

Corrigé

P1C2 - Découvrez Vitest et les niveaux de test



Commençons par installer les dépendances nécessaires :

```
Shell
npm install -D vitest @vitejs/plugin-react jsdom
```

Nous ajoutons ensuite les scripts dans `package.json` :

```
JSON
{
  "scripts": {
    "test": "vitest",
    "test:run": "vitest run"
  }
}
```

Créons maintenant un fichier `add.ts` :

```
TypeScript
export function add(a: number, b: number) {
  return a + b;
}
```

Puis le fichier `hello.test.ts` :

TypeScript

```
import { describe, it, expect } from "vitest";  
  
import { add } from "./add";  
  
describe("add", () => {  
  it("should return the sum of two numbers", () => {  
    expect(add(1, 2)).toBe(3);  
  });  
});
```

Enfin, lançons les tests :

Shell

```
npm run test
```

Vitest détecte automatiquement le fichier `hello.test.ts`, exécute le scénario et affiche un résultat positif.

Vous obtenez alors un indicateur vert signalant que le test est réussi.

Même si ce test semble très simple, il valide plusieurs points essentiels :

- Vitest est correctement installé ;
- le projet reconnaît les fichiers de test ;
- les imports fonctionnent ;
- le moteur d'assertion est opérationnel.

Vous disposez désormais d'un environnement prêt à accueillir les véritables tests de l'application de gestion de tâches.

C'est exactement ce que souhaitait votre tech lead avant de commencer le développement des scénarios de validation métier.