

Caractéristiques

Durée : 3 jours

Modalité : présentiel

Niveau : intermédiaire

Type : professionnalisation

Taux de satisfaction : 4,3/5

Méthodes pédagogiques

- Pédagogie par objectif
- Alternance de pédagogie par exposé et de pédagogie participative sur cas pratiques professionnels
- Productions individuelles

Modalités d'évaluation

- Évaluation et régulations individualisées
- Possibilité de certification suivant cursus
- Questionnaire d'évaluation et certificat de stage

Moyens et supports

- 5 salles pouvant accueillir entre 6 à 12 stagiaires
- Postes de travail en réseau sous Windows 10 Pro ou ultérieur
- Kit de vidéo projection ou grand écran connecté
- Supports de cours et d'exercices papiers et/ou numériques
- Supports complémentaires possibles par e-learning
- Formateur professionnel de plus de dix ans d'expérience et/ou expert de Bac+5 à Bac +7

Tarif

- Pour connaître les tarifs de la formation en Inter, en Intra, en CPF (si éligible) et nos délais d'accès à cette formation, merci de nous contacter

Accessibilité aux personnes handicapées

- En cas de formation réalisée en présentiel les locaux sont garantis accessibles aux PMR. Pour d'autres handicaps, nous contacter pour la mise en place des moyens de compensation
- Contactez notre référent PSH (cfa@osipro.re)

Objectifs

- De mettre en œuvre des tests unitaires automatisés
- D'améliorer la stabilité d'une application grâce aux techniques de tests

Prérequis et public concerné

- Tout développeur Java ou tout autre développeur familier avec un langage objet
- Tout testeur désireux d'automatiser ses suites de tests
- Bonne connaissance d'un langage objet : Java, C# ou PHP

Programme

Résumé des compétences à acquérir

- Connaître les différents types de test en informatique
- Comprendre les enjeux des tests automatisés
- Concevoir un cas de test
- Utiliser un framework de test (JUnit)
- Utiliser un framework de build automatisé (Maven)
- Mesurer la couverture de code
- Utiliser les doubles de test

Méthodologie

- Organiser les familles de tests
- Définir les responsabilités
- Organiser le test unitaire
- Connaître l'historique
- Comprendre les enjeux

Technique du test

- Connaître le vocabulaire
- Définir l'unité
- Connaître le Framework xUnit
- Connaître les limites

Java / Junit

- Connaître les versions
- Connaître les conventions
- Mettre des annotations
- Faire des assertions
- Faire des suppositions
- Définir les règles
- Définir les ressources
- Modifier les paramètres
- Matcher
- Poursuivre les tests
- Intégrer avec Maven

Pratiques additionnelles

- Faire une couverture de code
- Faire un TDD ou « Test Driven Development »
- Faire un BDD ou « Behavior Driven Development »
- Faire les doubles de tests

Ouverture

- Connaître les bonnes pratiques
- Connaître les problèmes communs
- Trouver des solutions
- Aller vers d'autres types de test