

Documentation Technique : Brief 8

1 : Quel fichier représente l'application web ?

Le fichier représentant l'application se nomme my-app.py

2 : A quoi sert la ligne USER uwsgi?

La ligne user sert à définir l'utilisateur du Dockerfile, si celui-ci n'est pas signifié, alors l'utilisateur sera le root.

3 : Que signifie la ligne EXPOSE 9090 9191?

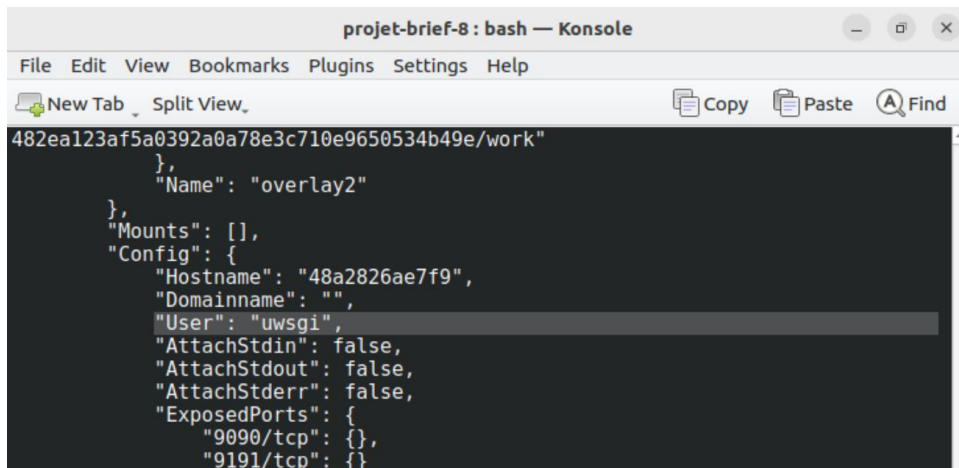
Cette ligne permet de signifier à but informatif les ports utilisés par le Dockerfile

4 : Lancez l'application et trouvez un moyen pour identifier l'utilisateur avec lequel tourne le conteneur

`docker inspect nom_du_conteneur | grep User`

```
fabio@Ubuntu-Docker:~$ docker inspect frontend | grep User
  "UsernsMode": "",
  "User": "uwsgi",
fabio@Ubuntu-Docker:~$
```

`docker inspect -f '{{ json .Config }}' nom_du_conteneur | jq`

A screenshot of a terminal window titled "projet-brief-8 : bash — Konsole". The window has a menu bar with "File", "Edit", "View", "Bookmarks", "Plugins", "Settings", and "Help". Below the menu bar is a toolbar with "New Tab", "Split View", "Copy", "Paste", and "Find". The terminal content shows a JSON object representing a Docker container configuration. The path "482ea123af5a0392a0a78e3c710e9650534b49e/work" is visible at the top. The JSON object includes fields for "Name", "Mounts", "Config", "Hostname", "Domainname", "User", "AttachStdin", "AttachStdout", "AttachStderr", "ExposedPorts", and "9090/tcp".

```
482ea123af5a0392a0a78e3c710e9650534b49e/work
},
  "Name": "overlay2"
},
  "Mounts": [],
  "Config": {
    "Hostname": "48a2826ae7f9",
    "Domainname": "",
    "User": "uwsgi",
    "AttachStdin": false,
    "AttachStdout": false,
    "AttachStderr": false,
    "ExposedPorts": {
      "9090/tcp": {},
      "9191/tcp": {}
    }
  }
}
```

Voici mon docker-compose.yml :

version: "3.7"

services:

frontend:

build: .

image: frontend

container_name: frontend

ports:

- "9090:9090"

- "9191:9191"

networks:

- projectnetwork

backend:

image: amouat/dnmonster:latest

ports:

- "8080"

container_name: backend

networks:

- projectnetwork

redis:

image: redis:latest

ports:

- "6379"

container_name: redis

networks:

- projectnetwork

networks:

projectnetwork:

driver: bridge

Pour que je puisse mettre en en-tête “backend” en lieu et place de dnmonster j’ai dû modifier le fichier my-app.py en remplaçant le dnmonster par backend :

Avant modification :

```
r = requests.get('http://dnmonster:8080/monster/' +  
name + '?size=80')
```

Après modification :

```
r = requests.get('http://backend:8080/monster/' +  
name + '?size=80')
```