

L'héritage en PHP (POO)

L'héritage est l'un des concepts fondamentaux de la programmation orientée objet (POO). Il permet de créer des relations entre les classes, ce qui favorise la réutilisation du code et permet de modéliser les relations "est un" entre les objets. En PHP, comme dans de nombreux autres langages de programmation orientée objet, l'héritage est implémenté à l'aide de la notion de classes parentes et de classes enfants.

Une classe parente (ou classe de base) est une classe qui définit des propriétés et des méthodes communes à plusieurs classes. Les classes enfants (ou sous-classes) héritent de ces propriétés et méthodes et peuvent les utiliser tels quels ou les modifier pour les adapter à leurs besoins spécifiques.

L'héritage permet de créer une hiérarchie de classes, où les classes enfants héritent des caractéristiques des classes parents tout en ajoutant leurs propres fonctionnalités supplémentaires. Cela favorise la modularité, la maintenabilité et la réutilisabilité du code.

Dans PHP, pour déclarer qu'une classe hérite d'une autre classe, on utilise le mot-clé "extends". Les propriétés et méthodes de la classe parente sont alors accessibles dans la classe enfant via le mot-clé "parent".

Supposons que nous ayons une classe parente appelée "Animal" et deux classes enfants appelées "Chien" et "Chat". La classe "Animal" contient des propriétés et des méthodes communes à tous les animaux, tandis que les classes "Chien" et "Chat" héritent de la classe "Animal" et peuvent avoir leurs propres propriétés et méthodes spécifiques.

```
<?php
// Classe parente Animal
class Animal {
    protected $nom;
    protected $age;

    // Constructeur
    public function __construct($nom, $age) {
        $this->nom = $nom;
        $this->age = $age;
    }

    // Méthode pour afficher les informations de l'animal
    public function afficherInfos() {
        echo "Nom : " . $this->nom . ", Age : " . $this->age . " ans";
    }
}

// Classe enfant Chien
class Chien extends Animal {
    private $race;

    // Constructeur
    public function __construct($nom, $age, $race) {
        parent::__construct($nom, $age);
        $this->race = $race;
    }

    // Méthode pour afficher les informations spécifiques du chien
    public function afficherInfos() {
        parent::afficherInfos();
        echo ", Race : " . $this->race;
```

L'héritage en PHP (POO)

```
}  
}  
  
// Classe enfant Chat  
class Chat extends Animal {  
    private $couleur;  
  
    // Constructeur  
    public function __construct($nom, $age, $couleur) {  
        parent::__construct($nom, $age);  
        $this->couleur = $couleur;  
    }  
  
    // Méthode pour afficher les informations spécifiques du chat  
    public function afficherInfos() {  
        parent::afficherInfos();  
        echo ", Couleur : " . $this->couleur;  
    }  
}  
  
// Création d'objets Chien et Chat  
$chien = new Chien("Rex", 3, "Labrador");  
$chat = new Chat("Minou", 5, "Gris");  
  
// Affichage des informations  
echo "Informations sur le chien : ";  
$chien->afficherInfos();  
echo "<br>";  
  
echo "Informations sur le chat : ";  
$chat->afficherInfos();  
?>
```

Dans cet exemple, la classe "Animal" est la classe parente avec les propriétés "nom" et "age" ainsi que la méthode "afficherInfos()". Les classes "Chien" et "Chat" sont les classes enfants qui héritent de la classe "Animal" et ajoutent leurs propres propriétés spécifiques ("race" pour le chien et "couleur" pour le chat) ainsi que leurs propres méthodes "afficherInfos()" qui affichent les informations spécifiques de chaque animal.