

Fiche de révision

Les Classes Statiques en PHP

1. Qu'est-ce que c'est ?

Une classe statique, dans le contexte de la programmation orientée objet, est une classe dont tous les membres (propriétés et méthodes) sont statiques : *vous n'avez pas besoin de créer une instance de la classe pour accéder à ses méthodes et propriétés*. Au lieu de cela, vous y accédez directement à l'aide du nom de la classe.

Voici quelques caractéristiques clés des classes statiques :

- **Pas d'instanciation** : Vous ne pouvez pas créer d'instance d'une classe statique avec l'opérateur « new » car le concept de classe statique est de fournir une sorte de "boîte à outils" de méthodes et de propriétés accessibles sans avoir besoin de créer un objet.
- **Accès aux membres** : Pour accéder aux méthodes et propriétés statiques, utilisez le nom de la classe suivi de l'opérateur de résolution de portée « :: ».
- **Utilisation** : Les classes statiques sont souvent utilisées pour regrouper des fonctions utilitaires ou des méthodes de service qui ne nécessitent pas de maintenir des états (pas de données spécifiques à une instance de classe).
- **État global** : Étant donné que les membres statiques ne sont pas liés à une instance spécifique, ils maintiennent un état global dans l'application. Cela signifie que si vous modifiez une propriété statique, le changement est reflété partout où cette propriété est utilisée.

Il est important de noter que l'utilisation excessive de classes statiques peut conduire à un code difficile à tester et à maintenir, car elles introduisent un état global et des dépendances cachées dans votre application.

Par conséquent, leur utilisation doit être équilibrée et réfléchie, en préférant souvent des instances de classe lorsque cela est possible pour maintenir un **couplage faible** et une **haute cohésion** dans votre code.

“Remember that there is no code faster than no code” - Taligent's Guide to Designing Programs

2. Comment faire ?

2.1. Définir la classe :

Commencez par définir votre classe en utilisant le mot-clé **class**, suivi du nom de la classe. Par convention, les noms de classe doivent commencer par une lettre majuscule.

```
<?php
class Utilitaire {
    // La classe Utilitaire contiendra des méthodes et des propriétés statiques
}
```

2.2. Ajouter une propriété statique

Les propriétés statiques sont accessibles sans instancier la classe. Déclarez une propriété statique en utilisant le mot-clé **static**.

```
<?php
class Utilitaire {
    public static $valeurStatique = "Je suis statique";
}
```

2.3. Ajouter une méthode statique

De même, les méthodes statiques peuvent être appelées sans créer une instance de la classe. Utilisez le mot-clé **static** pour déclarer ces méthodes.

```
<?php
class Utilitaire {
    public static $valeurStatique = "Je suis statique";

    public static function afficherValeur() {
        echo self::$valeurStatique;
    }
}
```

Utilisez **self** pour accéder aux propriétés ou aux méthodes statiques à l'intérieur de la classe.

2.4. Accéder aux éléments statiques

Pour accéder aux propriétés ou méthodes statiques, utilisez le nom de la classe suivi de l'opérateur de résolution de portée ::.

```
echo Utilitaire::$valeurStatique; // Affiche "Je suis statique"
Utilitaire::afficherValeur(); // Exécute la méthode afficherValeur
```

3. Bonnes Pratiques

- Utilisez les classes statiques avec modération. Elles peuvent rendre le code difficile à tester et à maintenir en raison de leur état global.
- Préférez les méthodes statiques lorsque la méthode en question n'a pas besoin d'accéder aux propriétés d'une instance de la classe.