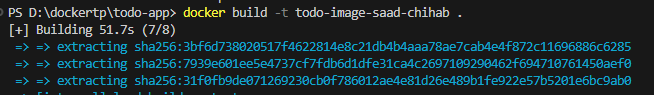
Docker

1. Créer un fork dans votre compte github et cloner localement le dépôt git suivant :

<https://github.com/najasoft/todo-app>

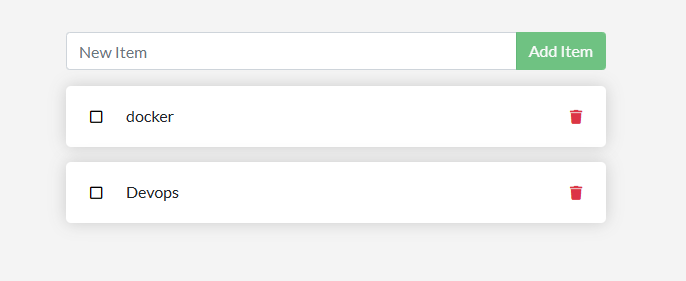
1. Créer un fichier Dockerfile dans le même dossier que le fichier package.json, et qui doit contenir les commandes suivantes pour créer une image pour l’application todo-app :
   * Image de base : node :12-apline (c’est une application nodejs)
   * Dossier de travail courant : /app
   * Copier le contenu répertoire local dans le dossier app de l’image : COPY . .
   * Exécuter la commande pour télécharger les dépendances du fichier package.json et installer l’application en mode production : yarn install –production
   * Exécuter l’application (node "/app/src/index.js") : CMD ["node","/app/src/index.js"]
2. Créer une image nommée todo-image-votre-nom (l’application doit être suffixée par votre nom, si le nom est composé de plusieurs mots alors il faut les séparer par un tiret : -)

**

1. Créer et exécuter un conteneur nommé todo-ctr-votre-nom, l’application utilise le port 3000 qui doit être mappé au port 3457, l’exécution doit être en mode détaché.

**

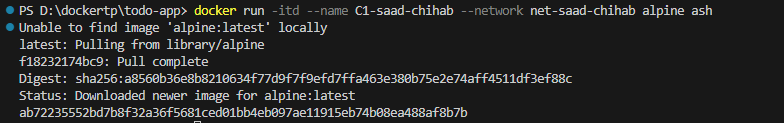
1. Tester l’application (localhost:3457) en ajoutant les deux tâches (ou items) suivantes : Docker , Devops

**

1. Créer un réseaux nommé net-votre-nom

**

1. Créer et exécuter un conteneur nommé C1-votre-nom , options : -itd, réseau : celui créé dans la question précédente, image : alpine , Almquist shell comme shell de démarrage : ash



1. Connecter le conteneur todo-ctr-votre-nom au réseau créé à la question 6.



1. Attacher le conteneur C1-votre-nom , et afficher l’application todo-app (la réponse http) :
   * Commande docker attach
   * Dans le conteneur C1, installer curl : apk add curl
   * curl lien\_todo\_app
   * exit

