



Caractéristiques

- Durée : 3 jours
- Niveau : débutant
- Type : professionnalisation

Méthodes pédagogiques

- Pédagogie par objectif
- Alternance de pédagogie par exposé et de pédagogie participative sur cas pratiques professionnels
- Productions individuelles
- Evaluation et régulations individualisées
- Possibilité de Certification suivant cursus

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 papier ou numérique
- Supports complémentaires possibles par e-learning
- Questionnaire d'évaluation et certificat de stage
- Formateur professionnel de plus de dix ans d'expérience et/ou expert de Bac+5 à Bac +7

Objectifs

A l'issue de ce stage, les participants seront capables :

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

Public concerné

- Tout développeur Java ou tout autre développeur familier avec les langages objets.
- Tout testeur désireux d'automatiser ses suites de tests.

Prérequis :

- bonne connaissance d'un langage objet (type Java, C#, Php)

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

- Familles de tests
- Responsabilités
- Test unitaire
- Historique
- Enjeux

Pratiques additionnelles

- Couverture de Code
- TDD ou « Test Driven Development »
- BDD ou « Behavior Driven Development »
- Les doubles de tests

Technique du test

- Vocabulaire
- Définir l'unité
- Frameworks xUnit
- Limites

Ouverture

- Bonnes pratiques
- Problèmes communs
- Solutions
- Vers d'autres types de test

Java / Junit

- Versions
- Conventions
- Les annotations
- Les assertions
- Les suppositions
- Les règles
- Les ressources
- Les paramètres
- Les matchers
- Suites de test
- Intégration avec Maven