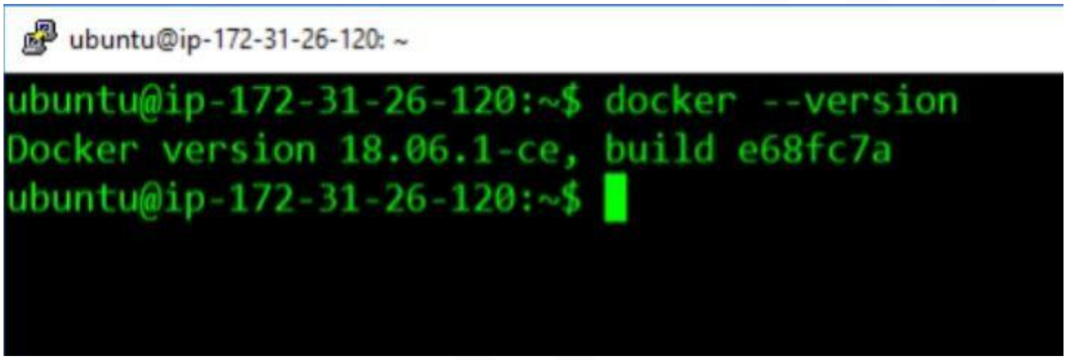
Common Docker Commands

CommonDockerCommands:

1. This commandhelpsyouknowtheinstalledversionoftheDockersoftware on your system

```
$ docker --version
```

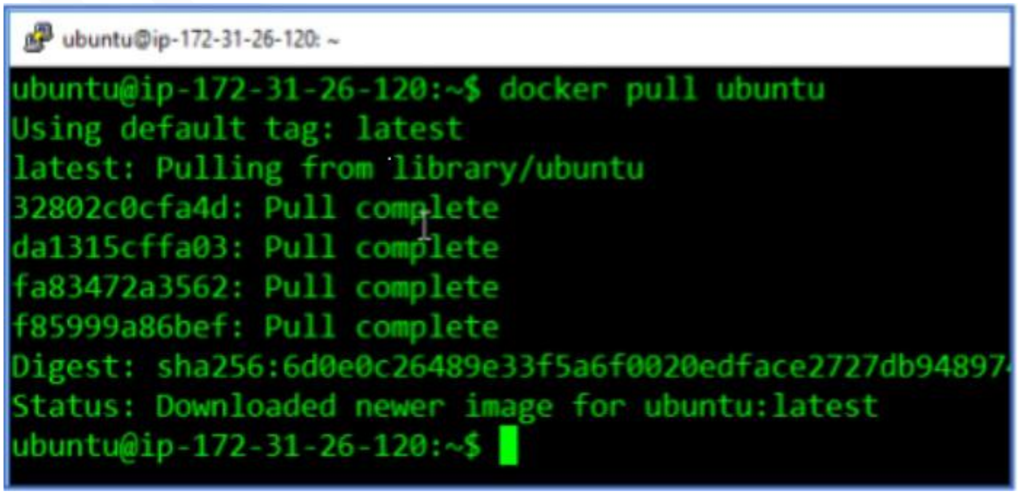


ubuntu@ip-172-31-26-120: ~

```
ubuntu@ip-172-31-26-120:~$ docker --version
Docker version 18.06.1-ce, build e68fc7a
ubuntu@ip-172-31-26-120:~$
```

2. ThiscommandhelpsyoupullimagesfromthecentralDockerrepository

```
$ docker pull <image-name>
```

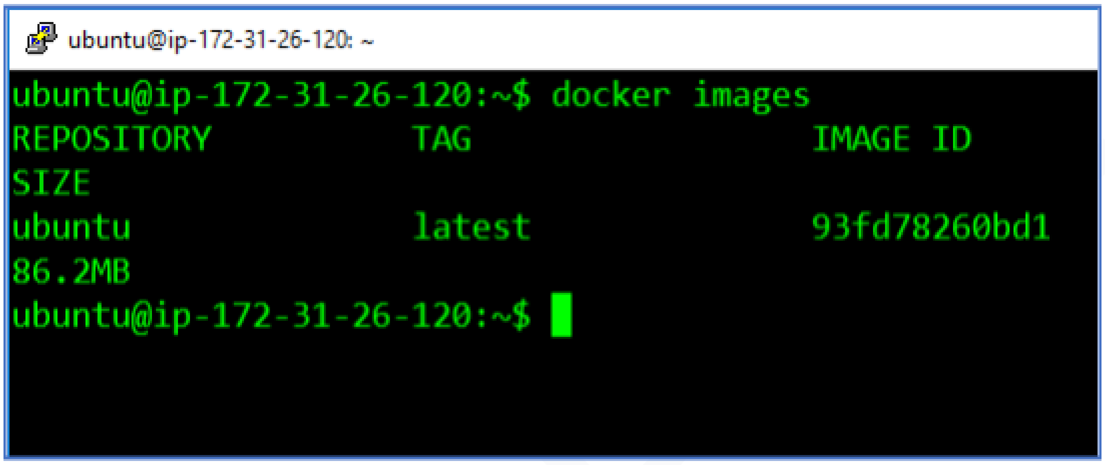


ubuntu@ip-172-31-26-120: ~

```
ubuntu@ip-172-31-26-120:~$ docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
32802c0cfa4d: Pull complete
da1315cffa03: Pull complete
fa83472a3562: Pull complete
f85999a86bef: Pull complete
Digest: sha256:6d0e0c26489e33f5a6f0020edface2727db94897
Status: Downloaded newer image for ubuntu:latest
ubuntu@ip-172-31-26-120:~$
```

3. This command helps you in listing all the Docker images downloaded onyour system

```
$ docker images
```

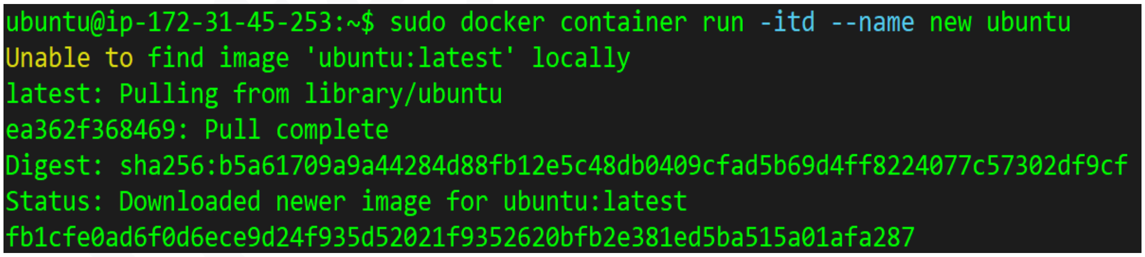


ubuntu@ip-172-31-26-120: ~

```
ubuntu@ip-172-31-26-120:~$ docker images
REPOSITORY          TAG                 IMAGE ID
SIZE
ubuntu              latest              93fd78260bd1
86.2MB
ubuntu@ip-172-31-26-120:~$
```

4. Thiscommandhelpsinrunningcontainersfromtheirimagename

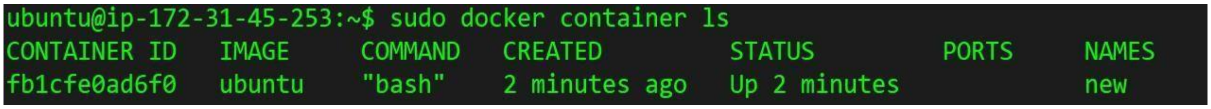
```
$ docker container run <image-name>
```



```
ubuntu@ip-172-31-45-253:~$ sudo docker container run -itd --name new ubuntu
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
fb1cfe0ad6f0d6ece9d24f935d52021f9352620bfb2e381ed5ba515a01afa287
```

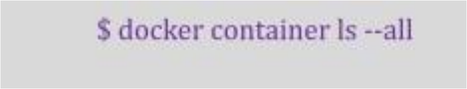
5. This command helps in listing all the containers which are running in thesystem

```
$ docker container ls
```

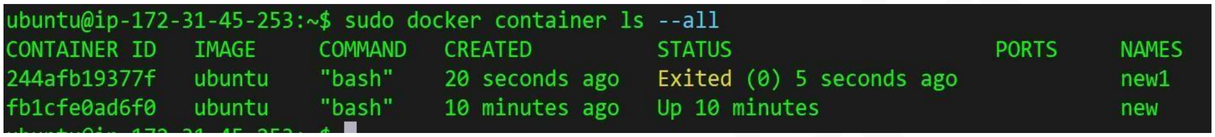


```
ubuntu@ip-172-31-45-253:~$ sudo docker container ls
CONTAINER ID   IMAGE     COMMAND    CREATED          STATUS          PORTS     NAMES
fb1cfe0ad6f0   ubuntu    "bash"     2 minutes ago    Up 2 minutes              new
```

6. If there are any stopped containers, they can be seen by adding the“--all” flag in this command

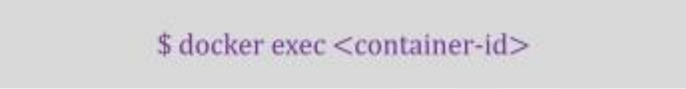


```
$ docker container ls --all
```

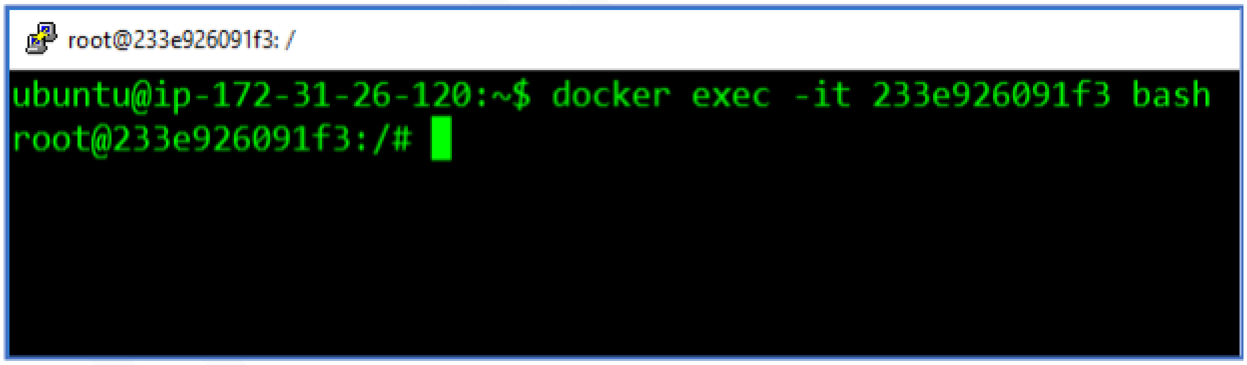


```
ubuntu@ip-172-31-45-253:~$ sudo docker container ls --all
CONTAINER ID   IMAGE    COMMAND   CREATED          STATUS                    PORTS     NAMES
244afb19377f   ubuntu   "bash"    20 seconds ago   Exited (0) 5 seconds ago            new1
fb1cfe0ad6f0   ubuntu   "bash"    10 minutes ago   Up 10 minutes                       new
```

7. Forlogginginto/accessingthecontainer,onecanusetheexeccommand



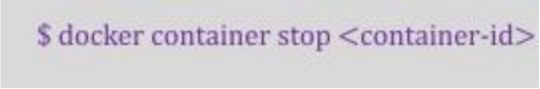
```
$ docker exec <container-id>
```



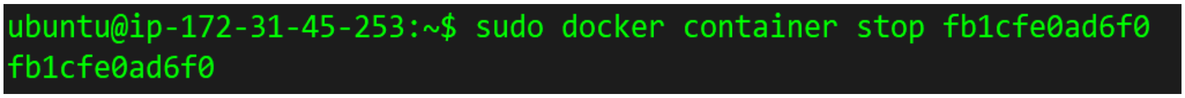
root@233e926091f3: /

```
ubuntu@ip-172-31-26-120:~$ docker exec -it 233e926091f3 bash
root@233e926091f3:/#
```

8. Forstoppingarunningcontainer,weusethe“stop”command

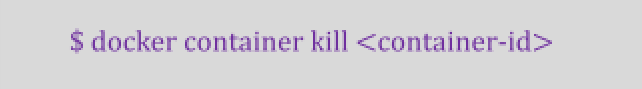


```
$ docker container stop <container-id>
```

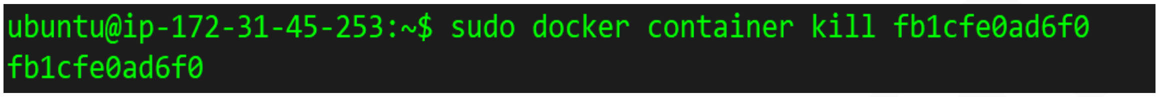


```
ubuntu@ip-172-31-45-253:~$ sudo docker container stop fb1cfe0ad6f0
fb1cfe0ad6f0
```

9. This command kills the container by stopping its execution immediately. The difference between ‘docker kill’ and ‘docker stop’. ‘docker stop’ gives the container time toshutdowngracefully.Insituationswhenitistakingtoo much time for getting the container to stop, one can opt to kill it.

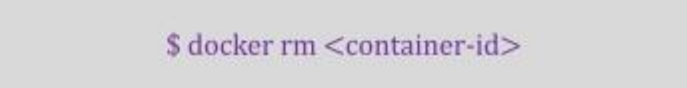


```
$ docker container kill <container-id>
```

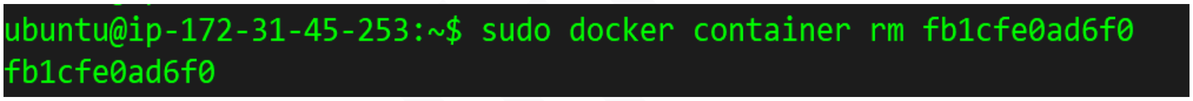


```
ubuntu@ip-172-31-45-253:~$ sudo docker container kill fb1cfe0ad6f0
fb1cfe0ad6f0
```

10. To remove a stopped container from the system, we use the “rm” command

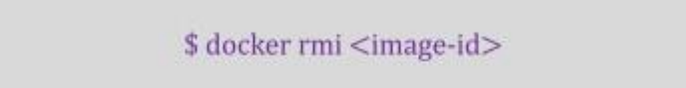


```
$ docker rm <container-id>
```

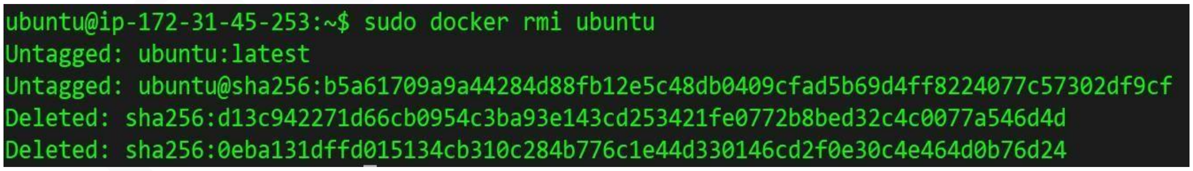


```
ubuntu@ip-172-31-45-253:~$ sudo docker container rm fb1cfe0ad6f0
fb1cfe0ad6f0
```

11. Toremoveanimagefromthesystemweusethecommand“rmi



```
$ docker rmi <image-id>
```

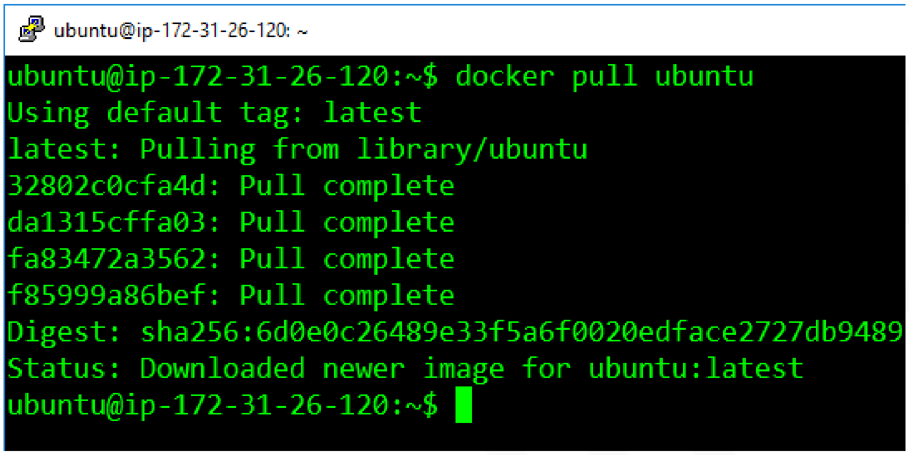


```
ubuntu@ip-172-31-45-253:~$ sudo docker rmi ubuntu
Untagged: ubuntu:latest
Untagged: ubuntu@sha256:b5a61709a9a44284d88fb12e5c48db0409cfad5b69d4ff8224077c57302df9cf
Deleted: sha256:d13c942271d66cb0954c3ba93e143cd253421fe0772b8bed32c4c0077a546d4d
Deleted: sha256:0eba131dffd015134cb310c284b776c1e44d330146cd2f0e30c4e464d0b76d24
```

Saving Change to a Docker Container

Step1: Pullthedockercontainerusingthecommand:

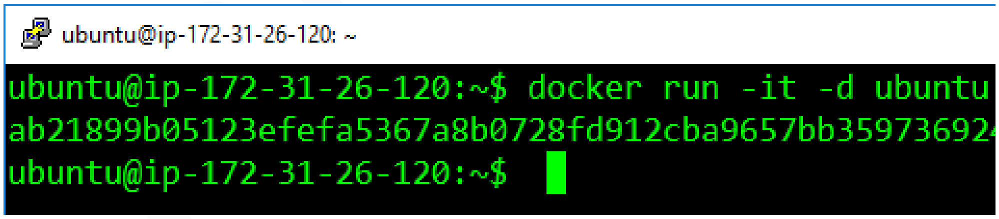
```
$ docker pull ubuntu
```



```
ubuntu@ip-172-31-26-120:~$ docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
32802c0cfa4d: Pull complete
da1315cffa03: Pull complete
fa83472a3562: Pull complete
f85999a86bef: Pull complete
Digest: sha256:6d0e0c26489e33f5a6f0020edface2727db9489
Status: Downloaded newer image for ubuntu:latest
ubuntu@ip-172-31-26-120:~$
```

Step2: Runthecontainerusingthecommands

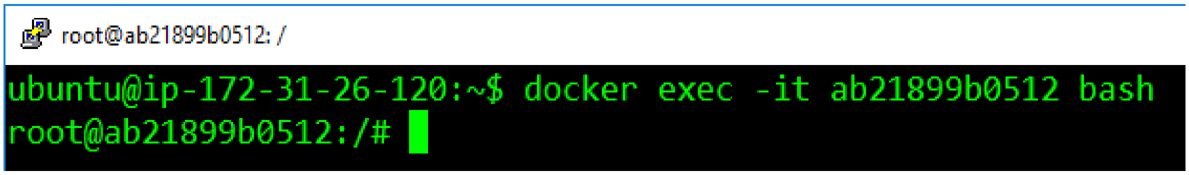
```
$ docker container run –it –d ubuntu
```



```
ubuntu@ip-172-31-26-120:~$ docker run -it -d ubuntu
ab21899b05123efefa5367a8b0728fd912cba9657bb35973692
ubuntu@ip-172-31-26-120:~$
```

Step3: Accessthecontainerusingthecommand:

```
$ docker exec –it <container-id> bash
```

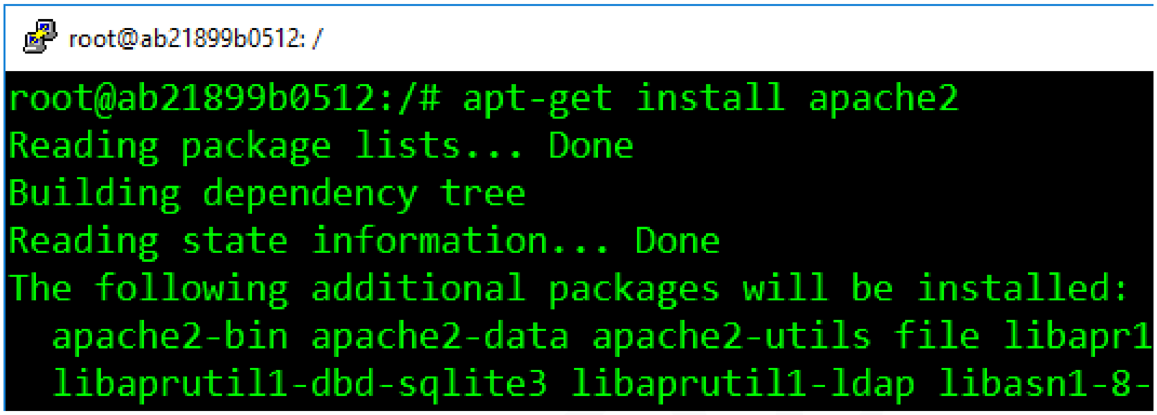


```
ubuntu@ip-172-31-26-120:~$ docker exec -it ab21899b0512 bash
root@ab21899b0512:/#
```

Step4: Installapache2onthiscontainer,usingthefollowing:

```
$ apt-get update

$ apt-get install apache2
```

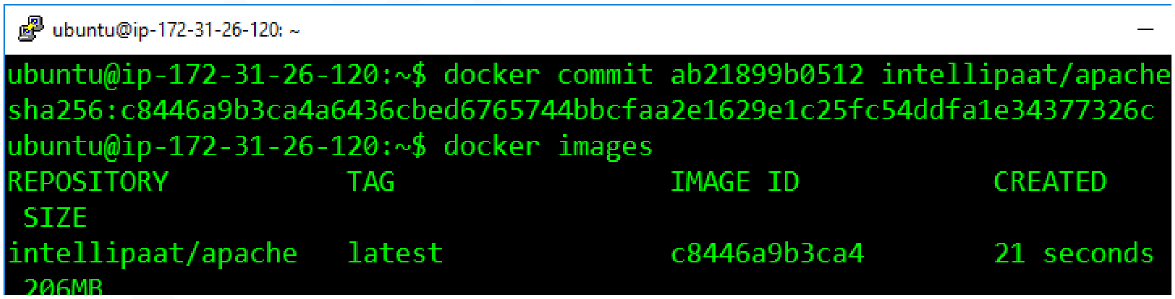


root@ab21899b0512: /

```
root@ab21899b0512:/# apt-get 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 file libapr1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libasn1-8-
```

Step5: Exitthecontainer,andsavethecontainerusingthiscommand. Thesavedcontainerwillbeconvertedintoanimagewiththenamespecifiedis

```
$ docker commit <container-id> <username>/<container-name>
```



ubuntu@ip-172-31-26-120: ~

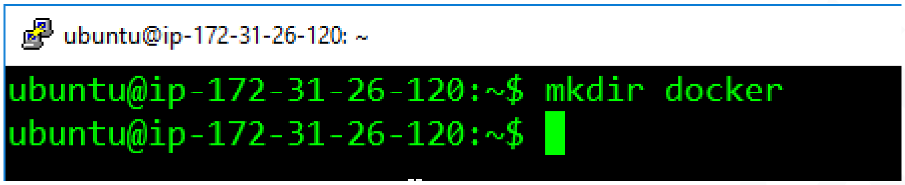
```
ubuntu@ip-172-31-26-120:~$ docker commit ab21899b0512 intellipaat/apache
sha256:c8446a9b3ca4a6436cbed6765744bbcfaa2e1629e1c25fc54ddfa1e34377326c
ubuntu@ip-172-31-26-120:~$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED
 SIZE
intellipaat/apache   latest              c8446a9b3ca4        21 seconds
 206MB
```

The usernamemustmatchwiththeusernamewhichyoucreatedonDockerHub. The container-name name can be anything.

Creating a Docker file

Step1: CreateafolderDockerinthehomedirectory

```
$ mkdir docker
```



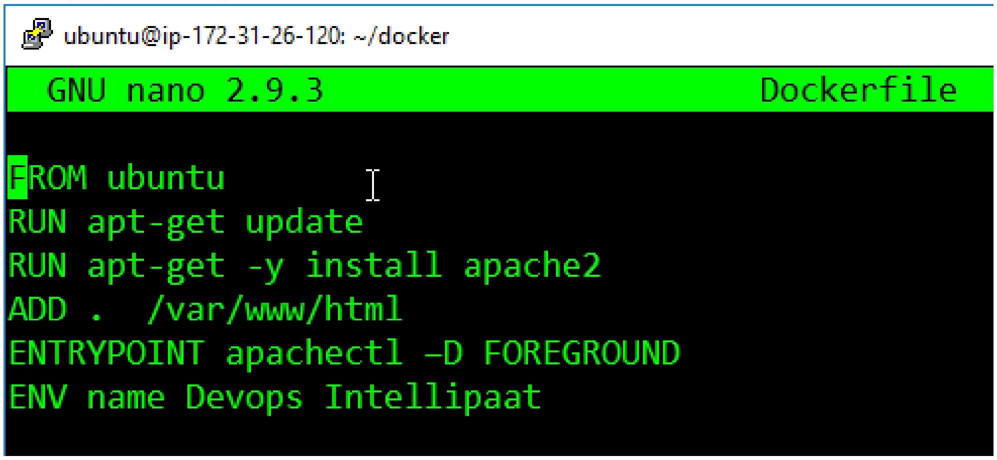
ubuntu@ip-172-31-26-120: ~

```
ubuntu@ip-172-31-26-120:~$ mkdir docker
ubuntu@ip-172-31-26-120:~$
```

Step 2: Enter into this directory and create a file called ‘Dockerfile’, with thesame contents as the Sample Dockerfile. Add the following content in the Dockerfile.

```
$ cd docker
$ sudo nano Dockerfile
```

```
FROM ubuntu
RUN apt-get update
RUN apt-get -y install apache2
  ADD . /var/www/html
ENTRYPOINT apachectl -D FOREGROUND
ENV name Devops Intellipaat
```



ubuntu@ip-172-31-26-120: ~/docker

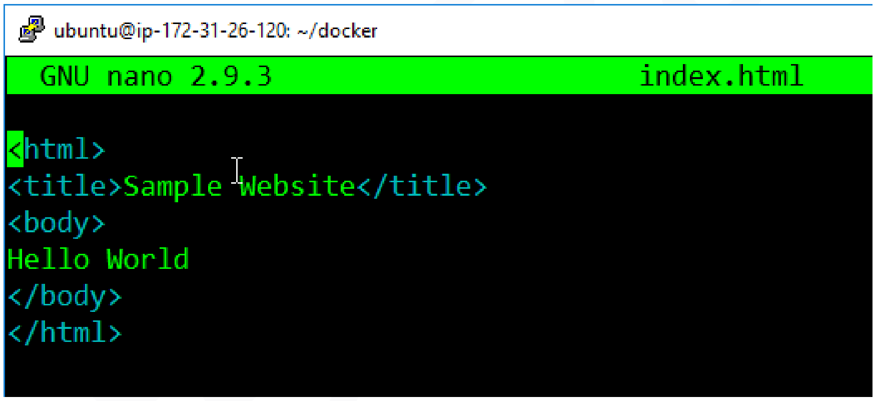
```
  GNU nano 2.9.3                      Dockerfile

FROM ubuntu
RUN apt-get update
RUN apt-get -y install apache2
ADD .  /var/www/html
ENTRYPOINT apachectl –D FOREGROUND
ENV name Devops Intellipaat
```

Step 3: Create one more file called index.html with the followingcontentswhich can verify the push on Docker Hub.

```
$ sudo nano index.html
```

```
<html>
<title> Sample Website </title>
<body>
  Hello World
</body>
</html>
```



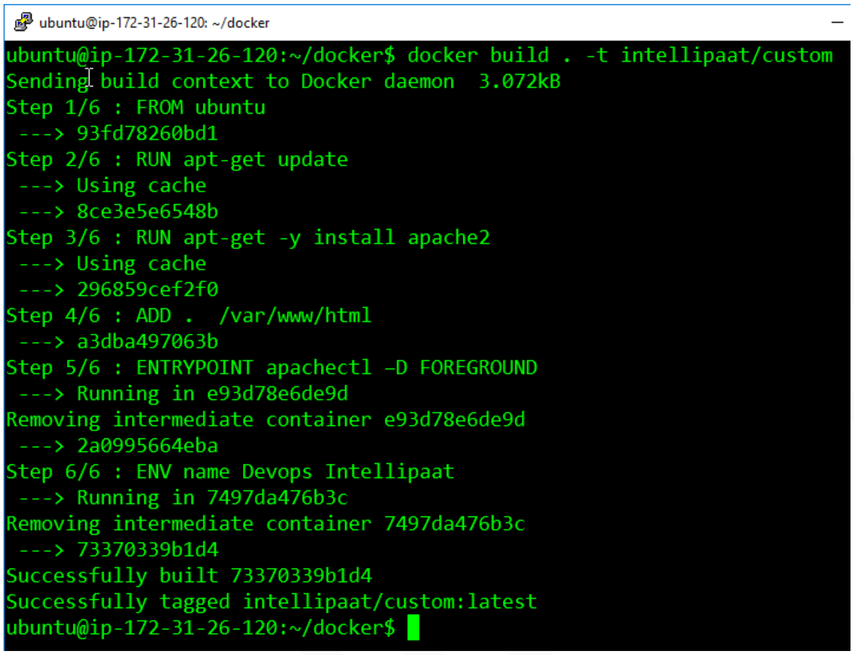
ubuntu@ip-172-31-26-120: ~/docker

```
  GNU nano 2.9.3                    index.html

<html>
<title>Sample Website</title>
<body>
Hello World
</body>
</html>
```

Step4: Nowpassthefollowingcommand:

```
$ docker build <directory-of-dockerfile> -t
          <name of container>
```

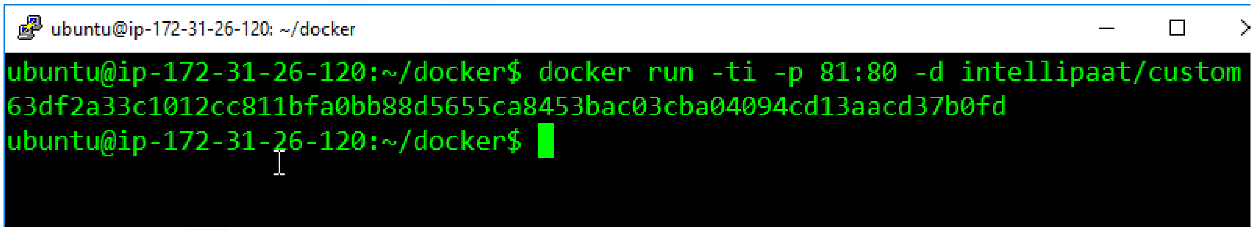


ubuntu@ip-172-31-26-120: ~/docker

```
ubuntu@ip-172-31-26-120:~/docker$ docker build . -t intellipaat/custom
Sending build context to Docker daemon  3.072kB
Step 1/6 : FROM ubuntu
 ---> 93fd78260bd1
Step 2/6 : RUN apt-get update
 ---> Using cache
 ---> 8ce3e5e6548b
Step 3/6 : RUN apt-get -y install apache2
 ---> Using cache
 ---> 296859cef2f0
Step 4/6 : ADD .  /var/www/html
 ---> a3dba497063b
Step 5/6 : ENTRYPOINT apachectl -D FOREGROUND
 ---> Running in e93d78e6de9d
Removing intermediate container e93d78e6de9d
 ---> 2a0995664eba
Step 6/6 : ENV name Devops Intellipaat
 ---> Running in 7497da476b3c
Removing intermediate container 7497da476b3c
 ---> 73370339b1d4
Successfully built 73370339b1d4
Successfully tagged intellipaat/custom:latest
ubuntu@ip-172-31-26-120:~/docker$
```

Step5: Finally,runthisbuiltimage,usingthefollowingcommand:

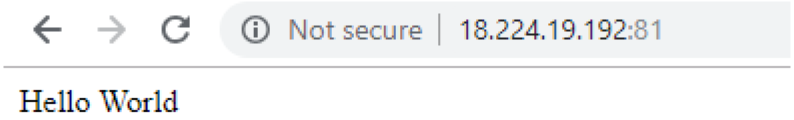
```
$ docker run –it –p 81:80 –d <name of container>
```



ubuntu@ip-172-31-26-120: ~/docker

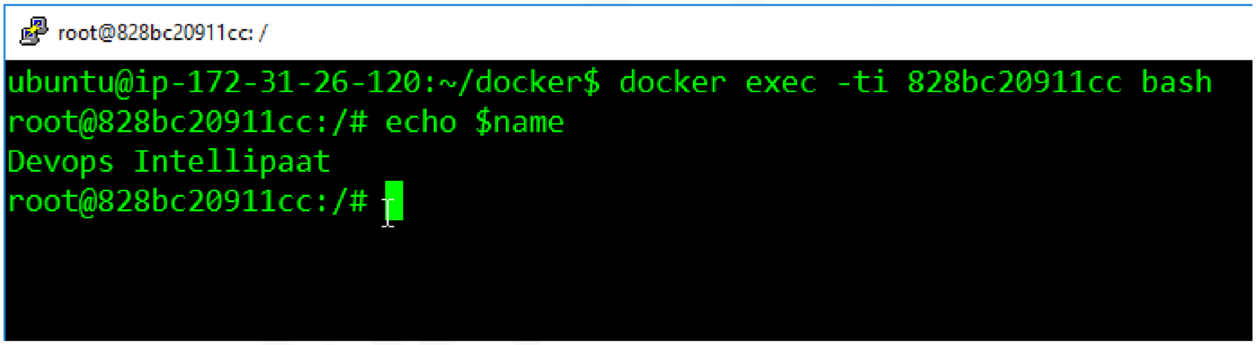
```
ubuntu@ip-172-31-26-120:~/docker$ docker run -ti -p 81:80 -d intellipaat/custom
63df2a33c1012cc811bfa0bb88d5655ca8453bac03cba04094cd13aacd37b0fd
ubuntu@ip-172-31-26-120:~/docker$
```

Step6: NownavigatetotheserverIPaddressonport81



Hello World

Step 7: Finally, login into the container and check the variable $name, it will have the same value, as given in the Dockerfile.

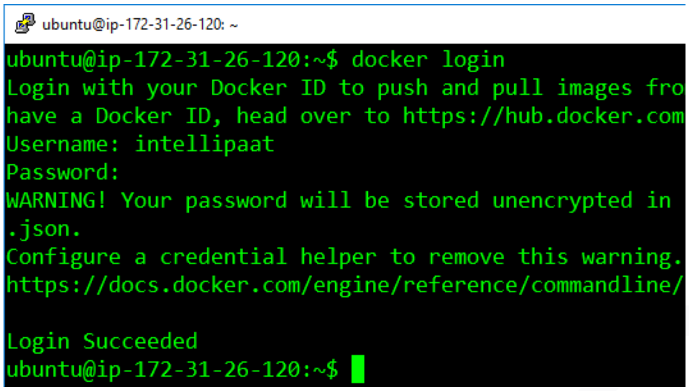


```
ubuntu@ip-172-31-26-120:~/docker$ docker exec -ti 828bc20911cc bash
root@828bc20911cc:/# echo $name
Devops Intellipaat
root@828bc20911cc:/# 
```

Pushing the Docker to Container Hub

Step1: Thefirststepistologin,itcanbedoneusingthefollowingcommand:

```
$ docker login
```

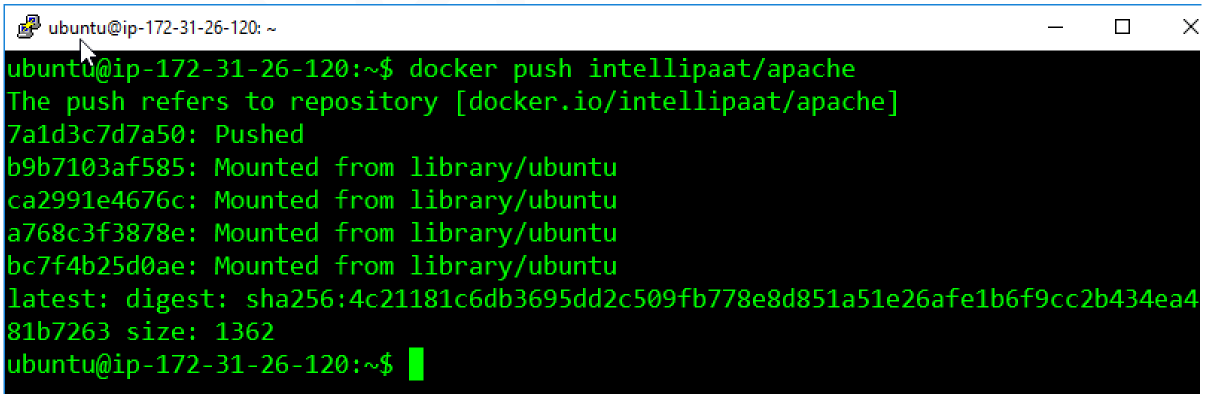


```
ubuntu@ip-172-31-26-120: ~
ubuntu@ip-172-31-26-120:~$ docker login
Login with your Docker ID to push and pull images fro
have a Docker ID, head over to https://hub.docker.com
Username: intellipaat
Password:
WARNING! Your password will be stored unencrypted in
.json.
Configure a credential helper to remove this warning.
https://docs.docker.com/engine/reference/commandline/

Login Succeeded
ubuntu@ip-172-31-26-120:~$
```

Step2: Finally,forpushingyourcontaineronDockerHub,usethefollowing command:

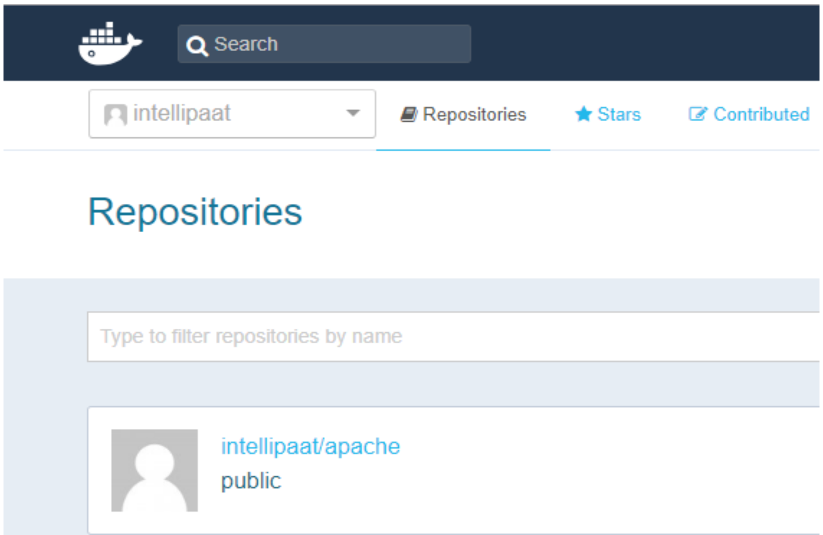
```
$ docker push <dockerhub-username>/<image-name>
```



```
ubuntu@ip-172-31-26-120: ~
ubuntu@ip-172-31-26-120:~$ docker push intellipaat/apache
The push refers to repository [docker.io/intellipaat/apache]
7a1d3c7d7a50: Pushed
b9b7103af585: Mounted from library/ubuntu
ca2991e4676c: Mounted from library/ubuntu
a768c3f3878e: Mounted from library/ubuntu
bc7f4b25d0ae: Mounted from library/ubuntu
latest: digest: sha256:4c21181c6db3695dd2c509fb778e8d851a51e26afe1b6f9cc2b434ea4
81b7263 size: 1362
ubuntu@ip-172-31-26-120:~$
```

Step3: YoucanverifythepushonDockerHub

```
$ docker exec –it <container-id> bash
```



Repositories

Type to filter repositories by name

intellipaat/apache
public

Now anyone who wants to downloadthiscontainercansimplypassthefollowing command:

```
docker pull intellipaat/apache
```