Play with Docker in virtual lab

Task 1: Run some simple docker containers

I.Run a single task in an Alpine Linux container

[node1] (local) root@192.168.0.8 ~

$ **git clone https://github.com/dockersamples/linux_tweet_app**

Cloning into 'linux_tweet_app'...

remote: Enumerating objects: 14, done.

remote: Total 14 (delta 0), reused 0 (delta 0), pack-reused 14

Receiving objects: 100% (14/14), 10.76 KiB | 10.76 MiB/s, done.

Resolving deltas: 100% (5/5), done.

[node1] (local) root@192.168.0.8 ~

$ **docker container run alpine hostname**

Unable to find image 'alpine:latest' locally

latest: Pulling from library/alpine

59bf1c3509f3: Pull complete

Digest: sha256:21a3deaa0d32a8057914f36584b5288d2e5ecc984380bc0118285c70fa8c9300

Status: Downloaded newer image for alpine:latest

8eeaeb998e0f

[node1] (local) root@192.168.0.8 ~

$ **docker container ls --all**

CONTAINER ID   IMAGE   COMMAND   CREATED   STATUS   PORTS   NAMES

8eeaeb998e0f   alpine   "hostname"   4 seconds ago   Exited (0) 3seconds ago   practical_blackburn

II.Run an interactive Ubuntu container

[node1] (local) root@192.168.0.8 ~

$ **docker container run --interactive --tty --rm ubuntu bash**

Unable to find image 'ubuntu:latest' locally

latest: Pulling from library/ubuntu

ea362f368469: Pull complete

Digest: sha256:b5a61709a9a44284d88fb12e5c48db0409cfad5b69d4ff8224077c57302df9cf

Status: Downloaded newer image for ubuntu:latest

root@cba5ddba7b86:/# **ls /**

bin dev home lib32 libx32 mnt proc run srv tmp var

boot etc lib lib64 media opt root sbin sys usr

root@cba5ddba7b86:/# **ps aux**

USER   PID %CPU %MEM   VSZ   RSS TTY   STAT START   TIME COMMAND

root   1 0.5 0.0 4108 3580 pts/0   Ss   07:16   0:00 bash

root   10 0.0 0.0 5896 3012 pts/0   R+   07:16   0:00 ps aux

root@cba5ddba7b86:/# **cat /etc/issue**

Ubuntu 20.04.3 LTS \n \l

root@cba5ddba7b86:/# **exit**

exit

[node1] (local) root@192.168.0.8 ~

$ **cat /etc/issue**

Welcome to Alpine Linux 3.12

Kernel \r on an \m (\l)

III.Run a background MySQL container

[node1] (local) root@192.168.0.8 ~

**$ docker container run **

**> --detach **

**> --name mydb **

**> -e MYSQL_ROOT_PASSWORD=my-secret-pw **

> mysql:latest

Unable to find image 'mysql:latest' locally

latest: Pulling from library/mysql

72a69066d2fe: Pull complete

93619dbc5b36: Pull complete

99da31dd6142: Pull complete

e5c707858ec0: Pull complete

fc41578cbf60: Pull complete

4785d896ef10: Pull complete

7d250cdc93be: Pull complete

309700f41983: Pull complete

45fd33301836: Pull complete

92f970c68b71: Pull complete

bb3544339a9e: Pull complete

f66ddf4c43fa: Pull complete

Digest:
sha256:d0507b008897c39f6cbc76285af1171d4551988475e00e91344060023cd9c55
3

Status: Downloaded newer image for mysql:latest

d050b2b9595ab4cdb62e2223ed4cd96fa17dcb3d13c02088efd452d1faa31e5c

[node1] (local) root@192.168.0.8 ~

$ **docker container ls**

CONTAINER ID   IMAGE   COMMAND   CREATED   STATUS
PORTS   NAMES

d050b2b9595a   mysql:latest   "docker-entrypoint.s..."   5 seconds ago   Up 4 seconds
3306/tcp, 33060/tcp   mydb

[node1] (local) root@192.168.0.8 ~

$ **docker container logs mydb**

2022-01-20 07:16:55+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL
Server 8.0.28-1debian10 started.

2022-01-20 07:16:55+00:00 [Note] [Entrypoint]: Switching to dedicated user
'mysql'

2022-01-20 07:16:55+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL
Server 8.0.28-1debian10 started.

2022-01-20 07:16:55+00:00 [Note] [Entrypoint]: Initializing database files

2022-01-20T07:16:55.858185Z 0 [System] [MY-013169] [Server] /usr/sbin/mysqld
(mysqld 8.0.28) initializing of server in progress as process 42

2022-01-20T07:16:55.872760Z 1 [System] [MY-013576] [InnoDB]
InnoDBinitialization has started.

2022-01-20T07:16:56.289965Z 1 [System] [MY-013577] [InnoDB] InnoDBinitialization has ended.

2022-01-20T07:16:57.758125Z 6 [Warning] [MY-010453] [Server] root@localhost is created with an empty password ! Please consider switching off the --initialize-insecure option.

2022-01-20 07:17:01+00:00 [Note] [Entrypoint]: Database files initialized

2022-01-20 07:17:01+00:00 [Note] [Entrypoint]: Starting temporary server

2022-01-20T07:17:01.878990Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 8.0.28) starting as process 91

2022-01-20T07:17:01.911302Z 1 [System] [MY-013576] [InnoDB] InnoDBinitialization has started.

2022-01-20T07:17:02.139892Z 1 [System] [MY-013577] [InnoDB] InnoDBinitialization has ended.

2022-01-20T07:17:02.443868Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.

2022-01-20T07:17:02.444139Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encrypted connections are now supported for this channel.

2022-01-20T07:17:02.445703Z 0 [Warning] [MY-011810] [Server] Insecure configuration for --pid-file: Location '/var/run/mysqld' in thepath is accessible to all OS users. Consider choosing a different directory.

2022-01-20T07:17:02.479054Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Socket: /var/run/mysqld/mysqlx.sock

2022-01-20T07:17:02.479376Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8.0.28' socket: '/var/run/mysqld/mysqld.sock' port: 0 MySQL Community Server - GPL.

2022-01-20 07:17:02+00:00 [Note] [Entrypoint]: Temporary server started.

[node1] (local) root@192.168.0.8 ~

$ **docker container top mydb**

PID   USER   TIME   COMMAND

1106   999   0:00   mysqld

[node1] (local) root@192.168.0.8 ~

**$ docker exec -it mydb **

> mysql --user=root --password=$MYSQL_ROOT_PASSWORD --version

mysql: [Warning] Using a password on the command line interface can be insecure.

mysql Ver 8.0.28 for Linux on x86_64 (MySQL Community Server - GPL)

[node1] (local) root@192.168.0.8 ~

$ docker exec -it mydb sh

 # mysql --user=root --password=$MYSQL_ROOT_PASSWORD --version

mysql: [Warning] Using a password on the command line interface can be insecure.

mysql Ver 8.0.28 for Linux on x86_64 (MySQL Community Server - GPL)

 # exit

Task 2: Package and run a custom app using Docker

Build a simple website image

[node1] (local) root@192.168.0.8 ~

$ cd ~/linux_tweet_app

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ cat Dockerfile

FROM nginx:latest

COPY index.html /usr/share/nginx/html

COPY linux.png /usr/share/nginx/html

EXPOSE 80 443

CMD ["nginx", "-g", "daemon off;"]

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ echo neethushanavas

neethushanavas

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ docker image build --tag neethushanavas/linux_tweet_app:1.0 .

Sending build context to Docker daemon 92.16kB

Step 1/5 : FROM nginx:latest

latest: Pulling from library/nginx

a2abf6c4d29d: Pull complete

a9edb18cadd1: Pull complete

589b7251471a: Pull complete

186b1aaa4aa6: Pull complete

b4df32aa5a72: Pull complete

a0bcbecc962e: Pull complete

Digest:
sha256:0d17b565c37bcbd895e9d92315a05c1c3c9a29f762b011a10c54a66cd53c9b31

Status: Downloaded newer image for nginx:latest

 ---> 605c77e624dd

Step 2/5 : COPY index.html /usr/share/nginx/html

 ---> cd6a44d0bba6

Step 3/5 : COPY linux.png /usr/share/nginx/html

 ---> 2be9c42ddf7c

Step 4/5 : EXPOSE 80 443

 ---> Running in b4e4def76d49

Removing intermediate container b4e4def76d49

 ---> d29157e692b1

Step 5/5 : CMD ["nginx", "-g", "daemon off;"]

 ---> Running in 37c7f4e66852

Removing intermediate container 37c7f4e66852

 ---> 5cd6e29d4b9b

Successfully built 5cd6e29d4b9b

Successfully tagged neethushanavas/linux_tweet_app:1.0

[node1] (local) root@192.168.0.8 ~/linux_tweet_app
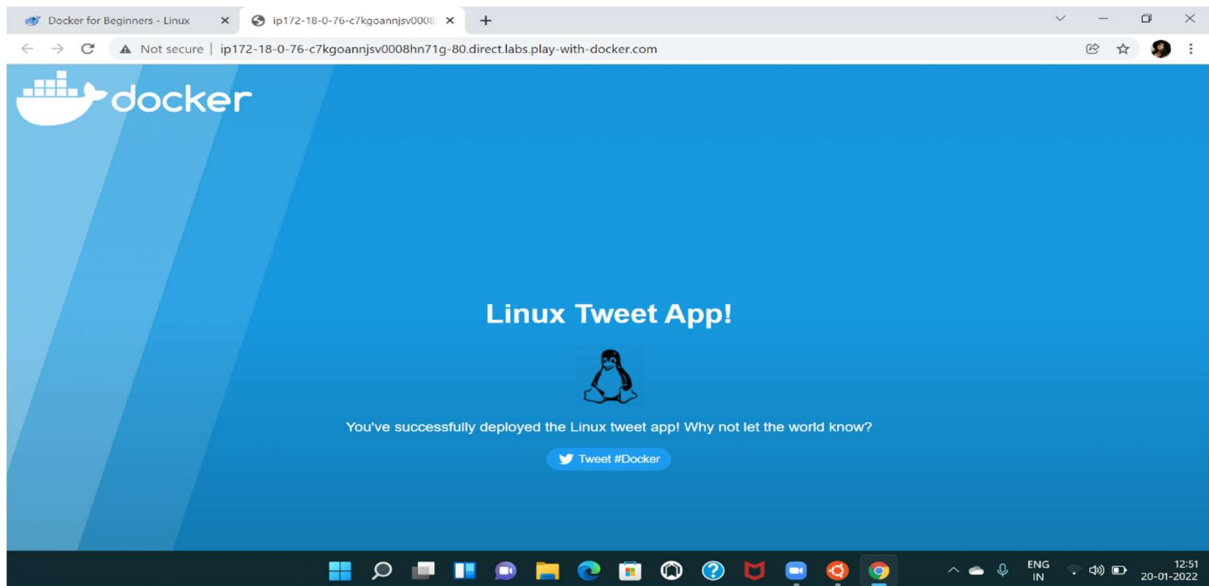
$ **docker container run **

> **--detach **

> **--publish 80:80 **

> **--name linux_tweet_app **

> **neethushanavas/linux_tweet_app:1.0**

a961feba236a4a4bb9d510b585f6aebf2358c969887529af3be2e8c2e4a17d0f

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ **docker container rm --force linux_tweet_app**
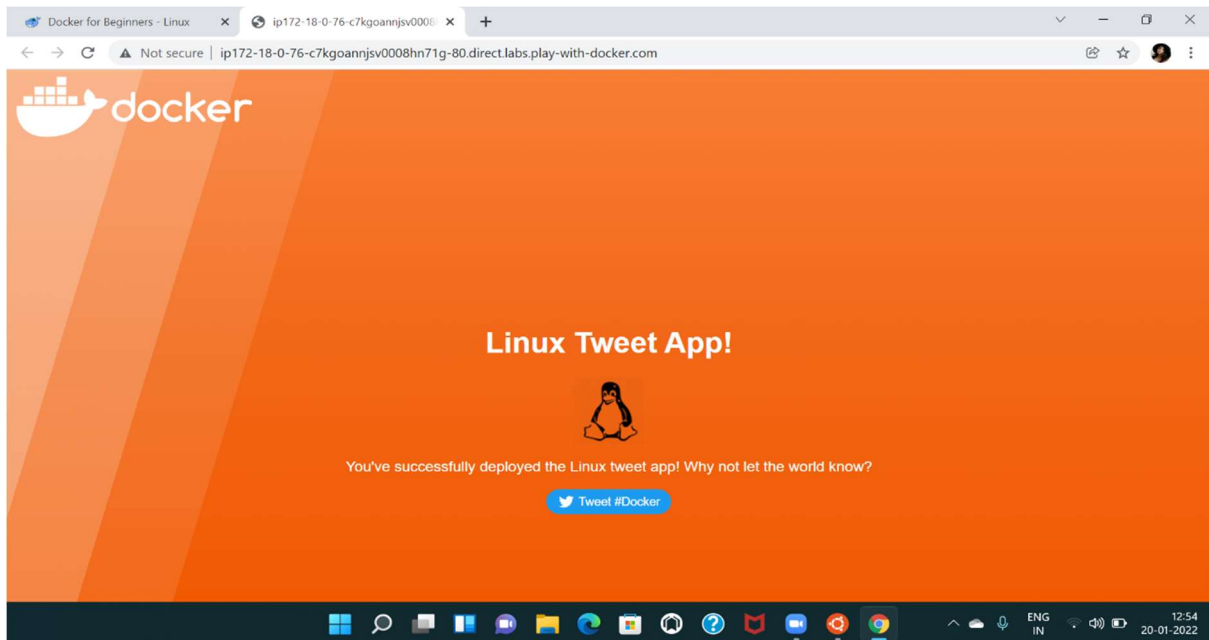
linux_tweet_app

Task3: Modify a running website

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

**$ docker container run **

**> --detach **

**> --publish 80:80 **

**> --name linux_tweet_app **

**> --mount type=bind,source="$(pwd)",target=/usr/share/nginx/html **

> neethushanavas/linux_tweet_app:1.0

f8b3a3b3ee8a3246b00d91afa9d5ade8bf8de17ab2475c1d12089d1501f6eec4

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ **cp index-new.html index.html**

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ **docker rm --force linux_tweet_app**

linux_tweet_app

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ **docker container run **

> **--detach **

> **--publish 80:80 **

> **--name linux_tweet_app **

> **neethushanavas/linux_tweet_app:1.0**

f842dd77a4ecd666c4849cd5633b43399aea7553e2a6c853202aed443213fd32

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ **docker rm --force linux_tweet_app**

linux_tweet_app

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ **docker image build --tag neethushanavas/linux_tweet_app:2.0 .**

Sending build context to Docker daemon 92.16kB

Step 1/5 : FROM nginx:latest

 ---> 605c77e624dd

Step 2/5 : COPY index.html /usr/share/nginx/html

 ---> 15ccbcd225fe

Step 3/5 : COPY linux.png /usr/share/nginx/html

 ---> 7107f915048f

Step 4/5 : EXPOSE 80 443

 ---> Running in e7781b1296f4

Removing intermediate container e7781b1296f4

 ---> 22442303f1d4

Step 5/5 : CMD ["nginx", "-g", "daemon off;"]

 ---> Running in 6724f1c4c229

Removing intermediate container 6724f1c4c229

 ---> eb0887159207

Successfully built eb0887159207

Successfully tagged neethushanavas/linux_tweet_app:2.0

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ **docker image ls**

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
neethushanavas/linux_tweet_app	2.0	eb0887159207	8 seconds ago	142MB
neethushanavas/linux_tweet_app	1.0	5cd6e29d4b9b	8 minutes ago	142MB
mysql	latest	5b4c624c7fe1	3 hours ago	519MB
ubuntu	latest	d13c942271d6	13 days ago	72.8MB
nginx	latest	605c77e624dd	3 weeks ago	141MB
alpine	latest	c059bfaa849c	8 weeks ago	5.59MB

[node1] (local) root@192.168.0.8 ~/linux_tweet_app
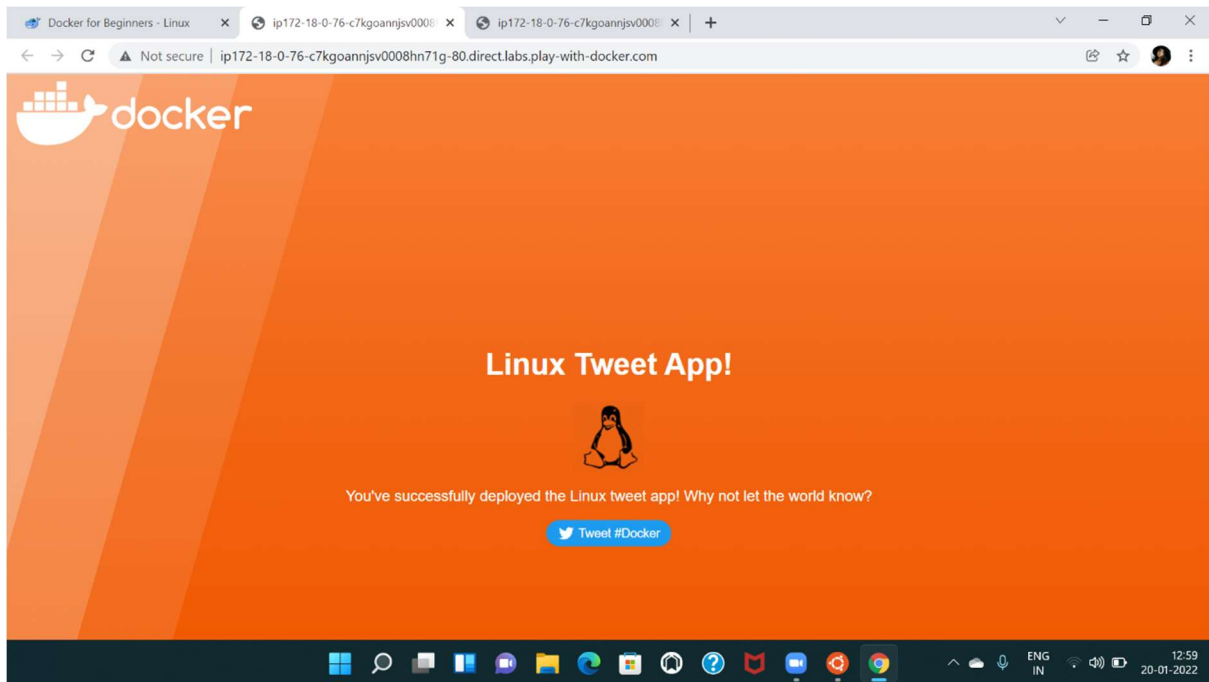
$ **docker container run **

> **--detach **

> **--publish 80:80 **

> **--name linux_tweet_app **

> **neethushanavas/linux_tweet_app:2.0**

ff40bfde754a48e8dc8b7e9645cbdbf18f1cc20b5c48e625b1886d4486683bc1

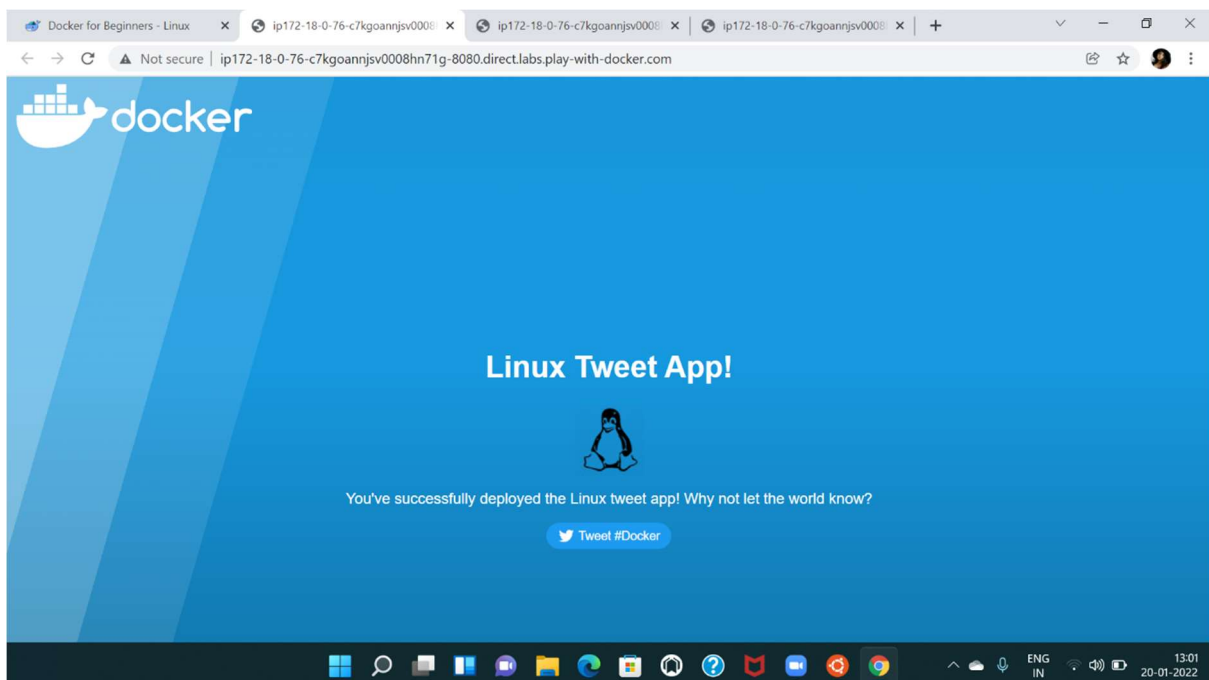[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ **docker container run **

> **--detach **

> **--publish 8080:80 **

> **--name old_linux_tweet_app **

> **neethushanavas/linux_tweet_app:1.0**

54c624cc8fa89572ec80ba84794182e6699b09920321355b940c2687f9b124ee

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ **docker image ls -f reference="neethushanavas/*"**

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
neethushanavas/linux_tweet_app	2.0	eb0887159207	5 minutes ago	142MB
neethushanavas/linux_tweet_app	1.0	5cd6e29d4b9b	13 minutes ago	142MB

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ **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.comto create one.

Username: neethushanavas

Password:

WARNING! Your password will be stored unencrypted in /root/.docker/config.json.

Configure a credential helper to remove this warning. See

https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ **docker image push neethushanavas/linux_tweet_app:1.0**

The push refers to repository [docker.io/neethushanavas/linux_tweet_app]

0a95c4775782: Pushed

6aec4374de4d: Pushed

d874fd2bc83b: Mounted from library/nginx

32ce5f6a5106: Mounted from library/nginx

f1db227348d0: Mounted from library/nginx

b8d6e692a25e: Mounted from library/nginx

e379e8aedd4d: Mounted from library/nginx

2edcec3590a4: Layer already exists

1.0: digest: sha256:968809d3e84d3850af3563300162a145dadab7c1c396c65926cb864704410010 size: 1985

[node1] (local) root@192.168.0.8 ~/linux_tweet_app

$ **docker image push neethushanavas/linux_tweet_app:2.0**

The push refers to repository [docker.io/neethushanavas/linux_tweet_app]

0a77456ada90: Pushed

7b3f6d6bee43: Pushed

d874fd2bc83b: Layer already exists

32ce5f6a5106: Layer already exists

f1db227348d0: Layer already exists

b8d6e692a25e: Layer already exists

e379e8aedd4d: Layer already exists

2edcec3590a4: Layer already exists

2.0: digest:
sha256:50547d06639d123aa58469554d15557c05442b2bbcc7a58d0bc0484813b07d
34 size: 1985

Application Containerization and Microservice Orchestration

I.Basic Link extractor script

[node1] (local) root@192.168.0.8 ~

$ git clone https://github.com/ibnesayeed/linkextractor.git

Cloning into 'linkextractor'...

remote: Enumerating objects: 144, done.

remote: Counting objects: 100% (4/4), done.

remote: Compressing objects: 100% (4/4), done.

remote: Total 144 (delta 0), reused 1 (delta 0), pack-reused 140

Receiving objects: 100% (144/144), 44.55 KiB | 5.57 MiB/s, done.

Resolving deltas: 100% (43/43), done.

[node1] (local) root@192.168.0.8 ~

$ cd linkextractor

[node1] (local) root@192.168.0.8 ~/linkextractor

$ git checkout demo

Branch 'demo' set up to track remote branch 'demo' from 'origin'.

Switched to a new branch 'demo'

[node1] (local) root@192.168.0.8 ~/linkextractor

$ git checkout step0

Branch 'step0' set up to track remote branch 'step0' from 'origin'.

Switched to a new branch 'step0'

[node1] (local) root@192.168.0.8 ~/linkextractor

$ **tree**

.

├── README.md

└── linkextractor.py

0 directories, 2 files

[node1] (local) root@192.168.0.8 ~/linkextractor

$ **cat linkextractor.py**

#!/usr/bin/env python

import sys

import requests

from bs4 import BeautifulSoup

res = requests.get(sys.argv[-1])

soup = BeautifulSoup(res.text, "html.parser")

for link in soup.find_all("a"):

   print(link.get("href"))

[node1] (local) root@192.168.0.8 ~/linkextractor

$ ./linkextractor.py http://example.com/

bash: ./linkextractor.py: Permission denied

[node1] (local) root@192.168.0.8 ~/linkextractor

$ **ls -l linkextractor.py**

-rw-r--r--   1 root   root   220 Jan 20 08:50 linkextractor.py

[node1] (local) root@192.168.0.8 ~/linkextractor

$ **python3 linkextractor.py**

Traceback (most recent call last):

  File "linkextractor.py", line 5, in <module>

   from bs4 import BeautifulSoup

ModuleNotFoundError: No module named 'bs4'

II.Containerised link extractor script

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **git checkout step1**

Branch 'step1' set up to track remote branch 'step1' from 'origin'.

Switched to a new branch 'step1'

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **tree**

.

├── Dockerfile

├── README.md

└── linkextractor.py

0 directories, 3 files

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **cat Dockerfile**

FROM   python:3

LABEL   maintainer="Sawood Alam <@ibnesayeed>"

RUN   pip install beautifulsoup4

RUN   pip install requests

WORKDIR   /app

COPY   linkextractor.py /app/

RUN   chmod a+x linkextractor.py

ENTRYPOINT ["./linkextractor.py"]

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker image build -t linkextractor:step1 .**

Sending build context to Docker daemon 107.5kB

Step 1/8 : FROM   python:3

3: Pulling from library/python

0e29546d541c: Pull complete

9b829c73b52b: Pull complete

cb5b7ae36172: Pull complete

6494e4811622: Pull complete

6f9f74896dfa: Pull complete

fcb6d5f7c986: Pull complete

3445db4c939c: Pull complete

def920d3ef5d: Pull complete

aabf25a1ee4b: Pull complete

Digest: sha256:a780eec51b47c0684d81315d48524b92e84873f93558f863733333e558d628b6

Successfully built bf00dcf797fa

Successfully tagged linkextractor:step1

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker image ls**

REPOSITORY   TAG   IMAGE ID   CREATED   SIZE

linkextractor   step1   bf00dcf797fa   4 seconds ago   927MB

python   3   cecf555903c6   37 hours ago   917MB

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker container run -it --rm linkextractor:step1 http://example.com/**

https://www.iana.org/domains/example

[node1] (local) root@192.168.0.28 ~/linkextractor

$ docker container run -it --rm linkextractor:step1 https://training.play-with-docker.com/

/

/about/

#ops

#dev

/ops-stage1

/ops-stage2

/ops-stage3

/dev-stage1

/dev-stage2

/dev-stage3

/alacart

https://twitter.com/intent/tweet?text=Play with Docker Classroom&url=https://training.play-with-docker.com/&via=docker&related=docker

https://facebook.com/sharer.php?u=https://training.play-with-docker.com/

https://plus.google.com/share?url=https://training.play-with-docker.com/

http://www.linkedin.com/shareArticle?mini=true&url=https://training.play-with-docker.com/&title=Play%20with%20Docker%20Classroom&source=https://training.play-with-docker.com

https://www.docker.com/dockercon/

https://www.docker.com/dockercon/

https://dockr.ly/slack

https://www.docker.com/legal/docker-terms-service

https://www.docker.com

https://www.facebook.com/docker.run

https://twitter.com/docker

https://www.github.com/play-with-docker/play-with-docker.github.io

III.Link extractor module with full url and anchor text

[node1] (local) root@192.168.0.28 ~/linkextractor

$ git checkout step2

Branch 'step2' set up to track remote branch 'step2' from 'origin'.

Switched to a new branch 'step2'

[node1] (local) root@192.168.0.28 ~/linkextractor

$ tree

.

├── Dockerfile

├── README.md

└── linkextractor.py

0 directories, 3 files

[node1] (local) root@192.168.0.28 ~/linkextractor

$ docker image build -t linkextractor:step2 .

Sending build context to Docker daemon 110.1kB

Step 1/8 : FROM   python:3

 ---> cecf555903c6

Step 2/8 : LABEL   maintainer="Sawood Alam <@ibnesayeed>"

 ---> Using cache

 ---> c331121749e2

Step 3/8 : RUN   pip install beautifulsoup4

 ---> Using cache

 ---> a39e63b3c35d

Step 4/8 : RUN   pip install requests

 ---> Using cache

 ---> b4b4b6e21710

Step 5/8 : WORKDIR   /app

 ---> Using cache

 ---> 64e4dac1df76

Step 6/8 : COPY   linkextractor.py /app/

 ---> 9b0295b4f3d2

Step 7/8 : RUN   chmod a+x linkextractor.py

 ---> Running in 0f81cef4a658

Removing intermediate container 0f81cef4a658

 ---> edb8ba03a62b

Step 8/8 : ENTRYPOINT ["./linkextractor.py"]

 ---> Running in 69c64a5b453d

Removing intermediate container 69c64a5b453d

 ---> 0bd5747091af

Successfully built 0bd5747091af

Successfully tagged linkextractor:step2

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker image ls**

REPOSITORY   TAG   IMAGE ID   CREATED   SIZE

linkextractor   step2   0bd5747091af   37 seconds ago   927MB

linkextractor   step1   bf00dcf797fa   17 minutes ago   927MB

python   3   cecf555903c6   38 hours ago   917MB

[node1] (local) root@192.168.0.28 ~/linkextractor

$ docker container run -it --rm linkextractor:step2 https://training.play-with-docker.com/

[Play with Docker classroom](https://training.play-with-docker.com/)

[About](https://training.play-with-docker.com/about/)

[IT Pros and System Administrators](https://training.play-with-docker.com/#ops)

[Developers](https://training.play-with-docker.com/#dev)

[Stage 1: The Basics](https://training.play-with-docker.com/ops-stage1)

[Stage 2: Digging Deeper](https://training.play-with-docker.com/ops-stage2)

[Stage 3: Moving to Production](https://training.play-with-docker.com/ops-stage3)

[Stage 1: The Basics](https://training.play-with-docker.com/dev-stage1)

[Stage 2: Digging Deeper](https://training.play-with-docker.com/dev-stage2)

[Stage 3: Moving to Staging](https://training.play-with-docker.com/dev-stage3)

[Full list of individual labs](https://training.play-with-docker.com/alacart)

[[IMG]](https://twitter.com/intent/tweet?text=Play with Docker Classroom&url=https://training.play-with-docker.com/&via=docker&related=docker)

[[IMG]](https://facebook.com/sharer.php?u=https://training.play-with-docker.com/)

[[IMG]](https://plus.google.com/share?url=https://training.play-with-docker.com/)

[[IMG]](http://www.linkedin.com/shareArticle?mini=true&url=https://training.play-with-docker.com/&title=Play%20with%20Docker%20Classroom&source=https://training.play-with-docker.com)

[[IMG]](https://www.docker.com/dockercon/)

[Sign up today](https://www.docker.com/dockercon/)

[Register here](https://dockr.ly/slack)

[here](https://www.docker.com/legal/docker-terms-service)

[[IMG]](https://www.docker.com)

[[IMG]](https://www.facebook.com/docker.run)

[[IMG]](https://twitter.com/docker)

[[IMG]](https://www.github.com/play-with-docker/play-with-docker.github.io)

[node1] (local) root@192.168.0.28 ~/linkextractor

$ docker container run -it --rm linkextractor:step1 https://training.play-with-docker.com/

/

/about/

#ops

#dev

/ops-stage1

/ops-stage2

/ops-stage3

/dev-stage1

/dev-stage2

/dev-stage3

/alacart

https://twitter.com/intent/tweet?text=Play with Docker Classroom&url=https://training.play-with-docker.com/&via=docker&related=docker

https://facebook.com/sharer.php?u=https://training.play-with-docker.com/

https://plus.google.com/share?url=https://training.play-with-docker.com/

http://www.linkedin.com/shareArticle?mini=true&url=https://training.play-with-docker.com/&title=Play%20with%20Docker%20Classroom&source=https://training.play-with-docker.com

https://www.docker.com/dockercon/

https://www.docker.com/dockercon/

https://dockr.ly/slack

https://www.docker.com/legal/docker-terms-service

https://www.docker.com

https://www.facebook.com/docker.run

https://twitter.com/docker

https://www.github.com/play-with-docker/play-with-docker.github.io

IV. Link extractor API Service

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **git checkout step3**

Branch 'step3' set up to track remote branch 'step3' from 'origin'.

Switched to a new branch 'step3'

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **tree**

.

├── Dockerfile

├── README.md

├── linkextractor.py

├── main.py

└── requirements.txt

0 directories, 5 files

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **cat Dockerfile**

FROM   python:3

LABEL   maintainer="Sawood Alam <@ibnesayeed>"

WORKDIR   /app

COPY   requirements.txt /app/

RUN   pip install -r requirements.txt

COPY   *.py /app/

RUN   chmod a+x *.py

CMD   ["./main.py"]

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker image build -t linkextractor:step3 .**

Sending build context to Docker daemon 115.2kB

Successfully built 4bd6f73e0c5e

Successfully tagged linkextractor:step3

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker container run -d -p 5000:5000 --name=linkextractor linkextractor:step3**

5f42db6dc2212823b2a78656df767c4042d504ae067a4154e756f18c0dcaf51d

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker container ls**

CONTAINER ID   IMAGE   COMMAND   CREATED STATUS   PORTS
NAMES

5f42db6dc221   linkextractor:step3   "./main.py"   20 seconds ago Up 18 seconds
0.0.0.0:5000->5000/tcp   linkextractor

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **curl -i http://localhost:5000/api/http://example.com/**

HTTP/1.0 200 OK

Content-Type: application/json

Content-Length: 79

Server: Werkzeug/2.0.2 Python/3.10.2

Date: Thu, 20 Jan 2022 09:45:37 GMT

[{"href":"https://www.iana.org/domains/example","text":"More information..."}]

[node1] (local) root@192.168.0.28 ~/linkextractor

$ docker container logs linkextractor

 * Serving Flask app 'main' (lazy loading)

 * Environment: production

   WARNING: This is a development server. Do not use it in a production
deployment.

   Use a production WSGI server instead.

 * Debug mode: off

 * Running on all addresses.

   WARNING: This is a development server. Do not use it in a production
deployment.

 * Running on http://172.17.0.2:5000/ (Press CTRL+C to quit)

172.17.0.1 - - [20/Jan/2022 09:45:37] "GET /api/http://example.com/ HTTP/1.1"
200 –

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker container rm -f linkextractor**

V.Link Extractor API and Web Front services

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **git checkout step4**

Branch 'step4' set up to track remote branch 'step4' from 'origin'.

Switched to a new branch 'step4'

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **tree**

```
.
├── README.md
├── api
│   ├── Dockerfile
│   ├── linkextractor.py
│   ├── main.py
│   └── requirements.txt
├── docker-compose.yml
└── www
    └── index.php
```

2 directories, 7 files

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **cat docker-compose.yml**

```
version: '3'
services:
  api:
    image: linkextractor-api:step4-python
    build: ./api
    ports:
      - "5000:5000"
  web:
```

```
  image: php:7-apache
  ports:
    - "80:80"
  environment:
    - API_ENDPOINT=http://api:5000/api/
  volumes:
    - ./www:/var/www/html
```

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker-compose up -d --build**

Creating network "linkextractor_default" with the default driver

Building api

Digest: sha256:729ad01c7d8e10fd992a6d4f3eb05dce3fb69bdf5c4fb4a9de4be4f4f5ae4dcc

Status: Downloaded newer image for php:7-apache

Creating linkextractor_web_1 ... done

Creating linkextractor_api_1 ... done

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker container ls**

CONTAINER ID IMAGE   COMMAND   CREATED   STATUS
PORTS   NAMES

ff1433ed7a41 php:7-apache   "docker-php-entrypoi..." 49 seconds ago
Up 46 seconds 0.0.0.0:80->80/tcp   linkextractor_web_1

26f8bc603b49 linkextractor-api:step4-python "./main.py"   49 seconds ago Up
47 seconds 0.0.0.0:5000->5000/tcp linkextractor_api_1

[node1] (local) root@192.168.0.28 ~/linkextractor

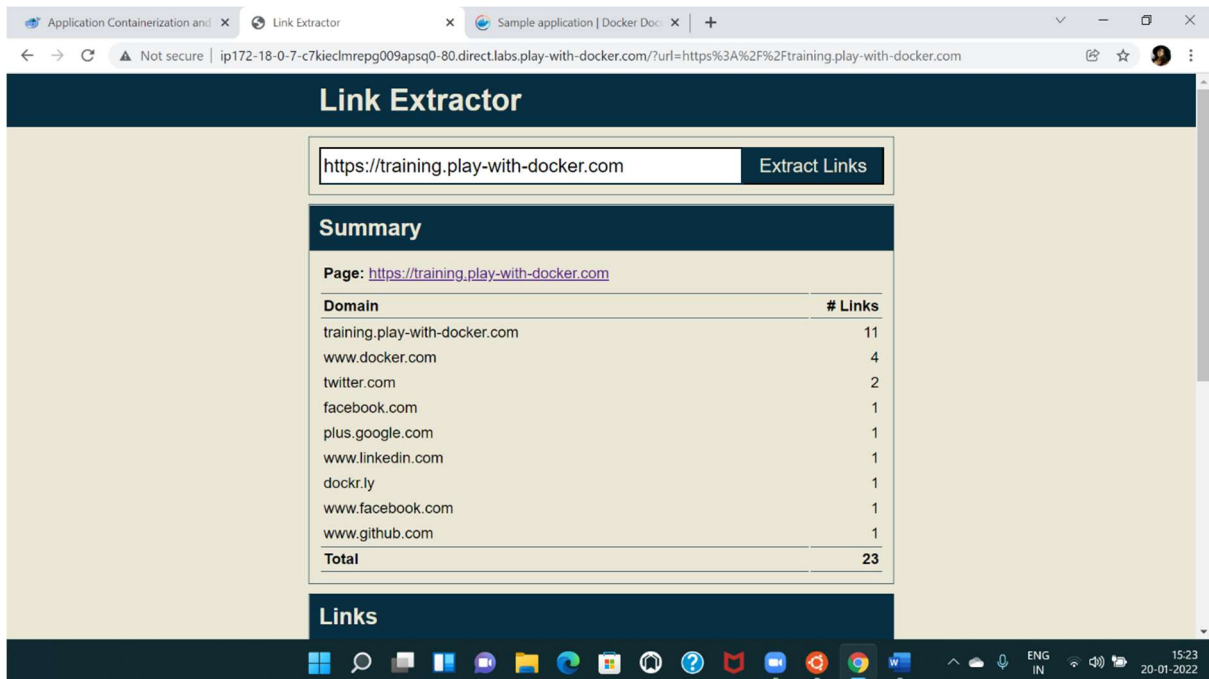$ **curl -i http://localhost:5000/api/http://example.com/**

HTTP/1.0 200 OK

Content-Type: application/json

Content-Length: 79
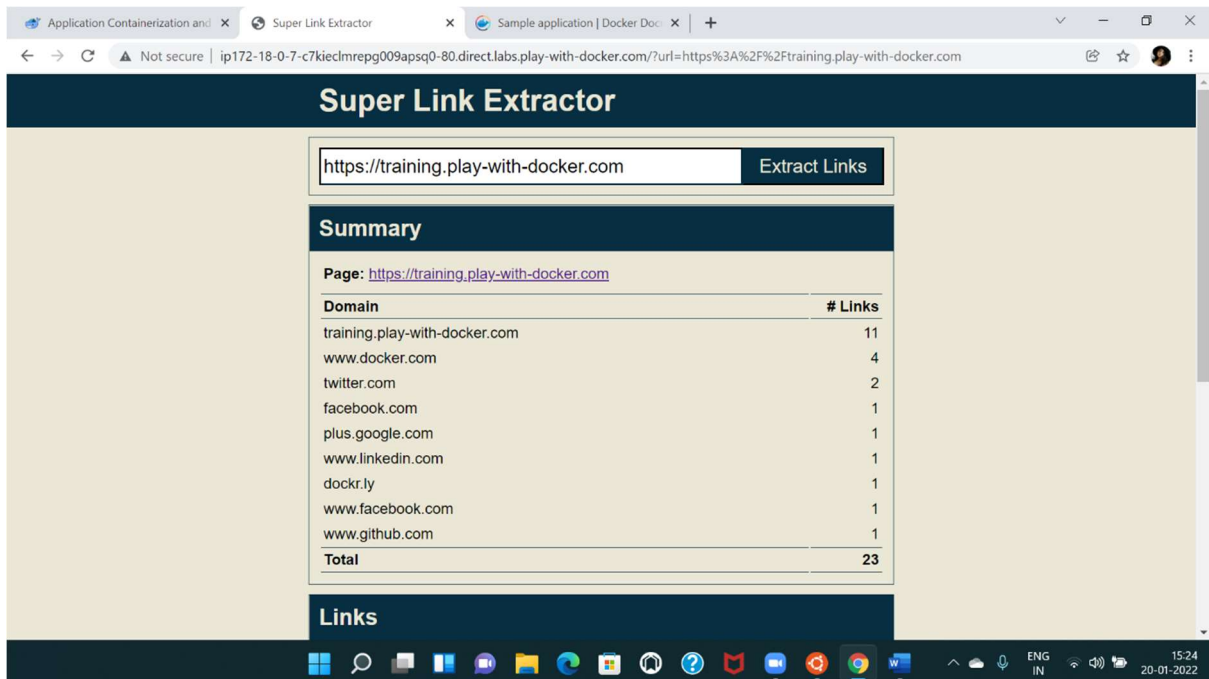
Server: Werkzeug/2.0.2 Python/3.10.2

Date: Thu, 20 Jan 2022 09:52:05 GMT

[{"href":"https://www.iana.org/domains/example","text":"More information..."}]

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **sed -i 's/Link Extractor/Super Link Extractor/g' www/index.php**

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **git reset --hard**

HEAD is now at 2a3ec3e Synchronize branch step4

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker-compose down**

Stopping linkextractor_web_1 ... done

Stopping linkextractor_api_1 ... done

Removing linkextractor_web_1 ... done

Removing linkextractor_api_1 ... done

Removing network linkextractor_default

VI.Redis Service for caching

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **git checkout step5**

Branch 'step5' set up to track remote branch 'step5' from 'origin'.

Switched to a new branch 'step5'

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **tree**

```
.
├── README.md
├── api
│   ├── Dockerfile
│   ├── linkextractor.py
│   ├── main.py
│   └── requirements.txt
├── docker-compose.yml
└── www
    ├── Dockerfile
    └── index.php
```

2 directories, 8 files

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker-compose up -d --build**

Creating network "linkextractor_default" with the default driver
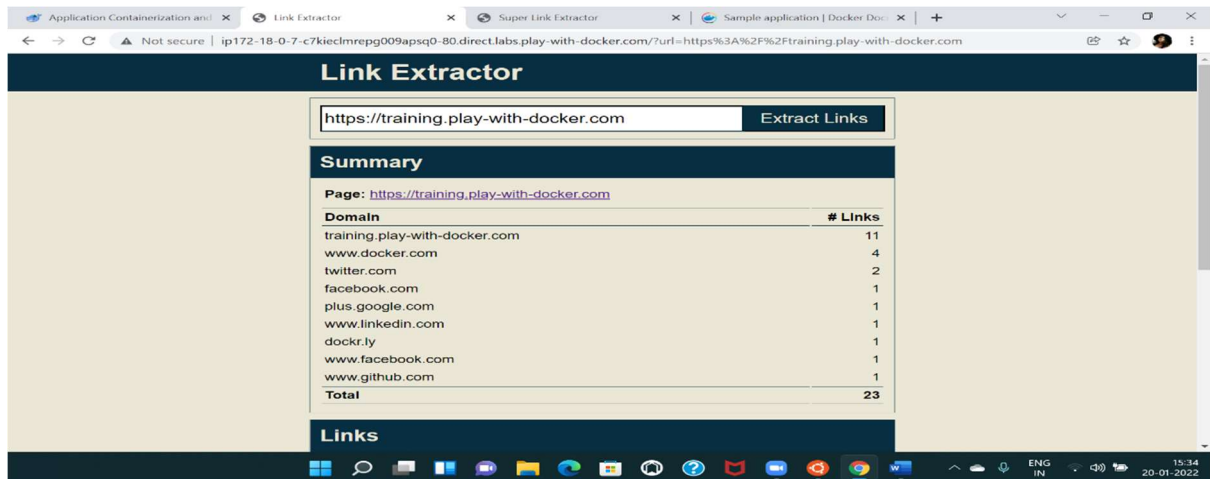
Building api

Digest: sha256:db485f2e245b5b3329fdc7eff4eb00f913e09d8feb9ca720788059fdc2ed8339

Status: Downloaded newer image for redis:latest

Creating linkextractor_api_1   ... done

Creating linkextractor_web_1   ... done

Creating linkextractor_redis_1 ... done

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker-compose exec redis redis-cli monitor**

OK

^C[node1] (local) root@192.168.0.28 ~/linkextractor

$ **sed -i 's/Link Extractor/Super Link Extractor/g' www/index.php**

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **git reset --hard**

HEAD is now at 3dbb7eb Synchronize branch step5

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker-compose down**

Stopping linkextractor_redis_1 ... done

Stopping linkextractor_web_1   ... done

Stopping linkextractor_api_1   ... done

Removing linkextractor_redis_1 ... done

Removing linkextractor_web_1   ... done

Removing linkextractor_api_1   ... done

Removing network linkextractor_default

VII.Swap python API service with Ruby

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **git checkout step6**

Already on 'step6'

Your branch is up to date with 'origin/step6'.

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **tree**

```
.
├── README.md
├── api
│   ├── Dockerfile
│   ├── Gemfile
│   └── linkextractor.rb
├── docker-compose.yml
├── logs
└── www
    ├── Dockerfile
    └── index.php
```

3 directories, 7 files

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker-compose up -d --build**

Building api

Successfully built b1f79a46150c

Successfully tagged linkextractor-web:step6-php

Creating linkextractor_api_1   ... done

Creating linkextractor_web_1   ... done

Creating linkextractor_redis_1 ... done

[node1] (local) root@192.168.0.28 ~/linkextractor

$ **curl -i http://localhost:4567/api/http://example.com/**

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: 97

X-Content-Type-Options: nosniff

Server: WEBrick/1.4.4 (Ruby/2.6.9/2021-11-24)

Date: Thu, 20 Jan 2022 10:13:03 GMT

Connection: Keep-Alive


```
[
 {
   "text": "More information...",
   "href": "https://www.iana.org/domains/example"
 }
```


[node1] (local) root@192.168.0.28 ~/linkextractor

$ **docker-compose down**

Stopping linkextractor_redis_1 ... done

Stopping linkextractor_web_1   ... done

Stopping linkextractor_api_1   ... done

Removing linkextractor_redis_1 ... done

Removing linkextractor_web_1   ... done

Removing linkextractor_api_1   ... done

Removing network linkextractor_default

[node1] (local) root@192.168.0.28 ~/linkextractor

```
$ cat logs/extraction.log

1642673583    MISS   http://example.com/

1642673642    MISS   https://training.play-with-docker.com

1642673669    MISS   https://facebook.com
```