Cours : Introduction au langage PHP

---

1. Qu'est-ce que PHP et à quoi sert-il ?

PHP (Hypertext Preprocessor) est un **langage de programmation côté serveur** utilisé principalement pour créer des **sites web dynamiques**.

Contrairement au HTML, qui affiche du contenu statique, PHP permet de **générer du contenu en fonction de données, de conditions ou d'interactions**.

PHP s'exécute sur le **serveur web** (Apache, Nginx, etc.) avant d'envoyer le résultat au navigateur.

C'est un langage **interprété**, **libre**, et **largement utilisé** pour le développement web.

---

2. Qu'est-ce qu'un langage de programmation ?

Un langage de programmation est un **ensemble de règles et d'instructions** permettant de **communiquer avec un ordinateur**.

Il permet d'**écrire des algorithmes** pour traiter des données, exécuter des calculs ou automatiser des tâches.

PHP appartient à la famille des **langages impératifs et procéduraux**, et peut également être utilisé de manière **orientée objet**.

---

3. Premier exemple PHP : "Hello World !"

L'exemple classique pour commencer en PHP affiche un simple message à l'écran :

```
<?php
echo "Hello World!";
?>
```

Lorsque ce code est exécuté sur un serveur web compatible (ex. XAMPP), il produit :

Hello World!

---

4. Qu'est-ce qu'une instruction ?

Une **instruction** est une **action complète** exécutée par le programme.

Chaque instruction se termine **obligatoirement par un point-virgule (;)** en PHP.

Exemple :

```
<?php
echo "Bonjour";
echo "Au revoir";
?>
```

Chaque ligne d'affichage est une instruction distincte.

5. Comment écrit-on du code PHP ?

Le code PHP est écrit **entre les balises suivantes** :

```
<?php
// Code PHP ici
?>
```

Tout ce qui est à l'extérieur de ces balises est considéré comme **du HTML**.

Exemple :

```
<html>
<body>
<?php
echo "Ceci est du code PHP";
?>
</body>
</html>
```

---

6. Plusieurs utilisations de echo

echo est une **instruction d'affichage**.

Elle peut afficher du texte, des nombres, des variables ou du code HTML.

Exemples :

```
echo "Bonjour";
echo 42;
echo "<h1>Titre principal</h1>";
```

Il est possible d'afficher plusieurs valeurs à la suite :

```
echo "Valeur : ", 10 + 5;
```

---

7. Les commentaires

Les **commentaires** servent à expliquer le code sans être exécutés.

- Commentaire sur une ligne :
- // Ceci est un commentaire
- Commentaire sur plusieurs lignes :
- /*
- Ceci est un commentaire
- sur plusieurs lignes.
- */

---

8. Les opérations arithmétiques

PHP gère les opérations de base :

Opérateur	Fonction	Exemple	Résultat
+	Addition	5 + 3	8
-	Soustraction	8 - 2	6
*	Multiplication	4 * 2	8
/	Division	10 / 2	5
%	Modulo (reste)	10 % 3	1

Exemple :

```
<?php
$a = 10;
$b = 4;
$resultat = $a + $b;
echo $resultat; // 14
?>
```

Exercices – Opérations arithmétiques

1. Calculer la somme, la différence, le produit et le quotient de deux nombres.
2. Afficher le reste de la division de 15 par 4.
3. Créer un petit calculateur qui affiche :
4. 10 + 5 = 15
5. 10 - 5 = 5
6. 10 * 5 = 50
7. 10 / 5 = 2

---

9. Les variables

Une **variable** stocke une valeur temporaire (nombre, texte, etc.).
En PHP, elle commence toujours par le symbole $.
Exemple :

```
<?php
$nom = "Alice";
$age = 20;
echo $nom;
echo $age;
?>
```

Les noms de variables sont sensibles à la casse ($Nom ≠ $nom).

Exercices – Variables

1. Créer trois variables : $nom, $age, $ville et les afficher dans une phrase.
2. Modifier la valeur d'une variable et afficher à nouveau le résultat.
3. Créer deux variables $x et $y et afficher la somme de leurs valeurs.

---

10. La structure if / if else

if permet d'exécuter un bloc d'instructions **si une condition est vraie**.
else définit le bloc à exécuter **si la condition est fausse**.
Exemple :

```
<?php
$age = 18;
if ($age >= 18) {
   echo "Majeur";
} else {
   echo "Mineur";
}
?>
```

Exercices – if / if else

1. Vérifier si un nombre est positif ou négatif.
2. Vérifier si une note est supérieure ou égale à 10.
3. Afficher un message selon la température :
    - < 0 : "Il gèle"
    - 0 à 20 : "Frais"
    - 20 : "Chaud"

---

11. La structure switch

switch permet de **tester plusieurs cas possibles** pour une même variable.
Exemple :

```
<?php
$jour = "lundi";

switch ($jour) {
   case "lundi":
      echo "Début de la semaine";
      break;
   case "vendredi":
      echo "Presque le week-end";
      break;
   default:
      echo "Jour ordinaire";
}
?>
```

Exercices – switch

1. Afficher un message selon le jour de la semaine.
2. Associer une note (A, B, C, D) à un commentaire (Excellent, Bien, Moyen, Faible).

---

12. La boucle for

La boucle for répète un bloc d'instructions un nombre déterminé de fois.

Syntaxe :

```
for ($i = 0; $i < 5; $i++) {
   echo $i;
}
```

Exercices – for

1. Afficher les nombres de 1 à 10.
2. Afficher la table de multiplication de 5.
3. Calculer la somme des nombres de 1 à 100.

---

13. La boucle while

La boucle while s'exécute **tant qu'une condition est vraie**.

Exemple :

```
<?php
$i = 0;
while ($i < 5) {
   echo $i;
   $i++;
}
?>
```

Exercices – while

1. Afficher les nombres pairs jusqu'à 20.
2. Créer un compteur de 10 à 0.
3. Afficher la somme des nombres jusqu'à 50.

---

14. Grand exercice de synthèse

Créer un script PHP complet qui :

1. Déclare deux variables $a et $b.
2. Calcule et affiche les quatre opérations arithmétiques.
3. Vérifie si $a est plus grand que $b et affiche un message approprié.
4. Utilise un switch pour afficher un message selon la valeur de $a.
5. Affiche les nombres de 1 à 10 avec une boucle for.
6. Calcule la somme des nombres pairs jusqu'à 20 avec une boucle while.

Exemple de structure attendue

```php
<?php
$a = 10;
$b = 5;

// Opérations
echo "Somme : " . ($a + $b) . "<br>";
echo "Différence : " . ($a - $b) . "<br>";
echo "Produit : " . ($a * $b) . "<br>";
echo "Quotient : " . ($a / $b) . "<br>";

// Condition
if ($a > $b) {
   echo "a est plus grand que b<br>";
} else {
   echo "a est plus petit ou égal à b<br>";
}

// Switch
switch ($a) {
   case 10:
      echo "Valeur égale à 10<br>";
      break;
   default:
      echo "Autre valeur<br>";
}

// Boucle for
for ($i = 1; $i <= 10; $i++) {
   echo $i . " ";
}

// Boucle while
$i = 0;
$somme = 0;
while ($i <= 20) {
   if ($i % 2 == 0) {
      $somme += $i;
   }
   $i++;
}
echo "<br>Somme des pairs : " . $somme;
?>
```