

# Les types de test

**Tests unitaires** : Les tests unitaires visent à tester des parties individuelles (unités) du code, comme des classes, des méthodes ou des fonctions, de manière isolée. Les frameworks de test populaires pour les tests unitaires en Java sont JUnit et TestNG.

**Tests d'intégration** : Les tests d'intégration vérifient comment différentes parties du système interagissent les unes avec les autres. Cela peut inclure des tests de composants, des tests de bases de données, et d'autres tests pour s'assurer que les différentes parties du logiciel fonctionnent ensemble correctement.

**Tests fonctionnels** : Les tests fonctionnels examinent le logiciel du point de vue de l'utilisateur final. Ils vérifient que le logiciel fonctionne comme prévu, en se concentrant sur les fonctionnalités et les scénarios d'utilisation.

**Test doubles** : est un terme général utilisé dans le développement logiciel et les tests pour décrire des objets ou des composants utilisés comme substituts d'autres composants du système lors des tests. Les "test doubles" sont utilisés pour isoler le code en cours de test et créer des environnements de test contrôlés.

# JUnit

JUnit framework de test unitaire open source pour Java. Il offre un ensemble de bibliothèques et de conventions pour écrire des tests unitaires de manière structurée et automatisée.

Les tests unitaires sont utilisés pour vérifier le bon fonctionnement des unités de code, telles que les classes, les méthodes ou les fonctions, de manière isolée.

# Mockito

Mockito est un framework de création de test doubles (mocking) open source pour la programmation Java. Il permet aux développeurs de créer des objets fictifs (ou "mocks") qui simulent le comportement d'objets réels.

Ces mocks sont utilisés dans les tests unitaires pour isoler l'unité de code testée de ses dépendances externes, telles que d'autres classes, interfaces ou services