Hey techies…👋

In this session , we're going to learn how to push custom images to docker private repositories.

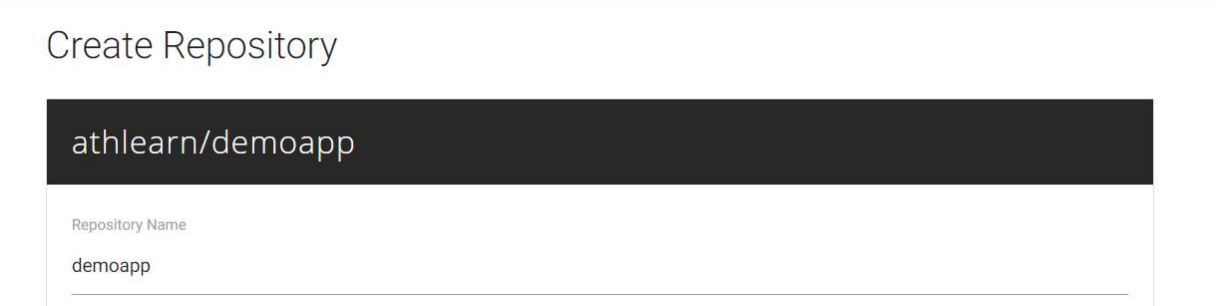
Before going to the hands-on lab , let's have some basic ideas on Docker.

📌A Docker image is a file used to execute code in a Docker container which acts as a set of instructions to build a Docker container, like a template.
📌A Docker container image is a lightweight, standalone, executable package of software that includes everything needed to run an application: code, runtime, system tools, system libraries and settings.
📌Docker Engine is an open source containerization technology for building and containerizing your applications.
📌A Docker registry is a system for versioning, storing and distributing Docker images.
Docker Control Plane is the enterprise-grade cluster management solution from Docker.

📌private Docker registry allows you to share your custom base images within your organization, keeping a consistent, private, and centralized source of truth for the building blocks of your architecture.

▶️▶️▶️ Let's get started….!!!

📌Here we're using Canister.io to create a private repository. I've created an instance using Google Cloud account to use CLI. You can opt on your convenience. I've created a repo in Canister.io named demoapp and athlearn is my account name.



I have cleaned up all the existing images,containers etc.

🖌To stop any running container → docker stop $(docker ps -q)
🖌To remove all the containers → docker rm $(docker ps -a -q)
🖌To remove all the images → docker rmi $(docker images -a -q)

Now we're going to build a new image called centos

🖌docker pull centos

```
[root@centos athirakk1827]# docker pull centos
Using default tag: latest
latest: Pulling from library/centos
a1d0c7532777: Already exists
Digest: sha256:a27fd8080b517143cbbbab9dfb7c8571c40d67d534bbdee55bd6c473f432b177
Status: Downloaded newer image for centos:latest
docker.io/library/centos:latest
[root@centos athirakk1827]#
```

And run the image in detached mode

🖊 docker run -dit centos

```
[root@centos athirakk1827]# docker run -dit centos
b07087e87e76ad801bbb4c0dbf23f10ba1241c9db73ce90568607c4b6c645fbd
[root@centos athirakk1827]#
```

Now list the running containers as shown below:

```
[root@centos athirakk1827]# docker ps -a
CONTAINER ID   IMAGE     COMMAND       CREATED          STATUS          PORTS     NAMES
b07087e87e76   centos    "/bin/bash"   21 seconds ago   Up 20 seconds             sleepy_dubinsky
[root@centos athirakk1827]#
```

Then login inside to the container as shown below:

```
[root@centos athirakk1827]# docker exec -it b07087e87e76 bash
[root@b07087e87e76 /]#
```

And install below packages:

🖊 yum install httpd net-tools vsftpd

```
Installed:
  apr-1.6.3-12.el8.x86_64                                  apr-util-1.6.1-6.el8.x86_64
  apr-util-bdb-1.6.1-6.el8.x86_64                          apr-util-openssl-1.6.1-6.el8.x86_64
  brotli-1.0.6-3.el8.x86_64                                centos-logos-httpd-85.8-2.el8.noarch
  httpd-2.4.37-43.module_el8.5.0+1022+b541f3b1.x86_64      httpd-filesystem-2.4.37-43.module_el8.5.0+1022+b541f3b1.noarch
  httpd-tools-2.4.37-43.module_el8.5.0+1022+b541f3b1.x86_64  logrotate-3.14.0-4.el8.x86_64
  mailcap-2.1.48-3.el8.noarch                              mod_http2-1.15.7-3.module_el8.4.0+778+c970deab.x86_64
  net-tools-2.0-0.52.20160912git.el8.x86_64                vsftpd-3.0.3-34.el8.x86_64

Complete!
[root@b07087e87e76 /]#
```

Also place a text document as shown below and enable below services:

🖊 systemctl enable httpd

🖊 systemctl enable vsftpd

```
[root@b07087e87e76 /]# echo "Always be work in progress" > /var/www/html/index.html
[root@b07087e87e76 /]# more /var/www/html/index.html
Always be work in progress
[root@b07087e87e76 /]#
```

Then come out from container as shown below : We can conclude commit is successful as we can see the output sha256:..

```
[root@centos athirakk1827]# docker commit -c 'CMD ["/usr/sbin/httpd","-D","FOREGROUND"]' b07087e87e76
sha256:c218045762fdf8a08cc77679ef40dc14b04b6b8377e03018ba8d3fffd4695ea8
[root@centos athirakk1827]#
```

We can see a new image has been added to images list:

```
[root@centos athirakk1827]# docker images
REPOSITORY    TAG       IMAGE ID       CREATED          SIZE
<none>        <none>    c218045762fd   32 seconds ago   281MB
centos        latest    5d0da3dc9764   23 months ago    231MB
[root@centos athirakk1827]#
```

Now tag the newly builded image as shown below by taking the image ID and copy the lines after docker pull from Cansister.io Docker CLI and give a tag v1.0 and again run docker images , so we can see the repository name which we just provided:

```
[root@centos athirakk1827]# docker images
REPOSITORY   TAG       IMAGE ID       CREATED              SIZE
<none>       <none>    c218045762fd   About a minute ago   281MB
centos       latest    5d0da3dc9764   23 months ago        231MB
[root@centos athirakk1827]# docker tag c218045762fd cloud.canister.io:5000/athlearn/demoapp:v1.0
[root@centos athirakk1827]# docker images
REPOSITORY                                  TAG       IMAGE ID       CREATED          SIZE
cloud.canister.io:5000/athlearn/demoapp     v1.0      c218045762fd   2 minutes ago    281MB
centos                                      latest    5d0da3dc9764   23 months ago    231MB
[root@centos athirakk1827]#
```

Now push the image to private repository as shown below:

```
[root@centos athirakk1827]# docker push cloud.canister.io:5000/athlearn/demoapp:v1.0
The push refers to repository [cloud.canister.io:5000/athlearn/demoapp]
86f43fca5575: Pushed
74ddd0ec08fa: Pushed
v1.0: digest: sha256:f622f945ba77b3466370f9603b1a88fef1051a36400ec6a41b3e8317d9cc5836 size: 741
[root@centos athirakk1827]#
```

We check the image is pushed correctly to repository in 2 ways.
1- By refreshing the https://cloud.canister.io/ page and check the image
2- Pull the image from Cansister.io and check it is successfully pulled using CLI

For that delete the images and remove all containers:

```
[root@centos athirakk1827]# docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
[root@centos athirakk1827]# docker images
REPOSITORY   TAG       IMAGE ID   CREATED   SIZE
[root@centos athirakk1827]#
```

Then run the below command and check the pull is working or not:

🖊️ docker pull cloud.canister.io:5000/athlearn/demoapp:v1.0

```
[root@centos athirakk1827]# docker pull cloud.canister.io:5000/athlearn/demoapp:v1.0
v1.0: Pulling from athlearn/demoapp
a1d0c7532777: Already exists
3ec4225bb259: Pull complete
Digest: sha256:f622f945ba77b3466370f9603b1a88fef1051a36400ec6a41b3e8317d9cc5836
Status: Downloaded newer image for cloud.canister.io:5000/athlearn/demoapp:v1.0
cloud.canister.io:5000/athlearn/demoapp:v1.0
[root@centos athirakk1827]#
```

Hence we can conclude the image push to private repository is successfully completed.

⭐ ⭐ ⭐ Enjoy your learning….!!! ⭐ ⭐ ⭐