

Formation Java, approfondissement

(Cours dédié aux actions collectives Atlas)

Objectifs de la formation Java approfondissement

Cette formation Java Approfondissement a pour objectif de consolider une formation de base en Java.

Elle peut servir notamment à rafraîchir les concepts vus lors d'une précédente formation Java suivie il y a plusieurs mois sans pratique ultérieure, ou consolider les connaissances de ceux ayant appris le langage par eux-mêmes.

Elle permet également aux participants d'envisager sereinement une préparation à la certification Oracle Certified Associate, Java SE 8 Programmer.

À qui s'adresse cette formation ?

Public :

Cette formation Java s'adresse aux développeurs qui souhaitent revisiter les concepts clés de Java, approfondir la syntaxe du langage et se préparer au passage de l'examen officiel de la certification Associate Java 8 Programmer (1ZO-808).

Prérequis :

Pour tirer pleinement profit de cette formation, il est conseillé d'avoir au préalable suivi la Formation Java, les fondamentaux ou de posséder des connaissances équivalentes.

Contenu du cours Java approfondissement

Introduction à UML

Présentation d'UML

Rôle du diagramme de classes

Représentation des relations entre classes : dépendance, généralisation, associations

Implémentation en Java des différentes relations

Travaux Pratiques :



ITgate

Training

Your Gateway to Excellence

Mise en œuvre des associations d'un vers un et un vers plusieurs
Mise en œuvre de la composition

Rappels sur la syntaxe de base

Création de tableaux
Anatomie d'une classe
Spécificateurs d'accès private et public
Création d'objets
Constructeurs, méthodes d'instance, méthodes static
Surcharge de méthodes
Passage d'arguments aux méthodes : par valeur vs par référence
Création d'énumérations

Travaux Pratiques :

Création de quelques classes avec constructeurs et méthodes surchargés
Manipulation d'instances de ces classes

Mise en œuvre de l'héritage et du polymorphisme

Héritage de classes et d'interfaces
Usage du mot-clé protected
Utiliser les mots-clés this et super pour l'accès aux objets et aux constructeurs
Distinction surcharge/redéfinition
Manipulation d'instances de sous-classes
Différencier le type de l'objet du type de sa référence
Conversions explicites de références
Utilisation de l'opérateur instanceof

Travaux Pratiques :

Parcours d'une hiérarchie d'objets sans, puis avec polymorphisme

Gestion des exceptions

Distinction exceptions contrôlées-non contrôlées
Mise en œuvre d'une gestion d'exceptions avec try-catch
Utilisation du finally
Mise en œuvre du try-with-resources
Propagation d'exceptions
Principe de la translation d'exceptions
Mise en œuvre d'exceptions personnalisées

Travaux Pratiques :



ITgate

Training

Your Gateway to Excellence

Mise en place d'une gestion d'exceptions avec try-catch dans un programme de lecture de fichiers

Mise en place d'une propagation d'exceptions dans un programme de lecture de fichiers

Utilisation des collections Java

Savoir choisir un type de collection parmi Set, List, Queue, et Map

Maîtrise de la généricité

Manipuler les principales classes de collections : ArrayList, HashSet, PriorityQueue, HashMap, Properties

Collections ordonnées vs triées

Comprendre le rôle des méthodes equals et hashCode

Savoir utiliser un itérateur

Connaître les principales méthodes de la classe Collections

Travaux Pratiques :

Mise en application des différentes collections en fonction de critères comme : facilité de recherche, conservation de l'ordre, temps d'accès à un élément

Gestion des dates et du temps

Manipulation des dates avec java.time.LocalDate, java.time.LocalDateTime

Manipulation du temps avec java.time.LocalTime

Calcul de durées avec java.time.Period

Formattage de dates avec java.time.format.DateTimeFormatter

Correspondances avec les anciennes classes java.util.Date et java.util.Calendar

Travaux Pratiques :

Calcul du nombre de jours avant une date anniversaire

Calcul de la date de règlement d'une facture

Calcul de l'heure locale d'arrivée d'un vol Paris-Sydney

Expressions Lambda

Comprendre les principes de la programmation fonctionnelle

Syntaxe des classes internes anonymes

Notion d'interface fonctionnelle

Anatomie d'une expression Lambda

Mise en œuvre de l'interface fonctionnelle Predicate

Travaux Pratiques :

Remplacement de classes internes anonymes par des expressions lambda

Mise en œuvre de l'interface fonctionnelle Predicate

Introduction à l'API Stream pour la manipulation des flux de données

Objectif de l'API Stream

Obtention et sortie d'un stream

Opérations intermédiaires et terminales

Mise en oeuvre des méthodes forEach, map, filter, findFirst

Travaux Pratiques :

Utilisation de quelques Stream séquentiels sur tableaux et collections