

Lab3:Publication d'image

1. Créer une page d'accueil personnalisée

`index.html` :

```
<!DOCTYPE html>
<html lang="fr">
<head>
    <meta charset="UTF-8">
    <title>My First Container</title>
</head>
<body>
    <h1>Mon Conteneur Nginx est Actif !</h1>
    <p>
        Félicitations ! Ce contenu a été servi par votre image Docker personnalisée
        qui utilise le serveur web Nginx.
    </p>
    <p>
        <small>Ce fichier remplace la page d'accueil par défaut (index.html).</small>
    </p>
</body>
</html>
```

2. Lancer un conteneur Nginx

```
docker run -d --name nginx-custom -p 8080:80 nginx
```

3. Copier la page HTML dans le conteneur

```
docker cp index.html nginx-custom:/usr/share/nginx/html/index.html
```

Cela remplace la page d'accueil par défaut à l'intérieur du conteneur.

Vérifier ici:

```
http://localhost:8080/
```

4. Créer une image à partir du conteneur modifié

```
docker commit nginx-custom monuserdocker/nginx-custom:v1
```

⚠ Remplace monuserdocker par ton nom Docker Hub.

- docker commit: Crée une nouvelle image à partir des modifications apportées au système de fichiers du conteneur.
- nginx-custom: C'est le **nom** (ou l'**ID**) du conteneur Docker en cours d'exécution (ou récemment arrêté) dont tu veux sauvegarder l'état.
- monuserdocker: Ton nom d'utilisateur Docker Hub (ou le dépôt cible).

- `nginx-custom:v1` → Le nom du dépôt de l'image et le tag de l'image

5. Se connecter à Docker Hub

`docker login`

6. Pousser l'image vers Docker Hub

`docker push monuserdocker/nginx-custom:v1`

Pourquoi ne pas utiliser docker commit ?

`docker commit` :

- ✗ ne garde *aucune trace* de ce que on a modifié
- ✗ impossible à versionner
- ✗ crée souvent des images plus lourdes
- ✗ complique le travail en équipe
- ✗ rend les builds automatiques impossibles (CI/CD)

Autre solution?