DOCKER VOLUMES,IMAGES, BUILD WITH DOCKER HUB

Docker documentation

Kaushik Dey

DOCKER ADVANCED COMMANDS

To check the volume

```
[root@ip-172-31-21-247 ~] # docker volume list

DRIVER    VOLUME NAME
```

To create the volume

```
[root@ip-172-31-21-247 ~] # docker volume create   kaushikvol1
kaushikvol1
[root@ip-172-31-21-247 ~] # docker volume list
DRIVER    VOLUME NAME
local     kaushikvol1
```

To inspect the volume

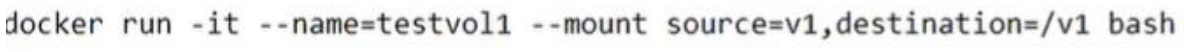
```
[root@ip-172-31-21-247 ~] # docker volume inspect kaushikvol1
[
    {
        "CreatedAt": "2023-05-03T01:21:29Z",
        "Driver": "local",
        "Labels": {},
        "Mountpoint": "/var/lib/docker/volumes/kaushikvol1/_data",
        "Name": "kaushikvol1",
        "Options": {},
        "Scope": "local"
    }
]
[root@ip-172-31-21-247 ~] #
docker run -it --mount source=kaushikvol1,destination=/kaushikvol1
centos bash
```

```
[root@ip-172-31-45-37 ec2-user]# docker volume list
DRIVER    VOLUME NAME
local     kaushikvol1
[root@ip-172-31-45-37 ec2-user]# docker inspect kaushikvol1
[
    {
        "CreatedAt": "2023-05-08T02:23:43Z",
        "Driver": "local",
        "Labels": {},
        "Mountpoint": "/var/lib/docker/volumes/kaushikvol1/_data",
        "Name": "kaushikvol1",
        "Options": {},
        "Scope": "local"
    }
]
[root@ip-172-31-45-37 ec2-user]# cd /var/lib/docker/volumes
[root@ip-172-31-45-37 volumes]# ls
backingFsBlockDev  kaushikvol1  metadata.db
[root@ip-172-31-45-37 volumes]# history
```

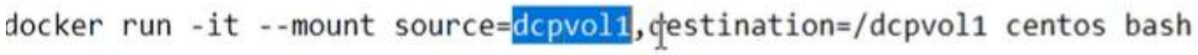
- ✓ clear
- ✓ docker --version
- ✓ yum install docker -y
- ✓ clear
- ✓ docker --version
- ✓ docker version
- ✓ systemctl status docker
- ✓ systemctl start docker
- ✓ systemctl status docker
- ✓ systemctl enable docker
- ✓ clear
- ✓ docker version
- ✓ clear
- ✓ docker images
- ✓ docker ps
- ✓ docker ps -a
- ✓ docker run ubuntu
- ✓ docker images
- ✓ docker ps -a
- ✓ docker ps
- ✓ docker pull nginx
- ✓ clear
- ✓ docker images
- ✓ docker run -it centos bash
- ✓ clear
- ✓ docker images
- ✓ docker volume list
- ✓ docker volume create kaushikvol1
- ✓ docker volume list
- ✓ docker volume inspect kaushikvol1
- ✓ docker volume list
- ✓ docker inspect kaushikvol1
- ✓ cd /var/lib/docker/volumes
- ✓ ls
- ✓ history

```
[root@ip-172-31-45-37 volumes]# cd..
bash: cd..: command not found
[root@ip-172-31-45-37 volumes]# cd ..
[root@ip-172-31-45-37 docker]# cd ..
[root@ip-172-31-45-37 lib]# cd
[root@ip-172-31-45-37 ~]# pwd
/root
[root@ip-172-31-45-37 ~]# cd /var/lib/docker/volumes
[root@ip-172-31-45-37 volumes]# pwd
/var/lib/docker/volumes
[root@ip-172-31-45-37 volumes]# ls
backingFsBlockDev  kaushikvol1  metadata.db
[root@ip-172-31-45-37 volumes]# cd kaushikvol1
[root@ip-172-31-45-37 kaushikvol1]# ls
_data
[root@ip-172-31-45-37 kaushikvol1]# pwd
/var/lib/docker/volumes/kaushikvol1
[root@ip-172-31-45-37 kaushikvol1]# cd _data
[root@ip-172-31-45-37 _data]# pwd
/var/lib/docker/volumes/kaushikvol1/_data
[root@ip-172-31-45-37 _data]# ls
fileformcontainer.txt
[root@ip-172-31-45-37 _data]# echo "newvol1" >> propertyfile.txt
[root@ip-172-31-45-37 _data]# ls
fileformcontainer.txt  propertyfile.txt
[root@ip-172-31-45-37 _data]# ll
total 8
-rw-r--r-- 1 root root 25 May  8 02:23 fileformcontainer.txt
-rw-r--r-- 1 root root  8 May  8 03:24 propertyfile.txt
[root@ip-172-31-45-37 _data]# docker run -it --mount source=kaushikvol1,destination=/kaushikvol1 centos bash
[root@94baf3926469 /]# ls
bin  etc   kaushikvol1  lib64       media  opt   root  sbin  sys  usr
dev  home  lib          lost+found  mnt    proc  run   srv   tmp  var
[root@94baf3926469 /]# cd kaushikvol1
[root@94baf3926469 kaushikvol1]# ls
fileformcontainer.txt  propertyfile.txt
[root@94baf3926469 kaushikvol1]#
```

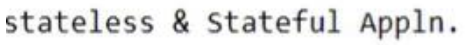
Please note that :
Container also running in state. In such scenario we required external volume.



```
docker run -it --name=testvol1 --mount source=v1,destination=/v1 bash
```

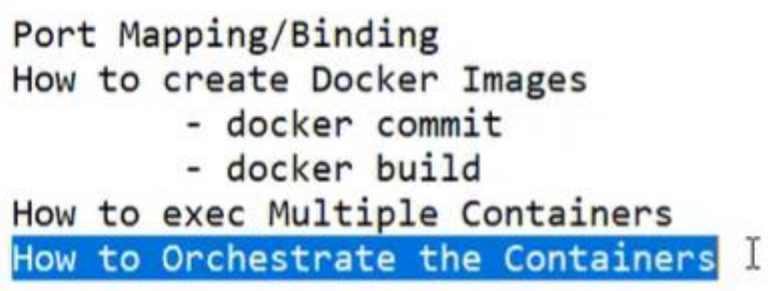


```
docker run -it --mount source=dcpvol1,destination=/dcpvol1 centos bash
```



stateless & Stateful Appln.

With the help of orchestration tool we can add volumes, external volumes and clear the volume

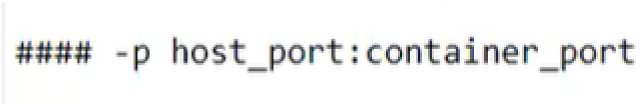


```
Port Mapping/Binding
How to create Docker Images
        - docker commit
        - docker build
How to exec Multiple Containers
How to Orchestrate the Containers
```

PORT MAPPING & BINDING

Run tomcat with port mapping

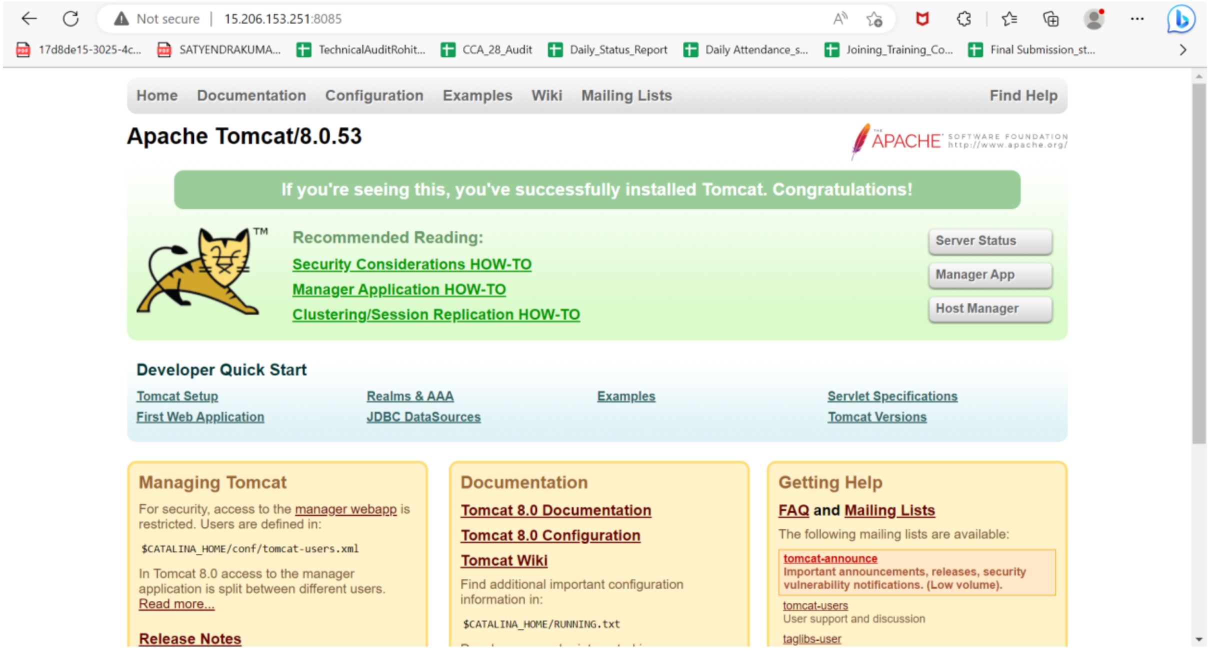
Host_port:container_port



```
#### -p host_port:container_port
```

docker run -it -p 8085:8080 tomcat:8.0

in this demo we never install tomcat in our VM and it is running inside the container so, that’s why port mapping and binding is required.



Apache Tomcat/8.0.53

If you're seeing this, you've successfully installed Tomcat. Congratulations!

Recommended Reading:
Security Considerations HOW-TO
Manager Application HOW-TO
Clustering/Session Replication HOW-TO

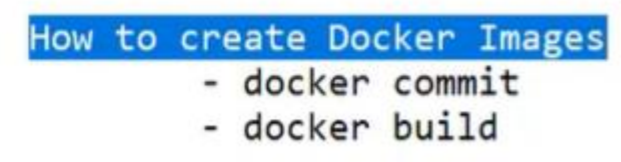
Developer Quick Start
Tomcat Setup
First Web Application
Realms & AAA
JDBC DataSources
Examples
Servlet Specifications
Tomcat Versions

Managing Tomcat
For security, access to the manager webapp is restricted. Users are defined in:
$CATALINA_HOME/conf/tomcat-users.xml
In Tomcat 8.0 access to the manager application is split between different users. Read more...
Release Notes

Documentation
Tomcat 8.0 Documentation
Tomcat 8.0 Configuration
Tomcat Wiki
Find additional important configuration information in:
$CATALINA_HOME/RUNNING.txt

Getting Help
FAQ and Mailing Lists
The following mailing lists are available:
tomcat-announce
Important announcements, releases, security vulnerability notifications. (Low volume).
tomcat-users
User support and discussion
taglibs-user

DOCKER IMAGES


```
How to create Docker Images
          - docker commit
          - docker build
```

```
[root@ip-172-31-45-37 ec2-user]# docker images
REPOSITORY   TAG      IMAGE ID      CREATED         SIZE
debian       latest   34b4fa67dc04   5 days ago      124MB
nginx        latest   6efc10a0510f   3 weeks ago     142MB
ubuntu       latest   08d22c0ceb15   2 months ago    77.8MB
centos       latest   5d0da3dc9764   19 months ago   231MB
tomcat       8.0      ef6a7c98d192   4 years ago     356MB
[root@ip-172-31-45-37 ec2-user]# docker run -it debian
root@83f56f24030b:/# ls
bin  dev  home  lib64  mnt  proc  run   srv  tmp  var
boot  etc  lib   media  opt  root  sbin  sys  usr
root@83f56f24030b:/# git --version
bash: git: command not found
root@83f56f24030b:/# apt-get update && apt-get install -y git
Install inside that debian container
root@83f56f24030b:/# git –version
```

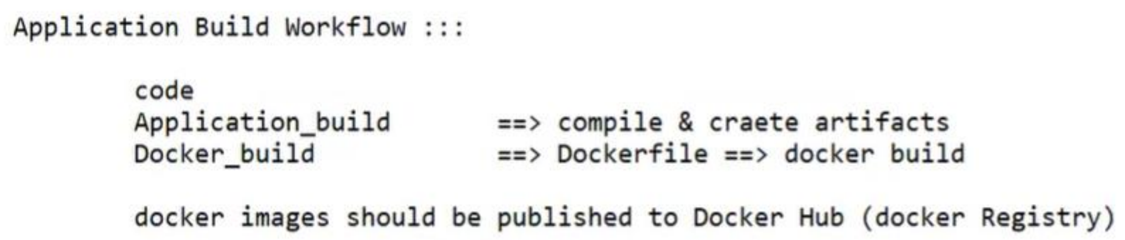
git version 2.30.2

in this debian container we added git install, so we can create our custom images from this debian container, so the command is

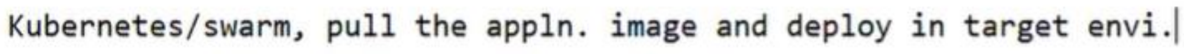
```
[root@ip-172-31-45-37 ec2-user] # docker commit 83f56f24030b iimdatascience1/dcpdebian-git:v1.0
[root@ip-172-31-45-37 ec2-user] # docker images
[root@ip-172-31-45-37 ec2-user] # docker run -it iimdatascience1/dcpdebian-git:v1.0
```

DOCKER Files

```
[root@ip-172-31-45-37 ec2-user] # mkdir dcp-docker

[root@ip-172-31-45-37 ec2-user] # cd dcp-docker

[root@ip-172-31-45-37 dcp-docker] # vi Dockerfile

[root@ip-172-31-45-37 dcp-docker] # cat Dockerfile

[root@ip-172-31-45-37 dcp-docker] # docker build -t iimdatascience1/dcp-deb-git-vim .

Successfully built 2c2aeb328b36

Successfully tagged iimdatascience1/dcp-deb-git-vim:latest

[root@ip-172-31-45-37 dcp-docker] # docker images
```



```
Application Build Workflow :::

        code
        Application_build       ==> compile & craete artifacts
        Docker_build            ==> Dockerfile ==> docker build

        docker images should be published to Docker Hub (docker Registry)
```



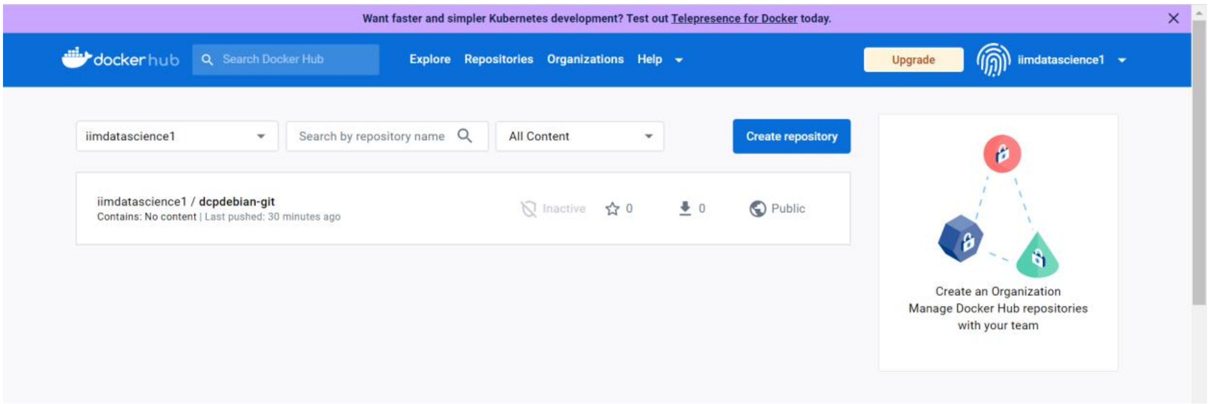
```
Kubernetes/swarm, pull the appln. image and deploy in target envi.
```

PUBLISH DOCKER IMAGES TO DOCKER HUB

Before login inside the docker hub from docker daemon first we need to create access token and with the help of access token we should connect with docker hub.

Step 1 : login in dockerhub

https://hub.docker.com/

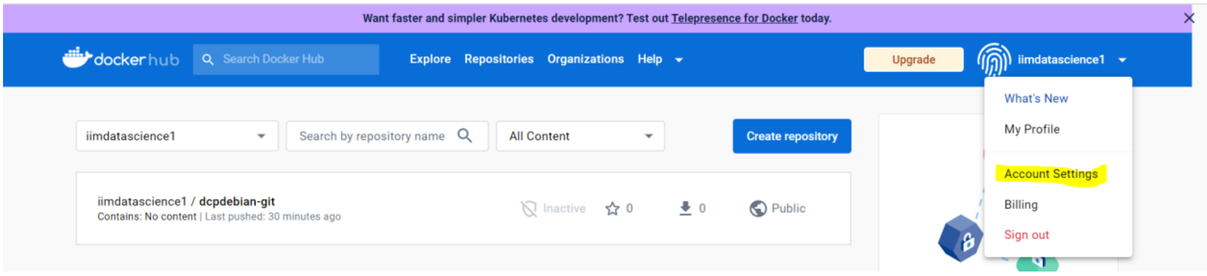


Want faster and simpler Kubernetes development? Test out Telepresence for Docker today.

iimdatascience1 / dcpdebian-git
Contains: No content | Last pushed: 30 minutes ago

Create an Organization
Manage Docker Hub repositories with your team

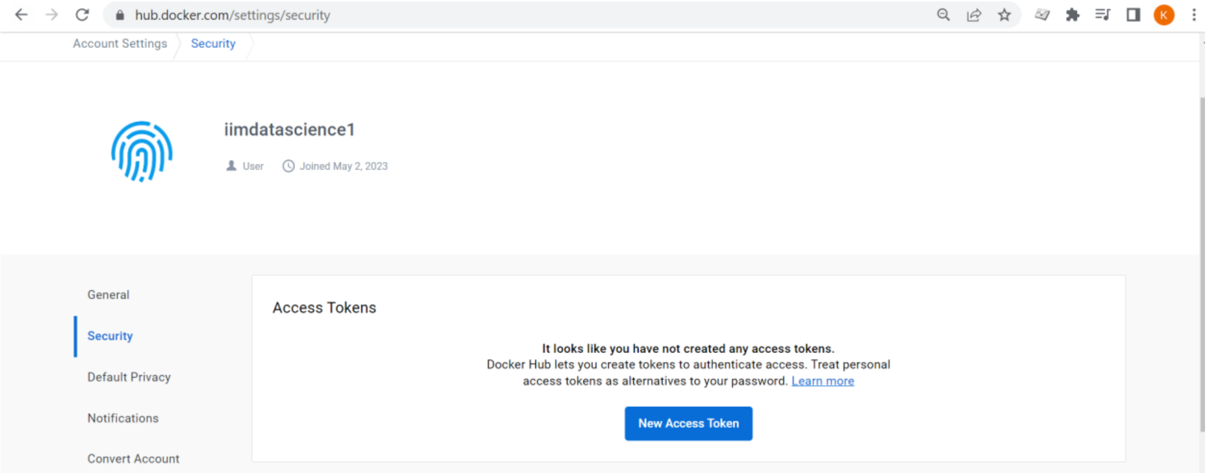
Step 2 : Now we should go with account settings, the screenshot is given below.



Want faster and simpler Kubernetes development? Test out Telepresence for Docker today.

iimdatascience1 / dcpdebian-git
Contains: No content | Last pushed: 30 minutes ago

Step 3 : Now we should create access token , the screenshot is given below.



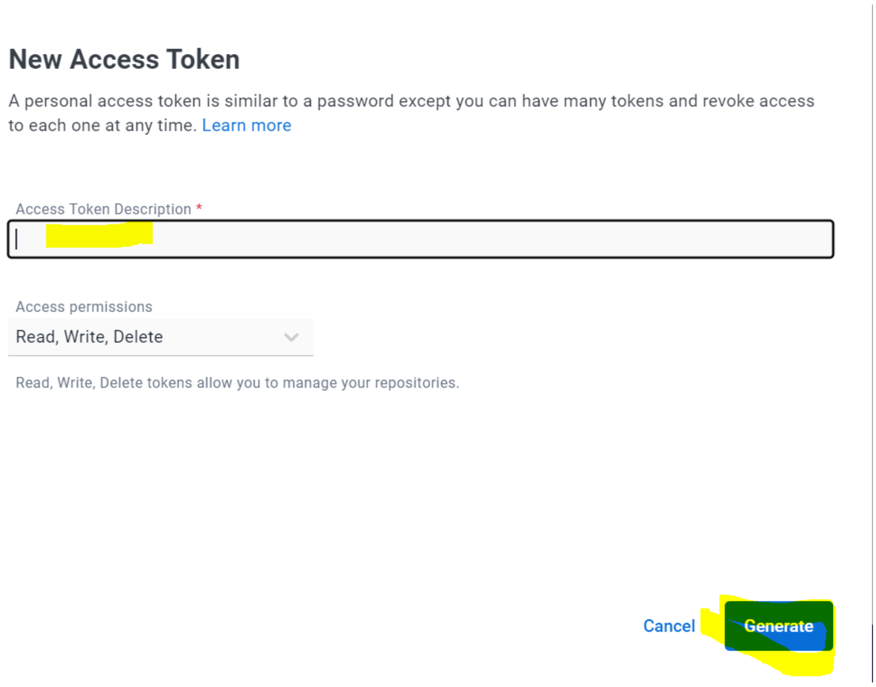
hub.docker.com/settings/security

iimdatascience1
User · Joined May 2, 2023

Access Tokens

It looks like you have not created any access tokens.
Docker Hub lets you create tokens to authenticate access. Treat personal access tokens as alternatives to your password. Learn more

Step 4 : Now click on New Access Token, the screenshot is given below.



New Access Token

A personal access token is similar to a password except you can have many tokens and revoke access to each one at any time. Learn more

Access Token Description *

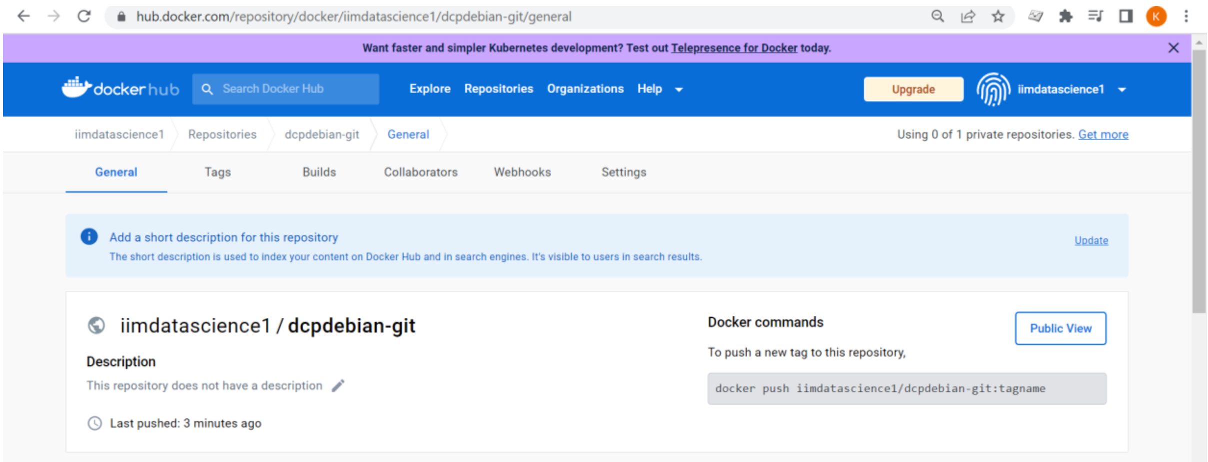
Access permissions: Read, Write, Delete

Read, Write, Delete tokens allow you to manage your repositories.

Step 5 : now we should run with following commands to see the effects.

```
[root@ip-172-31-45-37 dcp-docker]# docker login -u iimdatascience1
Password: dckr_pat_XR7_NEn8wxykHbQMEe4LWYOph_Y
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
[root@ip-172-31-45-37 dcp-docker] #
[root@ip-172-31-45-37 dcp-docker] # docker push iimdatascience1/dcpdebian-git:v1.0

The push refers to repository [docker.io/iimdatascience1/dcpdebian-git]
4d72adc2489a: Pushed
d925e0fae4e6: Mounted from library/debian
v1.0: digest: sha256:4bbcb5e5e26be0987f3dcbce08bb2845d7223effcc009c605fb8961af2ae3c95 size: 741
[root@ip-172-31-45-37 dcp-docker] #
```

Now, in our docker hub repo our images is pushed , the screenshot is given below.



hub.docker.com/repository/docker/iimdatascience1/dcpdebian-git/general
Want faster and simpler Kubernetes development? Test out Telepresence for Docker today.
docker hub
Search Docker Hub
Explore
Repositories
Organizations
Help
Upgrade
iimdatascience1
iimdatascience1
Repositories
dcpdebian-git
General
Using 0 of 1 private repositories. Get more
General
Tags
Builds
Collaborators
Webhooks
Settings
Update
Public View

Add a short description for this repository

The short description is used to index your content on Docker Hub and in search engines. It's visible to users in search results.

iimdatascience1 / **dcpdebian-git**

Description

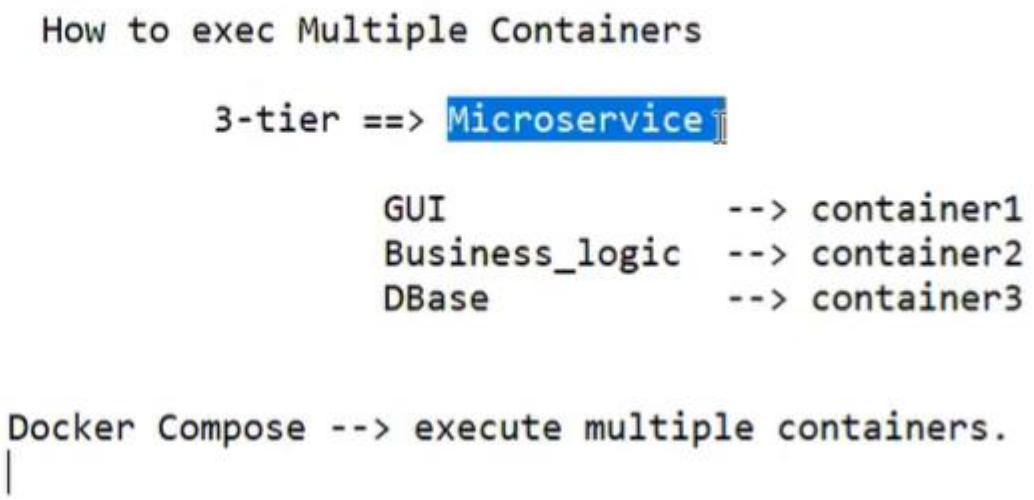
This repository does not have a description

Last pushed: 3 minutes ago

Docker commands

To push a new tag to this repository,

```
docker push iimdatascience1/dcpdebian-git:tagname
```



```
How to exec Multiple Containers

        3-tier ==> Microservice

                GUI               --> container1
                Business_logic    --> container2
                DBase             --> container3


Docker Compose --> execute multiple containers.
```