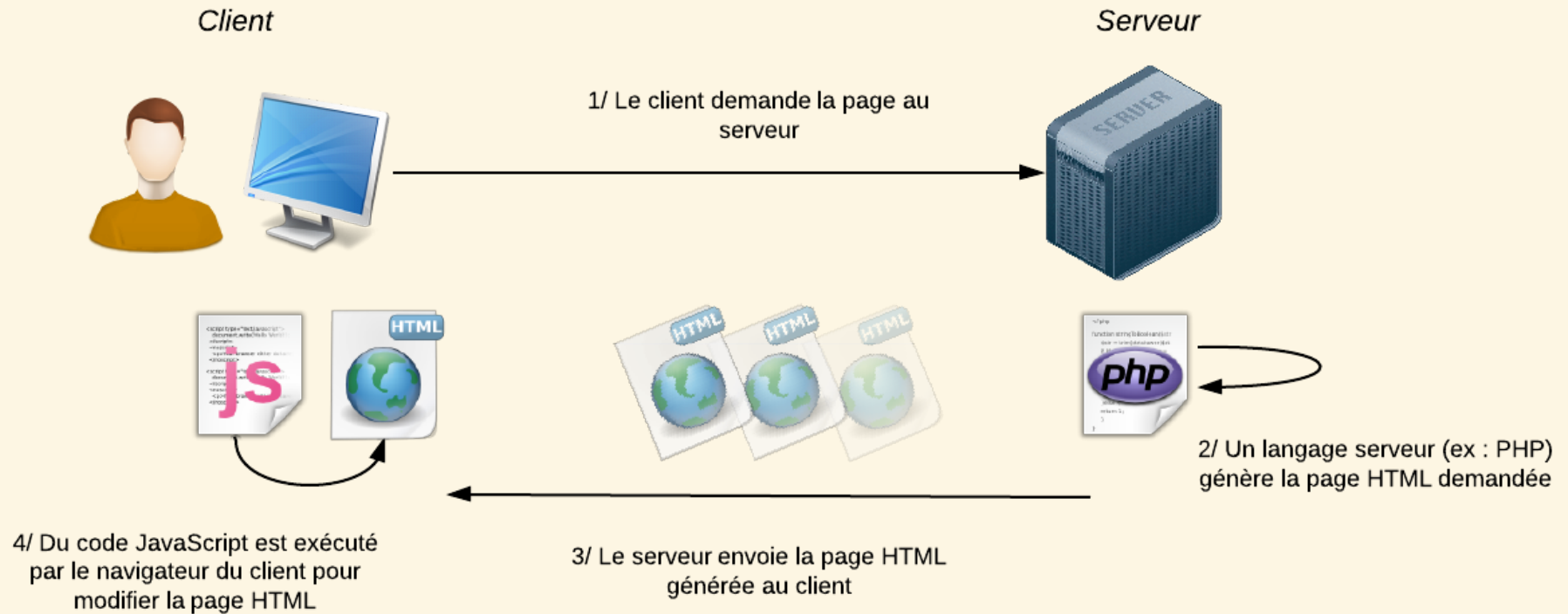
Responses: 0



Le PHP : Présentation

- **PHP = Hypertext Preprocessor**
- Open Source, conçu pour le développement web.
- Intégré facilement dans le HTML.
- Langage interprété, nécessite un serveur web.
- Connaissances requises : HTML, CSS.

Serveur / Résumé en image



Serveur web et cycle de vie d'une requête

- Pour pouvoir générer une page HTML avec un script PHP, il va être nécessaire d'avoir un serveur web qui :
 - Analyse l'url demandée par le visiteur
 - Si l'url doit appeler un script PHP
 - Dans ce cas, appeler le script PHP correspondant et l'exécuter
 - Renvoyer une page HTML

Serveur web

- On a besoin d'un serveur web et de php:
 - LAMP, WAMP, XAMPP
 - On peut lancer un serveur avec

```
php -S localhost:2023
```



Premier script PHP / Exemple

Source

```
1 <html>
2   <head>
3     <title>Test PHP</title>
4   </head>
5   <body>
6     <?php echo "<p>Bonjour le monde</p>"; ?>
7   </body>
8 </html>
```



Résultat

```
1 <html>
2   <head>
3     <title>Test PHP</title>
4   </head>
5   <body>
6     <p>Bonjour le monde</p>
7   </body>
8 </html>
```



Premiers pas avec la syntaxe

- Le code PHP doit nécessairement être inclus entre la balise ouvrante:
`<?php`
- et la balise fermante:
`?>`
- Chaque instruction (entendez chaque ligne) doit être terminée par un point virgule ;
- Les variables commencent toujours par le symbole dollar \$ pour être différenciées des fonctions.

Autre exemple

Source

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Bienvenue dans notre cours PHP</title>
5   </head>
6   <body>
7     <?php
8       $cours = ["PHP", "HTML", "SQL"];
9       echo "<ul>";
10      foreach ($cours as $cour) {
11        echo "<li>$cour</li>";
12      }
13      echo "</ul>";
14    ?>
15  </body>
16 </html>
```



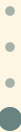
Résultat

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Bienvenue dans notre cours PHP</title>
5   </head>
6   <body>
7     <ul><li>PHP</li><li>HTML</li><li>SQL</li></ul></body>
8 </html>
```



Remarques

- Un fichier PHP va renvoyer le code qui n'est pas contenu dans les balises `<?php` et `?>` tel quel au navigateur
- Puisque le PHP renvoie le texte tel quel au navigateur, si le texte renvoyé est sous forme de balises html, alors le navigateur va interpréter ces balises et générer le html correspondant
- On peut maintenant utiliser toutes les fonctions PHP disponible pour générer du HTML.
- Dans cet exemple, on a utilisé un tableau avec des valeurs fixées, mais on aurait pu récupérer des valeurs depuis une base de données, un fichier json, etc.



Variables et types de données

- Déclaration : `$nom_variable = valeur;`
- Types principaux :
 - `string` : "Je suis une chaîne"
 - `integer` : 42
 - `float` : 3.14
 - `bool` : `true` / `false`
 - `null` : `NULL`
- En PHP, comme en Javascript, il n'est pas obligatoire de spécifier le type de données d'une variable. Il est aussi possible de changer le type de données de la variable en lui assignant une nouvelle valeur:

```
$une_variable = 2;      // integer  
$une_variable = "test"; // string  
$une_variable = false;  // bool
```



- On dit que PHP et Javascript sont des langages “non typés”, car ils permettent ces changements de type. Ceci apporte son lot d'avantages et d'inconvénients. Dans d'autres langages ceci n'est pas permis!

Chaînes de caractères (string)

- Déclaration :

```
$nom = "Pierre";  
$nom = 'Pierre';
```



- Échappement :

```
$variable = "Mon \"nom\" est Pierre";  
$variable = 'Je m\'appelle Pierre';
```



Affichage et concaténation

- Avec guillemets doubles :

```
$age = 17;  
echo "Le visiteur a $age ans";
```



- Avec guillemets simple, on doit utiliser la concaténation :

```
echo 'Le visiteur a ' . $age . ' ans';
```



Opérateurs et structures de contrôle

- Opérateurs:
 - Arithmétiques
 - +, -, *, /, %, ++, --
 - Comparaison
 - =, !=, <, >, <=, = >
 - Logiques
 - &&, ou and, ou AND,
 - ||, ou or, ou OR,
 - !
- Structures de contrôle
 - Conditionnelles
 - if, else, elseif, switch
 - Boucles
 - for, while, foreach



Conditions

- **if/elseif/else** :

```
if ($age >= 18) {  
    echo "Majeur";  
} else {  
    echo "Mineur";  
}
```



- **switch** :

```
switch ($couleur) {  
    case 'rouge':  
        echo "La couleur est rouge";  
        break;  
    case 'bleu':  
        echo "La couleur est bleu";  
        break;  
    default:  
        echo "Couleur inconnue";  
}
```



- **ternaire** :

```
echo ($age >= 18) ? "Majeur" : "Mineur";
```



Boucles

- **while** :

```
$i = 1;  
while ($i <= 100) {  
    echo "Ligne n°$i<br />";  
    $i++;  
}
```



- **for** :

```
for ($i = 1; $i <= 100; $i++) {  
    echo "Ligne n°$i<br />";  
}
```



Tableaux (Arrays)

- **Numérotés :**

```
$prenoms = array('François', 'Michel', 'Nicole');  
$prenoms[2]= 'Sophie';  
$prenoms[]= 'Julien'; // Ajoute "Julien" à la fin du tableau  
echo $prenoms[1]; // Affiche "Michel"
```



- **Associatifs :**

```
$coord = array(  
    'prenom' => 'François',  
    'ville' => 'Marseille'  
);  
echo $coord['ville']; // Affiche "Marseille"  
$coord['age'] = 45; // Ajoute une nouvelle entrée au tableau
```



La flèche => permet d'associer une clé à une valeur dans un tableau associatif.

Tableau (Arrays)

Les arrays numérotés permettent de stocker une série d'éléments du même type, comme des prénoms. Chaque élément du tableau contiendra alors un prénom.

Les arrays associatifs permettent de découper une donnée en plusieurs sous-éléments. Par exemple, une adresse peut être découpée en nom, prénom, nom de rue, ville...

Parcourir un tableau

- Avec for :

```
for ($i = 0; $i < count($prenoms); $i++) {  
    echo $prenoms[$i] . '<br />';  
}
```



- Avec foreach :

```
foreach($prenoms as $prenom) {  
    echo $prenom . '<br />';  
}
```



```
foreach($coord as $cle => $valeur) {  
    echo "$cle : $valeur<br />";  
}
```



Fonctions

- **Déclaration :**

```
function affichePersonne($personne) {  
    echo $personne['nom'] . " " . $personne['prenom'];  
}
```



- **Appel :**

```
$florian = array("nom" => "Rodriguez", "prenom" => "Florian");  
affichePersonne($florian);
```



print_r (débogage)

- Permet d'afficher le contenu d'une variable (tableau, objet, etc.)
- Exemple :

```
$fruits = array("pomme", "banane", "orange");  
print_r($fruits);
```



- Affiche :

```
Array (  
    [0] => pomme  
    [1] => banane  
    [2] => orange  
)
```



Ressources

- [Documentation PHP](<http://php.net/manual/fr/intro-what-is.php>)
- [XAMPP](<https://www.apachefriends.org/index.html>)