



Programmation orientée objet en PHP

[Accueil](#) ► [Mes cours](#) ► [Développement logiciel](#) ► [POO PHP](#) ► La programmation orientée objet : premiers pas ►
[Les méthodes magiques](#)

Les méthodes magiques

PHP propose un ensemble de méthodes prédéfinies dont le nom commence par un double underscore. Ces méthodes sont appelées des **méthodes magiques**. Nous avons déjà rencontré trois de ces méthodes magiques : `__construct()`, `__get()` et `__set()`. Même si ces méthodes sont prédéfinies par PHP et possèdent un comportement par défaut, il est possible, au sein de chaque classe, de redéfinir le comportement d'une ou plusieurs méthodes magiques. Par exemple dans la classe Humain nous avons redéfini les méthodes `__construct()`, `__get()` et `__set()`.

Nous pouvons aussi noter que les fonctions magiques sont rarement appelées directement, en général elles sont appelées "en sous main" par PHP. Par exemple la méthode `__construct()` est appelée par PHP quand nousinstancions une classe, la méthode `__get()` est appelée par PHP quand nous accédons directement à un attribut, la méthode `__set()` est appelée par PHP quand nous modifions directement la valeur d'un attribut.

Il existe d'autres méthodes magiques comme par exemple la méthode `__toString()` qui renvoie une chaîne de caractères représentant l'objet. Cette méthode est appelée lorsque l'on tente d'afficher un objet.

```
1 <?php
2
3 // Humain.php
4 class Humain {
5
6     private $age;
7     private $taille;
8     private $poids;
9
10    public function __construct($page, $ptaille, $ppoids) {
11        $this->age = $page;
12        $this->taille = $ptaille;
13        $this->poids = $ppoids;
14    }
15
16    public function __get($name) {
17        return $this->$name;
18    }
19
20    public function __set($name, $value) {
21        $this->$name = $value;
22    }
23
24    public function __toString() {
25        return "age : $this->age, taille : $this->taille, poids : $this->poids";
26    }
27
28 }
29
30 $sparrow = new Humain(42, 1.84, 75);
31 echo $sparrow; // appel en sous main de __toString()
```

Fin

NAVIGATION

[Accueil](#)



[Ma page](#)[Pages du site](#)[Mon profil](#)[Cours actuel](#)[POO PHP](#)[Participants](#)[Généralités](#)[La programmation orientée objet : premiers pas](#) [Introduction](#) [Classes et objets](#) [T.P. première classe](#) [Les classes en PHP](#) [T.P. Personne](#) [Le constructeur et this](#) [T.P. Personne v2](#) [Accesseurs et mutateurs](#) [Les méthodes magiques](#) [T.P. Personne v3](#) [Les constantes de classe](#) [Le modificateur static](#) [T.P. Personne v4](#) [Une Personne et une Adresse : la relation has-a](#) [Histoire de références](#) [T.P. formation](#) [T.P. formation v2](#) [Une Personne et plusieurs Adresse\(s\)](#) [T.P. formation v3](#) [Bricolage et dépendance](#)[L'héritage](#)[Les interfaces](#)[Le typage](#)[Les namespaces](#)[Les exceptions](#)[Les bases de données avec PDO](#)[Les tests avec PHPUnit](#)[Petite application version 2](#)[Petite application version 3](#)[Mes cours](#)**ADMINISTRATION**[Administration du cours](#)[Réglages de mon profil](#)Connecté sous le nom « [Arnaud Lemais](#) » ([Déconnexion](#))[POO PHP](#)