

Déroulé de l'action

• Modalités

Session Inter/Intra
En présentiel/Classe virtuelle

• Horaires

9H00-12H30 / 13H30-17H00

• Méthode pédagogique

Alternance exposés théoriques
et exercices pratiques
(80% de pratique)

• Suivi et assistance

Support de cours adapté
au logiciel étudié et
au niveau suivi
Assistance téléphonique
gratuite et illimitée

• Modalité d'évaluation

Passage de la certification TOSA
en fin de formation
Attestation de stage
Emargement quotidien d'une
feuille de présence

• Accessibilité aux personnes handicapées

Pour tout besoin d'adaptation,
retrouver le contact de notre
référént handicap et les
modalités d'accueil sur la page :
[Infos pratiques/Situation de
Handicap](#)

JAVA - Les Fondamentaux

Objectif : Cette formation a pour objectif d'introduire les différents concepts orientés-objets en Java et de permettre une initiation progressive aux différentes librairies et Framework du langage

À l'issue de la formation, le stagiaire sera capable de :

- Maîtriser la syntaxe et les principales APIs du langage Java
- Pouvoir consolider la compréhension des concepts orientés objet
- Savoir programmer en Java en toute autonomie
- Être capable de maîtriser les différentes APIs du langage (collections, I/O, JDBC)
- Être en mesure de manipuler des données à l'aide de JDBC et introduction à la persistance transparente avec JPA
- Savoir appliquer le paradigme de la programmation fonctionnelle
- Pouvoir tester et déboguer une application Java
- Découvrir les dernières évolutions de Java
- Apprendre à maîtriser l'utilisation d'un IDE Java

Public : Développeur / Chef de projet / Chargé de développement d'applications informatiques

Prérequis : Disposer d'une expérience d'un langage de programmation (C, C++, VB...) / Connaître les principes de la programmation orientée objet / Connaissance des concepts de bases de données relationnelles et du langage SQL.

- 5 jours -

Les concepts objet

- Programmation objet, les réutilisables
- Principe de l'encapsulation. Attributs et méthodes
- Accesseurs
- Différence entre objet et classe. Instanciation
- Conventions de nommage

Présentation de l'écosystème Java

- Introduction à Java et à ses tiers techniques J2SE/J2E/J2ME et la JVM
- Présentation de l'architecture d'une application respectant le patron d'architecture MVC et positionnement de Java
- Présentation des composants techniques de Java (JDK) ainsi que des principales APIs
- Les outils Java : JavaDoc, Jar et co
- Panorama des principaux IDE : Eclipse, NetBeans et IntelliJ

L'IDE Eclipse

- Notions de workspace, vues et perspectives
- Présentation des différents wizard
- Extensions avec Eclipse Marketplace
- Les accès aux bases de données et outils serveurs dans Eclipse

Déroulé de l'action

- **Modalités**

Session Inter/Intra
En présentiel/Classe virtuelle

- **Horaires**

9H00-12H30 /13H30-17H00

- **Méthode pédagogique**

Alternance exposés théoriques
et exercices pratiques
(80% de pratique)

- **Suivi et assistance**

Support de cours adapté
au logiciel étudié et
au niveau suivi
Assistance téléphonique
gratuite et illimitée

- **Modalité d'évaluation**

Passage de la certification TOSA
en fin de formation
Attestation de stage
Emargement quotidien d'une
feuille de présence

- **Accessibilité aux personnes handicapées**

Pour tout besoin d'adaptation,
retrouver le contact de notre
réfèrent handicap et les
modalités d'accueil sur la page :
[Infos pratiques/Situation de Handicap](#)

JAVA - Les Fondamentaux

Les bases du langage : la syntaxe

- Déclaration de variables
- Les constantes et les types énumérés
- Les bonnes pratiques d'écriture et de nommage
- Les opérateurs
- Instructions de contrôle
- Boucles et itérations
- Les tableaux
- La classe String
- La classe Scanner pour débiter avec les entrées/sorties
- Notions de visibilité
- Variable de classe vs. variable d'instance
- Les blocs "static"

Les concepts orientés Objet en Java

- La classe et ses attributs ainsi que les bonnes pratiques de nommage
- Les accesseurs
- Le mot-clé this
- Méthodes (nombre d'arguments fixe et variable)
- Les constructeurs avec et sans paramètres, notion de surcharge
- Destructeur avec le garbage collector et la méthode finalize()
- Les références
- La méthode toString()
- La méthode equals() pour vérifier l'égalité des objets
- L'encapsulation, les imports de packages, les imports static
- Les classes Wrappers
- Unboxing et l'autoboxing
- Le widening et narrowing (avec les notions de upcasting et downcasting)
- Les génériques en Java
- L'API Date
- La classe Optional

Factorisation du code avec le concept d'héritage

- Factorisation et spécialisation avec les mécanismes d'héritage
- Bonnes pratiques et conseils méthodologiques pour identifier les hiérarchies d'héritage
- Les mots clés : extends, instanceof, super, @override, protected
- Redéfinition des méthodes dans les classes filles
- Notion de polymorphisme
- Les classes abstraites

Les interfaces

- La notion de contrat et vision client vs. vision développeur
- Implémentation de plusieurs interfaces par une classe
- Les méthodes default, private et static depuis Java 9
- Les mots-clés, l'héritage multiple entre plusieurs interfaces

Déroulé de l'action

• Modalités

Session Inter/Intra
En présentiel/Classe virtuelle

• Horaires

9H00-12H30 /13H30-17H00

• Méthode pédagogique

Alternance exposés théoriques
et exercices pratiques
(80% de pratique)

• Suivi et assistance

Support de cours adapté
au logiciel étudié et
au niveau suivi
Assistance téléphonique
gratuite et illimitée

• Modalité d'évaluation

Passage de la certification TOSA
en fin de formation
Attestation de stage
Emargement quotidien d'une
feuille de présence

• Accessibilité aux personnes handicapées

Pour tout besoin d'adaptation,
retrouver le contact de notre
référént handicap et les
modalités d'accueil sur la page :
[Infos pratiques/Situation de Handicap](#)

JAVA - Les Fondamentaux

Les collections en Java

- Présenter l'API des collections en Java
- Savoir choisir entre les différentes interfaces SET, LIST et MAP
- Pratiquer les classes d'implémentations les plus connues
- Redéfinition des méthodes equals() et hashCode()
- Les interfaces comparable et Comparator pour les ensembles triés
- Parcours des collections
- La classe Stream et les filtres

Gestion des erreurs avec les exceptions

- Qu'est-ce que qu'une exception ?
- Les différentes exceptions en Java
- Les blocs try/catch/finally
- Gestion locale ou centralisée
- Créer ses propres exceptions métier

Les entrées/sorties

- Les entrées/sorties de fichiers
- Gérer les différents formats de flux d'E/S
- La sérialisation

Accès aux SGBDR avec JDBC

- Présentation de la librairie JDBC
- Connexion à une base de données, exécution d'une requête et itération sur les résultats avec les interfaces Connection, Statement et ResultSet
- Requêtes paramétrées et requêtes stockées
- Introduction à l'API JPA : objectifs et petite démo

Améliorer la qualité d'une application grâce aux tests unitaires

- Créer des tests unitaires avec JUnit
- Créer des suites de tests avec JUnit
- Générer des rapports
- Introduction à la notion de couverture de code et exemples de frameworks
- Options avancées de la perspective Debug

Programmation fonctionnelle et Lambda Expressions

- Introduction à la programmation fonctionnelle
- Exemples de Lambda Expressions
- Référentiels et méthodes

Introduction aux nouveautés Java depuis la version 9

- Présenter les nouveautés les plus marquantes depuis la version 9
- Donner des exemples pratiques ainsi que des petites démos

Travaux pratiques

- **Pour chaque point du programme abordé :**
Présentation et démonstration par le formateur, mise en pratique sur les fonctionnalités abordées par le stagiaire avec l'appui du formateur et du groupe, feedbacks du formateur tout au long de l'activité.