**Sample Dockerfile**

**~ /root/nabajyoti**

**vi Dockerfile**

*# Docker file for building image with httpd and tomcat in start mode and one index.html file customized in our httpd (webserver)*

**FROM ubuntu:latest**

**WORKDIR /root/nabajyoti**

**LABEL webserver="httpd"**

**LABEL appserver="tomcat"**

**MAINTAINER Nabajyoti Modak**

**RUN apt-get -y update &&\**

**apt-get -y upgrade**

**# ARG [variable=value]; Here we don't need**

*# Installing httpd and copying one index.html file to /var/www/html*

**RUN apt-get -y install httpd**

**RUN sudo service start -y httpd**

**COPY /root/naba/index.html /var/www/html**

*# Installing apache-tomcat in /usr/local/tomcat path*

**RUN mkdir /usr/local/tomcat**

**RUN wget http://tomcat.9.0.62.............**

**#RUN tar -xvzf apache-tomcat-9.0.62.tar.gz**

**#COPY /root/nabajyoti/apache-tomcat-9.0.62/\* /usr/local/tomcat**

*# Or else directly we can use* ***ADD*** *syntax also*

**ADD /root/nabajyoti/apache-tomcat-9.0.62.tar.gz /usr/local/tomcat**

*# Starting apache-tomcat*

**RUN chmod +x usr/local/tomcat/apache-tomcat-9.0.62/bin/sh**

**#ENV [username=admin, key=value]; Here it's not needed**

**ENTRYPOINT usr/local/tomcat/apache-tomcat-9.0.62/bin/sh**

**CMD startup.sh run**

*# If we want to avoid the entry point, we can give the CMD as below:*

**#CMD usr/local/tomcat/apache-tomcat-9.0.62/bin/startup.sh run**

**EXPOSE 8080**

The concept I gathered in Docker file from your class and browsing is the same that I wrote above. So here I combine both httpd and tomcat. In one place I found that we can run 2 apps. in 1 container but the maintenance become hectic. So, for sample purpose I prepared this.

**Kindly have a look in the file and please inform me, whether it is correct or not Sir……..**

**# Docker file for Nginx and copying a index.html file to nginx && Installing Jboss and running of Jboss**

**FROM Alpine**

**WORKDIR /root/naba**

**LABEL webserver="nginx"**

**LABEL appserver="Jboss"**

**MAINTAINER Nabajyoti Modak**

**RUN apk -y update &&\**

**apk -y add nginx**

**RUN service start -y nginx**

**COPY /root/naba/index.html /usr/share/nginx/html**

**RUN mkdir /opt/Jboss &&\**

**cd /opt/Jboss**

**RUN wget https://download.jboss.org/jbossas/7.1/jboss-as-7.1.1.Final/jboss-as-7.1.1.Final.zip**

**RUN tar -xvzf jboss-as-7.1.1.Final.zip**

**RUN chmod +x opt/jboss/jboss-as-7.1.1.Final/bin/sh**

**ENTRYPOINT [ "opt/jboss/jboss-as-7.1.1.Final/bin/sh" ]**

**CMD [ "run.sh run" ]**

**EXPOSE 8080:9900**

Installation of Nginx in Alpine Linux:

[apk update]

[apk add nginx] 🡪 Install nginx

[service nginx start] 🡪 Start nginx

**# Docker File for python App**

**FROM ubuntu-slim-buster As base**

**WORKDIR /root/naba/app**

**LABEL app="python"**

**LABEL Type="appserver"**

**LABEL App-file-name="app.py"**

**MAINTAINER Nabajyoti Modak**

**RUN apt-get update -y &&\**

**apt-get -y install sudo**

**RUN apt install software-properties-common -y**

**RUN add-apt-repository ppa:deadsnakes/ppa -y**

**ARG Version=3.8                                  # if needed**

**RUN apt-get install python -y                    # FROM python:3.8-slim-buster**

**FROM ubuntu-slim-buster As build**

**COPY requirements.txt requirements.txt**

**RUN pip3 install -r requirements.txt**

**FROM build As stage**

**RUN cd /opt/Jboss**

**RUN wget https://download.jboss.org/jbossas/7.1/jboss-as-7.1.1.Final/jboss-as-7.1.1.Final.zip**

**RUN tar -xvzf jboss-as-7.1.1.Final.zip**

**RUN chmod +x opt/jboss/jboss-as-7.1.1.Final/bin/sh**

**FROM base As final**

**ENTRYPOINT [ "opt/jboss/jboss-as-7.1.1.Final/bin/run.sh run" ]**

**COPY /root/appserver/app.py server/default/deploy**

**# Use ADD id the file is in zipped or tar format**

**CMD [ "python3", "-m" , "flask", "run" ]**

**EXPOSE 9090**

**# Dockerfile for Java Application**

**FROM openjdk:16-alpine3.13 As base-image**

**WORKDIR /root/naba/app**

**RUN apk -y add maven**

**COPY mvnw pom.xml ./**

**FROM openjdk:16-alpine3.13 As build-stage**

**RUN ./mvnw dependency:go-offline**

**RUN git clone https://git\_repository/naba-app.git**

**ENTRYPOINT [ "./mwnv" ]**

**FROM build As stage-1**

**RUN cd /opt/Jboss**

**RUN wget https://download.jboss.org/jbossas/7.1/jboss-as-7.1.1.Final/jboss-as-7.1.1.Final.zip**

**RUN tar -xvzf jboss-as-7.1.1.Final.zip**

**RUN chmod +x opt/jboss/jboss-as-7.1.1.Final/bin/sh**

**FROM base As final-build**

**ENTRYPOINT [ "opt/jboss/jboss-as-7.1.1.Final/bin/run.sh run" ]**

**RUN mv \*.war /app.war**

**COPY /root/appserver/app.war server/default/deploy**

**CMD [ "spring-boot:run" ]**

**EXPOSE 9091**