

Lab 4 – Dockerfile

Build docker image

Build image from scratch

1. Write a C program that displays "Hello DevOps !! ".

```
vagrant@Manager:~/Lab4$ cat app.c
#include <stdio.h>
#include <unistd.h>
int main() {
    int i;
    for(i=0;i<10;i++) {
        printf("Hello DevOps ! \n");
        sleep(3);
    }
    return 0;
}
vagrant@Manager:~/Lab4$
```

2. Compile the program with a static library.

```
vagrant@Manager:~/Lab4$ gcc app.c -o app -static
vagrant@Manager:~/Lab4$ ls -l
total 860
-rwxrwxr-x 1 vagrant vagrant 872080 Jun  8 10:29 app
-rw-rw-r-- 1 vagrant vagrant 158 Jun  8 10:28 app.c
-rw-rw-r-- 1 vagrant vagrant 43 Jun  8 10:20 Dockerfile
vagrant@Manager:~/Lab4$
```

3. Write a dockerfile file with the following content:

```
FROM      scratch
COPY      app  app
CMD       ["/app"]
```

```
vagrant@Manager:~/Lab4$ cat Dockerfile
FROM      scratch
COPY      app  app
CMD       ["/app"]
vagrant@Manager:~/Lab4$
```

4. Build image *myapp:1.0*

```
vagrant@Manager:~/Lab4$ docker build -t myapp:1.0 .
[+] Building 0.4s (5/5) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 80B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load build context
=> => transferring context: 872.27kB
=> [1/1] COPY APP APP
=> exporting to image
=> exporting layers
=> writing image sha256:6f1d802ea830804fd948051a16f05a08b8c64d806102899d275930927f8779d2
=> naming to docker.io/library/myapp:1.0
vagrant@Manager:~/Lab4$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
myapp	1.0	6f1d802ea830	7 seconds ago	872kB
mariaadb	latest	9a79847e85fb	6 days ago	403MB
nginx	latest	f9c14fe76d50	2 weeks ago	143MB

```
vagrant@Manager:~/Lab4$
```

rahim HAMDI

5. Create new container with builded image.

```
vagrant@Manager:~/Lab4$ docker run -d --name app1 myapp:1.0
c555ceeb8ff725d77e171dec9b42e29999602f272091cfcc881cc2c4dac8bf2f
vagrant@Manager:~/Lab4$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
c555ceeb8ff7   myapp:1.0  "./app"   4 seconds ago    Up 3 seconds           app1
vagrant@Manager:~/Lab4$
vagrant@Manager:~/Lab4$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
c555ceeb8ff7   myapp:1.0  "./app"   29 seconds ago    Up 28 seconds           app1
vagrant@Manager:~/Lab4$
vagrant@Manager:~/Lab4$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
c555ceeb8ff7   myapp:1.0  "./app"   36 seconds ago    Exited (0) 5 seconds ago           app1
vagrant@Manager:~/Lab4$ docker logs app1
Hello DevOps !
Hello DevOps !
Hello DevOps !
Hello DevOps !
Hello DevOps !
Hello DevOps !
Hello DevOps !
Hello DevOps !
Hello DevOps !
Hello DevOps !
Hello DevOps !
vagrant@Manager:~/Lab4$
```

Build mediawiki images

6. Clone the following mediawiki project from the remote repository :

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

Change to the wikimedia folder

```
vagrant@Manager:~/Lab4$ git clone https://github.com/brahimhamdi/wikimedia
Cloning into 'wikimedia'...
remote: Enumerating objects: 28, done.
remote: Counting objects: 100% (28/28), done.
remote: Compressing objects: 100% (21/21), done.
remote: Total 28 (delta 3), reused 28 (delta 3), pack-reused 0
Unpacking objects: 100% (28/28), 3.11 KiB | 167.00 KiB/s, done.
vagrant@Manager:~/Lab4$ cd wikimedia/
vagrant@Manager:~/Lab4/wikimedia$ tree
.
├── wiki_app
│   ├── Dockerfile
│   └── wiki.conf
└── wiki_db
    └── Dockerfile

2 directories, 3 files
vagrant@Manager:~/Lab4/wikimedia$
```

7. Describe the Dockerfile in the `wiki_app` folder.

```
vagrant@Manager:~/Lab4/wikimedia$ cd wiki_app/
vagrant@Manager:~/Lab4/wikimedia/wiki_app$ ls
Dockerfile  wiki.conf
vagrant@Manager:~/Lab4/wikimedia/wiki_app$ cat Dockerfile
FROM          nginx
MAINTAINER    brahim.hamdi.consult@gmail.com
COPY          wiki.conf /etc/nginx/conf.d/default.conf
RUN           apt update -y && apt install -y wget mariadb-client nano php-mbstring gzip unzip php-fpm php-xml php-intl php-mysql
RUN           usermod -a -G www-data nginx
ADD           https://releases.wikimedia.org/mediawiki/1.36/mediawiki-1.36.0-rc.0.tar.gz /
RUN           tar xzvf /mediawiki-1.36.0-rc.0.tar.gz && mv /mediawiki-1.36.0-rc.0 /usr/share/nginx/wiki && chown -R www-data:www-data /usr/s
hare/nginx/wiki
VOLUME        /var/log/nginx /usr/share/nginx/wiki /etc/nginx/conf.d/
CMD           service php7.4-fpm start && nginx -g "daemon off;"
vagrant@Manager:~/Lab4/wikimedia/wiki_app$
```

Based on this Dockerfile, build `wiki_app:1.0` image and then test it.

```
vagrant@Manager:~/Lab4/wikimedia/wiki_app$ docker build -t wiki_app:1.0 .
[+] Building 142.2s (12/12) FINISHED
=> [internal] load build definition from Dockerfile                                0.0s
=> => transferring dockerfile: 661B                                              0.0s
=> [internal] load .dockerignore                                                  0.1s
=> => transferring context: 2B                                                    0.0s
=> [internal] load metadata for docker.io/library/nginx:latest                  0.0s
=> [internal] load build context                                                  0.1s
=> => transferring context: 565B                                                  0.1s
=> https://releases.wikimedia.org/mediawiki/1.36/mediawiki-1.36.0-rc.0.tar.gz    38.5s
=> CACHED [1/6] FROM docker.io/library/nginx                                    0.0s
=> [2/6] COPY          wiki.conf /etc/nginx/conf.d/default.conf                  0.1s
=> [3/6] RUN           APT update -y && apt install -y wget mariadb-client nano php-mbstring gzip unzip php-fpm php-xml php-intl php-mysql 98.8s
=> [4/6] RUN           USERMOD -a -G www-data nginx                             1.5s
=> [5/6] ADD           HTTPS://RELEASES.WIKIMEDIA.ORG/MEDIAWIKI/1.36/MEDIAWIKI-1.36.0-RC.0.TAR.GZ / 1.5s
=> [6/6] RUN           TAR xzvf /mediawiki-1.36.0-rc.0.tar.gz && mv /mediawiki-1.36.0-rc.0 /usr/share/nginx/wiki && chown -R www-data:www-data 15.9s
=> exporting to image                                                            22.9s
=> => exporting layers                                                            22.5s
=> => writing image sha256:489c71e694c09f9ec28945bdd99de4acfb2185edf272aab9a306e3d13a23c289 0.1s
=> => naming to docker.io/library/wiki_app:1.0                                  0.1s
vagrant@Manager:~/Lab4/wikimedia/wiki_app$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
wiki_app      1.0       489c71e694c0  2 minutes ago  530MB
myapp         1.0       6f1d802ea830  15 minutes ago 872kB
mariadb       latest    9a79847e85fb   6 days ago    403MB
nginx         latest    f9c14fe76d50   2 weeks ago   143MB
vagrant@Manager:~/Lab4/wikimedia/wiki_app$
```

```
vagrant@Manager:~/Lab4/wikimedia/wiki_app$ docker volume ls
DRIVER    VOLUME NAME
vagrant@Manager:~/Lab4/wikimedia/wiki_app$ docker run -d --name app1 -p 8000:80 wiki_app:1.0
546deb8394f97f95d43ba5aac014fccc36d1493d47a9ee007fc5564dbb4cd6c8
vagrant@Manager:~/Lab4/wikimedia/wiki_app$ docker ps -a
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                               NAMES
546deb8394f9   wiki_app:1.0 "/docker-entrypoint. ..." 32 seconds ago Up 6 seconds   0.0.0.0:8000->80/tcp               app1
vagrant@Manager:~/Lab4/wikimedia/wiki_app$
vagrant@Manager:~/Lab4/wikimedia/wiki_app$ docker volume ls
DRIVER    VOLUME NAME
local     3dbfbceec8e680c74c77f932fec487cffd593daf1bb802eedd60cb573841461a5
local     9ad9913d3c9e2c1705ada73b21a530ad4b9d4ac2cc435028a98a989b5c396ba9
local     bc4951394aac6c92cff96019e3496f491e364cee66f2778776c7acc60ffa7329
vagrant@Manager:~/Lab4/wikimedia/wiki_app$ sudo ls /var/lib/docker/volumes/3dbfbceec8e680c74c77f932fec487cffd593daf1bb802eedd60cb573841461a5/_d
ata/
access.log  error.log
vagrant@Manager:~/Lab4/wikimedia/wiki_app$
```

8. Describe the Dockerfile in the `wiki_db` folder.

```
vagrant@Manager:~/Lab4/wikimedia/wiki_app$ cd ../wiki_db/
vagrant@Manager:~/Lab4/wikimedia/wiki_db$ cat Dockerfile
FROM          mariadb
MAINTAINER    brahim.hamdi.consult@gmail.com
ENV           MYSQL_ROOT_PASSWORD wiki
ENV           MYSQL_DATABASE DevOps_wiki
ENV           MYSQL_USER      brahim
ENV           MYSQL_PASSWORD  hamdi
CMD           ["mysqld"]
vagrant@Manager:~/Lab4/wikimedia/wiki_db$
```

Based on this Dockerfile, build `wiki_db:1.0` image and then test it.

```
vagrant@Manager:~/Lab4/wikimedia/wiki_db$ docker build -t wiki_db:1.0 .
[+] Building 0.7s (5/5) FINISHED
=> [internal] load build definition from Dockerfile                                0.2s
=> => transferring dockerfile: 222B                                              0.1s
=> [internal] load .dockerignore                                                 0.1s
=> => transferring context: 2B                                                  0.0s
=> [internal] load metadata for docker.io/library/mariadb:latest                0.0s
=> [1/1] FROM docker.io/library/mariadb                                         0.3s
=> exporting to image                                                            0.0s
=> => exporting layers                                                            0.0s
=> => writing image sha256:6d799b508be0f418ff55938bd043f6e69ef1f608a43b81429cdcf911fef206bb 0.0s
=> => naming to docker.io/library/wiki_db:1.0                                  0.0s
vagrant@Manager:~/Lab4/wikimedia/wiki_db$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
wiki_app      1.0       489c71e694c0   8 minutes ago  530MB
myapp         1.0       6f1d802ea830   22 minutes ago 872kB
mariadb       latest    9a79847e85fb   6 days ago    403MB
wiki_db       1.0       6d799b508be0   6 days ago    403MB
nginx         latest    f9c14fe76d50   2 weeks ago   143MB
vagrant@Manager:~/Lab4/wikimedia/wiki_db$ docker run -d --name db1 wiki_db:1.0
1162ef06575cffb16b2038bc05e0c06b1a4e9214d215744a199953a6ac97a37f
vagrant@Manager:~/Lab4/wikimedia/wiki_db$ docker volume ls
DRIVER    VOLUME NAME
local     3dbfbcec8e680c74c77f932fec487cffd593daf1bb802eadd60cb573841461a5
local     5ae3ce7bde6c186fe8af7ee77f578ce4620c7adfac0eafbae8894e286685d36
local     9ad9913d3c9e2c1705ada73b21a530ad4b9d4ac2cc435028a98a989b5c396ba9
local     bc4951394aac6c92cff96019e3496f491e364cee66f2778776c7acc60ffa7329
vagrant@Manager:~/Lab4/wikimedia/wiki_db$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED    STATUS    PORTS          NAMES
1162ef06575c   wiki_db:1.0    "docker-entrypoint.s..." 22 seconds ago Up 20 seconds 3306/tcp      db1
546deb8394f9   wiki_app:1.0   "/docker-entrypoint..." 6 minutes ago  Up 5 minutes  0.0.0.0:8000->80/tcp  app1
```

9. Install the application.



Build dockercoins images

10. Clone the following dockercoins project from the remote repository :

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

```

vagrant@Manager:~/Lab4$ git clone https://github.com/brahinhamdi/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 | 596.00 KiB/s, done.
Resolving deltas: 100% (45/45), done.
vagrant@Manager:~/Lab4$
vagrant@Manager:~/Lab4$ cd dockercoins/
vagrant@Manager:~/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
└── worker
    ├── Dockerfile
    └── worker.py

5 directories, 15 files
vagrant@Manager:~/Lab4/dockercoins$

```

11. Describe all Dockerfiles in the project.

```

vagrant@Manager:~/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@Manager:~/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@Manager:~/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@Manager:~/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@Manager:~/Lab4/dockercoins$

```


12. Based on Dockerfiles, build all images of this projects. And then test them.

```
vagrant@Manager:~/Lab4/dockercoins$ docker build -t hasher:v1 hasher/
[+] Building 146.2s (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@sha256:1df6125b0f90e087123698e1b2df1c6a544a40033a5a14bfa3ef7067863d3474
=> => resolve docker.io/library/ruby:alpine@sha256:1df6125b0f90e087123698e1b2df1c6a544a40033a5a14bfa3ef7067863d3474
=> => sha256:f55a9173cc83d16fc9d2fa4714aafff4971f72582aeb1e4a7202d6f7dd4385c9 7.24kB / 7.24kB
=> => sha256:8a49fdb3b6a5ff2bd8ec6a86c05b2922a0f7454579ecc07637e94dfd1d0639b6 3.40MB / 3.40MB
=> => sha256:4d3d25e339e798c7606304d1a00cff7f31a5876c66dec58866278476db2c83ea 4.13MB / 4.13MB
=> => sha256:75dd67f3ccc89561ff13877120b513941dc1e14c5ab32c4ce6d5585ee6ef9189 226B / 226B
=> => sha256:1df6125b0f90e087123698e1b2df1c6a544a40033a5a14bfa3ef7067863d3474 1.65kB / 1.65kB
=> => sha256:a50add1892fefca47fd3ae12cd3cc96c125a855b3d3e759e04843466ef8a9e96 1.36kB / 1.36kB
=> => sha256:02e0073c722e2b5a6b75ca6815e02a7eea9507b8e6e7395cf1c1fa637f095277 31.82MB / 31.82MB
=> => extracting sha256:8a49fdb3b6a5ff2bd8ec6a86c05b2922a0f7454579ecc07637e94dfd1d0639b6
=> => sha256:2f6f0e03d5f788eb85649a5771828e072560f5bc56e7d65caeab2551be52aaf1 173B / 173B
=> => extracting sha256:4d3d25e339e798c7606304d1a00cff7f31a5876c66dec58866278476db2c83ea
=> => extracting sha256:75dd67f3ccc89561ff13877120b513941dc1e14c5ab32c4ce6d5585ee6ef9189
=> => extracting sha256:02e0073c722e2b5a6b75ca6815e02a7eea9507b8e6e7395cf1c1fa637f095277
=> => extracting sha256:2f6f0e03d5f788eb85649a5771828e072560f5bc56e7d65caeab2551be52aaf1
=> [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 sha256:336a57c193e63c60741b53693b04a1d4cd9ea98c94f4626465cf69085a8ddeb
=> => naming to docker.io/library/hasher:v1
vagrant@Manager:~/Lab4/dockercoins$
```

```
vagrant@Manager:~/Lab4/dockercoins$ docker build -t rng:v1 rng/
[+] Building 29.7s (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
=> [1/3] FROM docker.io/library/python:alpine@sha256:995c7fcd9a10e0e1a4555861dac63436b456822a167f07b6599d4f105de6f
=> => resolve docker.io/library/python:alpine@sha256:995c7fcd9a10e0e1a4555861dac63436b456822a167f07b6599d4f105de6f
=> => sha256:0357922e53aa8671e175f58d46bfd245908047b81e66370f634863785fe0c70e 622.30kB / 622.30kB
=> => sha256:a5c7ada51cb6c797e043633c2ea90da1031cc4fe9e2ca00d37dc967d82932d67 14.62MB / 14.62MB
=> => sha256:9ebc4081ccad9b1c9760ff3af0d3f6417798d0d72f56ecc8108611f60344afa8 241B / 241B
=> => sha256:995c7fcd9a10e0e1a4555861dac63436b456822a167f07b6599d4f105de6fa0 1.65kB / 1.65kB
=> => sha256:a6ee91afa87ebe042adf0820c83745855f895addf70dfe9cb1c1e4e8bb436c47 1.37kB / 1.37kB
=> => sha256:35f8462d36fe25f117df6aa781567fe73dcd0796c859252c0894b1e308e27ce 6.26kB / 6.26kB
=> => sha256:839b27ab4d6651bf3819f976e92c30d5b00f6d0f4dc8779e82f04a1b4c88a1d0 3.09MB / 3.09MB
=> => extracting sha256:0357922e53aa8671e175f58d46bfd245908047b81e66370f634863785fe0c70e
=> => extracting sha256:a5c7ada51cb6c797e043633c2ea90da1031cc4fe9e2ca00d37dc967d82932d67
=> => extracting sha256:9ebc4081ccad9b1c9760ff3af0d3f6417798d0d72f56ecc8108611f60344afa8
=> => extracting sha256:839b27ab4d6651bf3819f976e92c30d5b00f6d0f4dc8779e82f04a1b4c88a1d0
=> [internal] load build context
=> => transferring context: 711B
=> [2/3] RUN pip install Flask
=> [3/3] COPY rng.py /
=> exporting to image
=> => exporting layers
=> => writing image sha256:f2d972dfef68b349bbb3999d41ef776fe172b185f393ce3b87ed50cfb398d2ba
=> => naming to docker.io/library/rng:v1
vagrant@Manager:~/Lab4/dockercoins$
```

```
vagrant@Manager:~/Lab4/dockercoins$ docker build -t worker:v1 worker/
[+] Building 14.6s (9/9) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 202B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/python:alpine
=> CACHED [1/4] FROM docker.io/library/python:alpine@sha256:995c7fcd9a10e0e1a4555861dac63436b456822a167f07t
=> [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:baae3122b8d9efb015fd16d8606a6d1d6aec346d46259ae7f234c4c21713aa08
=> => naming to docker.io/library/worker:v1
vagrant@Manager:~/Lab4/dockercoins$
```

```
vagrant@Manager:~/Lab4/dockercoins$ docker build -t webui:v1 webui/
[+] Building 60.5s (11/11) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 258B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/node:lts-slim
=> [1/6] FROM docker.io/library/node:lts-slim@sha256:b175cd7f3358c399f7bcee9b1032b86b71b1afa4cfb4dd0db55d6
=> => resolve docker.io/library/node:lts-slim@sha256:b175cd7f3358c399f7bcee9b1032b86b71b1afa4cfb4dd0db55d6
=> => sha256:d5b82e7bea47bbb6a1731885f2378af1dc5bcf4441826bec4ab517f41643981 6.84kB / 6.84kB
=> => sha256:0bfad7312c25e414fc3d0e3640a77bd4b5c74d2118595670999afef9758cb49b 4.19kB / 4.19kB
=> => sha256:4f7a205dec32e99b46ae10e5becb81c9fbe26d1e0f5491b8e9e9dc16b976aff3 46.00MB / 46.00MB
=> => sha256:36cdd2ca91d207e9d4cb63c6c4f289eece8673b728783bb11a227f6a8fa3f301 2.76MB / 2.76MB
=> => sha256:b175cd7f3358c399f7bcee9b1032b86b71b1afa4cfb4dd0db55d66f871475a3e 1.21kB / 1.21kB
=> => sha256:8463e2d7bacf0cb576453a0ea8425f3b3c87fa9dd5c8a84ab1908cfd407f3edd 1.37kB / 1.37kB
=> => extracting sha256:0bfad7312c25e414fc3d0e3640a77bd4b5c74d2118595670999afef9758cb49b
=> => sha256:4a5302452d0fc5cb8a4c4f13ebf2681de7f27949080f8ee32f255f31bd7fc680 452B / 452B
=> => extracting sha256:4f7a205dec32e99b46ae10e5becb81c9fbe26d1e0f5491b8e9e9dc16b976aff3
=> => extracting sha256:36cdd2ca91d207e9d4cb63c6c4f289eece8673b728783bb11a227f6a8fa3f301
=> => extracting sha256:4a5302452d0fc5cb8a4c4f13ebf2681de7f27949080f8ee32f255f31bd7fc680
=> [internal] load build context
=> => transferring context: 429.55kB
=> [2/6] WORKDIR /usr/app
=> [3/6] RUN npm install express
=> [4/6] RUN npm install redis
=> [5/6] COPY files/ ./files/
=> [6/6] COPY webui.js .
=> exporting to image
=> => exporting layers
=> => writing image sha256:87dfc182f7a4e53b145629fa1942e24dbf5fc9e5f41ac4e711f49aa522abff86
=> => naming to docker.io/library/webui:v1
vagrant@Manager:~/Lab4/dockercoins$
```

```
vagrant@Manager:~/Lab4/dockercoins$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
webui	v1	87dfc182f7a4	20 seconds ago	255MB
worker	v1	baae3122b8d9	About a minute ago	74MB
rng	v1	f2d972dfef68	2 minutes ago	73.5MB
hasher	v1	336a57c193e6	3 minutes ago	324MB
wiki_app	1.0	489c71e694c0	2 hours ago	530MB
myapp	1.0	6f1d802ea830	2 hours ago	872kB
mariadb	latest	9a79847e85fb	6 days ago	403MB
wiki_db	1.0	6d799b508be0	6 days ago	403MB
nginx	latest	f9c14fe76d50	2 weeks ago	143MB

```
vagrant@Manager:~/Lab4/dockercoins$
```

```

vagrant@Manager:~/Lab4/dockercoins$ docker run -d --name hasher -p 8001:80 hasher:v1
10dacf30d3071ffc54214ae57d18e3d41400ca95f4c6f2a7c4193f0777ed5803
vagrant@Manager:~/Lab4/dockercoins$ docker run -d --name rng -p 8002:80 rng:v1
3e180b7da1f03e486ca847d24e62ed6e3df1b4dfe4a0cd07672e4f9cc3a23772
vagrant@Manager:~/Lab4/dockercoins$ docker run -d --name worker worker:v1
1efc2081cc31b2b5c77b0d8306e373d82ea04bb203e94ac2baf0f8e585ed9b50
vagrant@Manager:~/Lab4/dockercoins$ docker run -d --name webui -p 8000:80 webui:v1
d0e91dd0ada91355544078c50b5b2f183a61db4db394b3bdef3824819a7f2671
docker: Error response from daemon: driver failed programming external connectivity on endpoint webui (07d5ec77f4932dc0fd016565098c7d0
0ef395dfd906e4046c7aaaf38): Bind for 0.0.0.0:8000 failed: port is already allocated.
vagrant@Manager:~/Lab4/dockercoins$
vagrant@Manager:~/Lab4/dockercoins$ docker ps

```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
1efc2081cc31	worker:v1	"python worker.py"	20 seconds ago	Up 19 seconds		worker
3e180b7da1f0	rng:v1	"python rng.py"	35 seconds ago	Up 34 seconds	0.0.0.0:8002->80/tcp	rng
10dacf30d307	hasher:v1	"ruby hasher.rb"	52 seconds ago	Up 49 seconds (healthy)	0.0.0.0:8001->80/tcp	hasher

Push docker images to remote registry

13. Create an account in *hub.docker.com/* registry.

14. Tags all images with your GitHub username and push them to dockerhub registry.

```

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@Manager:~/Lab4/dockercoins$

```

```

vagrant@Manager:~/Lab4/dockercoins$ docker tag rng:v1 brahimhamdi/rng:v1
vagrant@Manager:~/Lab4/dockercoins$ docker images | grep rng
brahimhamdi/rng      v1      f2d972dfef68   17 minutes ago   73.5MB
rng                  v1      f2d972dfef68   17 minutes ago   73.5MB
vagrant@Manager:~/Lab4/dockercoins$ docker push brahimhamdi/rng:v1
The push refers to repository [docker.io/brahimhamdi/rng]
7127afb85374: Pushed
eb3cf7bfc4bd: Pushed
31215aa3e6ae: Mounted from brahimhamdi/worker
a3941c1792e9: Mounted from brahimhamdi/worker
3b5ed29797e6: Mounted from brahimhamdi/worker
208977ac81d7: Mounted from brahimhamdi/worker
bb01bd7e32b5: Mounted from brahimhamdi/worker
v1: digest: sha256:6e089049dbea3c609d2e1cf0ac99f604e64d901187856051387859b8d74d45cf size: 1786
vagrant@Manager:~/Lab4/dockercoins$
vagrant@Manager:~/Lab4/dockercoins$ docker tag hasher:v1 brahimhamdi/hasher:v1
vagrant@Manager:~/Lab4/dockercoins$ docker push brahimhamdi/hasher:v1
The push refers to repository [docker.io/brahimhamdi/hasher]
82626f01751e: Pushed
b274485c6630: Retrying in 1 second
b6c3f30e1b42: Retrying in 1 second
d6b082f1b7b6: Pushing [=====] 69.1MB/244.8MB
21a01cb926ce: Mounted from library/ruby
df22cccb42bc: Retrying in 4 seconds
ca7012631066: Retrying in 1 second
04dfbe480e06: Waiting
bb01bd7e32b5: Waiting

```


15. Delete locally all pushed images.

```
vagrant@Manager:~/Lab4/dockercoins$ docker rmi worker:v1 brahimhamdi/worker:v1 rng:v1 brahimhamdi/rng:v1 hasher:v1 brahimhamdi/hasher:v1
Untagged: worker:v1
Untagged: brahimhamdi/worker:v1
Untagged: brahimhamdi/worker@sha256:cdec595f62ed841ea806aa0251e866045d175e394fda17f9c9b2510398a4ecef
Deleted: sha256:baae3122b8d9efb015fd16d8606a6d1d6aec346d46259ae7f234c4c21713aa08
Untagged: rng:v1
Untagged: brahimhamdi/rng:v1
Untagged: brahimhamdi/rng@sha256:6e089049d8ea3c609d2e1cf0ac99f604e64d901187856051387859b8d74d45cf
Deleted: sha256:f2d972dfe68b349bbb3999d41ef776fe172b185f393ce3b87ed50cfb398d2ba
Untagged: hasher:v1
Untagged: brahimhamdi/hasher:v1
Deleted: sha256:336a57c193e63c60741b53693b04a1d4cd9ea98c94f4626465cf69085a8ddebf
vagrant@Manager:~/Lab4/dockercoins$
```

Test pushed images.

```
vagrant@Manager:~/Lab4/dockercoins$ docker run -d --name worker brahimhamdi/worker:v1
Unable to find image 'brahimhamdi/worker:v1' locally
v1: Pulling from brahimhamdi/worker
8a49fdb3b6a5: Already exists
0357922e53aa: Already exists
a5c7ada51cb6: Already exists
9ebc4081ccad: Already exists
839b27ab4d66: Already exists
6c28a5a4585a: Already exists
53975e5528b9: Already exists
db752cca316c: Already exists
Digest: sha256:cdec595f62ed841ea806aa0251e866045d175e394fda17f9c9b2510398a4ecef
Status: Downloaded newer image for brahimhamdi/worker:v1
ab2d12bcf49fdca2100a7dc6778bd54901b3999818cabb6e70a9c973611f00a6
vagrant@Manager:~/Lab4/dockercoins$
vagrant@Manager:~/Lab4/dockercoins$ docker run -d --name rng -p 8001:80 brahimhamdi/rng:v1
Unable to find image 'brahimhamdi/rng:v1' locally
v1: Pulling from brahimhamdi/rng
8a49fdb3b6a5: Already exists
0357922e53aa: Already exists
a5c7ada51cb6: Already exists
9ebc4081ccad: Already exists
839b27ab4d66: Already exists
0f11e9dee518: Already exists
74b270a96ee8: Already exists
Digest: sha256:6e089049d8ea3c609d2e1cf0ac99f604e64d901187856051387859b8d74d45cf
Status: Downloaded newer image for brahimhamdi/rng:v1
f2420c57adf790e8b5895b0b214ea7bf710959df81175974c451483aa2853b8c
vagrant@Manager:~/Lab4/dockercoins$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
f2420c57adf7	brahimhamdi/rng:v1	"python rng.py"	8 seconds ago	Up 7 seconds	0.0.0.0:8001->80/tcp	rng
ab2d12bcf49f	brahimhamdi/worker:v1	"python worker.py"	42 seconds ago	Up 41 seconds		worker

```
vagrant@Manager:~/Lab4/dockercoins$
```