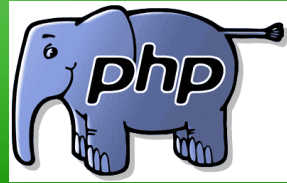


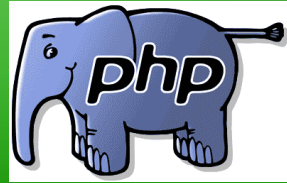
PHP première partie



- Définition
- La syntaxe
- Gestion des formulaires
- Conclusion



Définition



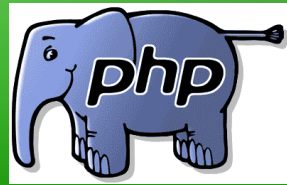
Première version de **PHP** : 1994 (Personal Home Pages de Rasmus Lerdorf)

Aujourd'hui **PHP 8.4** : depuis 2024

PHP (Hypertext Preprocessor depuis V4) est un ensemble de fonctionnalités qu'il est possible d'insérer directement dans le code d'une page HTML, comme le JavaScript, mais ce code va s'exécuter du **côté serveur**.



Définition

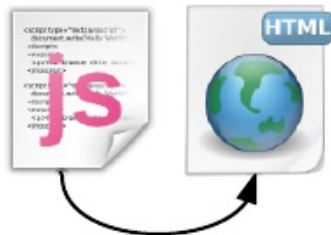
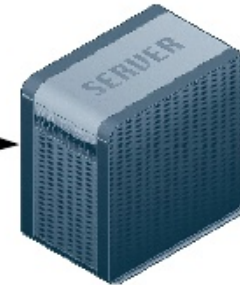


Cette technologie permet de réaliser un site dynamique
Par opposition à un site statique réalisé en HTML - CSS

Client

Serveur

1/ Le client demande la page au serveur

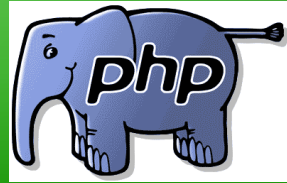


2/ Un langage serveur (ex : PHP) génère la page HTML demandée

4/ Du code JavaScript est exécuté par le navigateur du client pour modifier la page HTML

3/ Le serveur envoie la page HTML générée au client

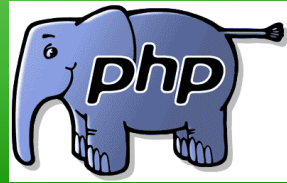
Définition



Objectif final : interagir avec une base de données pour plus d'adaptation et accès à un plus grand nombre d'informations



La Syntaxe



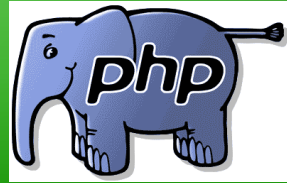
Le code PHP se trouve entre 2 balises HTML particulières.

<?php et ?> ou <? ?>

Ses balises peuvent se placer n'importe où dans le code HTML, il est également possible d'en mettre plusieurs fois dans une même page.

Le mieux est d'utiliser un fichier séparé avec une extension .php.

La Syntaxe



Le PHP est un langage objet.

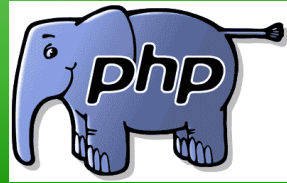
Les structures conditionnelles ou répétitives ont la même syntaxe qu'en langage C, C++.

C'est un langage faiblement typé. Il est possible d'utiliser une variable sans l'avoir déclaré au préalable et sans en préciser son type.

Une variable commence obligatoirement par \$

ATTENTION, ce \$ n'a rien à voir avec le \$ du JQuery

La Syntaxe



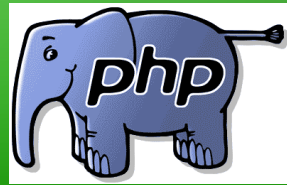
Exemple :

```
$chaine="ma chaine";
```

Les constantes se déclarent ainsi :

```
define("PI", 3.14 );  
echo "la valeur de pi est : ".PI;
```

La Syntaxe



Les tableaux :

Il existe plusieurs façon de créer un tableau :

```
<?php
```

```
$tab[ ] = "un" ;
```

```
$tab[ ] = "deux" ;
```

```
$tab[ ] = "trois" ;
```

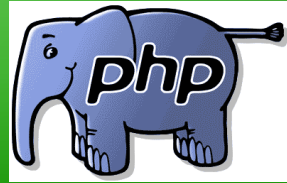
```
//Ou
```

```
$tab = array( "un", "deux", "trois" ) ;
```

```
// défini un tableau pouvant contenir 10 valeurs.
```

```
$tab = array(10) ;
```

La Syntaxe



Les tableaux :

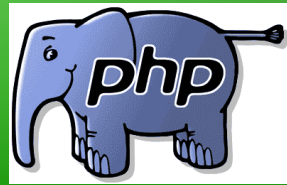
Il est possible d'utiliser des tableaux associatifs.

Exemple

```
$personne["nom"]=bidochon;  
$personne["prenom"]=robert;  
$personne["age"]=54;  
  
foreach ($personne as $champ=>$valeur)  
{  
    echo "$champ : $valeur";  
}
```

nom : bidochon
prenom : robert
age : 54

La Syntaxe



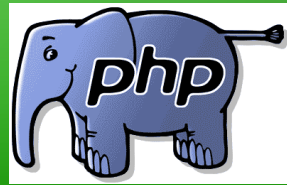
Les fonctions :

```
function nomFonction($arg1, $arg2, ..)
```

Les fonctions se mettent dans un fichier séparé ayant pour extension **.inc.php**

L'inclusion se fait avec **require_once** pour inclure qu'une seule et unique fois le fichier.

La Syntaxe



- page.php

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Bonjour</title>
    <meta charset="UTF-8">
  </head>

  <body>
    <?php
      require_once './fonctions.inc.php';
      afficherBonjour();
    ?>
  </body>
</html>
```

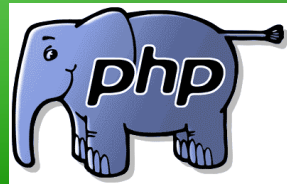
- fonctions.inc.php

```
<?php

function afficherBonjour()
{
    echo "Hello world";
}
```

Hello world

Configuration sous NetBeans



Project Properties - miseEnOeuvre_2

Categories:

- Sources
- Run Configuration
- Browser
- [-] • JavaScript Libraries
 - npm
 - Bower
 - CDNJS
- [-] • JavaScript
 - RequireJs
 - Oracle JET
- CSS Preprocessors
- Include Path

Configuration: <default> [v] [New...] [Delete]

Run As: PHP Built-in Web Server (running on built-in web server) [v]

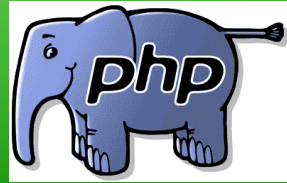
Hostname: localhost

Port: 8000
http://localhost:8000/

Router Script: [] [Browse...]

[OK] [Cancel] [Help]

Gestion formulaire

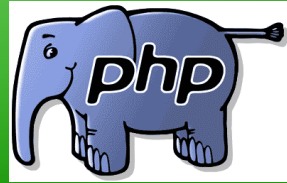


Les données en provenance d'un formulaire sont acheminées soit avec la méthode GET, soit avec la méthode POST.

Dans le premier cas, toutes les informations seront contenues dans le tableau associatif **`$_GET`**.

Dans le deuxième cas, c'est dans le tableau **`$_POST`** que seront les informations.

Gestion formulaire



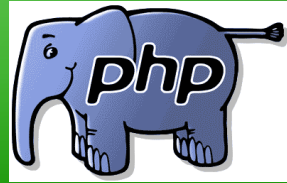
index.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Exemple Formulaire</title>
    <meta charset="UTF-8">
  </head>
  <body>
    <form action="traiter.php" method="get">
      <label for="login">Login</label>
      <input type="text" name="login" id="login"/>
      <br/>
      <label for="pwd">Mot de passe</label>
      <input type="password" name="pwd" id="pwd"/>
      <br/>
      <input type="submit" value="Envoyer"/>
    </form>
  </body>
</html>
```

Login

Mot de passe

Gestion formulaire



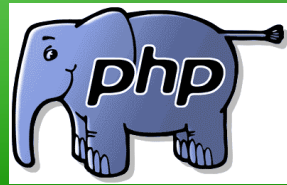
traiter.php

```
<!DOCTYPE html>
<html>
  <head>
    <title>Traitement php</title>
    <meta charset="UTF-8">
  </head>
  <body>
    <?php
    echo "Votre login est : " . $_GET["login"] . "<br/>";
    echo "Votre mot de passe est : " . $_GET["pwd"] . "<br/>";
    ?>
  </body>
</html>
```

Votre login est : toto

Votre mot de passe est : titi

Gestion formulaire avec vérification des champs



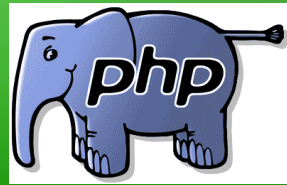
traiter.php

```
<!DOCTYPE html>
<html>
  <head>
    <title>Traitement formulaire sécurisé</title>
    <meta charset="UTF-8">
  </head>
  <body>
    <?php
      if (filter_input(INPUT_SERVER, 'REQUEST_METHOD') === 'GET') {
        echo "Votre login est : " . filter_input(INPUT_GET, 'login') . "<br/>";
        echo "Votre mot de passe est : " . filter_input(INPUT_GET, 'pwd') . "<br/>";
      }
    ?>
  </body>
</html>
```

Votre login est : toto
Votre mot de passe est : titi

La fonction **filter_input()** peut prendre un 3ième paramètre permettant d'appliquer un filtre de validation ou de "nettoyage" au champ reçu.

Exemple avec utilisation d'une fonction



```
<!DOCTYPE html>
<html>
  <head>
    <title>Traitement formulaire sécurisé</title>
    <meta charset="UTF-8">
  </head>
  <body>
    <?php
      require_once './fonctions.inc.php';

      if (filter_input(INPUT_SERVER, 'REQUEST_METHOD') === 'GET') {

        $login = filter_input(INPUT_GET, 'login');
        $motDePasse = filter_input(INPUT_GET, 'pwd');

        if (VerifierLogin($login, $motDePasse)) {
          echo "Utilisateur identifié";
        } else {
          echo "Mauvais login ou mot de passe";
        }
      }
    ?>
  </body>
</html>
```

traiter.php

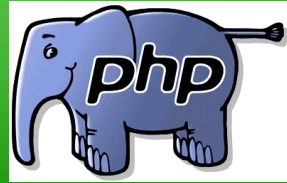
fonctions.inc.php

```
<?php

function VerifierLogin($log, $mdp) {
  $retour = FALSE;

  if ($log == "toto" && $mdp == "toto") {
    $retour = TRUE;
  }

  return $retour;
}
```



Login toto

Mot de passe

Envoyer

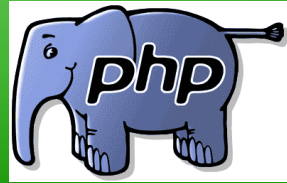
Login / Mot de passe non
conforme

Mauvais login ou mot de passe

Login / Mot de passe
conforme

Utilisateur identifié

Références



- <http://php.net/manual/fr/>
- <http://www.w3schools.com/php/>