

## Formation Design Patterns

■ <b>Durée :</b>	3 jours (21 heures)
■ <b>Tarifs inter-entreprise :</b>	1 745,00 CHF HT (standard) 1 396,00 CHF HT (remisé)
■ <b>Public :</b>	Tous
■ <b>Pré-requis :</b>	Connaissance de la programmation Objet
■ <b>Objectifs :</b>	Découvrir les Design Patterns - Concevoir des modèles de structures de données
■ <b>Modalités pédagogiques, techniques et d'encadrement :</b>	<ul style="list-style-type: none"><li>• Formation synchrone en présentiel et distanciel.</li><li>• Méthodologie basée sur l'Active Learning : 75 % de pratique minimum.</li><li>• Un PC par participant en présentiel, possibilité de mettre à disposition en bureau à distance un PC et l'environnement adéquat.</li><li>• Un formateur expert.</li></ul>
■ <b>Modalités d'évaluation :</b>	<ul style="list-style-type: none"><li>• Définition des besoins et attentes des apprenants en amont de la formation.</li><li>• Auto-positionnement à l'entrée et la sortie de la formation.</li><li>• Suivi continu par les formateurs durant les ateliers pratiques.</li><li>• Évaluation à chaud de l'adéquation au besoin professionnel des apprenants le dernier jour de formation.</li></ul>
■ <b>Sanction :</b>	Attestation de fin de formation mentionnant le résultat des acquis
■ <b>Référence :</b>	PRO67-F
■ <b>Note de satisfaction des participants:</b>	4,39 / 5
■ <b>Contacts :</b>	commercial@dawan.fr - 09 72 37 73 73
■ <b>Modalités d'accès :</b>	Possibilité de faire un devis en ligne (www.dawan.fr, moncompteformation.gouv.fr, maformation.fr, etc.) ou en appelant au standard.

■ <b>Délais d'accès :</b>	Variable selon le type de financement.
■ <b>Accessibilité :</b>	Si vous êtes en situation de handicap, nous sommes en mesure de vous accueillir, n'hésitez pas à nous contacter à <a href="mailto:referenthandicap@dawan.fr">referenthandicap@dawan.fr</a> , nous étudierons ensemble vos besoins

## **Introduction**

Rappels de programmation objet sur plusieurs langages  
Historique et ouvrages de référence  
Domaines d'application  
Comment appliquer les Design Patterns

## **La génération d'instances**

Factory et Abstract Factory pour la création sous condition  
Singleton et dérivé : maîtrise des ressources disponibles

## **Modèles de structures de données**

Le Composite, comment simplifier les listes  
Proxy et Adapter, les interfaces de l'accès aux méthodes  
La Facade : clarifier un composant

## **Maîtrise du comportement**

Strategy : l'usine à méthodes  
L'itérateur et ses implémentations existantes  
Observer : l'événementiel sans événements  
Chaîne de responsabilité et arbres de responsabilité  
Visiteur et accès : maîtrise de la collaboration  
Aperçu d'autres Design Patterns