

### Syntaxe de base

Code PHP	<?php //Contenu ?>
Commentaire sur une ligne	// Commentaire
Commentaires sur plusieurs lignes	/* Commentaire */
Fin d'instruction	;
Inclusion de fichier	require_once('nom_fichier.php');

### Variables et Constantes

\$nomVariable = "Chaîne de caractère";
\$nomVariable = 'Chaîne de caractère';
\$nomVariable = 5;
\$nomVariable = "Une {\$autreVariable} info";
echo \$nomVariable;
global \$varGlobale;
echo \$GLOBALS['varGlobale'];
define('NOMCONSTANTE', 'valeur');
echo NOMCONSTANTE;

### Fonctions sur les variables

Vérifier l'existence de la variable	isset(\$var);
Détruire une variable	unset(\$var);
Connaître le type	gettype(\$var);
Vérifier un type	is_[type](\$var); Ex : is_string(\$var);
Cast (changement de type)	\$var = (string) \$var;
Conversion de valeur	[type]val(\$var); Ex : intval(\$var); floatval(\$var);
Test si la variable est vide	empty(\$var);

### Tableaux

Création	\$tableau = array();
Ajout	\$tableau[] = "valeur";
Ajout sur un index	\$tableau[4] = "valeur";

### Tableaux (cont)

Ajout sur une clé	\$tableau["cle"] = "valeur";
Création numérique	\$tableau = array('valeur1', 'valeur2');
Création associative	\$tableau = array('cle1' => 'valeur1', 'cle2' => 'valeur2');
Ecriture depuis numérique	echo \$tableau[4];
Ecriture depuis association	echo \$tableau['cle1'];
Tableaux numériques	Les clés sont des chiffres
Tableaux associatifs	Les clés sont des chaînes de caractères
Matrice (tableau multi-dimensions)	\$matrice[2][3] = "valeur";

### Types

Booléen	boolean	true OU false
Entier	integer	nombre positif ou négatif
Nombre flottant	float	nombre à virgule positif ou négatif
Nombre flottant	double	nombre à virgule positif uniquement
Chaîne de caractère	string	chaîne de caractères

### Chaînes de caractères (string)

Délimités entre guillemet ou apostrophe	"chaîne de caractères" ou 'chaîne de caractères'
Entre guillemet, les variables sont interprétés	\$var = caractères; echo "chaîne de \$var";
Caractère d'échappement	\
Retour à la ligne	\n
Retour chariot	\r

### Chaînes de caractères (string) (cont)

Tabulation	\t
------------	----

### Fonction utilisateur

```
function multiplier($arg1, $arg2)
{
    return $arg1 * $arg2;
}
$params = 4;
$params = 8;
$resultat = multiplier($params, $params);
```

### Classes and Objects

```
class SomeClass {
    private $property;
    public $anotherProperty;
    protected $yetAnotherProperty = null;
    public function __construct($arg=null)
    {
        $this->property = $arg;
    }
    public function someMethod()
    {
        echo "Hi";
    }
    public function getProperty()
    {
        return $this->property;
    }
    public function setProperty($p)
    {
        $this->property = $p;
    }
}
$myObject = new SomeClass("123");
echo $myObject->getProperty(); // 123
$myObject->property; // ERROR:private
```



### Operateurs

Affectation	=	<code>\$var = 5;</code>
Affectation par référence	<code>&amp;=</code>	<code>\$nouveiVar &amp;= \$var;</code>
Addition	+	<code>\$var = \$var + 5;</code>
Soustraction	-	<code>\$var = \$var - 5;</code>
Multiplication	*	<code>\$var = \$var * 5;</code>
Division	/	<code>\$var = \$var / 5;</code>
Modulo	%	<code>\$var = \$var % 5;</code>
Incrémentation	++	<code>\$var = \$var++;</code>
Décrémentation	--	<code>\$var = \$var--;</code>
Opérateurs combinés	[opérateur]=	<code>\$var += 5;</code> <code>\$var *= 5;</code>
Concaténation	.	<code>echo \$var." chaîne";</code>
Concaténation et assignation	<code>.=</code>	<code>\$var .= " chaîne";</code>

### Operateurs de comparaison

Egalité	==	<code>if(\$var == 5)</code>
Inférieur à	<	<code>if(\$var &lt; 5)</code>
Inférieur ou égal à	<=	<code>if(\$var &lt;= 5)</code>
Supérieur à	>	<code>if(\$var &gt; 5)</code>
Supérieur ou égal à	>=	<code>if(\$var &gt;= 5)</code>
Différent de	!=	<code>if(\$var != 5)</code>
Strictement égal (valeur et type)	===	<code>if(\$var === 5)</code>
Différent en valeur ou en type	!==	<code>if(\$var !== 5)</code>

UtilisÃ©s pour les structures conditionnelles

### Operateurs logiques

Inversion	!	Retourne true si false, et inversement
ET	&&	Retourne true si 2 conditions à true
Un seul	^	Retourne true si une seule des conditions à true
OU		Retourne true si une condition à true
ET non prioritaire	AND	Similaire à && mais moins prioritaire
Un seul non prioritaire	XOR	Similaire à ^ mais moins prioritaire
OU non prioritaire	OR	Similaire à    mais moins prioritaire

UtilisÃ©s pour les structures conditionnelles

### Fonctions PHP utiles

Récupérer une partie d'une chaîne	<code>substr(\$string, start, length);</code>
Transformer une chaîne en tableau	<code>explode(',', \$string);</code>
Concaténer un tableau en chaîne	<code>implode(',', \$tableau);</code>
Retirer les espaces au début et à la fin d'une chaîne	<code>trim(\$string);</code>
Remplacer à par b dans une chaîne	<code>str_replace('a', 'b', \$string);</code>
Vérifier une expression régulière	<code>preg_match('regex', \$string);</code>
Remplacer une expression régulière par b	<code>preg_replace('regex', 'b', \$string);</code>
Arrêter le script PHP	<code>exit();</code>
Envoyer un mail	<code>mail(\$mailDest, \$sujet, \$message, 'From: '.\$mailEnvoi);</code>

### Expression reguliere

^	Début de chaîne
\d	Chiffre entre 0 et 9
\w	Caractère alphanumérique [0-9A-Za-z]
\s	Espace
.	N'importe quelle lettre, chiffre ou espace
\$	Fin de chaîne
()	Groupe
[]	Classe de caractères
{x} {x,}	Quantité = x   Supérieur ou égal à x
{x,y}	Entre x et y
*	Quantité de 0 ou plus
?	Quantité de 0 ou 1
+	Quantité de 1 ou plus
	OU
\	Caractère d'échappement

Exemple pour une syntaxe de mail :

```
^[w.-+]+@[w.-]+\.[a-zA-Z]{2,6}$
```

### Structure conditionnelle : IF

```
if (condition) {
// Instructions
}
elseif (condition) {
// Instructions
}
else {
// Instructions
}
if ( $something == true ) {
// Si $something vaut true
doSomething();
} elseif ( $something == false ) {
// Si $something vaut false
doSomethingElse();
} else {
// sinon, exécuter doNothing();
doNothing();
}
if(condition):
```



By **Zetura** (Zetura)  
[cheatography.com/zetura/](https://cheatography.com/zetura/)  
[atago.fr](https://atago.fr)

Published 19th February, 2014.  
Last updated 2nd June, 2014.  
Page 2 of 3.

Sponsored by **Readability-Score.com**  
Measure your website readability!  
<https://readability-score.com>

### Structure conditionnelle : IF (cont)

```
// Instructions
endif;
(condition)? instructions si true : instructions si
false;
```

### Structure conditionnelle : SWITCH

```
switch ($var) {
case 1:
// Instructions
break;
case "test":
// Instructions
break;
default:
// Instructions
break;
}
```

Peut être utilisé avec des chiffres ou chaînes de caractères

### Boucle WHILE

Tant que la condition est vraie, l'instruction est exécutée

```
while(condition){
// Instructions
}
$i = 1;
while($i < 10){
echo $i;
$i++;
}
Exécution au moins une première fois
$i = 1;
do{
echo $i;
$i++;
}
while($i < 10);
```

Attention aux boucles infinies

### Boucle FOR

Exécute la première expression lors de l'initialisation, puis tant que la condition est valide, exécute le contenu de la boucle et fini en exécutant la dernière expression

```
for(expression1; condition; expression2){
// Instructions
}
for($i = 1; $i < 10; $i++){
// Instructions
}
```

### Boucle FOREACH

A chaque itération dans la boucle assigne la valeur de l'élément courant à la variable et le pointeur interne du tableau est avancé d'un élément.

```
foreach($tableau as $element){
// Instructions
}
foreach($tableau as $key => $value){
// Instructions
}
$tableau = array(
'prenom' => 'Obi-wan',
'nom' => 'Kenobi',
'metier' => 'Jedi'
);
foreach($tableau as $contenu){
echo "Valeur : $contenu<br/>";
}
foreach($tableau as $cle => $valeur){
echo "Clé : $cle -> Valeur : $valeur<br/>";
}
```

Attention, la boucle fonctionne sur une copie du tableau spatialisé, pas sur le tableau lui-même

### CONTINUE

```
for ($i = 0; $i < 5; ++$i) {
if ($i == 2)
continue;
print "$i , ";
}
```

produces the following output:

0 , 1 , 3 , 4



By **Zetura** (Zetura)  
[cheatography.com/zetura/](http://cheatography.com/zetura/)  
[atago.fr](http://atago.fr)

Published 19th February, 2014.  
Last updated 2nd June, 2014.  
Page 3 of 3.

Sponsored by **Readability-Score.com**  
Measure your website readability!  
<https://readability-score.com>