

Jour 1 – Job– 01 Test Unitaire

Welcome to Unit Tests – Part 1

Code first or test first ?

Vos Premiers Tests avec Jest

Bienvenue dans le monde des tests unitaires !

Aujourd'hui vous allez faire vos premiers pas dans les tests unitaires avec Jest.

L'objectif est de comprendre l'intérêt des tests et d'écrire vos premières fonctions testées.



Objectif :

Créer un petit projet JavaScript avec Node.js, y intégrer Jest, écrire une fonction simple et rédiger des tests unitaires pour vérifier son bon fonctionnement.

- **Faites des captures d'écran pour illustrer un readme.md**
- **Les images seront intégré au readme**
- **Stocké dans un dossier image du projet**
- **Puis déposé sur github à chaque étape**
- **Avec des commits régulier**
- **Avec un nom et une description explicite et claire**

(Pas de nom du genre "mon commit" ou "commit pierre", un nom pro c'est "modification de la connection à la BDD" par exemple, pareil pour les branches "Connection à la BDD" ou "formulaire de contact")



- 1. Initialiser un projet Node.js :
 - Créer un dossier nommé `test-unitaire-debutant`.
 - Ouvre un terminal dans ce dossier et
 - Initialiser un projet Node.js
 - Installer Jest
 - Configurer le dans package.json
 - Créer un fichier de fonction nommé math.js
 - CRéer une fonction addition qui retourne la somme de deux nombres qui retourne le résultat
 - Créer le fichier math_test.js dans le même dossier
 - créer un test à l'intérieur en utilisant la syntaxe test() et expect() fournie par jest
 - Ecrire un test unitaire
 - Lancer le test
 - Vérifier qu'il passe
 - Modifier volontairement le code pour le faire échouer
 - corriger l'erreur
 - Repasser le test il doit être bon



Compétences visées

- Installer et configurer Jest dans un projet Node.js
- Écrire des fonctions simples en JavaScript
- Écrire des tests unitaires
- Comprendre les erreurs et les corriger
- Documenter son travail dans un README

Rendu

- Faites des captures d'écran à chaque étape pour montrer la commande et le résultat même si c'est un résultat en erreur
- Rendre un "readme.md" avec ces captures d'écran à l'intérieur stockées sur votre github dans un dossier image en annotant chaque capture en expliquant le détail de chaque action et commande de manière structurée en Markdown dans un readme.md.
- partager votre projet github avec votre formateur
- Un dépôt GitHub nommé `test-unitaire-debutant` contenant :
 - Tous les fichiers du projet
 - Un fichier `README.md` avec captures d'écran et explication en Markdown
 - Des commits réguliers et nommés proprement



Base de connaissances

- <https://jestjs.io/docs/getting-started>
- <https://developer.mozilla.org/fr/docs/Web/JavaScript/Guide>
- <https://nodejs.org/>
- <https://github.com/>
- <https://www.markdownguide.org/basic-syntax/>