

Brief 2 : Gestionnaire de Stock pour une Petite Entreprise

Objectif : Créer un gestionnaire de stock simple en Node.js capable de gérer les produits (ajout, affichage, mise à jour et suppression) en mettant en œuvre les concepts de base de JavaScript orienté objet et en introduisant les principes CRUD.

Instructions :

1. Initialisation du projet

- Configurer un projet Node.js avec `npm init`.
- Créer un fichier principal `inventory.js`.

2. Fonctionnalités

- Ajouter un produit au stock :
 - Les utilisateurs doivent entrer le nom, la description, la quantité et le prix du produit.
 - Afficher tous les produits en stock :
 - Affichez la liste des produits avec leurs propriétés (nom, description, quantité, prix total).
 - Mettre à jour un produit :
 - Permettez aux utilisateurs de modifier la quantité ou le prix d'un produit existant.
 - Supprimer un produit du stock :
 - Supprimez un produit basé sur son identifiant unique.
 - Persistance des données :
 - Les données doivent être stockées dans un fichier JSON pour être disponibles entre les exécutions.
-

3. Gestion des erreurs

- Validez les entrées utilisateur (noms non vides, quantités et prix comme nombres positifs).
 - Gérez les cas où un produit demandé pour la mise à jour ou la suppression n'existe pas.
-

4. Organisation du code

- Créer une classe **Product** pour modéliser un produit.
 - Créer une classe **Inventory** pour gérer les opérations sur les produits (ajout, affichage, mise à jour, suppression).
 - Implémentez les principes SOLID :
 - **Single Responsibility** : Une classe pour les produits, une autre pour gérer les opérations.
 - **Open/Closed** : Facilitez l'ajout de nouvelles fonctionnalités sans modifier les classes existantes.
-

Structure du Programme :

Menu principal :

Présentez les options disponibles :

1. Ajouter un produit.
2. Afficher tous les produits.
3. Mettre à jour un produit.
4. Supprimer un produit.
5. Quitter.

Méthodes spécifiques :

- **addProduct(name, description, quantity, price)** : Pour ajouter un produit.
- **listProducts()** : Pour afficher tous les produits.
- **updateProduct(id, quantity, price)** : Pour mettre à jour un produit existant.
- **deleteProduct(id)** : Pour supprimer un produit.

Boucle principale :

Le programme reste actif jusqu'à ce que l'utilisateur choisisse de quitter.

Livrables

1. **Code source** : Hébergé sur un dépôt GitHub, avec des commits clairs pour chaque fonctionnalité.
2. **Documentation** : Explication des fonctionnalités principales et instructions pour exécuter le programme.
3. **Gestion de projet avec Trello** :
 - Créer un tableau Trello pour organiser les tâches et suivre l'avancement du projet.
 - Ajouter les colonnes suivantes :

- À faire : Liste des tâches à accomplir.
 - En cours : Tâches en cours de réalisation.
 - En revue : Tâches terminées mais en attente de validation.
 - Terminé : Tâches finalisées.
-