

ANOVOS – Installation & Run

This document contains the following sections -

1. Docker Installation
2. Configuring Docker Settings
3. Downloading Docker Image
4. Running Docker Image
5. Downloading Anovos Report
6. Direct Installation & Run
7. Post Workshop Cleanup Steps

1. Docker Installation

To get started straight away, install Docker on your machine

For Windows 10 Pro / Mac OS X / Linux

1. Visit <https://docs.docker.com/engine/install/> and download Docker for Mac or Windows.
 - a. Windows OS - <https://docs.docker.com/desktop/windows/install/> or <https://hub.docker.com/editions/community/docker-ce-desktop-mac>
 - b. For Linux Distributions - <https://docs.docker.com/engine/install/>
 - c. Mac OS - <https://docs.docker.com/desktop/mac/install/> or <https://hub.docker.com/editions/community/docker-ce-desktop-mac>

Note: For Mac OS your machine maybe one of two hardware chips - either Intel chip or Apple chip (M1). The hardware chip information can be found by clicking on the Apple logo in the menu bar and choosing *About This Mac => Overview* as seen on Figure1, 2

If it is intel chip-based Mac then we suggest going ahead with Docker Installation in above link by choosing Intel chip when downloading docker desktop.

If it is Apple-chip based Mac (M1) then we suggest skipping the following sections : . and doing section 1. ANOVOS Local Installation for Direct Installation to install Spark, Python and run Anovos directly in local without using docker.

This is due to recent issues reported for running tensor flow in docker containers on M1 machines specifically.



Figure 1 Sample Intel Chip Information



Figure 2 Sample Apple Chip Information

2. Follow the onscreen instructions on the website to install Docker on your machine.
3. Start Docker.

Confirm Docker Installation

Once installed, open a command-line terminal (like PowerShell in case of Windows OS). The following commands can help confirm docker is installed and running correctly.

1. Run `docker version` to show the docker version.
2. Run `docker image ls` to show list of images.
3. Run `docker ps` to show list of current running containers.
4. Run `docker ps -a` to show list of all containers

```
$  
$  
$docker version  
Client:  
Cloud integration: 1.0.17  
Version: 20.10.7  
API version: 1.41  
Go version: go1.16.4  
Git commit: f0df350  
Built: Wed Jun 2 11:56:22 2021  
OS/Arch: darwin/amd64  
Context: desktop-linux  
Experimental: true  
  
Server: Docker Engine - Community  
Engine:  
Version: 20.10.7  
API version: 1.41 (minimum version 1.12)  
Go version: go1.13.15  
Git commit: b0f5bc3  
Built: Wed Jun 2 11:54:58 2021  
OS/Arch: linux/amd64  
Experimental: false  
containerd:  
Version: 1.4.6  
GitCommit: d71fcfd7d8303cbf684402823e425e9dd2e99285d  
runc:  
Version: 1.0.0-rc95  
GitCommit: b9ee9c6314599f1b4a7f497e1f1f856fe433d3b7  
docker-init:  
Version: 0.19.0  
GitCommit: de40ad0  
$  
$  
$docker image ls  
REPOSITORY TAG IMAGE ID CREATED SIZE  
$  
$docker ps  
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES  
$  
$docker ps -a  
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES  
$  
$  
$
```

Figure 3 Sample Screenshot showing Docker commands on Terminal

2. Configure Docker Settings

Once Docker is installed, next step is to configure your machine to allocate adequate resources to it to run Anovos.

The following resources are recommended minimally for Anovos to run:

Setting	Recommended Value
CPUs	6
Memory	8.0 GB
Swap	1 GB
Disk Image Size	24 GB

The **Resources** tab in docker desktop (for MacOS and Windows) allows you to configure CPU, memory, disk and other resources. Please find the screenshot below for the same:

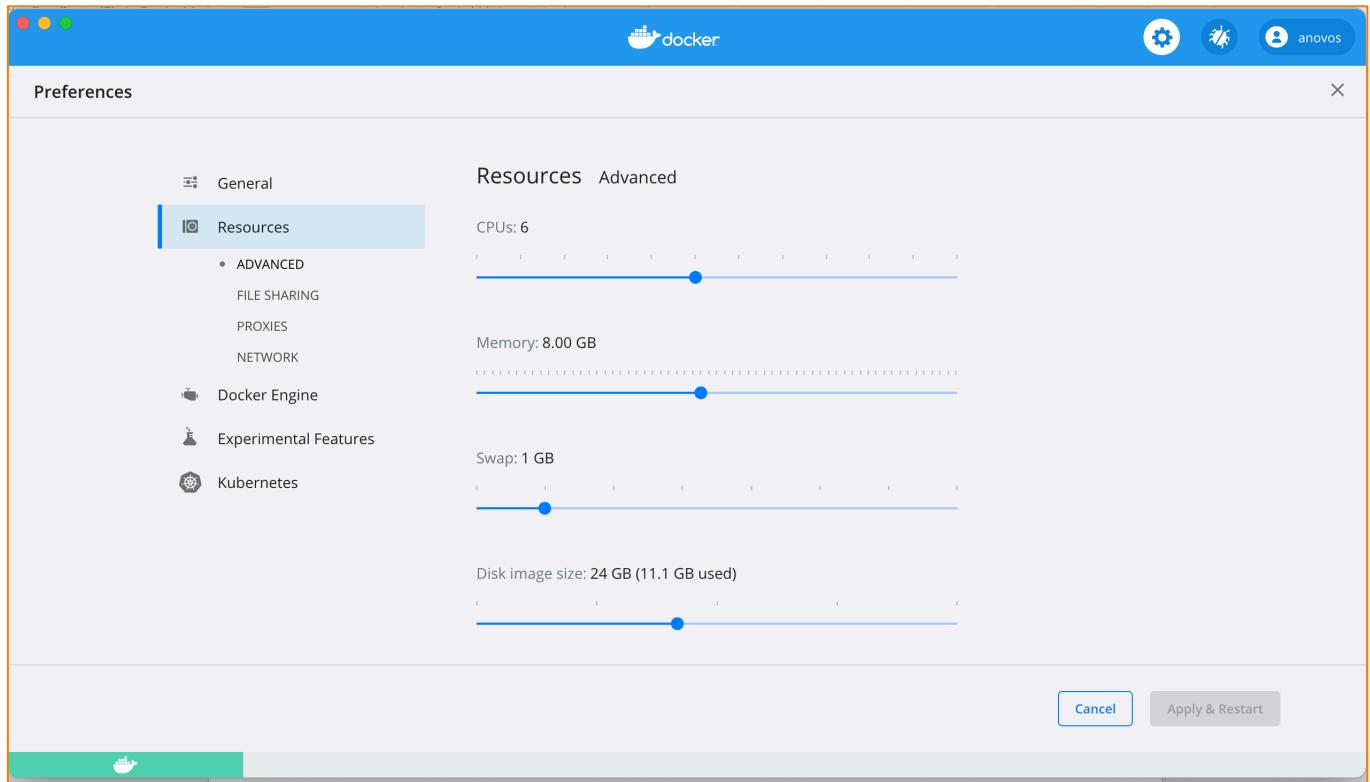


Figure 4 Screenshot showing preferred Docker Resource Allocation

Documentation :

<https://docs.docker.com/desktop/windows/#resources> (Windows)

<https://docs.docker.com/desktop/mac/#resources> (Mac OS)

Note : For native Linux distributions, Docker can run using all available host memory. There isn't a control or setting to limit or increase this. So the above setting is not applicable.

3. Downloading Docker Image

Once the docker installation and resource allocation steps have been completed, the next step is to download the Docker image for the workshop.

Run the following command on command line to download the latest workshop docker image -

```
docker pull anovos/anovos-workshop:latest
```

Once the image has been successfully pulled, kindly confirm by listing the images in your machine

```
docker image ls
```

Please find the screenshot below for the same:

```
mobilewalla@Nguyens-MacBook-Pro Documents % docker pull anovos/anovos-workshop:latest
latest: Pulling from anovos/anovos-workshop
2a750613653e: Already exists
cbc11eac65e1: Already exists
21568fa12a2e: Already exists
f06764e12582: Already exists
35f89b031d0f: Already exists
73b406e3a36a: Already exists
3a0238048a08: Already exists
cc54b2599eca: Already exists
462f6e7a3ab0: Already exists
6aff9a93d21e: Already exists
506374b8022f: Already exists
c02eeeba38cf: Already exists
244bfbdd0739: Already exists
a378658f78e7: Already exists
d8e9c08ebf40: Already exists
4fb7848e2941: Already exists
4f23222aebd4: Already exists
a2631ea8d683: Already exists
08629d84bfbc: Already exists
feea964d650e: Already exists
70edca9bf8c1: Already exists
bcfd911c9cf6: Already exists
7e361b788f3a: Already exists
b8801e919a5b: Already exists
50ff93a1e553: Already exists
015f0b39c06d: Already exists
1abc037e9bad: Already exists
baa0713828c4: Already exists
1ddf93a481f7: Already exists
76045f4daafc: Already exists
63000b8ab8b2: Already exists
a8a1b35ea83c: Already exists
Digest: sha256:41e6e5f4aeb1c65c1eebf3347285b5dee1bf39d8b68c38daa985a5ae29981d55
Status: Downloaded newer image for anovos/anovos-workshop:latest
docker.io/anovos/anovos-workshop:latest
mobilewalla@Nguyens-MacBook-Pro Documents %
mobilewalla@Nguyens-MacBook-Pro Documents % docker image ls
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
anovos/anovos-workshop  latest   f5d1920b639a  15 hours ago  9.49GB
mobilewalla@Nguyens-MacBook-Pro Documents %
```

Figure 5 Screenshot showing docker image successfully pulled to local

It is recommended to download the docker image in advance before the workshop to save time as the image size is ~10GB and could take a while to download.

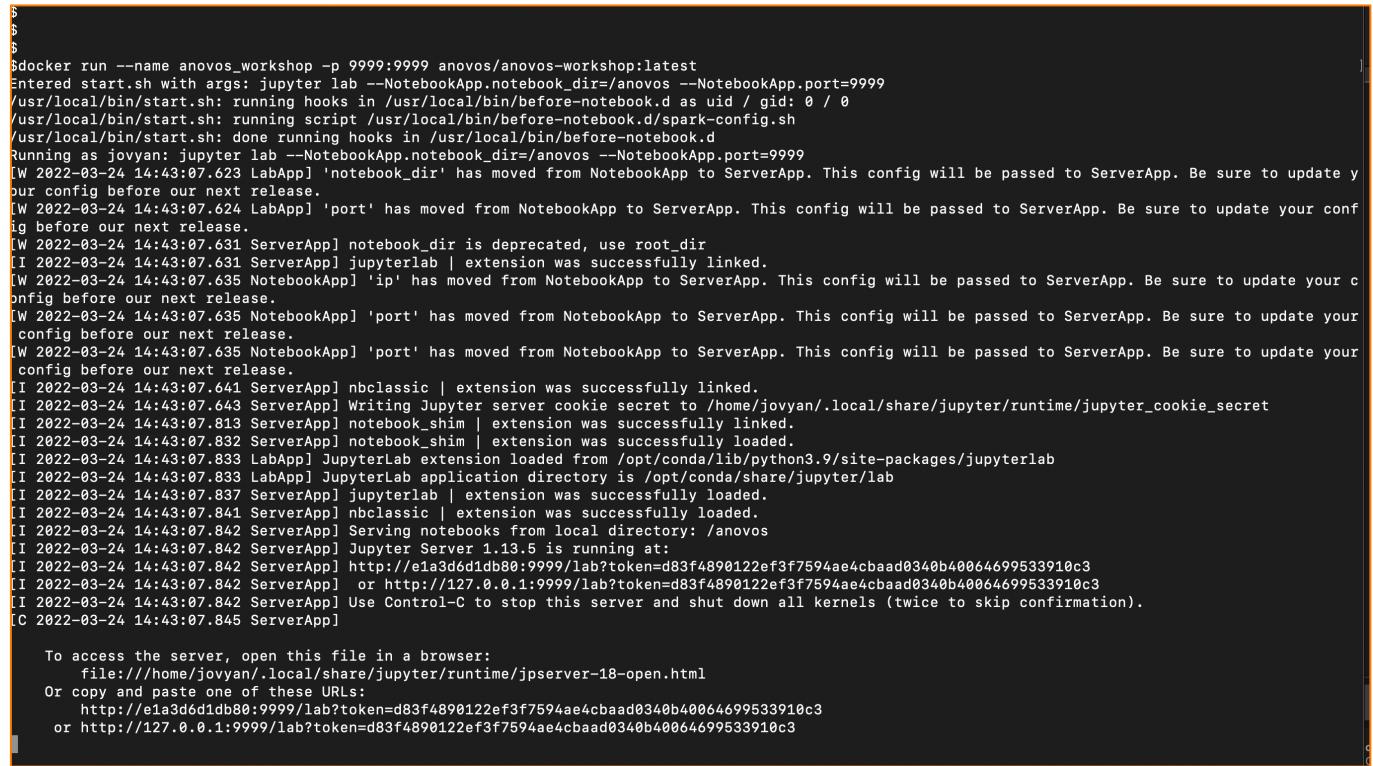
4. Running Docker Image

The final step is to run the docker image.

Note: This step is to be done during the planned workshop/tutorial

Run the following command on command line to run the docker image –

```
docker run --name anovos_workshop -p 9999:9999 anovos/anovos-workshop:latest
```



```
$  
$  
$ docker run --name anovos_workshop -p 9999:9999 anovos/anovos-workshop:latest  
Entered start.sh with args: jupyter lab --NotebookApp.notebook_dir=/anovos --NotebookApp.port=9999  
/usr/local/bin/start.sh: running hooks in /usr/local/bin/before-notebook.d as uid / gid: 0 / 0  
/usr/local/bin/start.sh: running script /usr/local/bin/before-notebook.d/spark-config.sh  
/usr/local/bin/start.sh: done running hooks in /usr/local/bin/before-notebook.d  
Running as jovyan: jupyter lab --NotebookApp.notebook_dir=/anovos --NotebookApp.port=9999  
[W 2022-03-24 14:43:07.623 LabApp] 'notebook_dir' has moved from NotebookApp to ServerApp. This config will be passed to ServerApp. Be sure to update your config before our next release.  
[W 2022-03-24 14:43:07.624 LabApp] 'port' has moved from NotebookApp to ServerApp. This config will be passed to ServerApp. Be sure to update your config before our next release.  
[W 2022-03-24 14:43:07.631 ServerApp] notebook_dir is deprecated, use root_dir  
[I 2022-03-24 14:43:07.631 ServerApp] jupyterlab | extension was successfully linked.  
[I 2022-03-24 14:43:07.635 NotebookApp] 'ip' has moved from NotebookApp to ServerApp. This config will be passed to ServerApp. Be sure to update your config before our next release.  
[W 2022-03-24 14:43:07.635 NotebookApp] 'port' has moved from NotebookApp to ServerApp. This config will be passed to ServerApp. Be sure to update your config before our next release.  
[W 2022-03-24 14:43:07.635 NotebookApp] 'port' has moved from NotebookApp to ServerApp. This config will be passed to ServerApp. Be sure to update your config before our next release.  
[I 2022-03-24 14:43:07.641 ServerApp] nbclassic | extension was successfully linked.  
[I 2022-03-24 14:43:07.643 ServerApp] Writing Jupyter server cookie secret to /home/jovyan/.local/share/jupyter/runtime/jupyter_cookie_secret  
[I 2022-03-24 14:43:07.813 ServerApp] notebook_shim | extension was successfully linked.  
[I 2022-03-24 14:43:07.832 ServerApp] notebook_shim | extension was successfully loaded.  
[I 2022-03-24 14:43:07.833 LabApp] JupyterLab extension loaded from /opt/conda/lib/python3.9/site-packages/jupyterlab  
[I 2022-03-24 14:43:07.833 LabApp] JupyterLab application directory is /opt/conda/share/jupyter/lab  
[I 2022-03-24 14:43:07.837 ServerApp] jupyterlab | extension was successfully loaded.  
[I 2022-03-24 14:43:07.841 ServerApp] nbclassic | extension was successfully loaded.  
[I 2022-03-24 14:43:07.842 ServerApp] Serving notebooks from local directory: /anovos  
[I 2022-03-24 14:43:07.842 ServerApp] Jupyter Server 1.13.5 is running at:  
[I 2022-03-24 14:43:07.842 ServerApp] http://e1a3d6dddb80:9999/lab?token=d83f4890122ef3f7594ae4cbaad0340b40064699533910c3  
[I 2022-03-24 14:43:07.842 ServerApp] or http://127.0.0.1:9999/lab?token=d83f4890122ef3f7594ae4cbaad0340b40064699533910c3  
[I 2022-03-24 14:43:07.842 ServerApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).  
[C 2022-03-24 14:43:07.845 ServerApp]  
  
To access the server, open this file in a browser:  
file:///home/jovyan/.local/share/jupyter/runtime/jpserver-18-open.html  
Or copy and paste one of these URLs:  
http://e1a3d6dddb80:9999/lab?token=d83f4890122ef3f7594ae4cbaad0340b40064699533910c3  
or http://127.0.0.1:9999/lab?token=d83f4890122ef3f7594ae4cbaad0340b40064699533910c3
```

Figure 6 Screenshot showing Docker run being executed

Open the link <http://127.0.0.1:9999/?token...> generated by the Jupyter NotebookApp in a browser to start the Jupyter notebook with Anovos installed.

On starting, You should see a folder titled `use_case_demo/`

Open `anovos_use_case_demo.ipynb` notebook inside the folder.

File Edit View Run Kernel Tabs Settings Help

Launcher anovos_use_case_demo.ipynb

Python 3 (ipykernel)

Anovos Use Case - Credit Risk

Table of Contents

- Introduction
- Exploratory Data Analysis (EDA)
- Feature Identification
- Data Transformation
- Dimensionality Reduction
- Summary

Introduction

Anovos is an open source library for feature engineering at scale. It covers various essential steps in ML pipeline such as exploratory data analysis, anomaly detection, missing value imputation, categorical encoding, drift detection etc.

```

graph TD
    BS[Batch Sources] --> AI[ANOVOS INGEST & ANALYZE]
    AI --> DI[Data Ingest]
    AI --> DA[Data Analysis]
    AI --> FER[Feature explorer/Recommender]
    AI --> LCF[Low-Code Feature Transformer]
    AI --> DPC[Data Pre-Processing, Cleaning & Treatment]
    AI --> DH[Data Health (Drift and Stability Index)]
    AI --> FH[Feature Health]
    
    DS1[Data Scientists] --> EDI[Explore data, clean data & identify drift & stability]
    DS1 --> CHS[Create high value stable features]
    
    DS2[Data Scientists / Data Engineers] --> GTSB[Generate Training Set and Build Models]
    DS2 --> FDC[Feature Drift Check]
    DS2 --> GTDP[Generate Test Data Features and Predict]
    DS2 --> DFS[Define and Store Features]
    
    FER --> FS[Feature Store]
    LCF --> FS
    FS --> P[Predictions]
    
    subgraph FeatureStore [Feature Store]
        FS
    end
    
    DI --> DS1
    DS1 --> CHS
    CHS --> DS2
    DS2 --> GTSB
    GTSB --> FDC
    FDC --> DS2
    DS2 --> GTDP
    DS2 --> DFS
    DFS --> FS
    FS --> P
    P --> DS2
    DS2 --> GTSB
    DS2 --> FDC
    DS2 --> GTDP
    DS2 --> DFS
    DS2 --> FS
    DS2 --> P
  
```

Figure 7 Anovos Use Case Demo Notebook

!!! Enjoy exploring ANOVOS !!!

5. Downloading Anovos Report

Anovos has the capability to generate multiple HTML reports such as Basic Report and Full report as part of its functions.

This will be explained thoroughly in detail during the workshop.

The reports can then be copied to the local machine from the Docker containers using the following command –

```
docker cp container-name:<path in container> <local_path>
```

For example, to copy basic_report.html from the path -

/anovos/use_case_demo/data/output/data_report/report_stats/basic_report.html in container anovos_workshop to local, the following command can be executed

```
docker cp  
anovos_workshop:/anovos/use_case_demo/data/output/data_report/report_stats/basic_report.html ./basic_report.html
```

The downloaded reports can then be opened in a browser (Google chrome

The screenshot shows a web-based report titled "ML-Anovos Report". At the top right is the Anovos logo. Below the title, there are three navigation tabs: "Descriptive Statistics" (which is active), "Quality Check", and "Attribute Associations". A sub-section titled "Global Summary" contains the following information:

- Total Number of Records: 10,149
- Total Number of Attributes: 30
- Number of Numerical Attributes : 23
- Numerical Attributes Name : EXT_SOURCE_2, FLAG_CONT_MOBILE, LIVE_CITY_NOT_WORK_CITY, APARTMENTS_MEDI, CNT_FAM_MEMBERS, CNT_CHILDREN, BASEMENTAREA_MEDI, AMT_REQ_CREDIT_BUREAU_DAY, AMT_INCOME_TOTAL, EXT_SOURCE_3, DAYS_BIRTH, SK_ID_CURR, DAYS_ID_PUBLISH, DAYS_EMPLOYED, AMT_REQ_CREDIT_BUREAU_MON, COMMONAREA_MEDI, REGION_RATING_CLIENT_W_CITY, FLAG_PHONE, AMT_GOODS_PRICE, AMT_CREDIT, TARGET, AMT_ANNUITY, EXT_SOURCE_1
- Number of Categorical Attributes : 7
- Categorical Attributes Name : NAME_INCOME_TYPE, NAME_CONTRACT_TYPE, ORGANIZATION_TYPE, NAME_TYPE_SUITE, CODE_GENDER, NAME_EDUCATION_TYPE, NAME_FAMILY_STATUS

At the bottom left, it says "basic_report created on 24/03/2022, 23:33:26" and at the bottom right is the Datapane logo.

Figure 8 Sample Screenshot of Anovos Basic Report

6. Direct Installation & Run

Note - Kindly skip this step if installation of Docker and run by Docker was successful.

However, for certain cases where a docker based run is not feasible (such as Mac OS on Apple Chip M1 machines),

Kindly install Anovos directly onto the end machine along with the required environment.

Anovos requires installation of supported Python, Java and Spark version as specified in the link.

Python Installation - <https://www.python.org/downloads/>

Spark Installation - <https://spark.apache.org/downloads.html>

Java Installation - <https://docs.oracle.com/en/java/javase/11/install/overview-jdk-installation.html#GUID-8677A77F-231A-40F7-98B9-1FD0B48C346A>

Anovos Installation - <https://github.com/anovos/anovos#using-anovos>

Also, for this workshop kindly install Jupyter Notebook as well - <https://jupyter.org/install>

Once all the above installation is done, kindly download the following .tgz to a local path and extract the tgz - https://mobilewalla-anovos.s3.amazonaws.com/workshop/use_case_demo.tgz

Open the folder on a Jupyter notebook local instance and start anovos_use_case_demo.ipynb

7. Post Workshop Cleanup

Once you have finished exploring Anovos, kindly stop the running docker container on command line by running `docker stop anovos_workshop` in a new terminal tab.

Then kindly check if there are no running containers using the command -

```
docker ps
```

and if the anovos-workshop container has exited properly using the command -

```
docker ps -a
```

 should show status as Exited (0)

Remove the old container using the command -

```
docker rm anovos_workshop
```

The image would still be loaded and by just following #4 Running Docker Image again, we can start a new container with Anovos installed.

If you choose to remove the image entirely from the system, kindly use the following command -

```
docker rmi anovos/anovos-workshop
```

```
$
$docker ps
CONTAINER ID   IMAGE          COMMAND           CREATED          STATUS          PORTS
 NAMES
9fff3f15be55   anovos/anovos-workshop:latest "tini -g -- /bin/bas..." 8 seconds ago   Up 6 seconds   8888/tcp, 0.0.0.0:9999->9999/tcp, :::9999->9999/
tcp   anovos_workshop
$ 
$ 
$ 
$ 
$docker stop anovos_workshop
anovos_workshop
$ 
$ 
$docker ps
CONTAINER ID   IMAGE          COMMAND           CREATED          STATUS          PORTS
$ 
$ 
$docker ps -a
CONTAINER ID   IMAGE          COMMAND           CREATED          STATUS          PORTS          NAMES
9fff3f15be55   anovos/anovos-workshop:latest "tini -g -- /bin/bas..." 30 seconds ago  Exited (0) 7 seconds ago
$ 
$ 
$ 
$docker rm anovos_workshop
anovos_workshop
$ 
$ 
$docker ps -a
CONTAINER ID   IMAGE          COMMAND           CREATED          STATUS          PORTS
$ 
$docker image ls
REPOSITORY      TAG      IMAGE ID      CREATED          SIZE
anovos/anovos-workshop  latest  d3951a7e336d  4 hours ago  9.170B
$ 
$
```

Figure 9 Screenshot showing steps to remove Docker container