Docker File Tags and commands:

1. **FROM**

# all images must have a FROM

# usually from a minimal Linux distribution like debian or (even better) alpine

# if you truly want to start with an empty container, use FROM scratch

1. **ENV:**

#optional environment variable that's used in later lines and set as envvar when container is running

1. **RUN**:

#optional commands to run at shell inside container at build time

1. **EXPOSE**:

# expose these ports on the docker virtual network

# you still need to use -p or -P to open/forward these ports on host

1. **CMD**:

# required: run this command when container is launched

# only one CMD allowed, so if there are multiple, last one wins

1. **WORKDIR:**

# to mention the working directory when container is up and running.

1. **COPY path/file conatiner\_path:**

# command is use to copy the files in the container from local host

If we don’t specify **EXPOSE** or **CMD** then it is taken from **FROM** command

Once we have our docker files ready. Once we start building image whatever we have in from command is downloaded in local cache. And commands are executed as mentioned in docker files.

**Building a docker image from docker\_file**

Command to build a docker image from dockerfile:

**Docker image build - f (docker\_dile\_path ) -t “name\_of\_docker\_image” “pth where to build the image”**

**Ex: docker image build -f [./Home/docker/Docker-File] -t custom-nginx .**

If -f is not mentioned it will start looking for a file with the name “Dockerfile” in the current directory.

\*\*\* note if we want to use the image locally we can name image anything.

But if we want to upload it to docker hub with docker push command

Name the image in format “**user\_name\_docker\_hub/repository\_name:version”**

By default to build the docker image from docker file docker file is read from top left to bottom right.