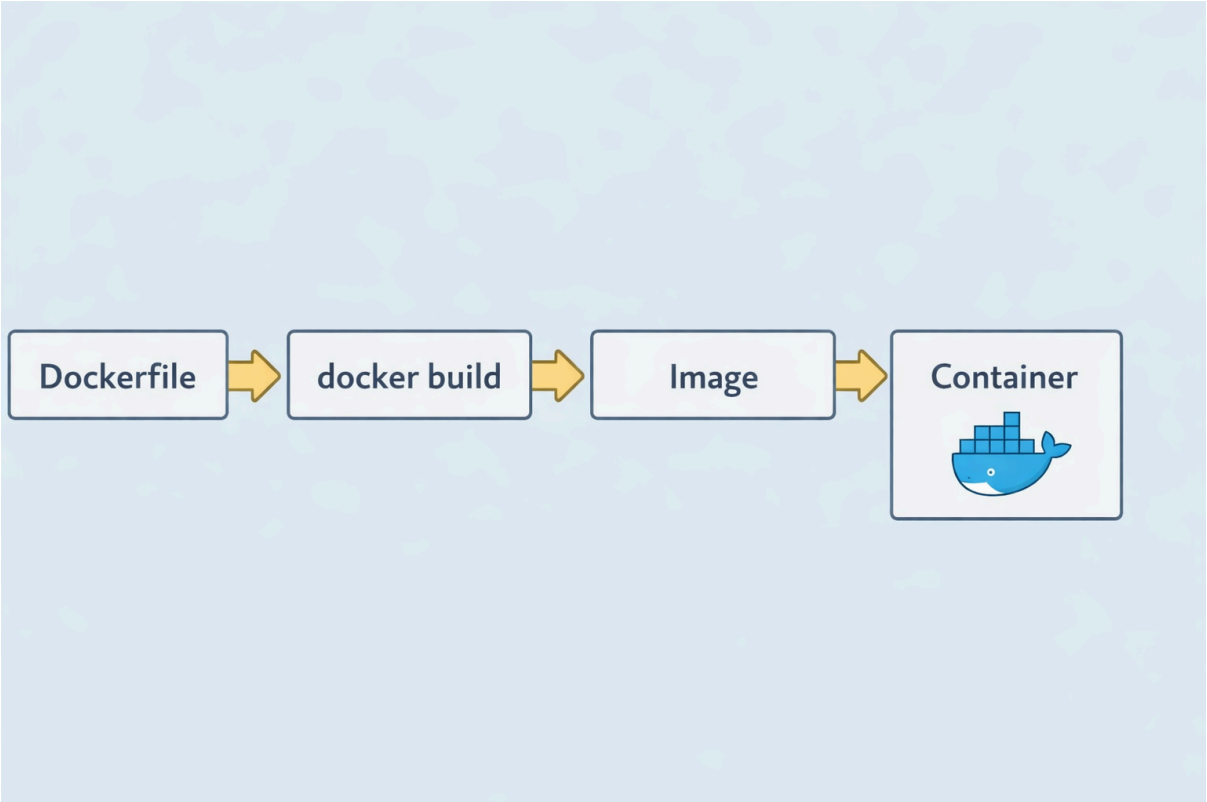
Dockerfile Creation:
What is a dockerfile?
A dockerfile is a text-based document that is used to create container images. It provides step by step instructions to create an image.

Key words:

```
Image = Blueprint (template)
Container = Running instance of image
```

Example:

```
docker build → creates IMAGE
docker run → creates CONTAINER
```



Key Dockerfile Instructions

Here are some key instructions:

Instruction	Purpose	Example
`FROM <image>`	Defines the base image to start the build process. Must be the first instruction.	`FROM ubuntu:latest`
`RUN <command>`	Executes any commands necessary to install packages or set up the environment *during* the image build.	`RUN apt-get update && apt-get install -y python3`
`CMD`	Provides defaults for executing a container. Only the *last* `CMD` instruction takes effect.	`CMD ["python3", "app.py"]`
`LABEL`	Adds metadata to an image (e.g., maintainer information).	`LABEL maintainer="user@example.com"`
`COPY <host-path> <image-path>`	Copies files or directories from the host machine (where the build is running) to the filesystem of the image.	`COPY . /app`
`ADD`	Similar to `COPY`, but can also handle remote URLs and automatically extract compressed files.	`ADD http://example.com/file.tar.gz /tmp/`
`EXPOSE`	Informs Docker that the container listens on the specified network ports at runtime. (It does not *publish* the port.)	`EXPOSE 8080`
`ENTRYPOINT`	Configures a container that will run as an executable. Often used with `CMD` to specify default arguments.	`ENTRYPOINT ["/usr/bin/nginx"]`
`WORKDIR <path>`	Sets the working directory for subsequent `RUN`, `CMD`, `ENTRYPOINT`, `COPY`, and `ADD` instructions.	`WORKDIR /app`
`ENV`	Sets environment variables. These variables are available to subsequent instructions and to the container at runtime.	`ENV PORT=8080`

Example 1:
We will make a simple image from a dockerfile. Hashtags # are for comments.
In this example we will take base layer ubuntu and install php inside it.

```
#Base Image
FROM ubuntu:latest
#Maintainer (optional)
LABEL maintainer="student@email.com"
#install Packages
RUN apt update && apt install -y php
#command when container runs
CMD ["echo", "Hello Students"]
```

Making new Directory using “mkdir”

```
sumitkumarmehta@sumits-MacBook-Air ~ % pwd
/Users/sumitkumarmehta
sumitkumarmehta@sumits-MacBook-Air ~ % ls
2026-study              Documents               Library                 Music                   Public
Desktop                 Downloads               Movies                  Pictures                docker-volume-demo
```

```
sumitkumarmehta@sumits-MacBook-Air ~ % cd docker-volume-demo
sumitkumarmehta@sumits-MacBook-Air docker-volume-demo % la
zsh: command not found: la
sumitkumarmehta@sumits-MacBook-Air docker-volume-demo % ls
hostfile.txt
sumitkumarmehta@sumits-MacBook-Air docker-volume-demo % mkdir dockerfile
sumitkumarmehta@sumits-MacBook-Air docker-volume-demo % cd dockerfile
```

Use any text editor (I’m using nano)

```
#base image
FROM ubuntu:latest

#maintainer (optional)
LABEL maintainer="Student Demo"

#install packages
```

```
LABEL maintainer="Student Demo"

#install packages
RUN apt update && apt install -y php

#default command when containers run
CMD ["echo", "Hello Students"]
```

Now we will build the image using command:

docker build .

docker build -t <image_name> .

-t: for tagging

“.” : path, “.” signifies current/present working directory

```
sumitkumarmehta@sumits-MacBook-Air dockerfile % nano dockerfile
sumitkumarmehta@sumits-MacBook-Air dockerfile % docker build -t ulb .
[+] Building 136.7s (6/6) FINISHED
```

```
[+] Building 136.7s (6/6) FINISHED                                                                                  docker:desktop-linux
 => [internal] load build definition from dockerfile                                                                                0.0s
 => => transferring dockerfile: 252B                                                                                                0.0s
 => [internal] load metadata for docker.io/library/ubuntu:latest                                                                    0.4s
 => [internal] load .dockerignore                                                                                                   0.0s
 => => transferring context: 2B                                                                                                     0.0s
 => [1/2] FROM docker.io/library/ubuntu:latest@sha256:cd1dba651b3080c3686ecf4e3c4220f026b521fb76978881737d24f200828b2b              0.0s
 => => resolve docker.io/library/ubuntu:latest@sha256:cd1dba651b3080c3686ecf4e3c4220f026b521fb76978881737d24f200828b2b              0.0s
 => [2/2] RUN apt update && apt install -y php                                                                                    132.0s
 => exporting to image                                                                                                              4.2s
 => => exporting layers                                                                                                             3.4s
 => => exporting manifest sha256:00ba525d1cbb37b631fd6d47a11f414458b30f57b06d6f84cc3d621606a091d9                                   0.0s
 => => exporting config sha256:c38ad982b474ec5857d2a88523aaee6f810e1f013846dcf64cbd2364f58dde80                                     0.0s
 => => exporting attestation manifest sha256:ebd4eed2f711bcfed0f436c203d21782e20858f6942589b634c647d3952ea894                       0.0s
 => => exporting manifest list sha256:195916186bb26dd964bb38545c9c7e4d44f0a9a3f6ac298526e8c6a642621900                              0.0s
 => => naming to docker.io/library/ulb:latest                                                                                       0.0s
 => => unpacking to docker.io/library/ulb:latest                                                                                    0.7s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/rz0mmrw9c6my7zbzgg2zrcf2d
```

```
sumitkumarmehta@sumits-MacBook-Air dockerfile % docker run ulb
Hello Students
sumitkumarmehta@sumits-MacBook-Air dockerfile % docker run -it ulb bash
root@8c31788e6a7b:/# php -v
PHP 8.3.6 (cli) (built: Jan  7 2026 08:40:32) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.3.6, Copyright (c) Zend Technologies
    with Zend OPcache v8.3.6, Copyright (c), by Zend Technologies
root@8c31788e6a7b:/# exit
exit
sumitkumarmehta@sumits-MacBook-Air dockerfile % docker images
```

i Info → U In Use

IMAGE	ID	DISK USAGE	CONTENT SIZE	EXTRA
docker/welcome-to-docker:latest	c4d56c24da4f	22.9MB	6.35MB	U
httpd:latest	dd178595edd6	207MB	47.8MB	U
mysql:latest	932fe8fbc04c	1.28GB	278MB	U
nginx:latest	13310a9cc1de	258MB	64.1MB	U
sakurawinter/ubuntusample:v1.1	067708b231d9	451MB	116MB	
ubuntu:latest	cd1dba651b30	141MB	30.8MB	U
ulb:latest	195916186bb2	451MB	116MB	U
welcome-to-docker:latest	250e3657631c	441MB	122MB	U