

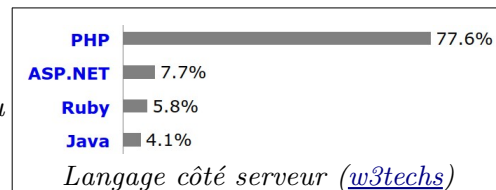
Introduction au langage PHP

I) Histoire et utilisations de PHP

Créé en 1994 en langage C par Rasmus Lerdorf pour son site web personnel sous le nom **PHP/FI** (pour *Personal Home Page Tools/Form Interpreter*)

En 1997, Deux étudiants [Andi Gutmans](#) et [Zeev Suraski](#) réécrivent le langage et le nomment *PHP: Hypertext Preprocessor*. Ils réécrivent le moteur de PHP et le nomment *Zend Engine* (*Zeev Andi*).

PHP est très utilisé dans le WEB pour propulser des sites WEB comme [Facebook.com](#), [Wikipedia.org](#), [Instagram.com](#) ou [zoom.us](#)



PHP est né comme un langage de scripts pour produire du HTML de manière dynamique et reste très utilisé à cette fin. Au fil du temps, PHP a évolué pour devenir un langage de programmation complet. Il permet de créer des applications web (wordpress, nextcloud, PHPmyadmin, GLPI...), de réaliser des scripts ou même des applications.

II) Syntaxe, entrées et sorties

Un script PHP a pour extension **.php** et la première ligne doit-être **<?php**

Chaque instruction se termine obligatoirement par le symbole **;**.

La commande **echo** permet un affichage dans la console.

Fichier monscript.PHP	Console
<pre><?php echo "Hello World!";</pre>	PS C:\> php monscript.php Hello World!

Variables

En PHP, les variables commencent par le symbole **\$**. Vous pouvez nommer vos variables comme vous voulez mais il est préférable de les écrire en minuscule en séparant les mots par un **_** (underscore).

Fichier exemple.PHP	Console
<pre><?php \$ma_variable = 7; echo \$ma_variable + 2;</pre>	PS C:\> php exemple.php 9

Entrée console: readline

On peut demander à l'utilisateur une information à l'aide de l'instruction **readline()**.

Dans l'exemple ci-contre, l'utilisateur a rentré le nombre **3** dans la console.

Fichier exemple.PHP	Console
<pre><?php \$a = readline("un nombre:"); echo \$a + 2;</pre>	PS C:\> php exemple.php un nombre: 3 5

Exercice 1: Créer un script demandant son année de naissance à l'utilisateur et affichant son age.

III) Structures conditionnelles

On peut effectuer certaines instructions seulement quand une condition est vérifiée à l'aide de la structure **if**. Le bloc **else** sera exécuté dans le cas contraire.

Exercice 2: Créer un script qui demande son année de naissance à l'utilisateur et qui affiche s'il est majeur ou non.

Syntaxe **if**

```
<?php
if (condition) {
    instructions;
}
else {
    instructions;
}
```

On peut également tester plusieurs conditions à l'aide de l'instruction **elseif**. Dans l'exemple ci-contre:

Si **condition1** est vérifiée: PHP effectue **instruction1** et sort du **if**.

Si **condition1** n'est pas vérifiée et que **condition2** est vérifiée: PHP effectue **instruction2** et sort du **if**.

Si **condition1** n'est pas vérifiée et que **condition2** non plus: PHP effectue **instruction3** et sort du **if**.

Syntaxe **elseif**

```
<?php
if (condition1){
    instructions1;
}
elseif (condition2){
    instructions2;
}
else {
    instructions3;
}
```

Exercice 3:

1. Que va afficher le script ci-contre?
2. Et si on change la deuxième ligne par **\$a = 39** ?
3. Quelle valeur doit-on mettre à **\$a** pour faire afficher D ?

Fichier **exercice.PHP**

```
<?php
$a = 42;
if ($a <= 40) { echo "A"; }
elseif ($a > 38) { echo "B"; }
elseif ($a > 41) { echo "C"; }
else { echo "D"; }
```

Comparaison en PHP

Nous avons vu ici les opérateurs de comparaison > et <= . En voici d'autres.

Pour la différence entre être égal et être identique, voir la [documentation PHP](#). Pour faire simple, être identique implique d'avoir le même type.

Opérateur	Sens
\$a == \$b	\$a est égal à \$b
\$a === \$b	\$a est identique à \$b
\$a < \$b	\$a est strictement inférieur à \$b
\$a <= \$b	\$a est inférieur ou égal à \$b

Si les conditions ne portent que sur la valeur d'une variable, on peut utiliser l'instruction **switch**.

Attention: Une fois que PHP rencontre un cas positif, il va effectuer toutes les instructions qui suivent dans le **switch**. Il est recommandé d'utiliser **break** à la fin d'un cas pour éviter cela. L'instruction **break** va sortir complètement au **switch**.

Syntaxe **switch**

```
<?php
switch($var){
    case a: #si $var == a
    case b: #si $var == b
    default: #tous les cas
}
```

Exercice 4:

1. Que va afficher le script suivant?
2. Si on change la deuxième ligne par `$var = 2` ?
3. Si on change la deuxième ligne par `$var = 42` ?

Fichier **exercice.PHP**

```
<?php
$var = 1
switch($var){
    case 1: echo 1;
    case 2: echo 2; break;
    default: echo 3;
}
```

Exercice TP 5:

Le prix d'une place de cinéma est de 10€ pour les plus de 20 ans, 7,5€ pour les personnes entre 5 et 19 ans et gratuit pour les autres. Écrire un algorithme qui demande l'année de naissance et affiche le prix à payer.

Exercice TP 6:

Écrire un script qui demande la date de naissance au format `jj/mm/aaaa` et qui dit s'il est majeur. Il faudra utiliser la fonction `date()` de PHP pour être précis au jour près!

III) Tableaux en PHP:

Le tableau en PHP (**array**) est un type qui associe une valeur à une clef. La clef est un entier ou une chaîne de caractère. On peut créer un tableau avec la structure de langage `array()`.

Exercice TP 7: Ne pas supprimer le code entre deux questions.

1. Effectuer le code php ci-contre. Corriger les erreurs.
2. Effectuer ensuite les instructions `echo $pairs;`
`var_dump($impairs);` et `print_r($pairs);`
3. Afficher la valeur de `current($pairs)` puis de `next($pairs)`.
4. Afficher de nouveau la valeur de `current($pairs)`
5. Afficher la valeur de `$impairs["clef3"]` et de `$pairs[3]`.

Syntaxe **array**

```
<?php
$pairs = array(0, 2, 4, 6, 8)
$impairs = array(
    "clef1" => 1,
    "clef2" => 3,
    "clef3" => 5,
```

Un tableau PHP a un pointeur interne qui commence au premier élément. On peut naviguer dans le tableau à l'aide des fonctions ci-contre.

On peut également obtenir la valeur d'un élément du tableau à l'aide de `$tab[clef]`.

Fonction PHP	Description
<code>current(\$tab)</code>	Renvoie l'élément courant du tableau
<code>next(\$tab)</code>	Avance le pointeur et renvoie l'élément
<code>prev(\$tab)</code>	Recul le pointeur et renvoie l'élément
<code>reset(\$tab)</code>	Place le pointeur au premier élément et le renvoie

Exercice TP 8:

1. Ouvrir l'url suivante dans un navigateur. Qu'y voit-on?
`https://data.rennesmetropole.fr/api/records/1.0/search/?dataset=prevision-meteo-rennes-arome&rows=1`
2. Dans un fichier PHP, utiliser `file_get_contents($url)` pour obtenir les données.
3. Convertir les données en tableau avec `json_decode($string, true)`.
4. Afficher les prévisions météorologiques de Rennes.