

# Le langage PHP

INSY2S  
Septembre 2018



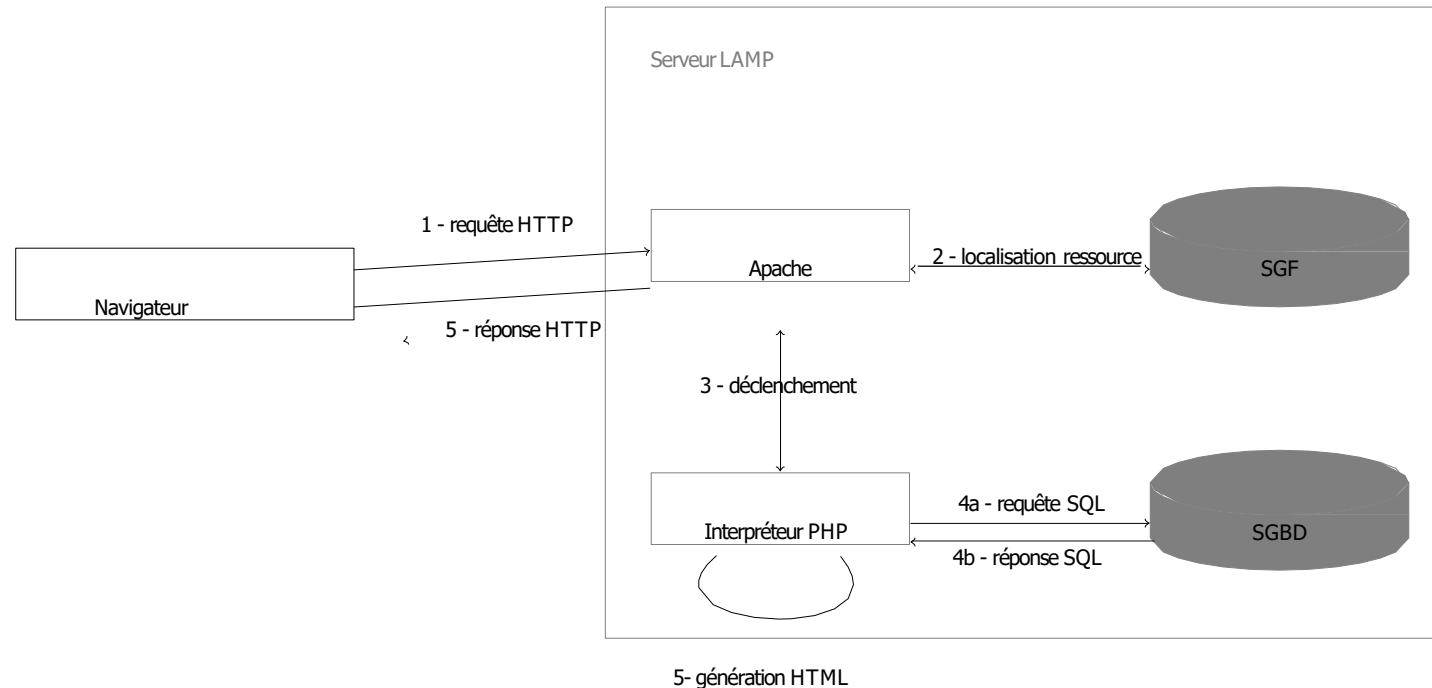
# Plan

- Introduction
- Variables
- Opérateurs
- Fonctions
- Conclusion
- Bibliographie

# Qu'est ce que le PHP ?

- PHP : acronyme récursif pour "Hypertext Preprocessor"
- PHP : naissance au milieu de l'année 1990
- PHP est un langage de script
- PHP est **interprété** du côté du serveur.
- PHP supporte de nombreux SGBD (MySQL, Oracle, PostgreSQL, etc.)
- PHP est langage de programmation **impératif**, qui permet de réaliser une séquence d'instruction
- PHP est un langage de programmation **procédurale**, qui permet de définir des routines et sous-routines (voire orienté objet)
- PHP est un langage libre
- PHP est adapté à la création de pages web **dynamique**
- **La version actuelle est PHP 7**

# Comment ça marche ?



- Le client web demande une page PHP
- Le serveur web identifie ce fichier dans son système de gestion de fichiers
- Le fichier est transmis au module d'interprétation PHP du serveur
- Le code HTML est généré par l'interpréteur à partir du code PHP
- Le serveur web répond au client

# Comment créer un script PHP ?

- Créer un fichier avec l'extension .php et y insérer du code HTML et/ou PHP.
- Le code PHP doit être délimité par l'une des balises :
  - `<script langage="php"> ... </script>`
  - `<?php ... ?>`

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8" />
  <title>Mon premier exemple</title>
</head>
<body>
  <?php phpinfo(); ?>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8" />
  <title>Mon second exemple</title>
</head>
<body>
  <?php
    echo "Hello World"; //Afficher un message
  ?>
</body>
</html>
```

# Comment inclure un script PHP ?

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8" />
  <title>Mon premier exemple</title>
</head>
<body>
  <?php include("monScript.php"); ?>
</body>
</html>
```

monScript.php

```
<?php
  //affiche les information sur l'interpreteur PHP

  phpinfo();
?>
```

# Les variables

- Pas de déclaration (de type) de variables
- Les types sont boolean, integer(int), float, double, string, array, ...
- Les noms des variables sont en ASCII (camelCase), sensible à la casse et précédés par \$

Par exemple

*<?php*

```
$a = -123; // Un nombre négatif
$b = 1.234;
$c = 1.2e3;
$d = false;
$e = true;
$f = 'aujourd'hui'; // chaine de caractère simple
$g = "Il fait beau $f"; // chaine de caractère évaluante
```

*?>*

# Conversion implicite de type

```
<?php
$foo = "0";
echo $foo;    // $foo est un string (ASCII 48)

$foo += 2;    // équivalent à $foo = $foo + 2;
echo $foo;    // $foo est maintenant an integer (2)

$foo = $foo + 1.3;
echo $foo;    // $foo est maintenant un float (3.3)

$foo = 5 + "10 ans";
echo $foo;    // $foo est un integer (15)

$foo = 5 . "10 ans";
echo $foo;    // $foo is string ("510 ans")
?>
```



# Les tableaux

- Un tableau est un ensemble de valeurs

```
<?php
$cars[0]="Saab";
$cars[1]="Volvo";
$cars[2]="BMW"; // $cars=array("Saab","Volvo","BMW");
unset($cars[2]); // $cars=array("Saab","Volvo");
echo $cars[0] . " and " . $cars[1] . " are Swedish cars.";
?>
```

# Les tableaux

- Un tableau est un ensemble de couples (*clef*, *valeur*)

```
<?php
```

```
$dalton = array(4=>"Joe", 5=>"William", 6=>"Jack",  
7=>"Averell");
```

```
//$dalton[4]="Joe";
```

```
//$dalton[5]="William";
```

```
//$dalton[6]="Jack";
```

```
//$dalton[7]="Averell";
```

```
echo $dalton[4].", ".$dalton[5].", ".$dalton[6]. " et  
".$dalton[7].".";
```

```
?>
```

# Les tableaux

- Un tableau est un ensemble de couples (*clef*, *valeur*)

```
<?php
$d =
array("aine"=>"Joe","cadet1"=>"William","cadet2"=>"Jack","
benjamin"=>"Averell");
//$d["aine"]="Joe";
//$d["cadet1"]="William";
//$d["cadet2"]="Jack";
//$d["benjamin"]="Averell";
echo $d["aine"].", ".$d["cadet1"].", ".$d["cadet2"]." et
".$d["benjamin"].".";
?>
```

## Les tableaux multidimensionnels

- Dans un tel tableau, chaque élément est un sous-tableau

```
<?php
```

```
$got = array ( "Lannister" =>  
    array ( "Pere" => "Tywin Lannister",  
            "Soeur" => "Cersei Lannister",  
            "Frere" => "Jaime Lannister"),  
    "Stark" =>  
    array ( "Pere" => "Robb Stark",  
            "Mere" => "Catelyn Stark",  
            "Fils" => "Jon Stark"),  
    );  
echo $got["Lannister"]["Pere"];  
?>
```

# Les opérateurs

- Opérateurs arithmétiques : +, -, \*, /, ++, --
- Opérateurs d'affectation : =, +=, -=, .=, /=, \*=
- Opérateur de concaténation de chaînes : .
- Opérateurs de concaténation de tableaux : +
- Reste de la division (modulo) : %
- Opérateurs de comparaison : ==, !=, <, >, <=, >=
- Opérateurs logiques : &&, ||, !(opérateur de négation)
- Opérateur conditionnel ternaire:  
\$a = (condition) ? value1 : value2 ;

<?php

```
$foo="Hello World!";  
$bar="What a nice day!";  
echo $foo . " " . $bar;  
$i=12;  
$j=24;  
$max=($i > $j) ? $i : $j;  
echo $max; // affiche 24
```

?>

# Instructions d'expressions

- Les affectations

```
<?php  
$i=2;  
$i*=3;  
?>
```

- Les incrémentations/décrémentations

```
<?php  
$cpt++;  
$cpt--;  
?>
```

- Les appels de fonctions

```
<?php  
function do($a : string){  
    echo 'Cette fonction ne retourne ' . $a ;  
}  
do('rien');  
?>
```

- Les blocs d'instructions

```
<?php  
{  
    $x=12;  
    $y=14;  
}  
?>
```

# Les instructions conditionnelles

- `if` (expression) instruction1 [`else` instruction2]

```
<?php
    $d="Monday";
    if ($d=="Friday")
        echo "Have a nice weekend!";
?>
```

```
<?php
    $d="Monday";
    if ($d=="Friday")
        echo "Have a nice weekend!";
    else
        echo "Have a nice day!";
?>
```

## Les instructions conditionnelles

- `if` (expression) instruction1 [`elseif` (expression) instruction2]  
[`else` instruction3]

```
<?php
$d="Monday";
if ($d=="Friday")
    echo "Have a nice weekend!";
elseif ($d=="Sunday")
    echo "Have a nice Sunday!";
else
    echo "Have a nice day!";
?>
```



# Les instructions conditionnelles

- `switch` (expression) { `case` valeur1 : [instruction1]; `case` valeur2 : [instruction2]}

*<?php*

```
$x = readline("Saisissez un nombre : ");
switch ($x) {
    case 1:
        echo "Number One";
        break; // permet de quitter le switch
    case 2:
        echo "Number Two";
        break;
    default:
        echo "Not 1 nor 2";
}
```

*?>*