120A3051

Shreya Idate

Batch: E3

Experiment No: 8

AIM: To understand Docker Architecture and Container Life Cycle, install Docker and execute docker commands to manage images and interact with containers.

THEORY: Docker Engine is an open source containerization technology for building and containerizing your applications. Docker Engine acts as a client-server application with:

- A server with a long-running daemon process dockerd.
- APIs which specify interfaces that programs can use to talk to and instruct the Docker daemon.
- A command line interface (CLI) client docker.

The CLI uses Docker APIs to control or interact with the Docker daemon through scripting or direct CLI commands. Many other Docker applications use the underlying API and CLI. The daemon creates and manage Docker objects, such as images, containers, networks, and volumes..

Installation of Docker:

To get started with Docker Engine on Ubuntu, make sure you meet the prerequisites, and then install Docker.

Prerequisites: OS requirements

To install Docker Engine, you need the 64-bit version of one of these Ubuntu versions:

- Ubuntu Hirsute 21.04
- Ubuntu Focal 20.04 (LTS)
- Ubuntu Bionic 18.04 (LTS)

Installation methods: You can install Docker Engine in different ways, depending on your needs:

1. Most users set up Docker's repositories and install from them
2. Some users download the DEB package and install it manually and manage upgrades completely manually.
3. In testing and development environments, some users choose to use automated convenience scripts to install Docker

Install using the convenience script: Docker provides a convenience script at get.docker.com to install Docker into development environments quickly and non-interactively. This example downloads the script from get.docker.com and runs it to install the latest stable release of Docker on Linux:

```
$ curl -fsSL https://get.docker.com -o get-docker.sh
$ sudo sh get-docker.sh
```

To get OS detail and version

it77@it77-OptiPlex-3050 :~$ lsb_release -a

Uninstall old versions

it77@it77-OptiPlex-3050 :~$ sudo su

t77@it77-OptiPlex-305i0 :~$ sudo apt-get remove docker docker-engine docker.io containerd runc

it77@it77-OptiPlex-3050 :~$ sudo apt install curl

root@it77-OptiPlex-3050:/home/it77# curl -fsSL https://get.docker.com -o get-docker.sh

Examine scripts downloaded from the internet

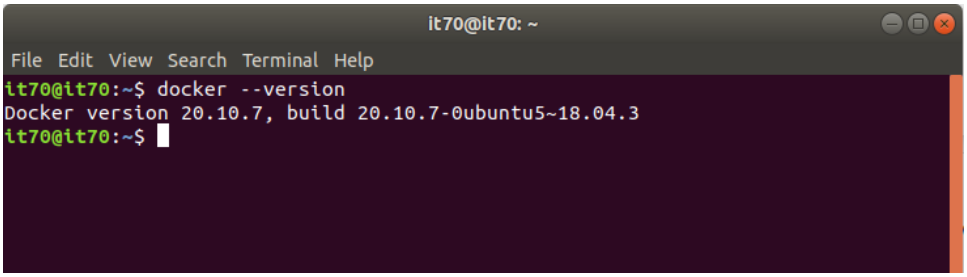
root@it77-OptiPlex-3050:/home/it77# ls

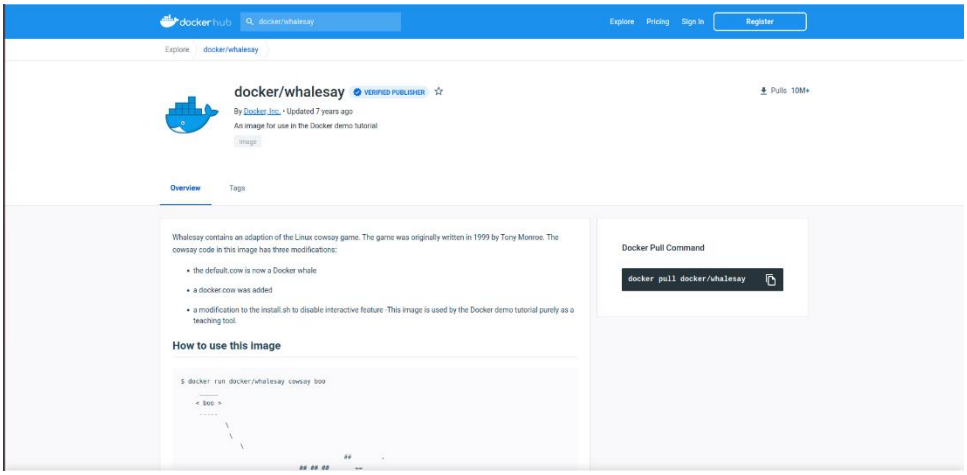
root@it77-OptiPlex-3050:/home/it77# sudo sh get-docker.sh

Basic Docker Commands:

Check the version of Docker installed

root@it77-OptiPlex-3050:/home/it77# docker –version



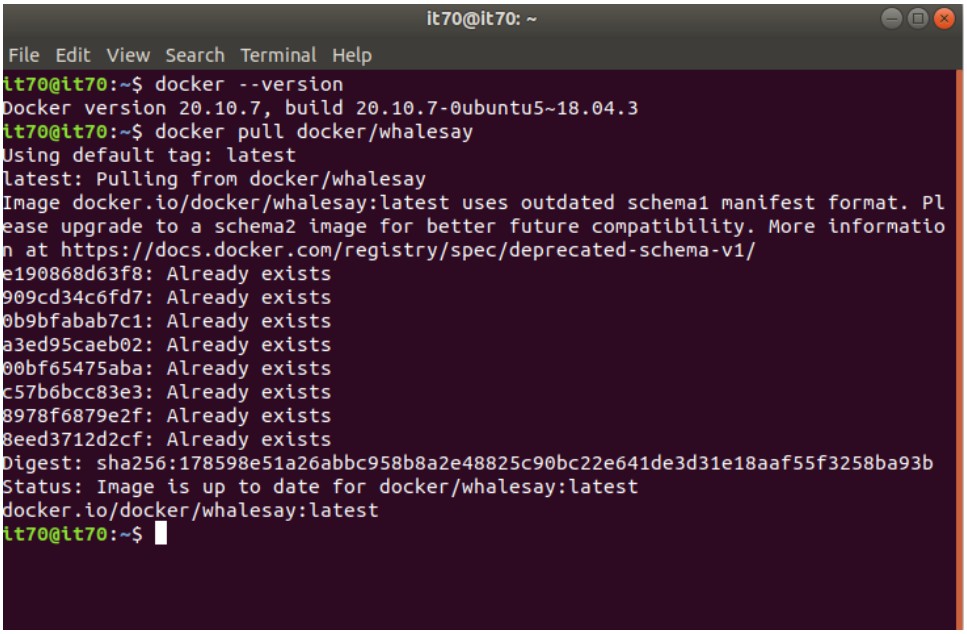


Running existing Docker images: Go to Docker public repository at https://hub.docker.com to get the official images available for testing purpose

Run docker image

root@it77-OptiPlex-3050:/home/it77# docker run docker/whalesay cowsay hello_you

root@it77-OptiPlex-3050:/home/it77# docker run docker/whalesay cowsay hello_me



```
it70@it70:~$ docker run docker/whalesay cowsay hello_you
 ___________
< hello_you >
 -----------
    \
     \
      \
                    ##        .
              ## ## ##       ==
           ## ## ## ##      ===
       /""""""""""""""""___/ ===
  ~~~ {~~ ~~~~ ~~~ ~~~~ ~~ ~ /  ===- ~~~
       \______ o          __/
        \    \        __/
          \____\______/
```

Check all pulled images

root@it77-OptiPlex-3050:/home/it77# docker images

Pull the sample images

root@it77-OptiPlex-3050:/home/it77# sudo docker pull postgres

root@it77-OptiPlex-3050:/home/it77# docker images

Check all running container

root@it77-OptiPlex-3050:/home/it77# docker ps // **note the container id**

root@it77-OptiPlex-3050:/home/it77# docker ps -a //**previously ran containers**

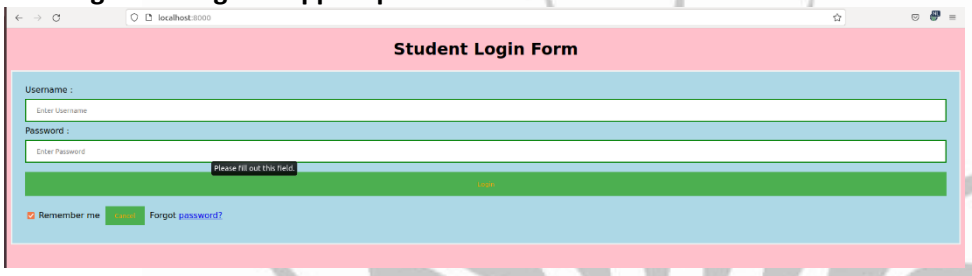
Pulling the created webapp project:

```
it70@it70:~$ docker pull bushsk/ngnix_webapp:v1
v1: Pulling from bushsk/ngnix_webapp
Digest: sha256:3c5f1755233d50bb1b02852386cd173d263bfe3ad646de5a0c5e35a104b73563
Status: Image is up to date for bushsk/ngnix_webapp:v1
docker.io/bushsk/ngnix_webapp:v1
```

Running the pulled image

```
it70@it70:~$ docker run -itd -p 8000:80 --name shreyanewestwebapp bushsk/ngnix_webapp:v1

de491ec4013f5b48ad048b2faa47116d6fc9fbba075443e5dbc1e69f71c588c5
it70@it70:~$
```

Running or testing webapp on port 8989:




```
it70@it70:~$ docker stop de4
de4
it70@it70:~$
```

Creating a new image from the running containers:

Pull the Ubuntu as a base image :

root@it77-OptiPlex-3050:/home/it77# docker pull ubuntu:latest

root@it77-OptiPlex-3050:/home/it77# docker images

```
root@puppet-agent:/home/it72# lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 20.04.4 LTS
Release:        20.04
Codename:       focal
```

```
root@puppet-agent:/home/it72# docker pull ubuntu:focal
focal: Pulling from library/ubuntu
fb0b3276a519: Pull complete
Digest: sha256:9c2004872a3a9fcec8cc757ad65c042de1dad4da27de4c70739a6e36402213e3
Status: Downloaded newer image for ubuntu:focal
docker.io/library/ubuntu:focal
```

```
root@puppet-agent:/home/it72# docker images
REPOSITORY                TAG               IMAGE ID       CREATED         SIZ
E
bash                      latest            9306da3708d9   3 days ago      13.
3MB
ubuntu                    focal             817578334b4d   5 days ago      72.
8MB
siesngnixservers          latest            d2f64f78f73e   7 days ago      171
MB
<none>                    <none>            eda4c3c72bbd   7 days ago      171
MB
sakshi123/myapp           v1                b47bacea66f2   11 days ago     171
MB
siesnginxservers          latest            720cd6ce6779   12 days ago     171
MB
<none>                    <none>            016948e25983   12 days ago     170
MB
img                       latest            da4873af6886   2 weeks ago     171
MB
```

Run the Ubuntu image with a command in a container: Getting a bash in Ubuntu
root@it77-OptiPlex-3050:/home/it77# docker run -it ubuntu:latest bash
root@it77-OptiPlex-3050:/home/it77# docker ps

```
root@puppet-agent:/home/it72# docker run -it ubuntu:focal bash
root@5a9309ce7f23:/# docker ps
bash: docker: command not found
root@5a9309ce7f23:/# apt update
Get:1 http://archive.ubuntu.com/ubuntu focal InRelease [265 kB]
Get:2 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:3 http://security.ubuntu.com/ubuntu focal-security/universe amd64 Packages [
916 kB]
Get:4 http://archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:5 http://archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
```

Note the <12 digit hash value> is the id of the shell.

Create an Apache Server and host index.html in the Containers

root@67e9bd16d77b:/# apt update

root@67e9bd16d77b:/# apt install apache2

root@67e9bd16d77b:/# cd

```
root@5a9309ce7f23:/# apt install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils ca-certificates file krb5-locales
  libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap
  libasn1-8-heimdal libbrotli1 libcurl4 libexpat1 libgdbm-compat4 libgdbm6
  libgssapi-krb5-2 libgssapi3-heimdal libhcrypto4-heimdal libheimbase1-heimdal
  libheimntlm0-heimdal libhx509-5-heimdal libicu66 libjansson4 libk5crypto3
```

```
it70@it70:~$ docker ps
CONTAINER ID   IMAGE                              COMMAND   CREATED          STAT
US          PORTS                                 NAMES
dbd607ee35c1   ubuntu:bionic                      "bash"    4 minutes ago    Up 4
 minutes                                          stoic_vaughan
a5c7adf67e3d   bushsk/dockerized_webapp:latest    "bash"    28 minutes ago   Up 2
8 minutes   0.0.0.0:8080->80/tcp, :::8080->80/tcp   shreyamywebapp
9f21c8d72cbe   bushsk/dockerized_webapp           "bash"    32 minutes ago   Up 3
2 minutes   0.0.0.0:8989->80/tcp, :::8989->80/tcp   mywebapp
it70@it70:~$
```

root@67e9bd16d77b:/var/www/html# nano index.html
root@67e9bd16d77b:/var/www/html# cat index.html

```
root@5a9309ce7f23:/# cd /var/www/html
root@5a9309ce7f23:/var/www/html# mv index.html index.backup
root@5a9309ce7f23:/var/www/html# ls
index.backup
root@5a9309ce7f23:/var/www/html# nano index.html
bash: nano: command not found
root@5a9309ce7f23:/var/www/html# apt install nano
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

```
root@5a9309ce7f23:/var/www/html# nano index.html
root@5a9309ce7f23:/var/www/html# cat index.html
<html>
<title> First Page </title>
<body bgcolor="pink">
OUR HOME PAGE
</body>
</html>
root@5a9309ce7f23:/var/www/html#
```

<title> First page</title>

<body bgcolor="pink">

Our home Page

</body>

</html>

root@67e9bd16d77b:/var/www/html# service apache2 start

root@67e9bd16d77b:/var/www/html# service apache2 status

```
root@5a9309ce7f23:/var/www/html# service apache2 start
 * Starting Apache httpd web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified doma
in name, using 172.17.0.3. Set the 'ServerName' directive globally to suppress t
his message
 *
root@5a9309ce7f23:/var/www/html# service apache2 status
 * apache2 is running
root@5a9309ce7f23:/var/www/html#
```

Committing an image:

Open a new terminal and run the foll. Commands:

Tagging <Ubuntu_apache:v1> image using image id:

root@it77-OptiPlex-3050:/home/it77# docker tag 27941809078c bushsk/ubuntu_apache:v1

27941809078c is the image id of the running container

```
root@puppet-agent:/home/it72# docker images
REPOSITORY                  TAG                 IMAGE ID       CREATED         SIZ
E
bash                        latest              9306da3708d9   3 days ago      13.
3MB
ubuntu                      focal               817578334b4d   5 days ago      72.
8MB
siesngnixservers            latest              d2f64f78f73e   7 days ago      171
MB
<none>                      <none>              eda4c3c72bbd   7 days ago      171
MB
sakshi123/myapp             v1                  b47bacea66f2   11 days ago     171
MB
siesnginxservers            latest              720cd6ce6779   12 days ago     171
MB
```

Checking its size and committing an image

```
root@puppet-agent:/home/it72# docker tag 817578334b4d shreya/ubuntu_apache:v1
root@puppet-agent:/home/it72# docker images
REPOSITORY                  TAG                 IMAGE ID       CREATED         SIZE
bash                        latest              9306da3708d9   3 days ago      13.3MB
bushsk/ubuntu_apache        v1                  817578334b4d   5 days ago      72.8MB
ubuntu                      focal               817578334b4d   5 days ago      72.8MB
shreya/ubuntu_apache        v1                  817578334b4d   5 days ago      72.8MB
siesngnixservers            latest              d2f64f78f73e   7 days ago      171MB
<none>                      <none>              eda4c3c72bbd   7 days ago      171MB
sakshi123/myapp             v1                  b47bacea66f2   11 days ago     171MB
```

Checking the size of committed image

```
root@puppet-agent:/home/it72# docker images
REPOSITORY                  TAG                 IMAGE ID       CREATED         SIZE
bash                        latest              9306da3708d9   3 days ago      13.3MB
bushsk/ubuntu_apache        v1                  817578334b4d   5 days ago      72.8MB
ubuntu                      focal               817578334b4d   5 days ago      72.8MB
shreya/ubuntu_apache        v1                  817578334b4d   5 days ago      72.8MB
siesngnixservers            latest              d2f64f78f73e   7 days ago      171MB
```

Allowing port 8888

```
root@puppet-agent:/home/it72# ufw allow 8888
Skipping adding existing rule
Skipping adding existing rule (v6)
root@puppet-agent:/home/it72#
```

Running a named <mywebsite> image Ubuntu_apache:v1 image and note container id

```
root@puppet-agent:/home/it72# docker exec -it 909c78fa06a4 service apache2 restart
```

Restarting an apache by attaching it into running containers using container id:

```
root@puppet-agent:/home/it72# ufw allow 8888
root@puppet-agent:/home/it72# docker run -itd -p 8888:80 --name shreyawebsite shreya/ubuntu_apache:v1
909c78fa06a4deeb36c8355342aead4b5adcff4877414f199dfb6dd5ad9a346b
root@puppet-agent:/home/it72#
```

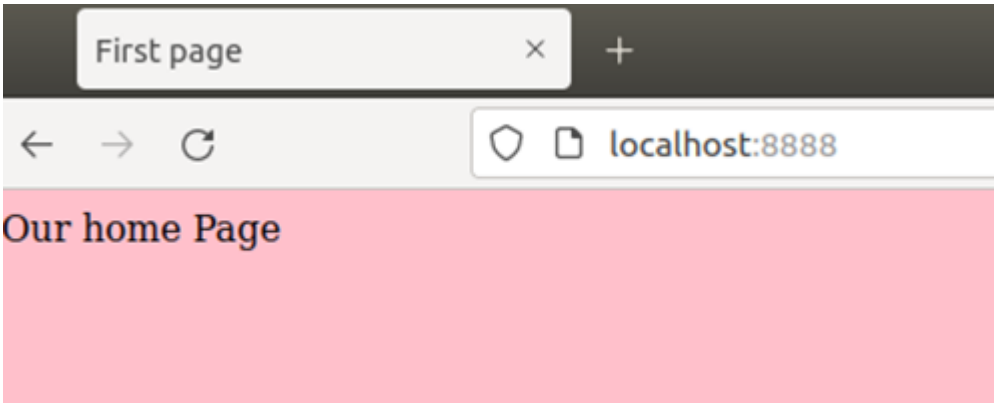
```
root@puppet-agent:/home/it72# docker ps
CONTAINER ID   IMAGE                     COMMAND                  CREATED              STATUS              PORTS                                       NAMES
909c78fa06a4   shreya/ubuntu_apache:v1   "nginx -g 'daemon of…"   About a minute ago   Up About a minute   0.0.0.0:8888->80/tcp, :::8888->80/tcp       shreyawebsite
5a9309ce7f23   ubuntu:focal              "bash"                   46 minutes ago       Up 46 minutes                                           awesome_dirac
43cd0e993d3d   bushsk/ngnix_webapp:v1    "nginx -g 'daemon of…"   55 minutes ago       Up 55 minutes       0.0.0.0:8989->80/tcp, :::8989->80/tcp       shreyangnixwebapp
root@puppet-agent:/home/it72#
```

Get the IP address of your system

root@it77-OptiPlex-3050:/home/it77# ifconfig

Test your web application:

Open a browser and put <IP: port number> (as 8888) or type localhost:8888



Login in hub

root@it77-OptiPlex-3050:/home/it77# docker login

```
root@puppet-agent:/home/it72# 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: shreya2806
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
```

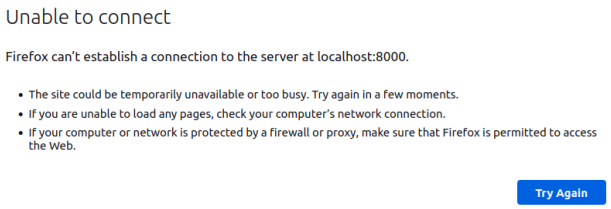
Pushing the image on Docker Hub

```
root@puppet-agent:/home/it72# docker push shreya/ubuntu_apache:v1
The push refers to repository [docker.io/shreya/ubuntu_apache]
d7cd17e156ea: Preparing
72ee6f0c8ac3: Preparing
f433e2c03768: Preparing
0cafcec33316: Preparing
822b0ae954c7: Preparing
4942a1abcbfa: Waiting
```

Stopping and removing containers using container id

root@it77-OptiPlex-3050:/home/it77# docker stop bd9fdf66daaf

```
it70@it70:~$ docker stop de4
de4
it70@it70:~$
```



root@it77-OptiPlex-3050:/home/it77# docker kill bd9fdf66daaf -- > not advisable

root@it77-OptiPlex-3050:/home/it77# docker rm bd9fdf66daaf

root@it77-OptiPlex-3050:/home/it77# docker ps -a

Remove multiple containers using container ids:

root@it77-OptiPlex-3050:/home/it77# docker rm bd9 abc

root@it77-OptiPlex-3050:/home/it77# docker rm $(docker ps –aq)

root@it77-OptiPlex-3050:/home/it77# docker ps -a

Deleting the images

root@it77-OptiPlex-3050:/home/it77# docker images

root@it77-OptiPlex-3050:/home/it77# docker rmi 5c6

Creating an image using docker file script:

DOCKER FILE

```
root@puppet-master:/etc/puppet/code/environments/production/modules/lamp/manifests# mkdir dockertutorial
root@puppet-master:/etc/puppet/code/environments/production/modules/lamp/manifests# cd dockertutorial
root@puppet-master:/etc/puppet/code/environments/production/modules/lamp/manifests/dockertutorial# nano index.html
```

root@it77-OptiPlex-3050:/home/it77# mkdir dockertutorial

root@it77-OptiPlex-3050:/home/it77# cd dockertutorial

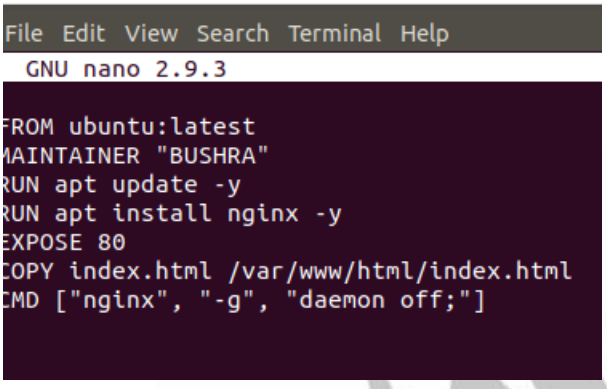
root@it77-OptiPlex-3050:/home/it77/dockertutorial# pwd

/home/it77/dockertutorial

root@it77-OptiPlex-3050:/home/it77/dockertutorial# nano index.html

root@it77-OptiPlex-3050:/home/it77/dockertutorial# ls

index.html



root@it77-OptiPlex-3050:/home/it77/dockertutorial# nano Dockerfile

FROM ubuntu:latest

MAINTAINER "BUSHRA"

RUN apt update -y

RUN apt install nginx -y

EXPOSE 80

COPY index.html /var/www/html/index.html

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

root@it77-OptiPlex-3050:/home/it77/dockertutorial# docker build -t siesnginxservers .

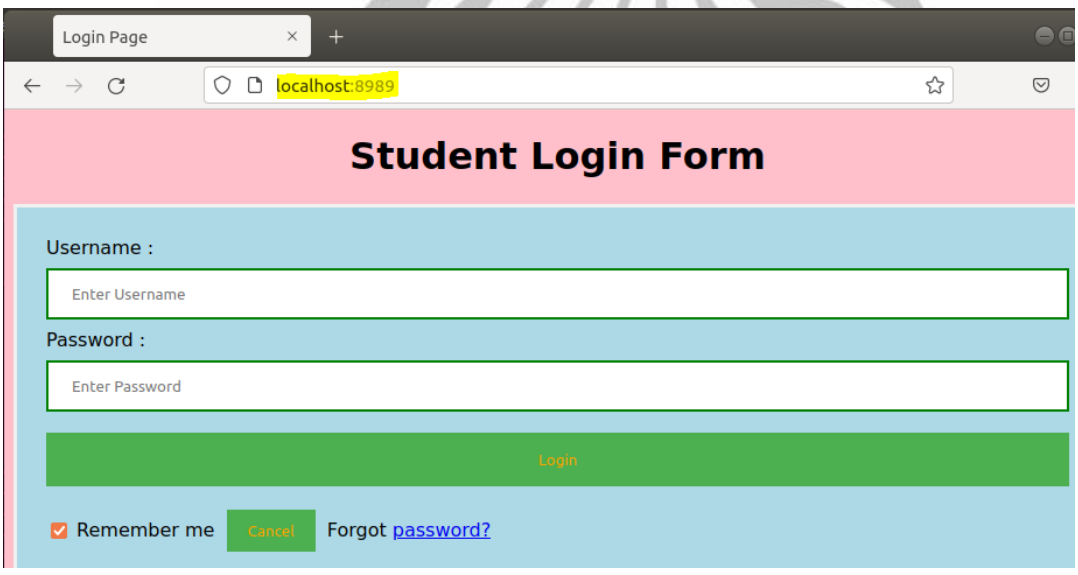
```
root@puppet-master:/etc/puppet/code/environments/production/modules/lamp/manifests/dockertutorial# docker build -t siesnginxservers .
Sending build context to Docker daemon  4.608kB
Step 1/7 : FROM ubuntu:latest
 ---> 27941809078c
Step 2/7 : MAINTAINER "BUSHRA"
 ---> Running in 961afab4cc50
Removing intermediate container 961afab4cc50
 ---> ed788c7e6ee5
Step 3/7 : RUN apt update -y
 ---> Running in c35a68d0ed38

WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy InRelease [270 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [238 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [94.9 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [212 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [109 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [4648 B]
Get:8 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [99.8 kB]
Get:9 http://archive.ubuntu.com/ubuntu jammy/main amd64 Packages [1792 kB]
Get:10 http://archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [17.5 MB]
Get:11 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [266 kB]
Get:12 http://archive.ubuntu.com/ubuntu jammy/restricted amd64 Packages [164 kB]
Get:13 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [4648 B]
Get:14 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [406 kB]
Get:15 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [162 kB]
Get:16 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [250 kB]
Get:17 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [5797 B]
Fetched 21.7 MB in 1min 13s (297 kB/s)
Reading package lists...
Building dependency tree...
Reading state information...
8 packages can be upgraded. Run 'apt list --upgradable' to see them.
Removing intermediate container c35a68d0ed38
 ---> 7781dce475ea
Step 4/7 : RUN apt install nginx -y
 ---> Running in 97d1caad1907
```

root@it77-OptiPlex-3050:/home/it77/dockertutorial# docker run -itd -p 8989:80 siesnginxservers

```
root@puppet-master:/etc/puppet/code/environments/production/modules/lamp/manifests/dockertutorial# docker run -itd -p 8989:80 siesnginxservers
fe691607ec5d38d948e63bd5380e5b0a4e9c57c6f3faaaab64dca1e7943c5ede
```

root@it77-OptiPlex-3050:/home/it77/dockertutorial# ufw allow 8989 //if required since already allowed

Open browser and put ip with port number as 8989



Conclusion: We are able to understand the concept of docker and have successfully pushed the repository on docker hub.