**ATTRIBUT STATIC**

Un attribut static en PHP change plusieurs aspects importants dans le comportement d'une propriété ou d'une méthode de classe :

1. **Portée de la propriété/méthode** : Une propriété ou méthode statique **appartient à la classe elle-même et non à une instance spécifique de cette classe.** Cela signifie qu'elle est partagée entre toutes les instances de la classe.

* **Accès sans instanciation** : Les propriétés et méthodes statiques peuvent être accédées directement via le nom de la classe, sans avoir à créer une instance de cette classe. On utilise l'opérateur de résolution de portée :: pour y accéder (ex: ClassName::$staticProperty ou ClassName::staticMethod()).
* **Absence du mot-clé** $this : À l'intérieur d'une méthode statique, le mot-clé $this fait référence à la classe elle-même et non à une instance. On ne peut donc pas accéder aux propriétés non statiques avec $this depuis une méthode statique.
* **Persistance des valeurs** : Les propriétés statiques conservent leur valeur entre les différentes instances et requêtes, contrairement aux propriétés non statiques qui sont réinitialisées à chaque nouvelle instance.
* **Utilisation typique** : Les propriétés et méthodes statiques sont souvent utilisées pour implémenter des fonctionnalités indépendantes de l'état d'un objet spécifique, comme des utilitaires, des constantes de classe, des méthodes d'usine, etc.

Utilisation d'une propriété statique :

php

class Compteur {

public static $count = 0; // Propriété statique

public function incrementer() {

self::$count++; // **Accès à la propriété statique via self::**

}

}

// **Accès à la propriété statique sans instanciation**

echo Compteur::$count; // Affiche 0

$obj1 = new Compteur();

$obj1->incrementer(); // Modifie la propriété statique partagée

$obj2 = new Compteur();

$obj2->incrementer();

echo Compteur::$count; // Affiche 2

Ici, la propriété $count est statique et partagée entre toutes les instances de Compteur. Chaque appel à incrementer() **modifie la même propriété statique, quelle que soit l'instance utilisée.** En résumé, l'attribut static en PHP permet de **définir des propriétés et méthodes appartenant à la classe plutôt qu'à une instance spécifique**, offrant un état partagé et un accès direct sans instanciation.

SELF

* self est un mot-clé qui représente la classe courante dans laquelle il est utilisé.
* Il est utilisé pour accéder aux propriétés et méthodes statiques de la classe.
* self ne fait pas référence à une instance spécifique de la classe, mais à la classe elle-même.