



Développez des sites web dynamiques en PHP & MySQL

Introduction au PHP

[TELECHARGER LE COURS](#)

Introduction au PHP

Qu'est-ce que PHP ?

- PHP est l'acronyme de "PHP : Hypertext Preprocessor"
- PHP est un langage de script open source largement utilisé Les scripts PHP sont exécutés sur le serveur
- PHP est gratuit à télécharger et à utiliser

Que peut faire PHP ?

- PHP peut générer du contenu de page dynamique
- PHP peut créer, ouvrir, lire, écrire, supprimer et fermer des fichiers sur le serveur
- PHP peut collecter des données de formulaire PHP peut envoyer et recevoir des cookies
- PHP peut ajouter, supprimer, modifier des données dans votre base de données
- PHP peut être utilisé pour contrôler l'accès des utilisateurs PHP peut chiffrer les données

Installation de Serveur PHP

Pour commencer à utiliser PHP, vous pouvez :

- Trouver un hébergeur avec support PHP et MySQL.
- Installez un serveur Web local sur votre propre PC, puis installez PHP et MySQL (WAMP-XAMP-MAMP)

Syntaxe PHP

Un script PHP peut être placé n'importe où dans le document.

Un script PHP commence par `<?php` et se termine avec `?>` .

L'extension de fichier par défaut pour les fichiers PHP est ".php" .

Un fichier PHP contient normalement des balises HTML et du code de script PHP.

Variables

En PHP, une variable commence par le **\$** signe, suivi du nom de la variable :

Exemple :

```
<?php  
$txt = "PHP";  
$x = 15;  
$y = 10.5;  
echo "I love $txt!"; *// I LOVE PHP!  
echo $x + $y; // 25.5  
?>
```

Type de données PHP

Les variables peuvent stocker des données de différents types. PHP prend en charge les types de données suivants :

- String
- Integer
- Float ou double
- Boolean
- Array
- Object
- NULL

STRING PHP

Une chaîne est une séquence de caractères, comme "Hello world!".

Une chaîne peut être n'importe quel texte entre guillemets.

Vous pouvez utiliser des guillemets simples ou doubles

```
<?php
```

```
$x = "Hello world!";
```

```
$y = 'Hello world!';
```

```
?>
```

INTEGER PHP

Un type de données entier est un nombre non décimal compris entre -2 147 483 648 et 2 147 483 647.

Règles pour les entiers :

- Un entier doit avoir au moins un chiffre
- Un entier ne doit pas avoir de point décimal Un entier peut être positif ou négatif

FLOAT OU DOUBLE PHP

Un flottant (nombre à virgule flottante) est un nombre avec un point décimal ou un nombre sous forme exponentielle.

```
<?php
```

```
$x = 10.365;
```

```
var_dump($x);
```

```
?>
```

BOOLEAN PHP

Un booléen représente deux états possibles : VRAI ou FAUX.

```
$x = true;
```

```
$y = false;
```

Les booléens sont souvent utilisés dans les tests conditionnels. Vous en apprendrez plus sur les tests conditionnels dans un chapitre ultérieur de ce cours.

ARRAY PHP

Un tableau stocke plusieurs valeurs dans une seule variable.

Dans l'exemple suivant, \$cars est un tableau.

Exemple:

```
<?php
```

```
$cars = array("Volvo","BMW","Toyota");
```

```
var_dump($cars);
```

```
?>
```

OBJECT PHP

Les classes et les objets sont les deux principaux aspects de la programmation orientée objet.

Une classe est un modèle pour les objets et un objet est une instance d'une classe.

NULL PHP

Null est un type de données spécial qui ne peut avoir qu'une seule valeur : NULL.

Une variable de type de données NULL est une variable à laquelle aucune valeur n'est affectée.

Conseil : Si une variable est créée sans valeur, la valeur NULL lui est automatiquement affectée.

Constants PHP

Pour créer une constante, utilisez la fonction `define()`. Syntaxe :

```
define(name, value, case-insensitive)
```

Paramètres:

- **name** : Spécifie le nom de la constante
- **value** : Spécifie la valeur de la constante
- **insensible à la casse** : spécifie si le nom de la constante doit être insensible à la casse.
- La valeur par défaut est `false`

Exemple :

```
<?php
```

```
define("MESSAGE", "BONJOUR");
```

```
echo MESSAGE; // BONJOUR
```

```
?>
```

Opérateurs PHP

Les opérateurs sont utilisés pour effectuer des opérations sur des variables et des valeurs.

PHP divise les opérateurs dans les groupes suivants :

- Opérateurs arithmétiques
- Opérateurs d'affectation
- Opérateurs de comparaison

- Opérateurs d'incrémentation/décrémentation
- Opérateurs logiques
- Opérateurs de chaîne
- Opérateurs de tableau
- Opérateurs d'affectation conditionnelle

Opérateurs arithmétiques PHP

Addition $x + y$

Soustraction $x - y$

Multiplication $x * y$

Division x / y

Module $x \% y$

Exponentiation $x ** y$

Opérateurs d'affectation PHP

$x = y$

Pareil

$x = y$

$x -= y$

Pareil

$x = x - y$

$x += y$

Pareil

$x = x + y$

$x *= y$

Pareil

$x = x * y$

$x /= y$

Pareil

 $x = x / y$
 $x \% = y$

Pareil

 $x = x \% y$

Opérateurs de comparaison PHP

== EGALE $\$x == \y renvoie true if \$x is equal to \$y

=== Identique $\$x === \y renvoie true (meme type meme valeur)

!= ou <> Différent de $\$x != \y Renvoie vrai si \$x n'est pas égal à \$y

!= Différent de $\$x != \y Renvoie vrai si \$x n'est pas égal à \$y

< Moins de $\$x < \y Renvoie vrai si \$x est inférieur à \$y

<= Inférieur ou égal à $\$x <= \y Renvoie vrai si \$x est inférieur ou égal à \$y

Opérateurs d'incrément PHP

++\$x Pré-incrémentation Incrémente \$x de un, puis renvoie \$x

\$x++ Post-incrément Renvoie \$x, puis incrémente \$x de un

--\$x Pré-décrémentation décrémente \$x de un, puis renvoie \$x

\$x++ Post-décrément Renvoie \$x, puis décrémente \$x de un

Opérateurs logiques PHP

AND/ & \$ x AND \$ y retourne Vrai si \$ x et \$ y sont vrais

OR / || \$ x OR \$ y retourne Vrai si \$ x ou \$ y est vrai

! NOT !\$x retourne Vrai si \$x n'est pas vrai

Opérateurs de chaîne PHP

. Concaténation \$txt1 . \$txt2 Concaténation de \$txt1 et \$txt2

.= Affectation de concaténation \$txt1 .= \$txt2 Ajoute \$txt2 à \$txt1

Instructions PHP if...else... elseif

En PHP, nous avons les instructions conditionnelles suivantes :

if instruction - exécute du code si une condition est vraie

if...else instruction - exécute du code si une condition est vraie et un autre code si cette condition est fausse

if...elseif...else déclaration - exécute des codes différents pour plus de deux conditions

switch instruction - sélectionne l'un des nombreux blocs de code à exécuter

Instructions PHP if

IF instruction exécute du code si une condition est vraie.

Syntaxe:

```
if (condition) {
```

```
code à execute si condition est vrai ;  
}
```

Instructions PHP if... else

if...else instruction exécute du code si une condition est vraie et un autre code si cette condition est fausse.

Exemple:

```
<?php  
$t = date("H");  
if ($t < "20"){  
    echo "BONJOUR!";  
} else {  
    echo "BONSOIR!";  
}  
?>
```

PHP - L'instruction if...elseif...else

if...elseif...else instruction exécute des codes différents pour plus de deux conditions.

Exemple:

```
<?php  
$t = date("H");  
  
if ($t < "10"){  
    echo "Have a good morning!";  
} elseif ($t < "20"){  
    echo "Have a good day!";  
} else {  
    echo "Have a good night!";  
}
```



```
}  
?>
```

PHP - L'instruction SWITCH

SWITCH instruction est utilisée pour effectuer différentes actions en fonction de différentes conditions.

Exemple:

```
<?php  
$favcolor = "rouge";  
  
SWITCH ($favcolor) { case "rouge":  
echo "Your favorite color is red!"; break;  
case "blue":  
echo "Your favorite color is blue!"; break;  
default:  
echo "Your favorite color is neither red, blue, nor green!";  
}  
?>
```