

# TP 1– Dockeriser l’application DockerCoins

## Créer les images de DockerCoins

1. Clonez le projet Dockercoins suivant à partir du dépôt distant :

<https://github.com/brahimhamdi/dockercoins>

```
vagrant@master:~/lab4$ git clone https://github.com/brahimhamdi/dockercoins
Cloning into 'dockercoins'...
remote: Enumerating objects: 147, done.
remote: Counting objects: 100% (147/147), done.
remote: Compressing objects: 100% (105/105), done.
remote: Total 147 (delta 45), reused 136 (delta 34), pack-reused 0
Receiving objects: 100% (147/147), 146.78 KiB | 719.00 KiB/s, done.
Resolving deltas: 100% (45/45), done.
vagrant@master:~/lab4$ cd dockercoins/
vagrant@master:~/lab4/dockercoins$ tree
.
├── hasher
│   ├── Dockerfile
│   └── hasher.rb
├── rng
│   ├── Dockerfile
│   └── rng.py
├── webui
│   ├── Dockerfile
│   ├── Dockerfile.old
│   └── files
│       ├── d3.min.js
│       ├── index.html
│       ├── jquery-1.11.3.min.js
│       ├── jquery.js
│       ├── rickshaw.min.css
│       └── rickshaw.min.js
└── webui.js
worker
├── Dockerfile
└── worker.py

5 directories, 15 files
vagrant@master:~/lab4/dockercoins$
```

2. Décrivez tous les Dockerfiles du projet.

```
vagrant@master:~/lab4/dockercoins$ cat worker/Dockerfile
FROM python:alpine
MAINTAINER brahim.hamdi.consult@gmail.com
RUN pip install redis
RUN pip install requests
COPY worker.py /
CMD ["python", "worker.py"]
vagrant@master:~/lab4/dockercoins$ ls worker/
Dockerfile worker.py
vagrant@master:~/lab4/dockercoins$
```

```

vagrant@master:~/lab4/dockercoins$ cat rng/Dockerfile
FROM python:alpine
MAINTAINER brahim.hamdi.consult@gmail.com
RUN pip install Flask
COPY rng.py /
CMD ["python", "rng.py"]
EXPOSE 80
vagrant@master:~/lab4/dockercoins$ ls rng/rng.py
rng/rng.py
vagrant@master:~/lab4/dockercoins$ 

```

```

vagrant@master:~/lab4/dockercoins$ cat hasher/Dockerfile
FROM ruby:alpine
MAINTAINER brahim.hamdi.consult@gmail.com
RUN apk add --update build-base curl
RUN gem install sinatra
RUN gem install thin
ADD hasher.rb /
CMD ["ruby", "hasher.rb"]
EXPOSE 80
HEALTHCHECK --interval=1s --timeout=2s --retries=3 --start-period=1s CMD curl http://localhost/ || exit 1
vagrant@master:~/lab4/dockercoins$ ls hasher/hashier.rb
hasher/hashier.rb
vagrant@master:~/lab4/dockercoins$ 

```

```

vagrant@master:~/lab4/dockercoins$ cat webui/Dockerfile
FROM node:lts-slim

# Needed for node 15+ see https://stackoverflow.com/a/65443098
WORKDIR /usr/app

RUN npm install express
RUN npm install redis
COPY files/ ./files/
COPY webui.js .
CMD ["node", "webui.js"]
EXPOSE 80
vagrant@master:~/lab4/dockercoins$ 

```

3. Créez les images des composants *worker*, *rng* et *hasher* de ce projet. Et puis testez-les.

```

vagrant@master:~/lab4/dockercoins$ docker build -t rng rng/
[+] Building 9.8s (8/8) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 181B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/python:alpine
=> CACHED [1/3] FROM docker.io/library/python:alpine@sha256:506eed442161ee54460830b5a4c282f84a97046d0222c917d62bcf069446c26a
=> [2/3] RUN pip install Flask
=> [internal] load build context
=> => transferring context: 711B
=> [3/3] COPY rng.py /
=> exporting to image
=> => exporting layers
=> => writing image sha256:bba301db52fca69da46f81c397f78b4b6e6f34a5c93df24aba22f8c01ab004f0
=> => naming to docker.io/library/rng
vagrant@master:~/lab4/dockercoins$ 
=> => sha256:ab9f45de3756d3acbcbac0257fc4686a28fc5ada48917ee9eb4a77afa9ce011d 3.08MB / 3.08MB
=> => extracting sha256:63b65145d645c1250c391b2d16ebe53b3747c295ca8ba2fcb6b0cf064a4dc21c
=> => extracting sha256:e9f9cf5029d271a7f108bc0ad050e3772c5047c61a6e1f7413bc751dc4f01653
=> => extracting sha256:d47ea573a7911457dc1b9f91c991dd22366e36b455251f21e6d0b3a056b3a4c4
=> => extracting sha256:37906e6a7a2e6f773bfef2bcb87c5fa77cb2150296f96a8d7ecb9cc5d4eea5a9
=> => extracting sha256:ab9f45de3756d3acbcbac0257fc4686a28fc5ada48917ee9eb4a77afa9ce011d
=> [internal] load build context
=> => transferring context: 1.74kB
=> [2/4] RUN pip install redis
=> [3/4] RUN pip install requests
=> [4/4] COPY worker.py /
=> exporting to image
=> => exporting layers
=> => writing image sha256:2b3a4aca54f86a20f9a605e9d352cbbc0401eba8a329b22731df1f5a2dafc79
=> => naming to docker.io/library/worker
vagrant@master:~/lab4/dockercoins$ 

```

```

vagrant@master:~/lab4/dockercoins$ docker build -t hasher hasher/
[+] Building 146.6s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 345B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/ruby:alpine
=> [1/5] FROM docker.io/library/ruby:alpine@sha256:ee54124793e62129c5cdf788e9f1164332587e7c8dab6bd592514cf5eddc3d33
=> => resolve docker.io/library/ruby:alpine@sha256:ee54124793e62129c5cdf788e9f1164332587e7c8dab6bd592514cf5eddc3d33
=> => sha256:b9b65e4e1280321d96ede55468a094294068fd0c4e8d8024462a61fd86340895 224B / 224B
=> => sha256:2da0d7c94605d01653020bb0147af661c19f5fe72cfa88b384974b7c49db98d 31.81MB / 31.81MB
=> => sha256:ee54124793e62129c5cdf788e9f1164332587e7c8dab6bd592514cf5eddc3d33 1.65kB / 1.65kB
=> => sha256:fee8f90378b13a5ac5ce376ef173955048f65c088e6a9994c2ae84c35df87a4e 1.36kB / 1.36kB
=> => sha256:ddecbf57a632bf57ed4f05e62db343c7d90242105e0c42fb032d18de062ab3f0 7.24kB / 7.24kB
=> => sha256:ea9e29845fca002fae49920d8d41b5b0744be6b08687585f8accce5dc121a488 4.13MB / 4.13MB
=> => sha256:70acdf5b55d2fd8f6d412ce2250758441084ac573a5f113a2a9e95385ddceb91 173B / 173B
=> => extracting sha256:ea9e29845fca002fae49920d8d41b5b0744be6b08687585f8accce5dc121a488
=> => extracting sha256:b9b65e4e1280321d96ede55468a094294068fd0c4e8d8024462a61fd86340895
=> => extracting sha256:2da0d7c94605d01653020bb0147af661c19f5fe72cfa88b384974b7c49db98d
=> => extracting sha256:70acdf5b55d2fd8f6d412ce2250758441084ac573a5f113a2a9e95385ddceb91
=> [internal] load build context
=> => transferring context: 315B
=> [2/5] RUN apk add --update build-base curl
=> [3/5] RUN gem install sinatra
=> [4/5] RUN gem install thin
=> [5/5] ADD hasher.rb /
=> exporting to image
=> => exporting layers
=> => writing image sha256:351e27aa43799ce4b6a33fc75b7ec0343f5aa3d80b9f94f7a6825e39587f356d
=> => naming to docker.io/library/hasher
vagrant@master:~/lab4/dockercoins$ 

```

```

vagrant@master:~/lab4/dockercoins$ docker images

```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
hasher	latest	351e27aa4379	13 minutes ago	324MB
rng	latest	bba301db52fc	16 minutes ago	72.9MB
worker	latest	2b3a4aca54f8	16 minutes ago	73.5MB
wiki_app	1.0	f3d33322ddcf	58 minutes ago	530MB
mariadb	latest	4a632f970181	8 days ago	401MB
wiki_db	1.0	ccf162a83226	8 days ago	401MB
nginx	latest	904b8cb13b93	3 weeks ago	142MB

```

vagrant@master:~/lab4/dockercoins$ 

```

# Poussez les images Docker vers le registre DockerHub

4. Créez un compte dans *hub.docker.com* .
5. Marquez les 3 images avec votre nom d'utilisateur DockerHub et transférez-les vers ce registre.

```
vagrant@master:~/lab4/dockercoins$ docker tag worker:latest brahimhamdi/worker:latest
vagrant@master:~/lab4/dockercoins$ docker tag rng:latest brahimhamdi/rng:latest
vagrant@master:~/lab4/dockercoins$ docker tag hasher:latest brahimhamdi/hasher:latest
vagrant@master:~/lab4/dockercoins$ docker tag wiki_app:1.0 brahimhamdi/wiki_app:1.0
vagrant@master:~/lab4/dockercoins$ docker tag wiki_db:1.0 brahimhamdi/wiki_db:1.0
vagrant@master:~/lab4/dockercoins$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
hasher	latest	351e27aa4379	17 minutes ago	324MB
brahimhamdi/hasher	latest	351e27aa4379	17 minutes ago	324MB
brahimhamdi/rng	latest	bba301db52fc	19 minutes ago	72.9MB
rng	latest	bba301db52fc	19 minutes ago	72.9MB
brahimhamdi/worker	latest	2b3a4aca54f8	20 minutes ago	73.5MB
worker	latest	2b3a4aca54f8	20 minutes ago	73.5MB
brahimhamdi/wiki_app	1.0	f3d33322ddcf	About an hour ago	530MB
wiki_app	1.0	f3d33322ddcf	About an hour ago	530MB
wiki_db	1.0	ccf162a83226	8 days ago	401MB
brahimhamdi/wiki_db	1.0	ccf162a83226	8 days ago	401MB
mariadb	latest	4a632f970181	8 days ago	401MB
nginx	latest	904b8cb13b93	3 weeks ago	142MB

```
vagrant@master:~/lab4/dockercoins$
```

```
vagrant@master:~/lab4/dockercoins$ docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username: brahimhamdi
Password:
WARNING! Your password will be stored unencrypted in /home/vagrant/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
vagrant@master:~/lab4/dockercoins$ docker push brahimhamdi/worker
Using default tag: latest
The push refers to repository [docker.io/brahimhamdi/worker]
9583317fd0ed: Pushed
ef3d14a92a4e: Pushed
54ba6a65492f: Pushed
73315ddf6270: Mounted from library/python
e7d1f4f1959a: Mounted from library/python
8a3287a52de9: Pushed
3efb846cc795: Mounted from library/python
7cd52847ad77: Pushed
latest: digest: sha256:03f2edcafc47204cea377e1ecb7da56cae419b83b499306720fce3c034109948 size: 1997
vagrant@master:~/lab4/dockercoins$
```

....