



## Docker - Installation on Linux

# Install process

- Connect to Linux
- Install Docker
- Start Docker
- Stop Docker
- Uninstall Docker

To install docker, OS should be 64-bit of any flavour

## Links

- <https://docs.docker.com/install/>
- <https://labs.play-with-docker.com/>
- <http://docs.aws.amazon.com/AmazonECS/latest/developerguide/docker-basics.html>

# Installing Docker engine on Ubuntu

`$uname -a` ---> to get OS(AMI) details

`#sudo su root` ----> to switch to root

`$sudo apt-get update` → if you didn't switch to root user, **use sudo before your command**

`#apt-get remove docker docker-engine` → to remove existing docker if available

`#curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -`  
--- for adding GPG key for docker repository

add the docker repository

```
#sudo add-apt-repository "deb [arch=amd64]
```

```
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"
```

```
#apt-get update → update package
```

```
#apt-get install docker-ce -y → install Docker
```

```
#service docker start → to start docker
```

```
#service docker stop → to stop docker
```

```
#docker --version → to know which version installed
```

```
#docker → to list all options
```

```
#docker info
```

```
#docker images → to list all images available
```

```
#docker ps → to list all not running docker containers
```

```
#docker ps -a → to list all running containers
```

#docker run <docker image name> to install and run container

#docker images → to list all images

# Installing docker on Linux

`$uname -r`

`$sudo yum -y update`

`$sudo yum install -y docker` → to install docker

`$docker -->` for docker commands

`$docker --version -->` for checking which version is installed

`$docker info -->` gives error, cause docker is not running

`$sudo service docker start` → to start docker

`$docker info --> for getting docker info`

`$sudo usermod -a -G docker ec2-user`

`$docker images --> to list all images`

`$docker ps --> to list all containers (not running)`

`$docker ps -a --> to list containers(running)`

`$docker run hello-world --> to run image / if image is not present in you hub, it goes and get from central`

`$docker images --> to list docker images`

`$sudo service docker stop → to stop docker`

# Docker Images

What are images

How to pull images

How to run a container using an image

Basic commands

How to write a dockerfile



`$docker -v` ---> to check docker is installed or not

`$docker images` → to list all images existing on this instance

`$docker images --help` → for getting more info regarding image command

Go to docker hub for checking images, and search for image and pull it

`$docker pull <image name>` → for pulling image (latest)

`$docker pull <image name>:tag` ---> for pulling image with particular image

Docker images are templates used to create docker containers. Container is a running instance of an image

`$docker images -q --> shows only numeric image ID`

`$docker images -f "dangling=false" ---> check images associated to a container running`

`$docker images -f "dangling=true" → check images associated to a container not running`

`$docker images -f "dangling=true" -q`

`$docker run --name MyUbuntu -it ubuntu bash --> you are now inside ubuntu`

`$docker ps -a`

\$docker inspect <image name> --> to see image file content

\$docker rmi <image name> --> to delete image (it is possible only when container is not running)

\$docker stop <container> --> to stop container

\$docker rmi -f <image name> → to remove forcefull