

Formation Java Initiation : Programmer en objet

■ Durée :	5 jours (35 heures)
■ Tarifs inter-entreprise :	2 375,00 € HT (standard) 1 900,00 € HT (remisé)
■ Public :	Tous
■ Pré-requis :	Connaissances informatiques de base, connaissances algorithmiques élémentaires
■ Objectifs :	Grands débutants : découvrir la programmation orientée objet au travers du langage Java Réaliser une première application complète
■ Modalités pédagogiques, techniques et d'encadrement :	• Formation synchrone en présentiel et distanciel. • Méthodologie basée sur l'Active Learning : 75 % de pratique minimum. • Un PC par participant en présentiel, possibilité de mettre à disposition en bureau à distance un PC et l'environnement adéquat. • Un formateur expert.
■ Modalités d'évaluation :	• Définition des besoins et attentes des apprenants en amont de la formation. • Auto-positionnement à l'entrée et la sortie de la formation. • Suivi continu par les formateurs durant les ateliers pratiques. • Évaluation à chaud de l'adéquation au besoin professionnel des apprenants le dernier jour de formation.
■ Sanction :	Attestation de fin de formation mentionnant le résultat des acquis
■ Référence :	JAV100976-F
■ Note de satisfaction des participants:	4,45 / 5
■ Contacts :	commercial@dawan.fr - 09 72 37 73 73

■ **Modalités d'accès :**

Possibilité de faire un devis en ligne (www.dawan.fr, moncompteformation.gouv.fr, maformation.fr, etc.) ou en appelant au standard.

■ **Délais d'accès :**

Variable selon le type de financement.

■ **Accessibilité :**

Si vous êtes en situation de handicap, nous sommes en mesure de vous accueillir, n'hésitez pas à nous contacter à referenthandicap@dawan.fr, nous étudierons ensemble vos besoins

Découvrir la plateforme Java

Historique, versions

Editions Java : Java SE, Java EE, Java ME

Compilation et interprétation par la JVM (Java Virtual Machine)

Technologies/frameworks Java et positionnement

Environnement de développement

Empaquetage et déploiement d'une application Java

Atelier : Installation du JDK (Java Development Kit) et d'un IDE (Eclipse/NetBeans) - Structure d'un projet, buildPath,...

Découvrir l'environnement de développement

Configuration

Principales vues, perspectives

Raccourcis utiles

Utilisation du debugger

Utiliser l'IDE pour gagner en productivité : snippets, refactoring, génération automatique de code

Maîtriser les bases

Utilisation de variables, constantes, opérateurs

Types simples et types références

Transtypage explicite, implicite, Wrappers

Priorités d'exécution

Expression de conditions : if/else, switch, opérateur ternaire

Utilisation de boucles : for, while, do while

Manipulation de tableaux

Factorisation de codes avec méthodes

Surcharge, arguments variables, récursivité

Commenter et documenter du code

Atelier : Multiples exemples de manipulation de structures de contrôles et de fonctions

Apprendre l'objet

Définition de classes, POJO vs JavaBean

Déclaration des membres d'instance / de classe (static)

Constructeurs et instanciation

Cycle de vie d'un objet en mémoire

Diagramme de classes (UML)

Agrégation d'objets (association)

Encapsulation : getters et setters

Extension de classes (Héritage)

Comparaison d'objets

Classes abstraites

Interfaces et implémentation

Polymorphisme

Atelier : Modélisation de problèmes en objet

Gérer les erreurs

Définition, types d'exceptions

Capturer et traiter une exception (try/catch/finally)

Lever/Remonter une exception (throw/throws)

Création d'exceptions

Atelier : Gestion des exceptions susceptibles d'être déclenchées dans une application

Utiliser des collections

Présentation de l'API disponible, generics

Comparatif, choix d'un type de collection

Classes essentielles : ArrayList, HashMap,...

Parcours, opérations sur des collections et tris

Atelier : Manipulation de collections d'objets

Manipuler des fichiers

Flux binaires / caractères

Lecture et écriture de fichiers

Utilisation de buffers

Manipulation de chemins, répertoires, surveillance

Sérialisation d'objets : binaire, XML

Externalisation de configuration dans des .properties

Gestion des logs : `java.util.logging`, Log4j

Atelier : Implémentation d'exports et imports depuis des fichiers

Utiliser les fonctions de base de Git dans Eclipse

Notions sur le dépôt de données et des branches

Enjeux de la gestion de configuration

Vérifier l'état de la copie de travail : `status`

Indexer ses modifications : `add`

Valider ses modifications : `commit`

Tirer, pousser et supprimer une branche distante