

---

## PHPC2 : Maîtrise de la programmation orientée Objet avec PHP et Les designs Patterns

### Informations

---

Durée : 2 jours soit 14 heures  
Code : PHPC2

Niveau : Intermédiaire  
Public : Développeurs-euses

### Prérequis

---

Maîtrise du langage PHP

### Conditions matérielles

---

PHP 7+ installé et fonctionnel

### Evaluation des compétences

---

L'évaluation des compétences sera effectuée tout au long de la formation par des cas pratiques qui permettent de valider les blocs de compétence. Une évaluation finale est effectuée en fin de stage.

### Moyens pédagogiques

---

- Slides de présentation en petit comité (max 8 personnes)
- Ateliers pratiques
- Remise des supports et des exercices corrigés

### Objectifs

---

Cette formation vous permettra :

- d'apprendre tous les concepts essentiels de la programmation orientée objet en PHP afin de devenir opérationnel sur des frameworks tels que Symfony, Silex, Laravel, Drupal ou Zend,
- d'apprendre à développer en orienté objet avec les toutes dernières versions de PHP et notamment PHP 7.



## Introduction à la programmation orientée objet

- Introduction aux concepts de POO
- Avantages de la POO
- Grands principes de la POO
- Apprendre à réfléchir en mode objet
- Nouveautés fonctionnalités majeures depuis PHP 5.3

## Fondamentaux de la programmation orientée objet

- Comprendre les notions de classes et d'objets
- Définir l'état d'un objet avec les propriétés
- Lire et modifier l'état d'un objet grâce aux méthodes
- Instancier et initialiser un objet avec son constructeur
- Détruire proprement un objet grâce à son destructeur
- Assurer l'encapsulation grâce aux portées des attributs et des méthodes
- Comprendre la différence la portée d'instance et la portée de classe
- Spécialiser des classes grâce au mécanisme d'héritage
- Tirer profit des classes pour définir de nouveaux types de données

## Concepts avancés de la programmation orienté objet

- Généraliser des types grâce aux classes et méthodes abstraites
- Empêcher la spécialisation de classes ou la redéfinition des méthodes
- Redéfinir et surcharger le comportement d'une méthode parente
- Gérer proprement les erreurs avec le mécanisme des exceptions
- Concevoir des contrats d'implémentation grâce aux interfaces
- Surcharger des comportements implicites d'objets avec les « méthodes magiques »
- Mettre en place des mécanismes de chargement automatique des classes
- Organiser les structures de données dans des espaces de nommage (« namespaces »)
- Factoriser des comportements réutilisables grâce au mécanisme des « traits »

## Tirer profit de la SPL de PHP 7

- Découvrir la « Standard PHP Library »
- Lever des exceptions spécifiques de la SPL
- Utiliser les nouvelles structures de données de la SPL
- Créer des itérateurs d'objets avec l'interface « Iterator »
- Dénombrer les objets d'une collection avec l'interface « Countable »
- Déguiser un objet en tableau PHP avec l'interface « ArrayAccess »

## Mettre en pratique les outils orientés objet de PHP 7

- Manipuler des bases de données relationnels avec les objets PDO
- Parcourir les données d'un fichier XML avec les objets SimpleXML
- Manipuler et formater des dates avec les objets DateTime
- Accéder aux informations des fichiers grâce aux objets SplFileInfo

## Acquérir les bonnes pratiques de développement objet

- Savoir commenter et annoter du code avec la « phpdoc »
- Maîtriser les grands principes de SOLID
- Programmer par rapport à des interfaces
- Favoriser la composition plutôt que l'héritage
- Découvrir quelques grands patrons de conception
- Introduction aux principes de tests unitaires