**Exercice : Création d'une API RESTful avec Node.js et Express**

**Objectif :**

Créer une API qui gère une liste de tâches (to-do list). L'API doit permettre d'ajouter, de lire, de mettre à jour et de supprimer des tâches.

**Instructions :**

**Étape 1 : Configuration du projet Backend**

1. **Créer un nouveau dossier pour le projet**.
2. **Initialiser un projet Node.js** avec npm init -y.
3. **Installer les dépendances nécessaires** :
   * express
   * body-parser
   * cors

**Étape 2 : Créer le fichier principal**

1. **Créer un fichier app.js**.

**Étape 3 : Configurer Express**

1. **Configurer un serveur Express basique**.
2. **Créer les routes suivantes** :
   * GET /tasks : Récupérer toutes les tâches.
   * POST /tasks : Ajouter une nouvelle tâche.
   * PUT /tasks/:id : Mettre à jour une tâche existante.
   * DELETE /tasks/:id : Supprimer une tâche.

**Étape 4 : Stockage des données**

1. Utiliser un tableau en mémoire pour stocker les tâches. Chaque tâche doit avoir un id, un title, et un completed (boolean).
2. Ajouter quelques tâches par défaut lors du démarrage du serveur.

**Étape 5 : Tester l'API**

Utiliser un outil comme Postman ou curl pour tester les différentes routes de l'API.

**Étape 6 : Création du Frontend avec React.js**

1. **Créer un nouveau projet React** avec create-react-app.
2. **Créer les composants suivants** :
   * Un composant pour afficher la liste des tâches.
   * Un formulaire pour ajouter une nouvelle tâche.
   * Des boutons pour mettre à jour et supprimer des tâches.

**Étape 7 : Intégration avec l'API**

1. Utiliser fetch ou axios pour interagir avec l'API backend.
2. Afficher les tâches récupérées et permettre les opérations CRUD via l'interface.

**Résultat attendu**

1. Une API backend fonctionnelle pour gérer les tâches.
2. Une interface utilisateur en React.js pour interagir avec l'API et gérer les tâches.