

php

Présentation du langage

PHP est un langage de script conçu pour le développement d'application web.

Il s'intègre facilement dans du contenu html PHP est un langage qui s'exécute côté serveur et permet la génération de page web dynamique.

L'interpréteur PHP va alors générer une page web html.

Environnement de travail

Pour développer en PHP nous allons avoir besoin d'un serveur, wamp (Windows) ou Lamp (Linux) suivant notre environnement de travail.

- Apache (serveur web pour héberger nos différents fichiers)
- MySQL (serveur de base de données, pour héberger nos bdd)
- PHP (interpréteur PHP)

Tp-1 installer wamp

<https://www.wampserver.com/>

Pensez à installer tout les prérequis windows !

<https://wampserver.aviatechno.net/?lang=fr&prerequis=afficher>

Syntaxe

En PHP nous écrivons nos scripts à l'intérieur de fichier. php.

Exemple : Index.php

Syntaxe

```
<!-- Nos scripts php devront être rédigé entre les balises : -->
<?php
?>
<!-- Exemple du code d'une page php : -->
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>ma première page php</title>
</head>
<body>
  <h1>mon premier programme</h1>
  <?php
//le script php se trouvera entre ces balises
  ?>
</body>
</html>
<!-- Chaque ligne de nos scripts devra se terminer par un ; -->
<?php
// Mon code
?>
```

Commentaires

<?php

//commentaire sur une ligne

*/**

Commentaire sur plusieurs lignes

**/*

Tp-1

Créez votre premier fichier php et faites à l'intérieur des balises php un « hello world » qui s'affichera en html.

Indice : recherchez la fonction echo().

Tp-1-echo correction

```
<!DOCTYPE html>
<html lang="fr">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>echo</title>
</head>
<body>
    <?php
        echo "hello world !"
    ?>
</body>
</html>
```

Les variables

Les différents types de variables.

- String = « chaine de caractères »
- Int = « nombre entier »
- Float= « nombre décimale »
- Booléen = « vrai ou faux »
- Array = « tableau »
- Object = « objet »
- Null = vide

Pour déclarer une variable on utilise le « \$ » exemple :

```
$maVariable = 152;
```

Les constantes

```
<?php  
define("URL", 'https://www.jeuxvideo.com')
```

Echo URL

Tp-2-les variables

Créez une variable de type :

- Int
- Bool
- String
- Null
- Float

Affichez leur contenu ainsi que leur type afin de voir si vous avez réussi .

Indice : Plusieurs méthodes possibles.

Tp-2-les variables-correction

```
<?php
    $string = "coucou";
    $bool = true;
    $null = null;
    $int = 22;
    $float = 22.22;

    var_dump($string,$bool,$null,$int,$float);
?>
```

les opérateurs

Pour effectuer des opérations mathématiques sur des types numériques (int, long, float etc...)

On utilise les opérateurs mathématiques suivant :

```
<?php
    $a = 10;
    $b = 2;

    $a + $b; // ADDITION
    $a - $b; // SOUSTRACTION
    $a * $b; // MULTIPLICATION
    $a / $b; // DIVISION
    $a % $b; // MODULO (reste de la division)
    $a ** $b; // exponentielle
?>
```

Tp-3- les opérateurs

```
$a = 10;  
$b = 2;
```

En partant de ce code, permutuez la valeur de \$a avec celle de \$b (je vous laisse trouver l'algorithme) SANS ECRIRE \$b = 10 et \$a=2.

Et ensuite faites une division de \$a par \$b.

Tp-3- les opérateurs correction

```
<?php
```

```
$a = 10;  
$b = 2;
```

```
$c = $a;  
$a = $b;  
$b = $c;
```

```
$result= $a/$b;  
var_dump($a,$b);  
var_dump($result);
```

```
?>
```


Les bases : les conditions

En php il existe plusieurs moyen de faire des conditions,

If – if else – else

Switch

Ternaire

Exemple :

```
<?php
$maNote = 13;

if ($maNote < 12) {
    echo "<p>tu es nul</p>";
} else {
    echo "<p>tu es bon (if)</p>";
}

echo $maNote < 12 ? "<p>tu es nul</p>" : "<p>tu es bon (ternaire)</p>";

switch ($maNote) {
    case $maNote < 12:
        ?><p>vous êtes nul</p>
        <?php
        case $maNote > 12:
            ?><p>vous êtes bon (switch)</p>
        <?php
    }
    ?>
```

Les bases : les tableaux

Les tableaux

en php il y a deux type de tableau:

tableau normaux :

```
C:\wamp64\www\coursPhp\revision-tableau.php:21:
array (size=4)
  0 => string 'benjamin' (length=8)
  1 => string 'Ophélie' (length=8)
  2 => string 'Julien' (length=6)
  3 => string 'Justine' (length=7)
```

tableau associatif :

```
C:\wamp64\www\coursPhp\revision-tableau.php:23:
array (size=3)
  'nom' => string 'bailly' (length=6)
  'prenom' => string 'benjamin' (length=8)
  'age' => string '27' (length=2)
```

```
<?php
$normalTab = ["benjamin", "Ophélie", "Julien", "Justine"]
;
$associativeTab = [
    "nom"=>"bailly",
    "prenom"=>"benjamin",
    "age"=>"27"
];
?>
```

Les bases : les boucles - for

```
<?php
for($i=0; $i<=20; $i++){
?>
<p>vous êtes le groupe <?=$i?> </p>
<?php
}
?>
```

Le premier paramètre est l'initialisation d'une variable.

Le deuxième paramètre est la condition à vérifier.

Le 3^{ème} est l'incrément.

vous êtes le groupe 0
vous êtes le groupe 1
vous êtes le groupe 2
vous êtes le groupe 3
vous êtes le groupe 4
vous êtes le groupe 5
vous êtes le groupe 6
vous êtes le groupe 7
vous êtes le groupe 8
vous êtes le groupe 9
vous êtes le groupe 10
vous êtes le groupe 11
vous êtes le groupe 12
vous êtes le groupe 13
vous êtes le groupe 14
vous êtes le groupe 15
vous êtes le groupe 16
vous êtes le groupe 17
vous êtes le groupe 18
vous êtes le groupe 19
vous êtes le groupe 20

Les bases : les boucles - while

```
$i = 0;
while ($i < 20) {
    $i++;
    ?>
    <p>vous êtes le groupe <?= $i ?> </p>
    <?php
}

?>
```

vous êtes le groupe 0
vous êtes le groupe 1
vous êtes le groupe 2
vous êtes le groupe 3
vous êtes le groupe 4
vous êtes le groupe 5
vous êtes le groupe 6
vous êtes le groupe 7
vous êtes le groupe 8
vous êtes le groupe 9
vous êtes le groupe 10
vous êtes le groupe 11
vous êtes le groupe 12
vous êtes le groupe 13
vous êtes le groupe 14
vous êtes le groupe 15
vous êtes le groupe 16
vous êtes le groupe 17
vous êtes le groupe 18
vous êtes le groupe 19
vous êtes le groupe 20

Les bases : les boucles - foreach

```
$associativeTab = [  
    "nom"=>"bailly",  
    "prenom"=>"benjamin",  
    "age"=>"27"  
];
```

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

```
nom : bailly  
prenom : benjamin  
age : 27
```

La syntaxe plus haut permet de stocker la valeur et la clé dans des variables, il est possible de stocker uniquement la valeur .
Pour concaténer en php il suffit de mettre un « . ».

Les bases : les fonctions

Liste de toutes les fonctions php :

<https://www.php.net/manual/fr/indexes.functions.php>

Il est également possible de créer vos propres fonctions :

```
<?php
```

```
function hello($nom){  
    echo "<p> bonjour ".$nom." .";  
}  
hello("Jean");
```

Tp-4- les bases

Créez une fonction, qui prendra en argument un tableau. Vous pouvez passé n'importe quel numéro dans ce tableau, la fonction retournera le plus grand. Si un tableau n'est pas passé, il faudra affiché un message précisant que vous voulez un tableau.

Fonctions utiles : `is_array()` et `empty()`

Je ne vous en dis pas plus je vous laisse réfléchir à un algo et un code qui part à toutes les éventualités.

Correction sur discord .

Tp-4- les bases(2)

Créer une fonction qui prend 2 arguments :

1 – une string qui sera le titre d'une liste

2 – un tableau qui représente les éléments de cette liste

La fonction devra retourner une liste HTML. Chaque élément de cette liste viendra dynamiquement du tableau passé en paramètre.

Si la liste ou le tableau est vide la fonction retournera null;

```
echo superList("formateur", ["Benjamin", "Ophelie", "Julien"]);
```

Correction sur discord .

formateur

- Benjamin
- Ophelie
- Julien