SETC Training Corporate Logo

Introduction à Java: familiarisation avec les objets (JINTROFR, 4 jours)

Description

Ce cours certifié SETC a pour but d'introduire les programmeurs à la programmation en Java. Le cours débute par la base de la programmation en présentant les principes modernes du développement orienté objet en utilisant le langage Java ainsi que sa bibliothèque de classes. Le cours couvre également les notions d'héritage et de polymorphisme. Les participants acquièrent une expérience pratique par la réalisation d'exercices impliquant la création de code Java en utilisant l'environnement de développement Eclipse. Une introduction détaillée des classes, des objets, des primitives et des énoncés de contrôle suit.

Tarifs

- Tarification: $3,750/person
- Rabais de 10% lorsque vous inscrivez 3 personnes.

Plan de cours

Qu'est-ce que Java ?

Code compatible binaire, compatible source et interprété

L'architecture d'une plate-forme interprétée

Comparaison entre .NET et Java

Forces et faiblesses de Java

Les spécifications de la machine virtuelle Java et l'indépendance de la plate-forme

Les origines de Java

Le compilateur Java

Travailler avec Eclipse

Déboguer avec Eclipse

Les composants de Java : Types de données, opérateurs arithmétiques et éléments de langage

Classes et paquets

Les types de données primitifs : byte, short, int, long, float, double, char et boolean.

Conversion des types de données primitifs

Variables : déclaration et instanciation

Variables : l'opérateur d'affectation

Variables : utilisation des objets et des constructeurs

Déclarer des constantes

Opérations arithmétiques : +, -, *, /, %

Opérateurs composés : +=, -=, *=, /=, %=

Préséance des opérateurs

Opérateur d'incrémentation et de décrémentation : formes postfixe et préfixe

Opérateurs arithmétiques et types de données

Casting de type

Création de fonctions

Passage de paramètres aux fonctions

Contrôle du flux : Opérateurs logiques et logique conditionnelle

Les éléments de la logique : De l'anglais à la programmation

Opérateurs logiques fondamentaux : ET, OU, NON

La précédence des opérateurs

Traduire de l'anglais à la logique formelle

Logique conditionnelle : l'instruction if : else : (si : sinon :)
Logique conditionnelle : l'instruction switch

Contrôle du flux : Répétition

Que sont les boucles ?
La boucle for
La boucle while
Utilisation de break et continue
Comparaison entre la boucle for et la boucle while

Chaînes de caractères, caractères et tableaux

Déclarer des caractères et des chaînes de caractères
Utilisation des opérateurs de base avec les chaînes de caractères
Les fonctions des chaînes de caractères
Jetons et expressions : Analyser et analyser les chaînes de caractères
Déclarer des tableaux : De une à plusieurs dimensions
Tableaux et boucles
Un exemple d'utilisation des tableaux

RUP et analyse orientée objet

Le processus unifié de Rational
De la conception à la transition
Analyse orientée objet
Rédaction de cas d'utilisation complets
Identifier les objets
Définition des associations
Ajout d'attributs
Rédaction d'un modèle de domaine

Conception orientée objet

Qu'est-ce que la conception ?
Syntaxe de base UML
Identifier les généralisations
Identification des agrégations et des compositions
Création d'un diagramme de classes de conception

Concepts et classes orientés objet

Qu'est-ce qu'une programmation orientée objet ?
Définition de concepts efficaces et réutilisables
Conception orientée objet : Une approche générale
L'objet en Java : la classe
Définir les variables membres
Encapsulation : La nécessité des variables membres privées
Création de propriétés : Définir les fonctions d'inspecteur
Création d'un objet simple
Création de méthodes
Définir la portée d'un membre : private, public et protected
Le cas particulier du constructeur
Définir les constructeurs
Explorer la gestion de la mémoire et les finaliseurs

Héritage et polymorphisme

Conception orientée objet dans un système complexe
Création et utilisation de packages
Un exemple de classes multiples
Héritage : la relation is-a

Utilisation de l'héritage pour implémenter un modèle objet
Polymorphisme
Définition des classes abstraites
Définition des classes finales
Utilisation des interfaces

Introduction à la programmation objet

À propos des types de référence et des conversions
Utilisation de Reflection
Gestion des exceptions
Travailler avec les génériques
Création de génériques
Introduction à la sérialisation
À propos des entrées-sorties de fichiers