

 **DURÉE : 2 JOURS**

Référence : RJAV01

OBJECTIFS

Comprendre ce qu'est le refactoring de code et l'identifier dans du code legacy
Découvrir les différentes méthodes de refactoring et leurs cas d'applications

PUBLIC

Développeurs Java

PRÉ-REQUIS

Expérience du développement en Java
Connaître les notions de base de la programmation orientée objet
Connaître les notions de tests unitaires


MÉTHODES PÉDAGOGIQUES


40 % de théorie / 60 % de démonstrations et mise en pratique


MODALITÉS D'ÉVALUATION


Un questionnaire d'évaluation à chaud sera adressé aux participants à la fin de la formation. Cette évaluation mesure le niveau 1 de Kirkpatrick.

CONTACT

 89, quai Panhard et Levassor
75013 Paris

 +33 1 44 75 42 55

 +33 1 44 75 05 25

 training@soat.fr

DESCRIPTION

Le code propre c'est bien. Mais comment faire lorsque l'on travaille avec une application comportant un historique de plusieurs années.

Cette formation va vous permettre d'analyser votre code « Legacy » et vous montrer sur quels éléments vous appuyer afin de démarrer un cycle d'amélioration de votre qualité de code et la maintenabilité de votre application.

PROGRAMME

Introduction ?

- Qu'est-ce que le refactoring
- Pourquoi refactorer
- Quand refactorer

Legacy Code

- Bad Smells
- Code non testable

Méthodes de refactoring

- Clean the Deck
- Extract or Inline
- Move methods
- Substitution