1….FROM

The base image to use in the build. This is mandatory and must be the first command in the file.

…………FROM Ubuntu……………………

2.MAINTAINER

An optional value for the maintainer of the script

……………………….sekharreddychanda@gmail.com……………..

3.RUN

Executes a command and save the result as a new layer

FROM ubuntu

RUN apt-get update update apt-get install php5

..

….RUN touch Sekhar.txt

RUN mkdir sekharreddy

CMD ping localhost

EXAMPLE:

FROM ubuntu:trusty

CMD ["/bin/ping","localhost"]

Rebuild the image and look at the command that is generated for the running container:

$ docker build -t demo .

[truncated]

$ docker run -d demo

90cd472887807467d699b55efaf2ee5c4c79eb74ed7849fc4d2dbfea31dce441

$ docker ps -l

CONTAINER ID IMAGE COMMAND CREATED

90cd47288780 demo:latest "/bin/ping localhost" 4 seconds ago

4..CMD

The command that runs when the container starts

CMD "echo" "Hello World!"

5..COPY

COPY SEKHAR.txt /usr/temp

6…ENTRYPOINT

An ENTRYPOINT helps you to configure a container that you can run as an executable.

RUN apt-get install python3

CMD echo "Hello world"

ENTRYPOINT echo "Hello world"

ENV name John Dow

ENTRYPOINT echo "Hello, $name"

**Exec form**

This is the preferred form for CMD and ENTRYPOINT instructions.

<instruction> ["executable", "param1", "param2", ...]

Examples:

RUN ["apt-get", "install", "python3"]

CMD ["/bin/echo", "Hello world"]

ENTRYPOINT ["/bin/echo", "Hello world"]

When instruction is executed in *exec* form it calls executable directly, and shell processing does not happen. For example, the following snippet in Dockerfile

ENV name John Dow

ENTRYPOINT ["/bin/echo", "Hello, $name"]

when container runs as docker run -it <image> will produce output

Hello, $name

Note that variable *name* is not substituted.

**How to run bash?**

If you need to run *bash* (or any other interpreter but sh), use *exec* form with /bin/bash as executable. In this case, normal shell processing will take place. For example, the following snippet in Dockerfile

ENV name John Dow

ENTRYPOINT ["/bin/bash", "-c", "echo Hello, $name"]

when container runs as docker run -it <image> will produce output

Hello, John Dow

7….USER

**USER** SEKHAR

8..ADD

9..ENV

**ENV** CONT\_IMG\_VER v1.0.0