

# Le langage PHP

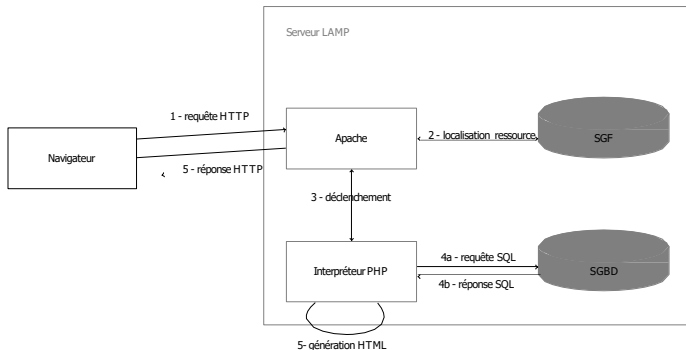
INSY2S  
2022



## 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 8**

# 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 P H P ?

Créer un fichier avec l'extension .php ;

Y insérer du code HTML et/ou PHP.

Le code PHP doit être délimité par les balises

- `<script language="php"> ... </script>`
- `<?php ... ?>`
- 

```
<!DOCTYPE html>
<html lang="fr">
<head>
<meta charset="utf-8" />
<meta name="description"
content="Exemple de cours" />
<meta name="author"
content="Mohamed SACI"/>
<title>Mon premier exemple</title>
</head>
<body>
<?php phpinfo(); ?>
</body>
</html>
```

`<!DOCTYPE html>`

```
<html lang="fr">
<head>
<meta charset="utf-8" />
<meta name="description"
content="Exemple de cours" />
<meta name="author"
content="Mohamed SACI"/>
<title>Mon second exemple</title>
</head>
<body>
<?php
echo "Hello World"; //Afficher un message
?>
</body>
</html>
```

## Comment créer un script P H P ? (bis)

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8" />
  <meta name="description"
        content="Exemple de cours" />
  <meta name="author"
        content="Mohamed SACI"/>
  <title>Mon premier exemple</title>
</head>
<body>
  <?php include("monScript.php"); ?>
</body>
</html>
```

```
<?php
/*
affiche les information
sur l'interpréteur PHP
*/
phpinfo();
?>
```

# Plan

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

# Les variables

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

Par exemple

```
<?php
$a=-123; // a negative number
$b = 1.234;
$c = 1.2e3;
$d = FALSE;
$e = TRUE;
$f ='Hardy dit \'le gros\'';
$g="Laurel et $f";
?>
```

# Conversion implicite de type

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

$foo += 2;
echo $foo; // $foo is now an integer (2)

$foo = $foo + 1.3;
echo $foo; // $foo is now a float (3.3)

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



# 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."  
?>
```

- 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].".";  
?>
```

```
<?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  
+ ou - ou \* ou / ou % ou ++ ou --
- Opérateurs d'affectation  
= ou += ou -= ou .= ou /= ou %=
- Opérateur de concaténation de chaînes .
- Opérateurs de concaténation de tableaux +
- Opérateurs de comparaison  
== ou != ou < ou > ou <= ou >=
- Opérateurs logiques  
&& ou || ou !
- Opérateur conditionnel  
variablename=(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;  
?>
```

# Instructions d'expressions

- Les affectations

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

- Les incrémentations/décrémentations

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

- Les appels de fonctions

```
<?php  
function unset($a[]){  
    Bla bla bla  
}  
unset($a[2]);  
?>
```

- 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!";
?>
```

```
<?php
switch ($x) {
    case 1:
        echo "Number 1 ";
        break;
    case 2:
        echo "Number 2 ";
        break;
    default:
        echo "No number between 1 and 3 ";
}
?>
```

```
<?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 itératives

- **while** (expression) instruction
- **do** instruction **while** (expression)

```
<?php
$i=1;
while($i<=5){
    echo "The number is " . $i . "<br />";
    $i++;
}
?>
```

```
<?php
$i=1;
do{
    $i++;
    echo "The number is " . $i . "<br />";
}while ($i<=5);
?>
```

- **for** (initialisation ; test ; incrément) instruction

```
<?php
for ($i=1; $i<=5; $i++){
    echo "The number is " . $i . "<br />";
}
?>
```

## Les instructions itératives (2)

- `foreach ($tabName as &$amp;content)` instruction

```
<?
$myTab=array(1, 2, 3, 4);
foreach ($myTab as &$myValue) {
    $myValue = $myValue* 2;
}
// $myTab is now array(2, 4, 6, 8)
?>
```

- `foreach ($tabName as $key => $value)`  
instruction

```
<?php
$a=array(1 => 'one', 2=> 'two', 3=> 'three');
foreach ($a as $key =>$value) {
    echo "Key: $key Value: $value";
}
?>
```

## Un peu de programmation procédurale

```
<?php  
function add($x,$y){  
    $total=$x+$y;  
    return $total;  
}  
echo "1 + 16 =" . add(1,16);  
?>
```



# Fonction de manipulation de chaînes

- int `strlen(string $ch)` longueur de \$ch
- int `strcmp(string $ch1, string $ch2)` compare deux chaînes
- string `trim(string $ch)` supprime les espaces en début/fin
- string `ltrim(string $ch)` supprime les espaces au début
- string `rtrim(string $ch)` supprime les espaces à la fin
- string `ucfirst($ch)` la première lettre en majuscule
- string `ucwords($ch)` la première lettre de chaque mot
- string `strtolower(string $ch)` tout en minuscules
- string `strtoupper(string $ch)` tout en majuscules
- string `nl2br(string $string)` remplace \n par <br>

# Fonction de manipulation de chaînes

- `int strpos(string $ch1, mixed $ch2)`
  - ❶ vérifie si `$ch2` est sous-chaîne de `$ch1` et
  - ❷ retourne la position de la 1ere occurrence de `$ch1` dans `$ch2`
- `int strrpos(string $ch1, mixed $ch2)`
  - ❶ vérifie si `$ch2` est sous-chaîne de `$ch1` et
  - ❷ retourne la position de la demiere occurrence de `$ch1` dans `$ch2`
- `mixed str_replace(mixed $ch1, mixed $ch2, mixed $ch3)`

toute les occurrences de `$ch2` dans `$ch1` on été remplacé par `$ch3`

## Fonction de manipulation de chaînes : exemple

```
<?php  
echo strlen("Hello world!");//12  
?>
```

```
<?php  
echo strpos("Hello world!","world");//6  
?>
```

# Fonction de manipulation de tableaux

- `int count(mixed $tab)` compte le nombre d'éléments
- `array explode(string $d, string $ch)`  
retourne le tableau des sous-chaînes de `$ch` divisé par `$d`
- `int array_push(array &$tab, mixed $v1[, mixed $v2...])` empile `$v1`, `$v2` dans `&$tab`
- `mixed array_pop(array $tab)` dépile le tableau
- `void unset(mixed $var)` supprime un élément du tableau
- `array array_unique(array $tab)` supprime les doublons
- `array array_merge(array $t1[, array $t2, array $t3...])` fusionne les tableaux
- `array array_intersect(array $t1[, array $t2,...])` retourne l'intersection des arguments
- `array array_diff(array $t1, array $t2[, array $t3...])`  
retourne la différence ensembliste entre `$t1` et `$t2...`

## Fonction de manipulation de tableaux (bis)

Les fonctions pour trier :

- les valeurs du tableau par ordre croissant  
`bool asort(array &$tab)`
- les valeurs du tableau par ordre décroissant `bool arsort(array &$tab)`
- les clefs du tableau par ordre croissant  
`bool ksort(array &$tab)`
- les clefs du tableau par ordre décroissant  
`bool krsort(array &$tab)`

# À emporter

- **Objectifs :**
  - › Langage de programmation
  - › Langage de programmation interprété sur le serveur
- **Contenu : PHP**
  - › La syntaxe basique
  - › <http://www.php.net>
  - › Fonctions standards
- **Perspectives :**
  - › PHP et formulaire/SQL

# Bibliographie



— Data, R. W3C  
schools.

[http ://www.w3schools.com/](http://www.w3schools.com/).



— Group, T. P.  
PHP.net.

[http ://php.net](http://php.net).



— Sklar, D. and Trachtenberg, A. (2002).  
*PHP Cookbook*.  
Éditions O'Reilly.