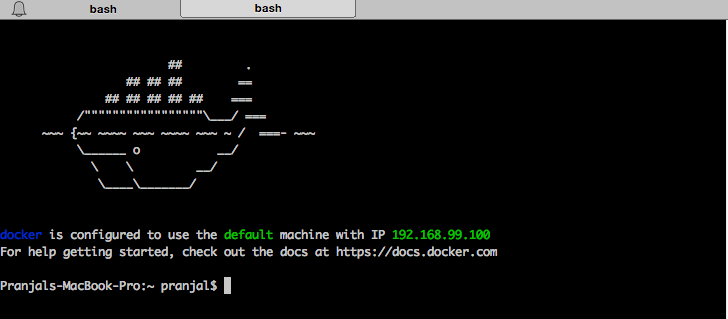
Installation

1. Go to <https://www.docker.com/products/overview> and download docker based on your operating system.
2. Install docker like a normal software program.

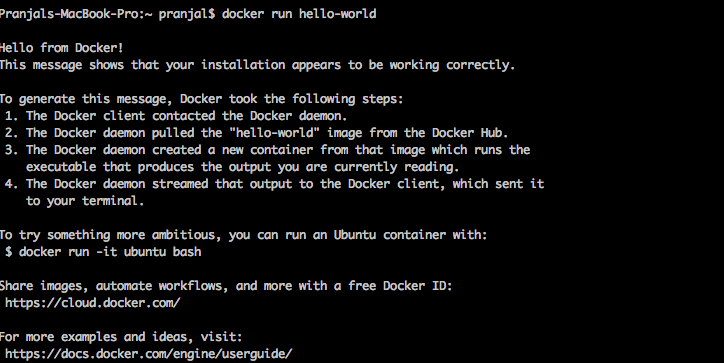
How to start docker?

1. Type docker in quickstart terminal . It starts the docker in terminal. First time it will install the docker image , create a virtualbox to manage various containers.



2. To check the installation was successful. Let’s run the hello-world container from docker.

|  |
| --- |
| docker **run** hello-world |



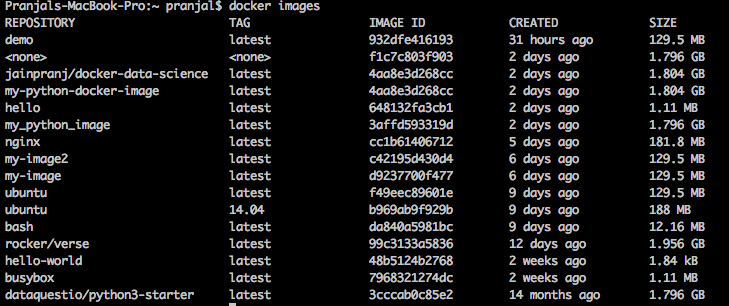
3. Run a docker container with ubuntu image.

|  |
| --- |
| docker **run** -ti unbuntu bash |



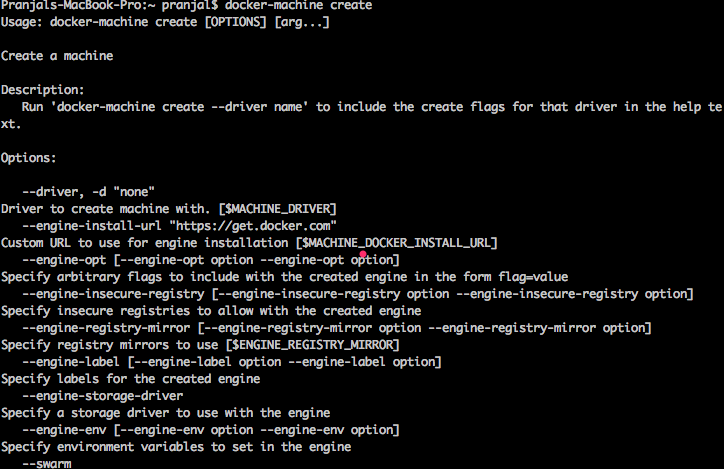
4. List all the docker images

|  |
| --- |
| docker images |



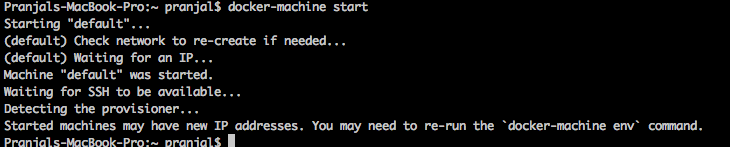
5. Create a new docker machine

|  |
| --- |
| docker-machine create |



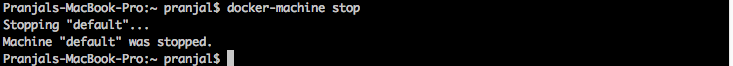
6. Start docker-machine

|  |
| --- |
| docker-machine start |



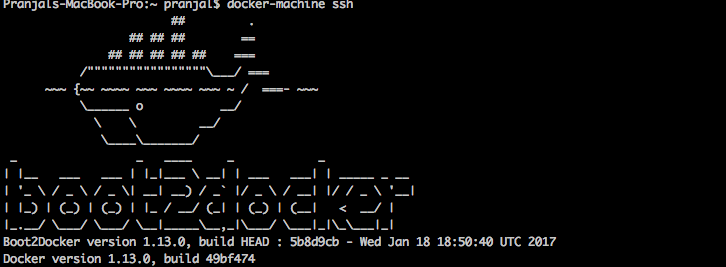
7. Stop docker-machine

|  |
| --- |
| docker-machine stop |



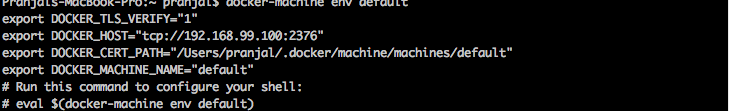
8. Connect to docker machine

|  |
| --- |
| docker-machine ssh |



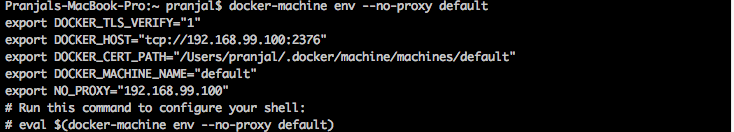
9. Setting docker machine as default.

|  |
| --- |
| docker-machine **env** default |



10. Enabling internet access on docker machine by setting it to no proxy.

docker-machine env --no-proxy default



11. Run a nginx server on docker machine

docker run -d -p 8000:80 nginx

-p implies the port number. The first port number is internal port and other is external port.

-d-implies running container in detached mode.



12. Connecting to nginx server

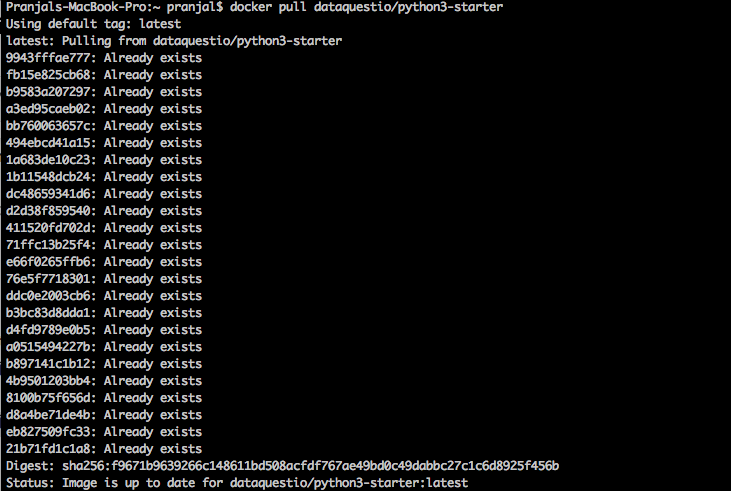
$(docker-machine ip default):8000

Data Science setup with docker.

Connecting to jupyter notebooks using docker container.

1. Download python image. I am downlading image from dataquest.io

docker pull dataquestio/python3-starter



2. Creating a local directory to store python notebooks

|  |
| --- |
| cd docker/ mkdir notebooks\_docker |

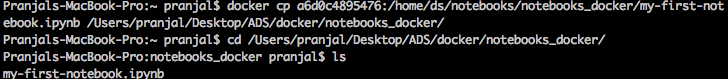
3. Running juypter from the image downloaded

|  |
| --- |
| docker **run** -d -p 8888:8888 -v /Users/pranjal/Desktop/ADS/docker:/home/ds/notebooks dataquestio/python3-starter |



4. Copying notebooks from docker container to local.

|  |
| --- |
| docker cp a6d0c4895476:/home/ds/notebooks/notebooks\_docker/my-first-notebook.ipynb /Users/pranjal/Desktop/ADS/docker/notebooks\_docker/ |

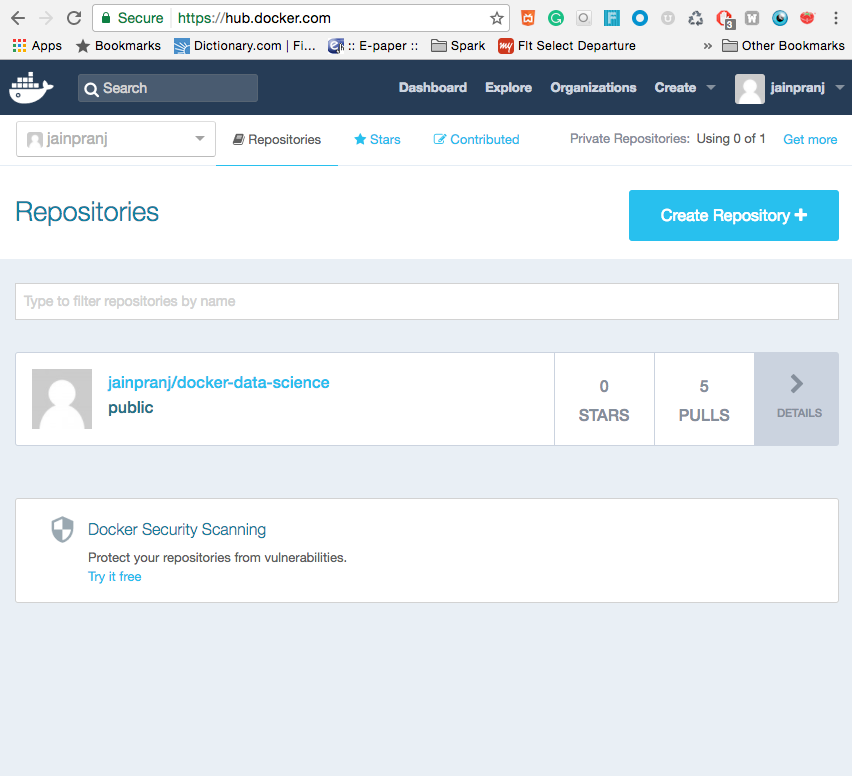


Pushing a docker image to docker hub.

Docker Hub

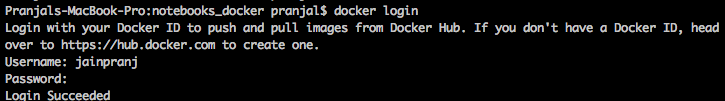
It is similar to Github, a place to push and pull docker images.

1. Create a account on docker Hub. Create a repository.



2. Connect to docker hub from docker. It prompts for username and password.

docker login



3. First commit the image with changes using

|  |
| --- |
| docker commit a6d0c4895476192605cd0309d5f8181c49f2117185d1841e6f16e8ccb621ef13 |

|  |
| --- |
| docker commit container-name |



4. Tagging the committing image with repository

|  |
| --- |
| docker tag my-python-docker-image jainpranj/docker-data-science:latest |

|  |
| --- |
| docker tag <image-to-be-committed> <repository name> |



5. Pushing the image to docker hub

|  |
| --- |
| docker push <repository-name> docker push jainpranj/docker-data-science:latest |



Creating a Dockerfile

1. Create a file and enter following command

|  |
| --- |
| touch Dockerfile **FROM** debian:sid **RUN** echo "building simple docker image" **CMD** echo "Hello Container" |

|  |
| --- |
| **FROM**- implies repository **from** which image needs to be downloaded **RUN**- to run the docker command **CMD**- To run other linux command **EXPOSE**- to **expose** any ports **ENV**- to set environment variables |

2. Building a docker file

|  |
| --- |
| docker build -t hello . |

|  |
| --- |
| docker build -t <container-name> |

. -implies current directory>

3. Run the container

|  |
| --- |
| docker **run** hello |