

# PHP, maîtriser la Programmation Orientée Objet

## De la conception à l'implémentation des classes pour les versions PHP 8, 7 et 5

Cours Pratique de 3 jours - 21h

Réf : POG - Prix 2024 : 2 070€ HT

Vous découvrirez la Programmation Orientée Objet en PHP et vous apprendrez à concevoir une application "Objet" en vous basant sur la notation UML. Vous y réaliserez un site Web en mettant en œuvre des concepts avancés tels que les objets métiers, les couches d'abstraction ou les design patterns.

### OBJECTIFS PÉDAGOGIQUES

À l'issue de la formation l'apprenant sera en mesure de :

Spécifier les exigences en fonction des cas d'usage

Maîtriser les concepts de base de l'implémentation Objet

Gérer la persistance des objets

Mettre en œuvre des Design Patterns

### MÉTHODES PÉDAGOGIQUES

Pédagogie active basée sur des échanges, des exemples, des exercices pratiques et une évaluation tout au long de la formation.

### TRAVAUX PRATIQUES

Apprendre à interpréter des diagrammes de classes UML et à concevoir, développer ou réécrire des applications PHP en respectant le paradigme objet.

## LE PROGRAMME

dernière mise à jour : 10/2023

### 1) Approche objet dans un contexte Web

- Expression initiale des besoins et exigences.
- Cas d'utilisation et processus unifié simplifié.
- UML : diagrammes de séquences et de classes.
- Présentation des IHM.

*Réflexion collective* : Sur la base d'un cahier des charges, définir les objets de l'application à réaliser.

### 2) Concepts de base de l'implémentation Objet

- Eclipse : un IDE adapté à la POO en PHP.
- Définir une classe et la générer à partir d'UML.
- Espaces de nommage PHP.
- Instancier une classe.
- Propriétés, statiques et constantes : This, Self et ORP.
- Visibilité et protection. Accesseurs, mutateurs et surcharge selon PHP.
- Constructeur et destructeur.
- Déclaration des propriétés dans le constructeur.
- Les attributs : métadonnées structurées.
- Maîtriser l'instanciation : Factory et Singleton.

*Travaux pratiques* : Depuis un logiciel UML, concevoir et générer les classes participantes.

### PARTICIPANTS

Développeurs désirant améliorer la pérennité et la maintenabilité de leurs développements.

### PRÉREQUIS

Bonnes connaissances du PHP procédural ou connaissances équivalentes à celles apportées par le cours "PHP, développer un site Web dynamique" (réf. PHH).

### COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

### MODALITÉS D'ÉVALUATION

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques...

Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

### MOYENS PÉDAGOGIQUES ET TECHNIQUES

- Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les stages pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.
- À l'issue de chaque stage ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.
- Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le stagiaire a bien assisté à la totalité de la session.

### MODALITÉS ET DÉLAIS D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

### ACCESSIBILITÉ AUX PERSONNES HANDICAPÉES

Vous avez un besoin spécifique d'accessibilité ? Contactez Mme FOSSE, référente handicap, à l'adresse suivante psh-accueil@orsys.fr pour étudier au mieux votre demande et sa faisabilité.

### 3) POO : concepts avancés et apports de PHP5

- Typage Objet et interfaces.
- Les interfaces de la librairie standard (SPL).
- Egalités et tris d'objets, Design Pattern Strategy.
- Référence, clonage, héritage, agrégation et association.
- Polymorphisme, collections et Design Pattern Iterator.
- Classes abstraites et finales. Les traits.

*Travaux pratiques* : Implémentation de classes mettant en œuvre les concepts avancés.

### 4) Persistance des objets

- Export et auto-sérialisation.
- Enregistrement en base de données et en session.
- JSON pour les requêtes client asynchrones.
- Transmission http.

*Travaux pratiques* : Classes pour la persistance des données. Accès asynchrone depuis une interface Ajax-XHR.

### 5) Refactoring MVC

- Chargement des classes : automatique ou paramétré.
- Classes ADO, modèle, vue. Gestion des fichiers.
- Les exceptions : gestion d'erreur orientée objet.

*Travaux pratiques* : Mise en œuvre du design pattern Chaîne de Responsabilité. Refactoring du projet en MVC.

### 6) Rétro-ingénierie

- Introspection et classe Reflection.
- Framework de librairies PEAR.
- Outils de documentation.
- XMI et diagrammes de classes.

*Travaux pratiques* : Générer les fichiers XMI pour l'ensemble des classes du projet et visualiser les diagrammes UML.

## LES DATES

---

Nous contacter