Aller sur Docker Desktop

Faire docker run hello-world (tester si notre docker marche)

Docker run –name web-server (nom de l’image qu’on recupere sur dockerHub, ici nginx)

* Cela va nous créer le conteneur avec l’image nginx, notre ‘web server’

Pour afficher la liste des conteneur sur ma machine faire : docker ps (si on a lancé la commande d’au dessus, ouvrir un nouveau terminal).

Pour afficher la liste des images sur ma machine faire : docker images

* On peut préciser la recherche de l’image ( si on a pleins dimages et qu’on veut chercher nginx) : docker images | grep nginx

Pour supprimer un conteneur : docker rm -f

Docker run –name web-server -d -p 80 : 80 nginx

* -d pour détacher comme (& en prog) , -p 80 :80 (HOST : CONTENEUR)
* Apres avoir fait la dernière commande, on a ouvert un serveur en local, on peut aller sur google et on tape : <https://localhost:80> ou juste :80 jsp

On cherche après des templates gratuits sur google (free bootstrap), on filtre avec html et free. ON choisit le templates qui nous interesse. Puis on va sur le lien git qui est associé et on copié le lien git du projet .

Puis on fait git clone ‘le lien du git’

Ca cree des dossiers dont celui du site, on va dans ce dossier puis dans le dossier dist ou exe ou jar (dossier qui permet d’afficher le site)

On va essayer de rentrer dans le conteneur qu’on a créer tout à lheure. (Le prof a créer un conteneur avec comme id 9d77…, donc la on tape 9d7)

Docker exec -it 9d7 bash

On voit ensuite qu’on est rentré dans le conteneur, en bash. On est comme dans une VM mtn, exemple si on fait ls, on voit les fichiers du dossier. On voit qu’on a acces au conteneur du coup.

Mtn on va mettre le site dans notre docker. Juste on a un dossier dans le conteneur qui permet de mettre les interfaces web cest : /usr/share/nginx/html.

On va mettre le contenu de dist dans le conteneur à /usr/share/nginx/html.

Donc on va dans le dossier du site ici (gray-scale jsp quoi), quand on fait ls, on voit bien dist et on copie ca dans le conteneur :

Docker cp dist/. 9d7 :usr/share/nginx/html/ le dist/. Permet de dire copier tout le dossier dist et 9d7 : cest pour dire qu’on va dans le conteneur de l’id 9d7.

Mtn on a l’interface du site dans le dossier html du conteneur. On peut mettre le local host sur nternet et accéder au site mtn.

On va mtn aller sur vs code pour créer des images. On ouvre le dosier de notre site gray-scale … Dans le sous dossier source on cree un fichier Dockerfile.

Dans ce fichier dockerfile on ecrit ca :

From nginx

COPY dist/. /usr/share/nginx/html/

Puis dans le terminal de vscode dans le fichier gray-scale on tape pour créer notre propre image :

Docker build -t esiea-nginx . (le . est important car ca signifie le repertoire courant)

On peut regarder nos images avec docker images.

Mtn qu’on à l’image on va l’associer à notre propre conteneur :

Docker run –name esiea-web-server -d -p 80 :80 esiea-nginx

Si on relance sur google notre localhost :80 : on va voir le site internet gray-scale !!!

Mtn qu’on a notre image en local, on va la mettre en ligne pour que tout le monde puisse l’utiliser. ON va la push sur notre Dockeurhub

Tjr sur vs code, on va renommer l’image pour la foutre sur dockeurHub :

Docker tag esiea-nginx user\_name\_dockeurhub(moi romdka jcrois)/esiea-nginx

Mtn on va push cette image :

🡪D’abord on se co à dockerhub : docker login

* Puis on fou notre user et mdp, puis on est connecté.
* On peut mtn faire : docker push nom\_image (soit ici romdka/esiea-nginx)

Si on va sur dockerHub, on rafraichit et on voit sur notre compte cette image.