



by Thom Ives, PhD

Consolidated Docker Commands Master Procedure

NOTE: I use NAME_i for the image name, and NAME for the container name. This way I can QUICKLY edit this file for new image / container pairs.

- To start clean on your local machine
 - `sudo docker image rm NAME_i:latest`
 - `sudo docker image rm docker_hub_username/NAME_i:latest`
- Build the image from your latest Dockerfile
 - `sudo docker image build -t NAME_i:latest .`
- Give the image a tag that works for Docker Hub
 - `sudo docker tag NAME_i:latest docker_hub_username/NAME_i:latest`
- Log into Docker Hub IF necessary
 - `sudo docker login`

- Push the image up to Docker Hub
 - `sudo docker push docker_hub_username/NAME_i:latest`
- Go to the server that needs the image and pull the image down to it from Docker Hub
 - `sudo docker pull docker_hub_username/NAME_i:latest`
- Run the image with the necessary flags
 - `sudo docker container run --name NAME --restart unless-stopped -p host_port_number:container_port_number -v /host/directory/path:/container/directory/path -d docker_hub_username/NAME_i`
- Stop the container and remove it when you need to rerun a new image
 - `sudo docker container stop NAME`
 - `sudo docker container rm NAME`
- Rarely used
 - `sudo docker container restart NAME`
- [How to Interact with a Running Container](#)
 - <https://stackoverflow.com/questions/63663307/start-docker-container-interactively>
 - `sudo docker exec -it NAME /bin/bash`
- [How to Detach from your Interactive Session with a Running Container](#)
 - <https://stackoverflow.com/questions/25267372/correct-way-to-detach-from-a-container-without-stopping-it>
 - use CTRL-p, CTRL-q to exit the bash of the container
- [How to Install New Things in Container and Actually Save Them!](#)
 - <https://stackoverflow.com/questions/63027514/install-package-in-running-docker-container>
- [The IP address of the Host of a Docker Container IS 172.17.0.1](#)
 - <https://stackoverflow.com/questions/22944631/how-to-get-the-ip-address-of-the-docker-host-from-inside-a-docker-container>

Another Format for Docker Commands

Naming Tip from Thom

For the commands below, I like to name the image closely to the same name as the container. However, I choose a container name first, and then name the image `container_name_i`. So, my image name would usually be `container_name_i` with a tag of `latest` = `container_name_i:latest`. Then, my container name is `container_name`. Also, I can copy and paste the commands below to a new directory for a new Docker image / container project, and then just replace the `container_name` for these commands below, and then I have all my commands ready for copy and paste - voila!

To start clean on your local machine

```
sudo docker image rm image_name:tag sudo docker image rm
docker_hub_username/image_name:tag
```

Build the image from your latest Dockerfile

```
sudo docker image build -t image_name:tag .
```

Give the image a tag that works for Docker Hub

```
sudo docker tag image_name:tag docker_hub_username/image_name:tag
```

Log into Docker Hub IF necessary

```
sudo docker login
```

Push the image up to Docker Hub

```
sudo docker push docker_hub_username/image_name:tag
```

Go to the server that needs the image and pull the image down to it from Docker Hub

```
sudo docker pull docker_hub_username/image_name:tag
```

Run the image with the necessary flags

```
sudo docker container run --name container_name --restart unless-stopped
-p 5001:80 -v /home/tives@hq.echogl.net/elf/ai_data:/usr/share/nginx/html
-d docker_hub_username/image_name:tag
```

Stop the container and remove it when you need to rerun a new image

```
sudo docker container stop container_name sudo docker container rm
container_name
```

Rarely used

```
sudo docker container restart container_name
```

POWER Docker Commands

Docker Removal Commands

Modify these for your specific needs ...

To STOP running containers: `sudo docker container stop $(sudo docker container ls -aq)`

To STOP SOME containers: `for container_name in 'name1' 'name2' 'name3'; do sudo docker container stop $container_name; done`

To remove ALL Stopped Docker Containers: `sudo docker container prune`

To remove SOME Stopped Docker Containers: `for container_name in 'name1' 'name2' 'name3'; do sudo docker container rm $container_name; done`

To remove ALL Docker Images: `sudo docker image rm -f $(sudo docker images -aq)`

To Remove ALL Unused Docker Images: `sudo docker image prune`

To Pull Multiple Images at one time: `for image in docker_hub_username/name1_i:latest docker_hub_username/name2_i:latest docker_hub_username/name3_i:latest docker_hub_username/name4_i:latest docker_hub_username/name5_i:latest docker_hub_username/name6_i:latest; do sudo docker pull $image; done`

`for image in docker_hub_username/name1_i:latest docker_hub_username/name2_i:latest docker_hub_username/name3_i:latest; do sudo docker pull $image; done`

To Run Multiple Containers at one time: `for run_cmd in '--name name1 --restart always -p 8000:8000 -d docker_hub_username/name1_i' '--name file_io --restart always -p 8001:8001 -d docker_hub_username/name2_i' '--name name3 --restart always -p 8002:8002 -v /host/dir:/container/dir -d docker_hub_username/name3_i' '--name name4 --restart always -p 8003:8003 -d docker_hub_username/name4_i' '--name name5 --restart always -p 5001:80 -v /host/dir:/container/dir -d docker_hub_username/name5_i' '--name name6 --restart always -p 5000:80 -d docker_hub_username/name6_i'; do sudo docker container run $run_cmd; done`

`for run_cmd in '--name name1 --restart always -p 8000:8000 -d docker_hub_username/name1' '--name name2 --restart always -p 8001:8001 -d docker_hub_username/name2_i' '--name name3 --restart always -p 8002:8002 -v /host/dir:/container/dir -d docker_hub_username/name3_i'; do sudo docker container run $run_cmd; done`

`for run_cmd in '--name name4 --restart always -p 8002:8002 -v /host/dir:/`

```
container/dir -d docker_hub_username/name4_i' '--name name5 --restart  
unless-stopped -p 5001:80 -v /host/dir:/container/dir -d  
docker_hub_username/name5_i'; do sudo docker container run $run_cmd; done
```

Other STRONG commands are: * sudo docker volume prune * sudo docker network
prune * sudo docker system prune