

## Formation PHP Intermédiaire : Programmation Orientée Objet

<b>Durée :</b>	3 jours
<b>Public :</b>	Développeur PHP ou Développeur de langages modernes (Java, C#) ayant une bonne connaissance de la syntaxe de PHP
<b>Pré-requis :</b>	Bonnes connaissances de PHP
<b>Objectifs :</b>	Maîtriser les fondamentaux de la programmation orientée objet en PHP
<b>Sanction :</b>	Attestation de fin de stage mentionnant le résultat des acquis
<b>Taux de retour à l'emploi:</b>	Aucune donnée disponible
<b>Référence:</b>	PHP1085-F
<b>Note de satisfaction des participants:</b>	5,00 / 5

### Introduction

Rappels de PHP  
Appréhender l'orienté objet, lier conception et programmation  
Comprendre l'apparition progressive de l'objet dans PHP au fil des versions  
Structurer les traitements au travers d'objets

### Créer des scénarios web via une organisation d'objets

### Syntaxe fondamentale de l'objet en PHP

Définition : objet, instance et classe  
Grouper des traitements : les classes et méthodes  
Passage de paramètres et retour : le «type hinting»  
Partager des données entre méthodes : les attributs, notion d'encapsulation  
Instancier et faire appel à des objets  
La visibilité des membres : méthodes, des attributs et constantes  
Accès statiques

### Implémenter les scénarios via des classes et les tester

### Travailler avec plusieurs objets

Association : principe et notion de dépendance  
Manipuler un objet via un autre : l'agrégation  
Contrôler le cycle de vie d'un objet via la composition  
Les espaces de nommages : namespaces  
Charger automatiquement des définition : autoload

### Lier les objets entre eux, définir des niveaux d'abstraction

## **Gestion du typage**

Passer un contrat via la déclaration d'interface  
Étendre les comportement d'un objet par héritage  
Factoriser des comportements au travers d'une classe parent  
Factoriser des comportements au travers d'un trait  
Bloquer l'instanciation grâce aux classes abstraites  
Comprendre les notions de polymorphisme et de généricité

## **Interactions entre classes, gestion de la cardinalité**

## **Composer**

PHP en ligne de commande CLI  
Installation de composer  
Déclaration du projet  
Déclaration de dépendances, contraintes de version  
Dépendances relatives à la plate-forme  
Mise à jour et installation des dépendances  
Auto-chargement des définitions (PSR-0, PSR-4, classmap, files)  
Débogage des incompatibilités (why, why-not)

## **Déclarer son projet avec composer, importer une bibliothèque**

## **Passage de la certification (si prévue dans le financement)**