```
docker pull nginx #Pull Nginx
docker run --name docker-nginx -p 80:80 nginx #Expose Nginx 80 Port
docker run --name docker-nginx -p 8080:80 -d nginx #Expose 8080
docker run -P nginx
docker run -d -P nginx

docker build -t friendlyname .              # Create image using this directory's Dockerfile
docker run -p 4000:80 friendlyname          # Run "friendlyname" mapping port 4000 to 80
docker run -d -p 4000:80 friendlyname       # Same thing, but in detached mode
docker run --name test-ubuntu -it ubuntu:16.04 ./bin/bash
docker exec -it [container-id] bash         # Enter a running container
docker ps                                   # See a list of all running containers
docker stop <hash>                          # Gracefully stop the specified container
docker ps -a                                # See a list of all containers, even the ones not running
docker kill <hash>                          # Force shutdown of the specified container
docker rm <hash>                            # Remove the specified container from this machine
docker rm $(docker ps -a -q)                # Remove all containers from this machine
docker images -a                            # Show all images on this machine
docker rmi <imagename>                      # Remove the specified image from this machine
docker rmi $(docker images -q)              # Remove all images from this machine
docker logs <container-id> -f               # Live tail a container's logs
docker login                                # Log in this CLI session using your Docker credentials
docker tag <image> username/repository:tag  # Tag <image> for upload to registry
docker push username/repository:tag         # Upload tagged image to registry
docker run username/repository:tag          # Run image from a registry
docker system prune                         # Remove all unused containers, networks, images (both dangling and unreferenced), and optionally, volumes. (Docker 17.06.1-ce and superior)
docker system prune -a                      # Remove all unused containers, networks, images not just dangling ones (Docker 17.06.1-ce and superior)
docker volume prune                         # Remove all unused local volumes
docker network prune                        # Remove all unused networks

cd usr/share/nginx/html/

docker volume create my_vol                 # Create a volume
docker volume ls
docker volume inspect my_vol                # troulbeshooting
docker volume rm my_vol




##Setup Docker in EC2
Allows access to port 80 (HTTP) from anywhere
HTTP  TCP  80 Anywhere
sudo yum update -y
sudo yum install -y docker
sudo service docker start
sudo usermod -aG docker ec2-user
```