Base OS (Ubuntu or Alpine)
Application code
Dependencies
Environment Variables
startup commands

} Image

This is written in Dockerfile.

Docker reads the dockerfile and creates an image.

An Image is → Efficient
Reusable
Cache-friendly

(i) read only
(ii) Blueprint / template
(iii) stored locally or in registry (like Docker hub)

Docker runs a container from the image.
when you run the image, Docker creates a container.

A container is

(i) A running instance of an image.
(ii) Isolated from other containers.
(iii) Has its own filesystem, network and processes.

Container is a running instance of an image, executable and writable.
We can start it, stop it, restart it, Delete it.

Difference between Image and container:-

Image	container
Blueprint/template	Running Instance.
Read only	writable
static.	Dynamic

simple real world analogy

Concept	Real World Example
Image	Receipe
Container	Dish made from recipe.
Docker Engine	Kitchen.

Why docker is powerful:-

(i) Eliminates "its work on my machine" problem.

(ii) Light weight.

(iii) Faster than VMs.

(iv) Easy scaling.

(v) Great for microservices.

Docker Vs VM

Docker

Lightweight.

Starts in seconds.

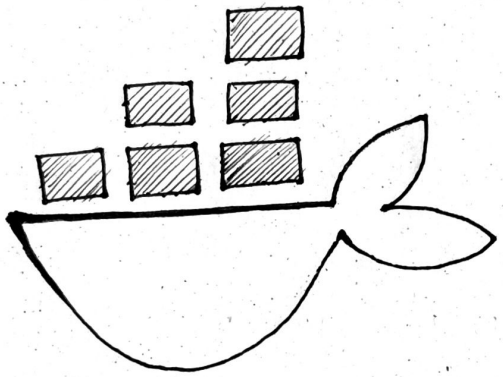
Uses fewer resources.

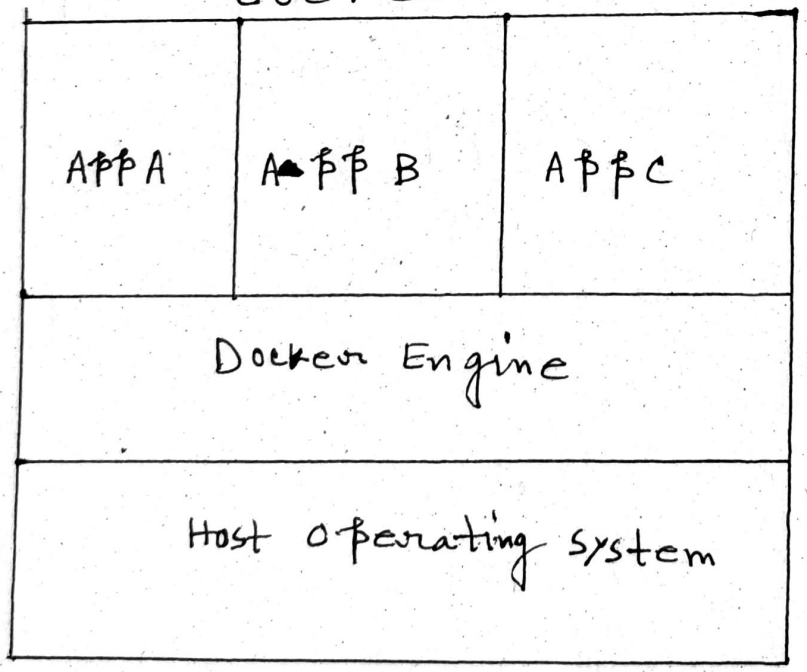
VM

Heavy weight.

Starts in minutes.

Uses more resources.





Apps are the Containers