



DOCKERFILE

PART : 4



BY : JAYA SHARMA AWASTHI



**A DOCKERFILE IS A TEXT
DOCUMENT THAT CONTAINS ALL
THE COMMANDS A USER COULD
CALL ON THE COMMAND LINE TO
ASSEMBLE AN IMAGE.**



Syntax for writing a Dockerfile step by step

1

FROM:

A FROM statement defines which image to download and start from. It must be the first command in your Dockerfile

2

MAINTAINER

This statement is a kind of documentation, which defines the author who is creating this Dockerfile

3

RUN

The RUN statement defines running a command through the shell, waiting for it to finish, and saving the result.

Syntax for writing a Dockerfile step by step

4

ADD

If we define to add some files, ADD statement is used. It basically gives instructions to copy new files, directories, or remote file URLs and then adds them to the filesystem of the image.

5

ENV

ENV statement sets the environment variables both during the build and when running the result. It can be used in the Dockerfile and any scripts it calls.

6

ENTRYPOINT

It specifies the starting of the expression to use when starting your container.

Syntax for writing a Dockerfile step by step

7

CMD

CMD specifies the whole command to run. We can say CMD is the default argument passed into the ENTRYPOINT. The main purpose of the CMD command is to launch the software required in a container.

8

EXPOSE

EXPOSE statement maps a port into the container. The ports can be TCP or UDP but by default, it is TCP.

9

VOLUME

The VOLUME statement defines shared volumes or ephemeral volumes depending upon whether you have one or two arguments.

Syntax for writing a Dockerfile step by step

10

WORKDIR

As the name suggests, WORKDIR sets the directory that the container starts in. Its main purpose is to set the working directory for all future Dockerfile commands.

11

USER

It sets which user's container will run as. This can be useful if you have shared network directories involved that assume a fixed username or a fixed user number.

EXAMPLE OF DOCKERFILE FOR A NODE PROJECT IMAGE

```
FROM node:18.7.0

WORKDIR /code

COPY package.json package.json
COPY package-lock.json package-lock.json

RUN npm install

COPY . .

CMD [ "node", "server.js" ]
```