

Formation Java Avancé : Programmation réactive

■ Durée :	2 jours (14 heures)
■ Tarifs inter-entreprise :	1 475,00 CHF HT (standard) 1 180,00 CHF HT (remisé)
■ Public :	Développeurs Java
■ Pré-requis :	Avoir suivi le stage "Java initiation+approfondissement" ou posséder les connaissances équivalentes - Notions en programmation fonctionnelle
■ Objectifs :	Comprendre l'intérêt de la programmation réactive - Connaître la spécification Reactive Stream et ses implémentations (Reactor, RxJava, Java 9 Flow)- Maîtriser la librairie Reactor- Traiter des problèmes de programmation concurrente - S'appuyer sur un modèle de communication asynchrone
■ Modalités pédagogiques, techniques et d'encadrement :	<ul style="list-style-type: none">• 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 :	<ul style="list-style-type: none">• 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 :	JAV101968-F
■ Note de satisfaction des participants:	5,00 / 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

Comprendre la programmation réactive

Programmation réactive : concepts, intérêt
Tour d'horizon de l'écosystème

Découvrir la spécification Reactive Stream

La programmation fonctionnelle
Utilisation des lambdas
Pattern Observer
Les opérateurs
Publisher et Subscriber
Flux d'événements asynchrone
Implémentation par Reactor (Flux/Mono) et RxJava

Atelier : Multiples exemples d'implémentations réactives

Maîtriser les concepts avancés de Reactive Stream

La souscription avec subscriber() Gestion de la contre-pression (backpressure) Gestion du multithreading et la concurrence avec Reactor Les Publishers de Reactor Eager vs lazy : comparatifs entre just(), defer() et ses dérivés push vs pull : stratégies de gestion de la contre-pression (backpressure) hot vs cold : cas d'usage avec les processors Comparatif avec RxJava Les opérateurs de Reactor Transformer et filtrer les événements Combiner plusieurs sources Écrire son propre opérateur Tests unitaires avec StepVerifier

Atelier : Mise en place de Reactor - Utilisation des interfaces publisher et subscriber - Utilisation des Schedulers - Gestion des erreurs et tests unitaires