

---

---

# PHP

Les Bases

---

---

# C'est Quoi PHP

- L'un des langages de programmation les plus utilisés pour la programmation web
- Créé en 1994 par Rasmus Lerdorf
- Open source
- Permet de créer des pages web dynamiques
- PHP est exécuté (interprété) côté serveur
- Utilisé dans plusieurs systèmes de gestion de contenu CMS (content management systems) comme Wordpress, Drupal and Joomla
- Plusieurs Frameworks robustes écrits en PHP : Laravel, Symfony ...

# Installation

Pour réaliser les différentes section de ce cours on à besoin de :

1. Un serveur web : apache
2. Un interpréteur PHP
3. Un serveur de base de donnée mySql

Les trois softwares sont installable à l'aide d'un seul package: XAMP

Cross-platform (X), Apache web server (A), MarieDB database server (M), PHP (P) and Perl (P)

# php.ini

- Au cours de développement on a besoin de voir les erreurs dans notre navigateur. Pour cela il faut apporter quelques modifications à ***/opt/lampp/etc/php.ini***
- php.ini (fichier de configuration de php)
- Ajouter ces deux lignes à la fin du fichier
  - error\_reporting=E\_ALL
  - display\_errors=On
- Enregistrer et redémarrer apache

# Page Zero

- Vérifiez que apache est ON
- Dans le dossier **htdocs** (/opt/lampp/htdocs) créez un fichier **hello.php** avec le contenu ci-contre.
- Dans votre navigateur visitez le lien **<http://localhost/hello.php>**

Code interpreter par PHP (`<?php .... ?>`) et transformer en HTML



**`/opt/lampp/htdocs/tps/hello.php`**

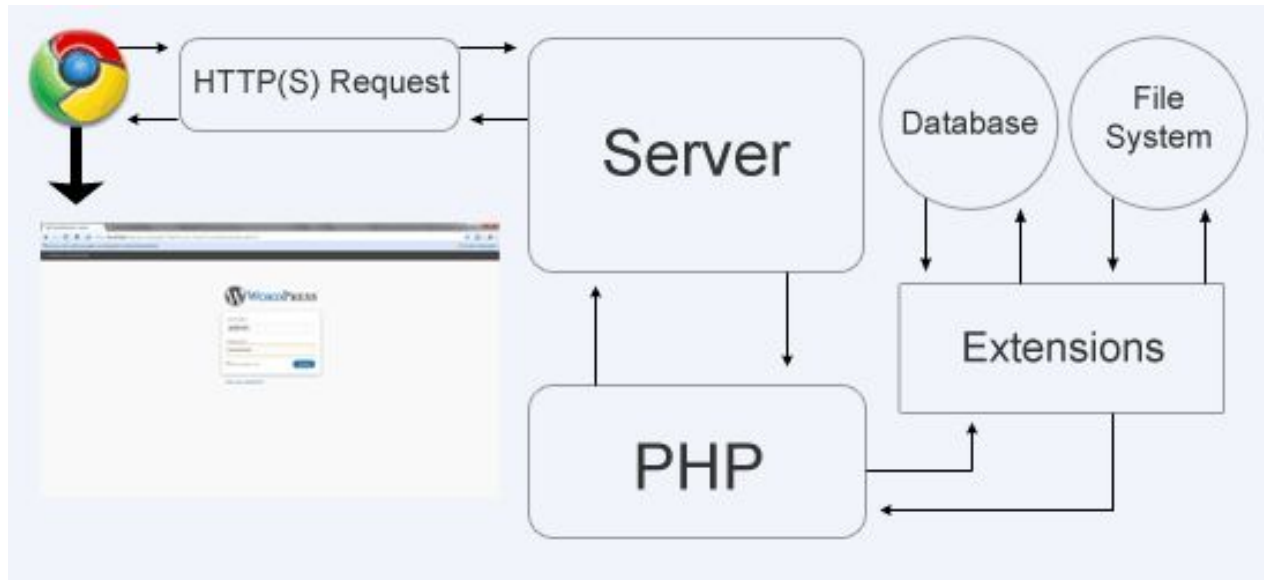
```
<!--DOCTYPE html>
<html>
  <head>
    <title>My first PHP page</title>
  </head>
  <body>
    <h1>Ma première page PHP</h1>
    <?php
      # un commentaire php
      echo "Bonjour Monde!";
    ?>
  </body>
</html>
```

# Page Zero

Fichier reçu par le navigateur

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4         <title>My first PHP page</title>
5     </head>
6     <body>
7         <h1>Ma première page PHP</h1>
8         Bonjour Monde!     </body>
9 </html>
```

# PHP workflow



Source: <https://padhmanaban.com/workflow-of-php/>

# Php: Output

`/opt/lampp/htdocs/outputs.php`

```
<?php
    echo '<h3>Bonjour tous !!!</h3>';
    echo ('un text <br>');
    echo "bonjour ", "tous", "<br>";
    print "Une autre output avec 'print'<br>";
    print " \\ est utilisé pour echapper des caractère spéciaux comme \\ et \" ";
```



localhost/web/00-intro/outputs.php



**Bonjour tous !!!**

un text

bonjour tous

Une autre output avec 'print'

\ est utiliser pour echapé des caractère specieaux comme \"



# Réutilisation du code

- Dans PHP les script sont généralement indépendants les uns des autres.
- Pour pouvoir inclure un code HTML ou PHP d'un fichier dans un autre fichier on utilise:
  - ***include*** : inclure un fichier avec un ***warning*** si ce dernier est ***introuvable***
  - ***require***: inclure un fichier avec une ***Erreur*** si ce dernier est ***introuvable***
  - ***Include\_once***: idem que ***include*** mais évite l'inclusion d'un fichier plusieurs fois
  - ***Require\_once***: idem que ***require*** mais évite l'inclusion d'un fichier plusieurs fois

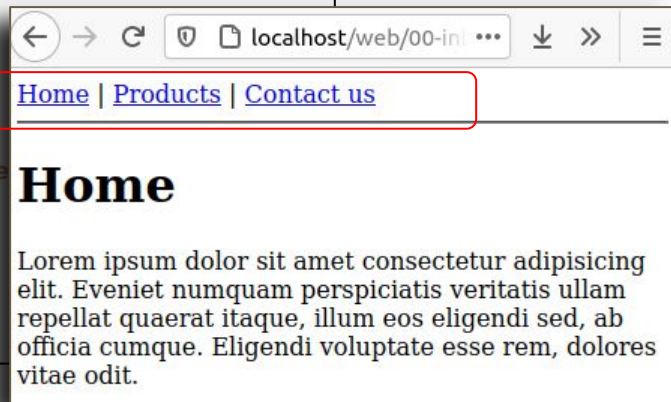
# Include

htdocs/home.php

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Home</title>
</head>
<body>
  <?php
    include "../views/header.html";
    echo "<h1> Home </h1>";
    echo "Lorem ipsum dolor sit amet consectetur adipisicing elit. Eveniet numquam perspiciatis veritatis ullam repellat quaerat itaque, illum eos eligendi sed, ab officia cumque. Eligendi voluptate esse rem, dolores vitae odit.";
  ?>
</body>
</html>
```

htdocs/views/header.html

```
<a href="home.html">Home</a> |
<a href="products.html">Products</a> |
<a href="contact.html">Contact us</a>
<hr>
```



# Redirection des utilisateurs: La fonction `header()`

- L'appel de la fonction ***header()*** doit être effectué ***avant*** tout autre instruction qui génère une sortie au navigateur:

```
<?php
```

```
header('Location: http://google.com');
```

```
?>
```

```
<!DOCTYPE html>
```

```
....
```

Le chargement de ce document redirrigera l'utilisateur vers <http://google.com>

# Redirection des utilisateurs: La fonction `header()`

Utilisation *inapproprié* de la fonction *header()*

## Exemple 1

```
<?php  
echo 'Hello';  
header('Location: http://example.com');  
?>
```

## Exemple 2

```
<p>This is some text.</p>  
<?php  
header('Location: http://example.com');  
?>
```



***Ces exemples peuvent passer sans problème en développement local mais pas dans un serveur de production***

# PHP: Constantes

- En PHP on peut définir une constante à l'aide de la fonction ***define(nom, valeur)***

```
define("MY_CONST", 1234);
```

```
echo MY_CONST;
```

```
echo '<BR>';
```

```
define("MY_CONST", 5678); # warning: Constant MY_CONST already defined ...
```

```
echo MY_CONST; # 1234
```

- La valeur de ***MY\_CONST*** reste inchangé dans tout le scripte

# PHP: Variables

- Les variables servent à stocker des données
- Le contenu d'une variable est modifiable
- En PHP le nom d'une variable commence par \$ suivi d'une alphabet puis tout autre caractère alphanumérique et \_ (A-z, 0-9, and \_)
- Les noms des variables sont sensibles à la casse \$nom et \$Nom sont deux variables

```
$nom  
$elem1  
$my_name  
$e1245
```



```
nom  
$1elem  
$my var  
$my_name!
```



*Essayez toujours d'utiliser des noms de variables significatifs et évitez des nom comme \$a, \$b ..*

# PHP: Variables

```
<?php
```

```
$taille= 76; # déclaration et initialisation d'une variable
```

```
echo $taille; # afficher le contenu d'une variable
```

```
$taille= 176; # modifier le contenu de la variable $taille
```

```
echo 'Ma taille est: $taille'; # affiche: Ma taille est $taille !!!
```

```
echo "Ma taille est: $taille"; # affiche: Ma taille est 176
```

```
?>
```

# Data types

- En PHP une variable peut être utilisée pour stocker différents types de valeurs:
  - **int** ou **integer**
  - **float** ou **double**
  - **bool** ou **boolean**
- Pour vérifier le type d'une variable on utilise la fonction ***var\_dump(\$nomVar)***
- Pour changer le type d'une variable (casting) on ajoute (int); (float) ou (bool) avant le nom de la variable
  - ***\$a= 2.7***
  - ***var\_dump(\$a)***    *# affiche float(2.7)*
  - ***(int)\$a***
  - ***var\_dump(\$a)***    *# affiche int(2)*
- On peut aussi utiliser (string); (array) ou (object)



# Les opérateurs de comparaison

Operator	Name	Example	Explanation
==	Equal	\$a == \$b	If \$a is equal to \$b, returns true
===	Identical	\$a === \$b	If \$a is equal to \$b and the same type, returns true
!=	Not equal	\$a != \$b	If \$a is not equal to \$b ,returns true
<>	Not equal	\$a <> \$b	If \$a is not equal to \$b ,returns true
!==	Not identical	\$a !== \$b	If \$a is not equal to \$b and not the same type, returns true
>	Greater than	\$a > \$b	If \$a is greater than \$b, returns true
<	Less than	\$a < \$b	If \$a is less than \$b, returns true
>=	Greater than or equal to	\$a >= \$b	If \$a is greater than or equal to \$b , returns true
<=	Less than or equal to	\$a <= \$b	If \$a is less than or equal to \$b, returns true
<=>	Spaceship	\$a <=> \$b	returns an integer less than, equal to, or greater than zero, depending on if \$a is less than, equal to, or greater than \$b.

# Les operateurs logique

Operator	Name	Example	Explanation
and	And	\$a and \$b	Return true if \$a and \$b both are true
or	Or	\$a or \$b	Return true if either \$a or \$b is true
xor	Xor	\$a xor \$b	Return true if either \$a or \$b is true but not both
&&	And	\$a && \$b	Return true if \$a and \$b both are true
	Or	\$a    \$b	Return true if either \$a or \$b is true
!	Not	!\$a	Return true if \$a is not true

# Les opérateurs de chaînes de caractères

Operator	Name	Example	Explanation
.	Concatenation	\$a . \$b	Concatenation of \$text1 and \$text2
.=	Concatenation and assignment	\$ .= \$b	appends \$text2 to the \$text1

# Les opérateurs d'affectation

Operator	Like as..	Explanation
A = B	A = B	Assign value from right side operand to left side.
A += B	A = A + B	Add both operands and then assign to left side
A -= B	A = A - B	Subtract the operands and result assign to left side
A *= B	A = A * B	Multiply both operands and then result assign to left side
A /= B	A = A / B	Division of operands and result assign to left side
A %= B	A = A % B	Modulus of two operands and assign to left side

# Les opérateurs conditionnels

Operator	Name	Example	Explanation
?:	Ternary	<code>\$a = expr1 ? expr2 : expr3</code>	Return the value of \$a. The value of \$a is <i>expr2</i> if <i>expr1</i> = TRUE. The value of \$a is <i>expr3</i> if <i>expr1</i> = FALSE
??	Null coalescing	<code>\$a = expr1 ?? expr2</code>	Returns the value of \$a. The value of \$a is <i>expr1</i> if <i>expr1</i> exists, and is not NULL. If <i>expr1</i> does not exist, or is NULL, the value of \$a is <i>expr2</i> .

# Les opérateurs d'incrémentation/décrémentation

Operator	Name	Explanation
<code>++\$a</code>	Pre-Increment	Increase the value of <code>\$a</code> by one, returns <code>\$a</code>
<code>\$a++</code>	Post-increment	Returns <code>\$a</code> , then increase value of <code>\$a</code> by one
<code>--\$a</code>	Pre-decrement	Decrease the value of <code>\$a</code> by one, returns <code>\$a</code>
<code>\$a--</code>	Post-decrement	Returns <code>\$a</code> , then decrease value of <code>\$a</code> by one

# Les chaînes de caractères

```
$msg = 'Hello';
```

```
$greeting = "Good Morning";
```

```
$emptyStr = "";
```

```
$areacode = "(208)";
```

```
$contact = '+1' . $areacode . '1234567';
```

```
echo $contact;
```

# Les chaînes de caractères

```
$str1 = 'Good Day!';  
echo strlen($str1);  
$str2 = 'Hello World';  
$str3 = strtolower($str2);  
$str4 = strtoupper($str2);  
echo '<BR>'.$str2;  
echo '<BR>'.$str3;  
echo '<BR>'.$str4;
```

```
$str5 = ' is ';  
echo 'PHP'.$str5.'Fun<BR>';  
echo 'PHP'.trim($str5).'Fun<BR>';  
$str6 = '**Hello**World**';  
echo trim($str6, '*');  
$str7 = 'ABCDEF';  
echo substr($str7, 2). '<BR>';  
echo substr($str7, -3). '<BR>';  
echo substr($str7, 2, 1);
```



# Les chaînes de caractères

```
echo strtotime("next Monday");
```

```
echo date('d-M-Y', strtotime("+ 10 hours"));
```

```
echo date('d-M-Y', strtotime("+ 10 hours"));
```

Dans php.ini : date.timezone=America/New\_York

```
date_default_timezone_set('Africa/Casablanca');
```

# Arrays

## Tableaux indexés

```
$firstArr = array();  
  
$secondArr = array(11, 16, 4, 9, 12);  
  
$fruitsArr = array('Apple', 'Banana',  
    'Coconut');  
  
echo $secondArr[3]; # Accéder au quatrième élément  
  
$secondArr[3] = 20; # Modifier le quatrième élément
```

## Tableaux associatifs

```
$assocArr = array(  
    'key1' => 11,  
    'key2' => 16,  
    'key3' => 12  
);  
  
echo $assocArr['key2'];
```

# Arrays

```
$simpleMDArr = array(  
    array(1, 2, 1, 4, 5),  
    array(0, 5, 1, 3, 4),  
    array(4, 1, 7, 8, 9)  
);  
  
echo $simpleMDArr[2];  
echo $simpleMDArr[2][3];
```

```
$assocMDArr = array(  
    "first array" => array(1, 2, 6, 1, 3),  
    "second array" => array(3, 5, 1, 8, 9),  
    "third array" => array(1, 0, 9, 4, 7)  
);  
  
echo $assocMDArr["first array"];  
echo $assocMDArr["first array"][2];
```

# Array

```
$anotherAssocMDArr = array(  
    "first player" => array("name" => 'John', "age"  
=> 25),  
    "second player" => array("name" => 'Tim', "age"  
=> 35)  
);  
  
echo $anotherAssocMDArr["first player"];  
echo $anotherAssocMDArr["first player"]["age"];
```

# Arrays

```
$myArray = array(2, 5.1, 'PHP', 105);
```

```
var_dump($myArray); # array(4) { [0]=> int(2) [1]=> float(5.1) [2]=> string(3) "PHP" [3]=> int(105) }
```

```
print_r($myArray); # Array ( [0] => 2 [1] => 5.1 [2] => PHP [3] => 105 )
```

# Arrays

- Ajouter un élément

```
$addDemo = array(1, 5, 3, 9);
```

```
$addDemo[] = 7;
```

```
$addDemoAssoc = array('Peter'=>20, 'Jane'=>15);
```

```
$addDemoAssoc['James'] = 17;
```

- Supprimer un élément

```
$colors =
```

```
array("red", "black", "pink", "white");
```

```
array_splice($colors, 2);
```

```
$awardwinners = array(
```

```
"Gold"=>"Max",
```

```
"Silver"=>"Boots",
```

```
"Bronze"=>"Dora");
```

```
array_splice($awardwinners, 1);
```

```
$pets = array("corgi", "poodle",  
"golden retriever", "jack russell");
```

```
array_splice($pets, 1, 2);
```

# PHP Superglobals

- Les superglobals sont des variables prédéfinie dans PHP et sont accessible dans tous les scripts du projet:
  - ***`$GLOBALS`***
  - ***`$_SERVER`***
  - ***`$_GET`***
  - ***`$_POST`***
  - ***`$_SESSION`***
  - ***`$_COOKIE`***
  - ***`$_REQUEST`***
  - ***`$_ENV`***
  - ***`$_FILES`***

<https://www.php.net/manual/en/language.variables.superglobals.php>

# PHP Superglobals: \$GLOBALS

**\$GLOBALS**: Un tableau associatif pour référencer toutes les dans le scop globale du scripte en cours

```
<?php

function test() {

    $foo = "local variable";

    echo 'foo in global scope: ' . $GLOBALS["foo"] . "\n";    # Affiche: $foo in global scope: Example content

    echo 'foo in current scope: ' . $foo . "\n";              # Affiche : $foo in current scope: local variable

}

$foo = "Example content";

test();

?>
```



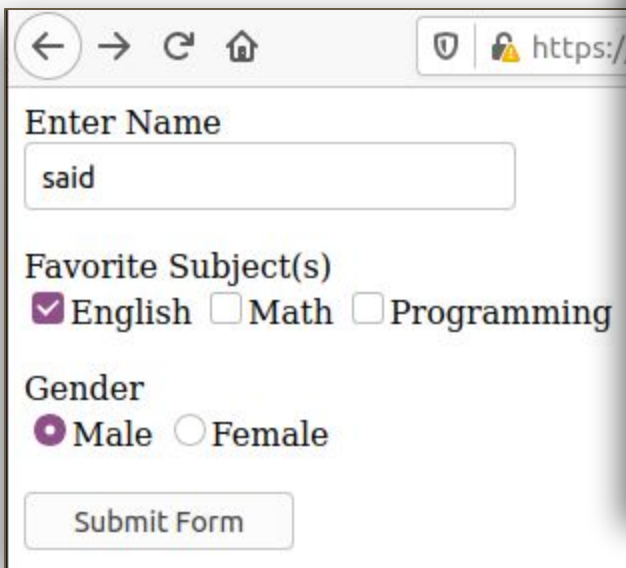
# PHP Superglobals: \$\_SERVER

**\$\_SERVER:** Un tableau associatif contenant des information sur l'environnement d'exécution du scripte

- **\$\_SERVER['PHP\_SELF']:** le chemin du scripte en cours d'exécution par exemple si le script est *http://example.com/foo/bar.php* alors on obtient */foo/bar.php*
- **\$\_SERVER['SERVER\_ADDR']:** L'adresse IP du serveur
- **\$\_SERVER['REMOTE\_ADDR']:** L'adresse IP du client
- **\$\_SERVER['SERVER\_NAME']:** Le nom du serveur
- **\$\_SERVER['REQUEST\_METHOD']:** La méthode utilisé pour accéder à la page: GET, POST ...
- ...

# PHP Superglobals: \$\_GET, \$\_POST

*htdocs/web/01-superglobals/form.html*



Enter Name

said

Favorite Subject(s)

☒ English ☐ Math ☐ Programming

Gender

☒ Male ☐ Female

Submit Form

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>PHP Form Handling</title>
5   </head>
6   <body>
7     <form action="process.php" method="get">
8       Enter Name <br />
9       <input type="text" name="studentname" value="Your Name" />
10      <br /><br />Favorite Subject(s)<br />
11      <input type="checkbox" name="subj[]" value="EL" />English
12      <input type="checkbox" name="subj[]" value="MA" />Math
13      <input
14        type="checkbox"
15        name="subj[]"
16        value="PG"
17      />Programming<br /><br />Gender <br />
18      <input type="radio" name="gender" value="M" />Male
19      <input type="radio" name="gender" value="F" />Female<br /><br />
20      <input type="submit" name="sm" value="Submit Form" />
21    </form>
22  </body>
23 </html>
```

# PHP Superglobals: \$\_GET, \$\_POST

- Un clique sur le bouton submit permet d'envoyer les données à l'aide plusieurs méthode:
  - GET: les données sont joint à la l'URL sous le nom de "query string":  
*<https://localhost/web/01-superglobals/process.php?studentname=said&subj%5B%5D=EL&subj%5B%5D=PG&gender=M&sm=Submit+Form>*
  - POST: Les données sont envoyé dans le body de la requête
  - ...
- Pour récupérer les données du formulaire nous avons recours aux superglobals:
  - *`$_GET["studentname"] : "said"`*
  - *`$_GET["subj"] : array("EL", "PG")`*
  - *`$_GET["gender"] : "M"`*
  - *`$_GET["sm"] : "Submit Form"`*
  - Si la méthode et POST on utilise *`$_POST[]`*

# TP N2

- Forms
  - Creating form
  - Getting values using \$\_POST and \$\_GET
  - The isset() function
  - Keeping The Values in The Form
  - Filtering User Input
    - filter\_var()
  - Cross-Site Scripting
    - htmlspecialchars()

# PHP Superglobals: \$\_SESSION

- \$\_SESSION est un tableau associatif utilisé pour partager des données entre plusieurs pages

*htdocs/web/01-superglobals/session1.php*

```
1 <?php
2     session_start();
3     $_SESSION['myFavFood'] = 'Couscous';
4     $_SESSION['myFavDrink'] = 'Cola';
5     $_SESSION['myFavColor'] = 'Orange';
6     $_SESSION['myFavDrink'] = 'Tea'; #updating a session variable
7     unset($_SESSION['myFavColor']);#deleting a session variable
8
```

*htdocs/web/01-superglobals/session2.php*

```
1 <?php
2     session_start();
3     echo '<BR>Food: ' . $_SESSION['myFavFood']; # affiche: Food: Couscous
4     echo '<BR>Drink: ' . $_SESSION['myFavDrink']; # affiche: Drink: Tea
5     echo 'Color: ' . $_SESSION['myFavColor']; # Warning: Undefined array key "myFavColor"
6 ?>
```