

TP1 : Installation et prise en main Docker

Environnement de test Docker

Installation de Docker

- Téléchargez et installez Docker Desktop selon votre système d'exploitation :
<https://www.docker.com/products/docker-desktop>
- Vérifiez l'installation en exécutant la commande suivante dans votre terminal :

`docker version`

Création de votre première image Docker

Récupérer le ZIP app.zip

Créez un fichier nommé Dockerfile à la racine de votre projet avec le contenu suivant pour héberger l'application suivante.

```
FROM node:14-alpine
```

```
WORKDIR /app
```

```
COPY ./
```

```
RUN npm install
```

```
EXPOSE 3000
```

```
CMD [ "npm", "run", "watch" ]
```

TP2 : Différence entre CMD et ENTRYPOINT

Différence CMD et ENTRYPOINT

Utilisons le package mtr comme démonstrateur, celui ci permet d'obtenir un traceroute agréable visuellement.

Créer deux images docker pour tester ENTRYPOINT et CMD

Exemple de commande :

CMD ["mtr", "8.8.8.8"]

ENTRYPOINT ["mtr"]

Lancer les images de cette façon maintenant :

`docker run -it mtr-cmd 8.8.8.8`

`docker run -it mtr-entrypoint 8.8.8.8`

Que remarquez vous ?

Quelles sont les différences entre CMD et ENTRYPOINT

TP2 : Déployer un site Wordpress sur Docker et Docker compose

Publication d'un site wordpress avec docker

Créez un fichier nommé `docker-compose.yml` dans un répertoire de votre choix.

Dans ce fichier, configurez deux services : un service pour la base de données MySQL et un service pour l'application WordPress.

Pour le service MySQL, utilisez l'image Docker officielle de MySQL et définissez les variables d'environnement nécessaires pour créer la base de données WordPress et l'utilisateur WordPress.

Pour le service WordPress, utilisez l'image Docker officielle de WordPress et définissez les variables d'environnement nécessaires pour se connecter à la base de données MySQL

Publication d'un site wordpress avec docker

Lancez le conteneur WordPress en exécutant la commande `docker-compose up`.

Vérifiez que votre site WordPress est en cours d'exécution en accédant à l'url dans votre navigateur.

TP3 : Créer une pipeline de build d'image Docker
avec les outils CI/CD Gitlab

Publication d'un site wordpress avec docker

Création d'un compte Gitlab

- Si vous n'en avez pas déjà un, créez un compte sur Gitlab.com :
https://gitlab.com/users/sign_in#register-pane

Création d'un nouveau projet

- Créez un nouveau projet Gitlab et poussez-y l'application de votre choix
- Assurez-vous qu'un Dockerfile est présent dans votre repository ou créez-en un.

Publication d'un site wordpress avec docker

Configuration de la pipeline de build avec Gitlab CI/CD

- Créez un fichier `.gitlab-ci.yml` depuis l'interface web de Gitlab et choisissez "Docker" comme template.
- Vérifiez attentivement le fichier `.gitlab-ci.yml` pour vous assurer que la pipeline de build est correctement configurée.

Publication d'un site wordpress avec docker

Commit du fichier de configuration de la pipeline

- Effectuez un commit du fichier `.gitlab-ci.yml` pour déclencher la pipeline de build.

Vérification de la pipeline de build

- Vérifiez l'exécution de la pipeline de build sur le portail de Gitlab pour vous assurer qu'elle s'est déroulée avec succès.

Vérification de l'image Docker générée

- Vérifiez dans la section "Container Registry" de Gitlab que l'image Docker générée a bien été poussée.