**Here explicitly this document focuses on the steps needed to create Docker image.**

**Step 1:** Install Docker Desktop on CentOS Linux release (7.9.2009)

Docker Official page to download Docker Desktop executable file and install on Linux.

<https://docs.docker.com/desktop/install/linux-install/>

**Step 2:**

Once installation completes you very the docker version as shown below



Step 3:

Get the application build zip file "**application.zip**" and unzip it

**command:** unzip application-2.0.0.zip -d applictionfilesdirectorysss

Step 3:

Now create a **Dockerfile** to create docker Image, here you need to define all the configuration/dependencies/required files script.

# Use Alpine Linux as the base image

FROM alpine:latest

COPY applicationfilesdirectory /opt

WORKDIR /opt

# Update package lists and install necessary packages

RUN apk update && \

    apk add --no-cache curl && \

    rm -rf /var/cache/apk/\*

RUN apk add --no-cache bash && \

    apk add --no-cache dos2unix && \

    apk add --no-cache python3 && \

    apk --update add fontconfig ttf-dejavu

RUN chmod 777 -R /opt

RUN apk --no-cache add openjdk17

WORKDIR /opt/bin

#set the entry point to execute run.sh file

ENTRYPOINT ["./run"]

Step4:

To create docker image out of above defined “Dockerfile” need to execute below commands on linux VM.

1. **cd <specific directory where Dockerfile resides>**
2. Execute “**sudo docker build .”** (it builds the image successfully)
3. **sudo docker images** (checks images in docker)
4. **sudo docker tag <docker\_imageid><space****>****<dockerimage\_name>:<Tag\_name>** (Give name and tag to your docker image)

for example : sudo docker tag 482705f00b10 webapp:2.0.0

1. **sudo docker images** (now check the image name and tag you assigned)
2. **sudo docker run -it** **<dockerimage\_name>:<Tag\_name>** (run the docker image instance in interactive mode, it makes you to login to docker container, by executing “exit” command you can log out from docker container)

example: sudo docker run -it webapp:2.0.0

1. sudo docker ps (to check the docker container status)
2. sudo docker container stop <docker\_containerID>(to stop docker container)
3. sudo docker container start <docker\_containerID>(to start docker container)

**Now Pushing created docker image to nexus**

1. sudo docker login -u=<NexusUser\_tokenid> -p=<NexusPassword\_tokenid> <Nexus\_url> (log in to nexus)

example: sudo docker login -u=Username -p=pwdtoken maven-dev-hosted.nexus.india.com

1. sudo docker tag <docker\_imagename:tag> <Nexus\_url>/<docker\_imagename:tag> (tag the dock er image in nexus)

example: sudo docker tag webapp:2.0.0 maven-dev-hosted.nexus.india.com / webapp:2.0.0

1. sudo docker push <Nexus\_url>/<docker\_imagename>:<tag\_name>(push image to nexus)
2. sudo docker pull <Nexus\_url>/<docker\_imagename>:<tag\_name>(pull image from nexus)