

Programmation experte en Java: Design Patterns, ORM, MVC et autres sujets mystiques (JEXPFR, 4 jours)

Description

Ce cours certifié SETC est conçu pour amener les développeurs Java avancés vers un niveau supérieur. Les participants apprendront comment construire de meilleures applications Java par l'utilisation des meilleures pratiques et par une meilleure connaissance des modèles de conceptions (design patterns). Le cours permet une exploration de la bibliothèque de classes afin d'aider le développeur à passer du mode «écriture de code» vers un mode «conception de solutions». Le cours couvre également la réutilisation de code par l'utilisation de technologies puissantes telles que Hibernate et Struts qui permettent de réaliser plus rapidement des applications complexes. Après avoir complété ce cours, le développeur sera mieux équipé pour utiliser les technologies de développement Java les plus récentes et ainsi produire du code de très haute qualité avec moins d'erreurs.

Tarifs

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

Plan de cours

Construire de bons logiciels

Vue d'ensemble

Caractéristiques d'une bonne solution logicielle

Comment construire un bon logiciel

Développement itératif

Gestion des exigences

Utilisation d'architectures basées sur des composants

Vérification continue de la qualité du logiciel

Contrôle des modifications du logiciel

Introduction à l'UML

Passage de l'analyse à la conception

Notation des classes UML

Généralisation

Agrégations

Classes de collecte

Diagrammes de séquences

Diagrammes d'état

Diagrammes d'activité

Interfaces, modèles et patrons

Héritage et polymorphisme

Héritage

Héritage et constructeurs

Polymorphisme

Fonctions et classes abstraites

Utilisation des interfaces dans la programmation quotidienne

Programmation Java

Travailler avec les types de référence

Aperçu du traitement des exceptions

Traitement des exceptions

Événements

Utilisation des génériques

Création de génériques

Modèles de conception

Aperçu de la structure des classes cibles

À propos de la création d'objets

À propos de la structure des objets

Considérations de refactoring

Usine abstraite

Singleton

Examen du code client

Passerelle

Constructeur

Façade

Prototype

Adaptateur

Composite

Autres modèles

Considérations sur le remaniement

Technologies de développement avancées

À propos des frameworks d'entreprise

Cadres d'entités commerciales

Cadres de validation

Développement d'applications dans un cadre d'entreprise

Rendre un objet liable

Programmation basée sur la configuration

Accès aux données

Traitement des exceptions

Journalisation

Validation

Bases d'Hibernate

Vue d'ensemble

Le problème du Sql codé à la main

À propos de l'ORM avec Hibernate

Espaces de noms d'Hibernate

Interfaces Hibernate

L'ISessionFactory

Les instances de session

Identité

Configuration des applications Hibernate

Cartographie d'objets

Vue d'ensemble

Le modèle de domaine

Principes de base du mappage

Objets de mappage

Gestion de l'identité

Types de cartographie

Mapping : Types de valeurs et types d'entités

Cartographie des hiérarchies de classes

Cartographie des associations

Mapping des collections

Mapping et polymorphisme

Utilisation des frameworks basés sur Struts

Architecture Modèle 1 et Modèle 2

MVC pour le Web

Le rôle du contrôleur

Le fichier de configuration

Mise en œuvre de la vue

L'action