

PROJET DOCKER:

Membres du groupe :

1-AMETEPE Malthus

2-ODAH Alliance

3-BAHN'GOBI Soumaila

4-MIGAN Bénie

5-SAIBOU Afiss

Pour chacune de ces étapes ci-dessous, coller le contenu de vos fichiers et vos commandes accompagner de captures d'écran dans un document que vous rendrez au format PDF ou .DOCX

1- Créer un Dockerfile pour dockeriser le projet nodejs-MySQL-links

2- Construire (Build) l'image à partir du Dockerfile précédent et le nommer node-app avec comme tag v1.0

3- Créer un network du nom de tpl2

4- Démarrer le conteneur à partir de cette image, lui spécifier le nom que vous souhaitez et le rajouter au network tpl2

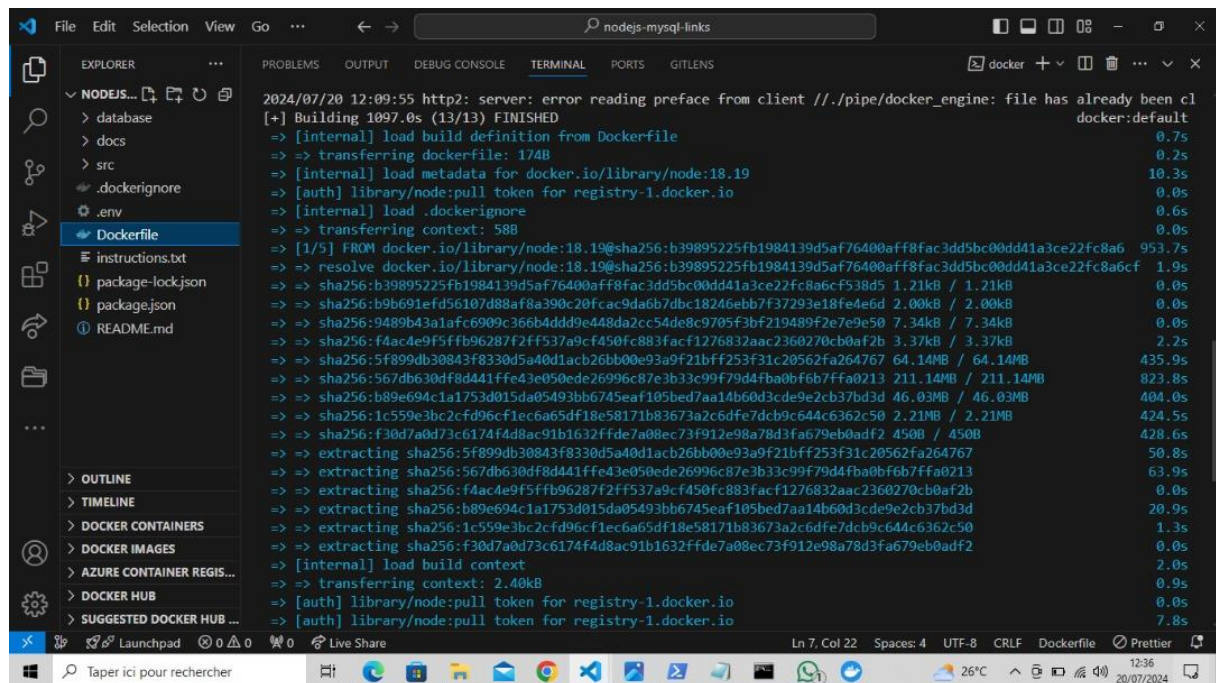
5- Démarrer un conteneur mysql avec le tag 5.7 en le rajoutant au network précédemment créer et exposer le port par défaut à un port disponible de votre DOCKER HOST

6- Créer un fichier docker-compose.yml qui démarre les 2 conteneurs

1- Créer un Dockerfile pour dockeriser le projet nodejs-MySQL-links

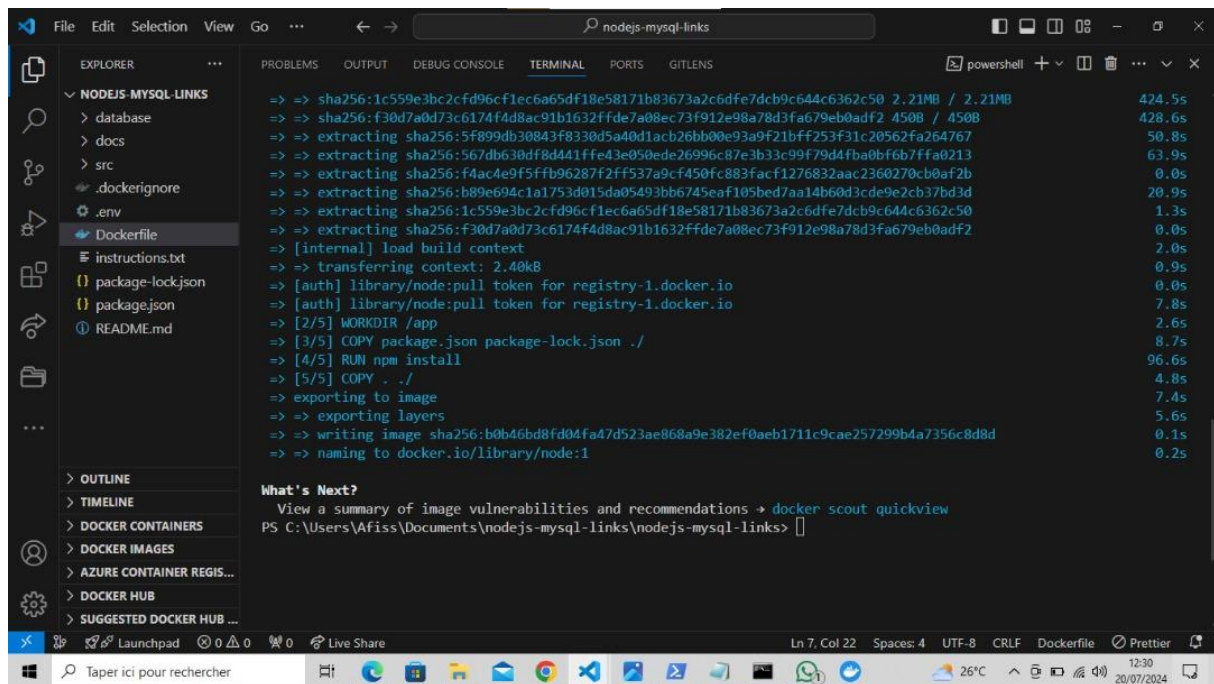
```
FROM node:18.19
WORKDIR /app
COPY package.json package-lock.json ./
RUN npm install
COPY . ./
EXPOSE 4000
CMD ["npm", "start"]
```

2- Construire (Build) l'image à partir du Dockerfile précédent et le nommer node-app avec comme tag v1.0



The screenshot shows the Visual Studio Code interface with the Dockerfile build process running in the terminal. The Explorer panel on the left shows the project structure, including the Dockerfile. The terminal output shows the build steps, including resolving the Dockerfile, transferring it, and building the image. The build is successful, and the image is named node-app with tag v1.0.

```
2024/07/20 12:09:55 http2: server: error reading preface from client //./pipe/docker_engine: file has already been cl
[+] Building 1097.0s (13/13) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 174B
=> [internal] load metadata for docker.io/library/node:18.19
=> [auth] library/node:pull token for registry-1.docker.io
=> [internal] load .dockerignore
=> => transferring context: 588
=> [1/5] FROM docker.io/library/node:18.19@sha256:b39895225fb1984139d5af76400aff8fac3dd5bc00dd41a3ce22fc8a6 953.7s
=> => resolve docker.io/library/node:18.19@sha256:b39895225fb1984139d5af76400aff8fac3dd5bc00dd41a3ce22fc8a6cf 1.9s
=> => sha256:b39895225fb1984139d5af76400aff8fac3dd5bc00dd41a3ce22fc8a6cf538d5 1.21kB / 1.21kB 0.0s
=> => sha256:b9b691efd56107d88af8a390c20fcac9da6b7dbc18246ebb7f37293e18fe4e6d 2.00kB / 2.00kB 0.0s
=> => sha256:9489b43a1afc6909c366b4ddd9e448da2cc54de8c9705f3bf219489f2e7e9e50 7.34kB / 7.34kB 0.0s
=> => sha256:f4ac4e9f5fffb96287f2ff537a9cf450fc883facf1276832aac2360270cb0af2b 3.37kB / 3.37kB 2.2s
=> => sha256:5f899db30843f8330d5a40d1acb26bb00e93a9f21bff253f31c20562fa264767 64.14MB / 64.14MB 435.9s
=> => sha256:567db630df8d441ffe43e050ede26996c87e3b33c99f79d4fba0bf6b7ffa0213 211.14MB / 211.14MB 823.8s
=> => sha256:b89e694c1a1753d015da05493bb6745eaf105bed7aa14b60d3cde9e2cb37bd3d 46.03MB / 46.03MB 404.0s
=> => sha256:1c559e3bc2cfd96cf1ec6a65df18e58171b83673a2c6dfe7dcb9c644c6362c50 2.21MB / 2.21MB 424.5s
=> => sha256:f30d7a0d73c6174f4d8ac91b1632ffde7a08ec73f912e98a78d3fa679eb0adf2 450B / 450B 428.6s
=> => extracting sha256:5f899db30843f8330d5a40d1acb26bb00e93a9f21bff253f31c20562fa264767 50.8s
=> => extracting sha256:567db630df8d441ffe43e050ede26996c87e3b33c99f79d4fba0bf6b7ffa0213 63.9s
=> => extracting sha256:f4ac4e9f5fffb96287f2ff537a9cf450fc883facf1276832aac2360270cb0af2b 0.0s
=> => extracting sha256:b89e694c1a1753d015da05493bb6745eaf105bed7aa14b60d3cde9e2cb37bd3d 20.9s
=> => extracting sha256:1c559e3bc2cfd96cf1ec6a65df18e58171b83673a2c6dfe7dcb9c644c6362c50 1.3s
=> => extracting sha256:f30d7a0d73c6174f4d8ac91b1632ffde7a08ec73f912e98a78d3fa679eb0adf2 0.0s
=> [internal] load build context
=> => transferring context: 2.40kB
=> [auth] library/node:pull token for registry-1.docker.io
=> [auth] library/node:pull token for registry-1.docker.io 7.8s
```



3- Créer un network du nom de tpl2

```
C:\Users\Afiss\Documents\nodejs-mysql-links\nodejs-mysql-links>docker network create tpl2
0f93f24ddd8ea57d26329efda4f6bd3b6df3fc7fec7d3da486531828e69f15b9
```

4- Démarrer le conteneur à partir de cette image, lui spécifier le nom que vous souhaitez et le rajouter au network tpl2

5- Démarrer un conteneur mysql avec le tag 5.7 en le rajoutant au network précédemment créer et exposer le port par défaut a un port disponible de votre DOCKER HOST

```
C:\WINDOWS\system32\cmd.exe
See 'docker run --help'.

C:\Users\Afiss\Documents\nodejs-mysql-links\nodejs-mysql-links> --network tpl2 \
'--network' n'est pas reconnu en tant que commande interne
ou externe, un programme exécutable ou un fichier de commandes.

C:\Users\Afiss\Documents\nodejs-mysql-links\nodejs-mysql-links> -p 4000:4000 \
'-p' n'est pas reconnu en tant que commande interne
ou externe, un programme exécutable ou un fichier de commandes.

C:\Users\Afiss\Documents\nodejs-mysql-links\nodejs-mysql-links> node-app:v1.0
La syntaxe du nom de fichier, de répertoire ou de volume est incorrecte.

C:\Users\Afiss\Documents\nodejs-mysql-links\nodejs-mysql-links>
C:\Users\Afiss\Documents\nodejs-mysql-links\nodejs-mysql-links>docker run -d --name node / --network tpl2 -p 4000:4000 / node:1
docker: invalid reference format.
See 'docker run --help'.

C:\Users\Afiss\Documents\nodejs-mysql-links\nodejs-mysql-links>docker run -d --name node --network tpl2 -p 4000:3306 node:1
0f56095fcf03b5a17849a2967742d9680d47150f2c50c7ee9944e1fbd4e3942a

C:\Users\Afiss\Documents\nodejs-mysql-links\nodejs-mysql-links>_
```

6- Créer un fichier docker-compose.yml qui démarre les 2 conteneurs

version: '3.8'

services:

node-app:

image: node-app:v1.0

container_name: node_app_container

networks:

- tpl2

ports:

- "4000:4000"

mysql:

image: mysql:5.7

container_name: mysql_container

environment:

MYSQL_ROOT_PASSWORD: mypass

MYSQL_DATABASE: linksdb

networks:

- tpl2

ports:

- "3306:4000"

networks:

tpl2:

driver: bridge