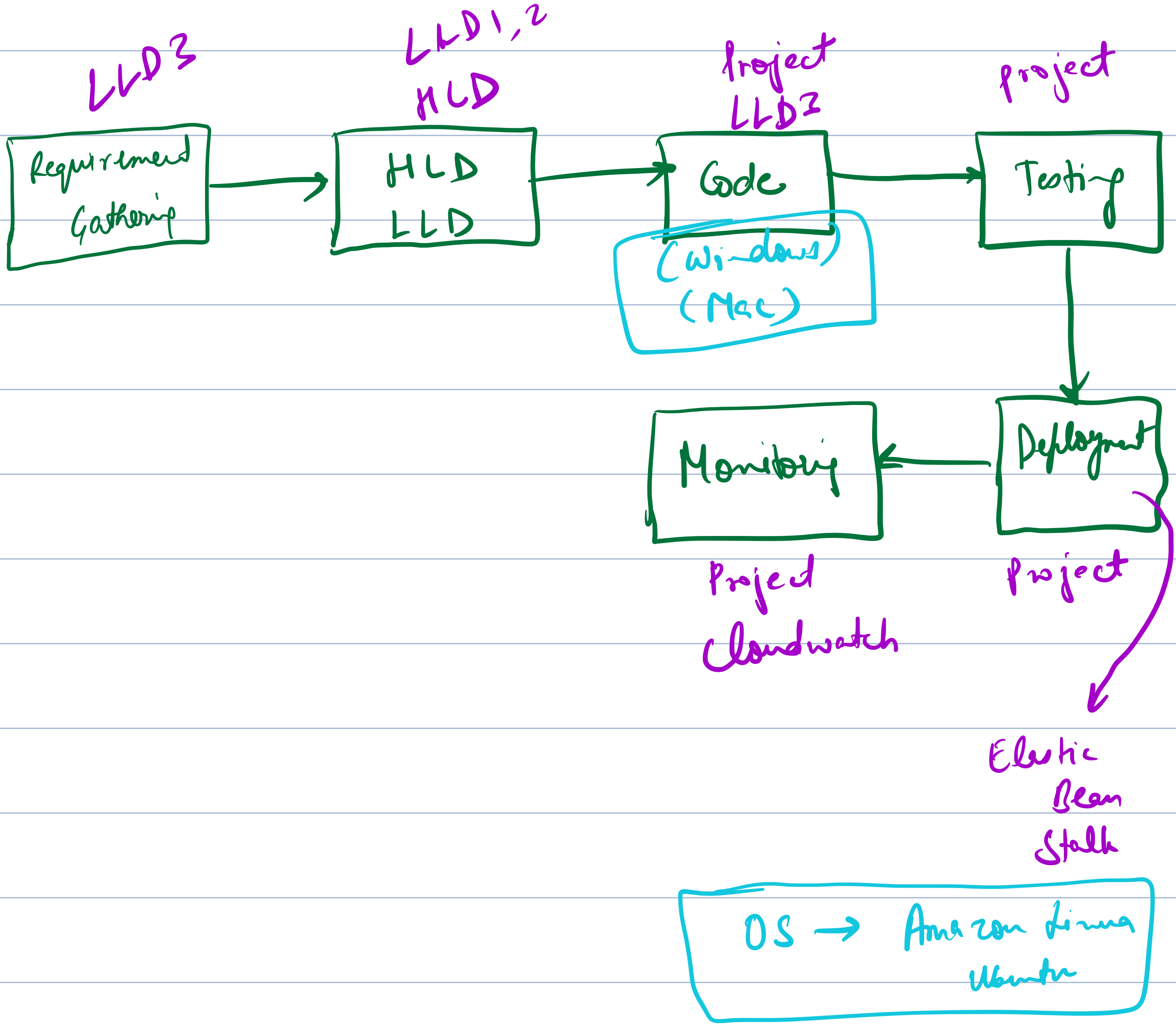
Docker →

- why do we need Docker
- Docker Terms
- Demo (Docker In Action)



EC2 →

install python
install virtual env
created virtual env

get code in virtual env
install dependency / flask
ran the code

Enough?

EC2
(cloud)

Problems during Deployment

✓ Development Different versions using of software of Python 2 and 3; Server

Python 2 Problem 1 Python 3

If I am referencing to a file in some path location, path mismatch

✓ File in location Problem 2 Missing File

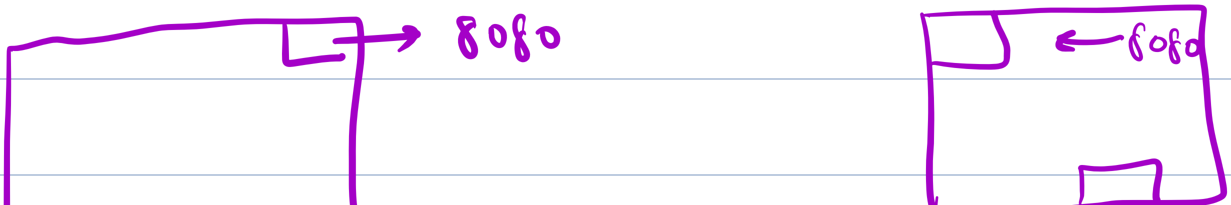
Path Mismatch

version incompatibilities happens

✓ Version Incompatible → Problem 3

Lets say we are referring to some configuration, which has dependency on OS, ubunutu or linux architetcure is too different

✓ OS Mismatch → Problem 4



Problem 5 ← Port Occupancy → Big challenge

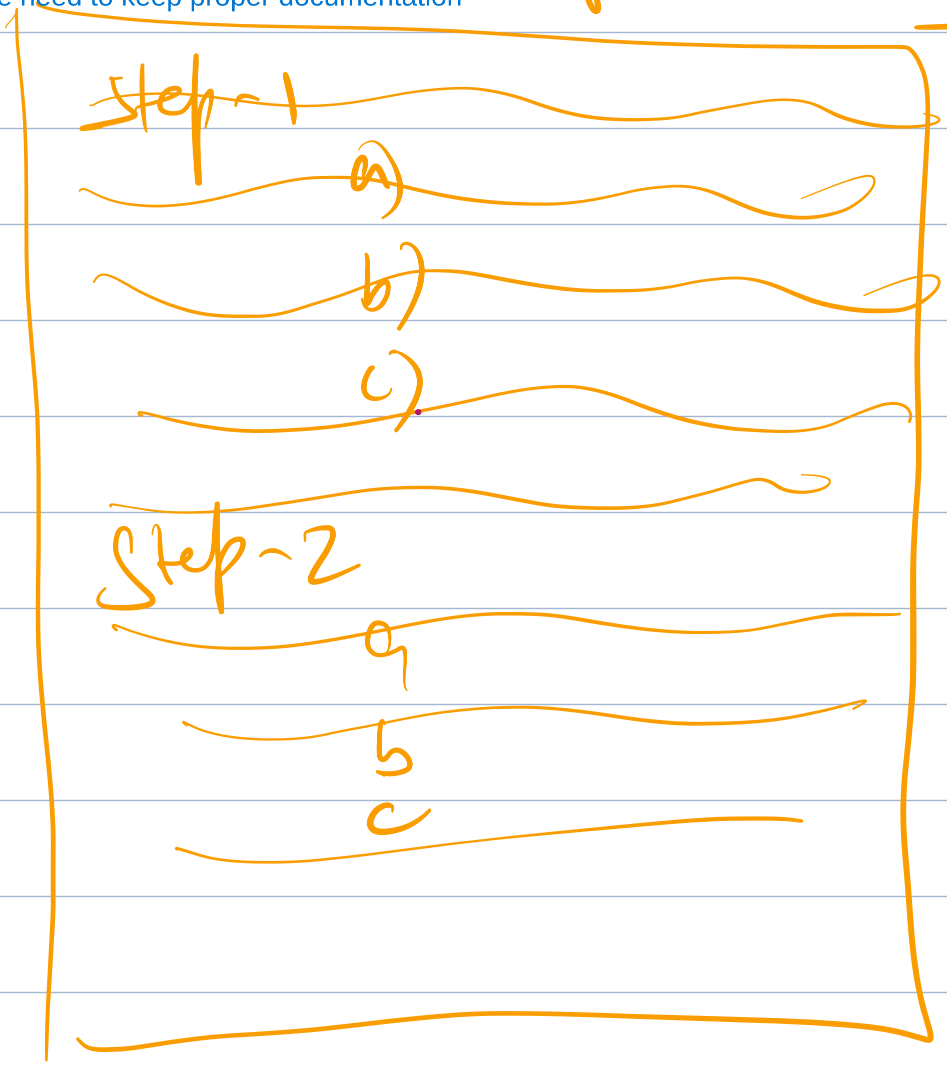
Lets say we have given a port - 8080 in our system code since it is free in my system but in EC2 is not empty and might be used by some other code

Admin Rights → Problem 6

Solutions →

Documentation

To avoid these problems -> we need to keep proper documentation



DevOps ppl needs to see everything is fine

Dev Ops

Dev ↓ Develop

Ops → operations

1000 changes per day

↓

X Machines

machines

If I have to deploy 1000 changes in a day on X machines -> thats not scalable

- Scalability ✓
- Human Error / Human

✓ Human Error / Human

Behavior
Based
Error

- Same type (Repetitive Job)

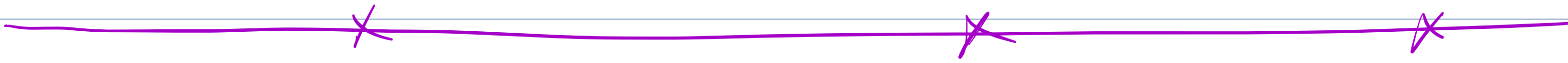
Same type of job -> devops person will not be happen

Solution is

Solution 2

Automation

More Reliable than Dev



What developers want?

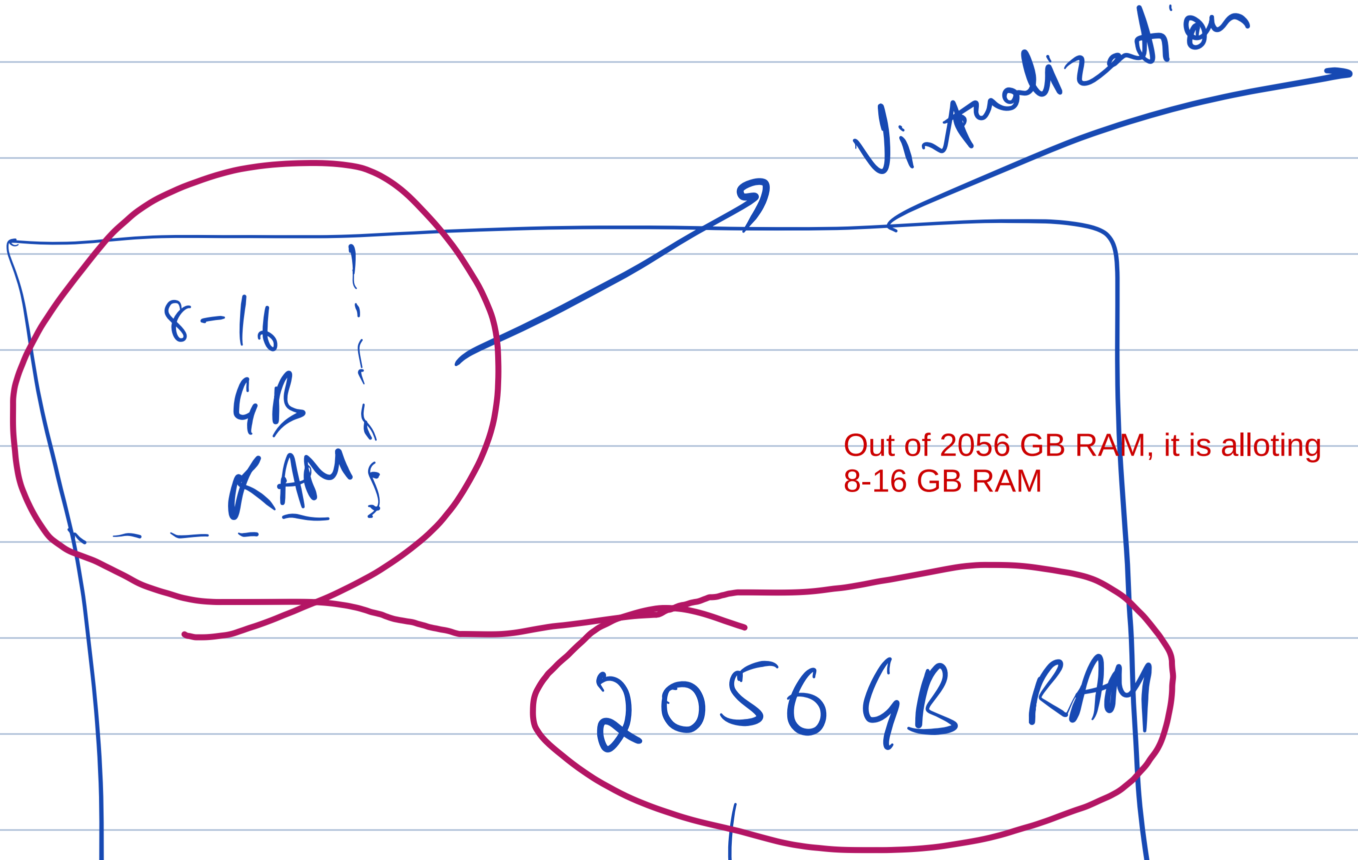
1. Mac OS ⟷ deployment Servers

Mac OS cannot be used for deployment purposes. Companies won't want their servers/prod servers to run on Mac OS since Mac Os is too costly

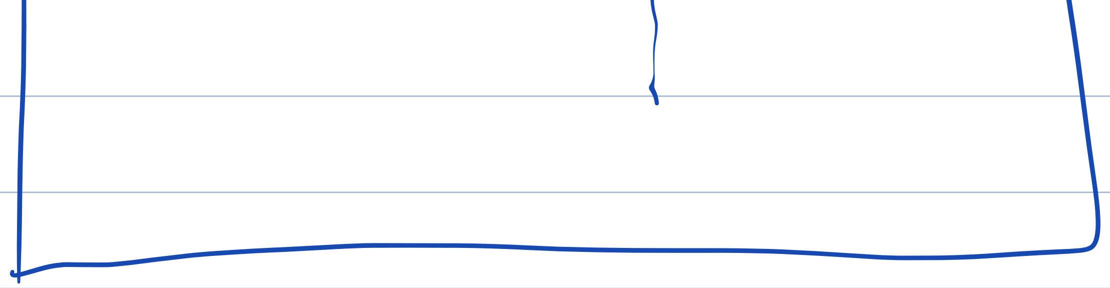
2. Running One service on one Machine doesn't look like a wise decision

Running one service on one Machine -> doesn't look as a good idea

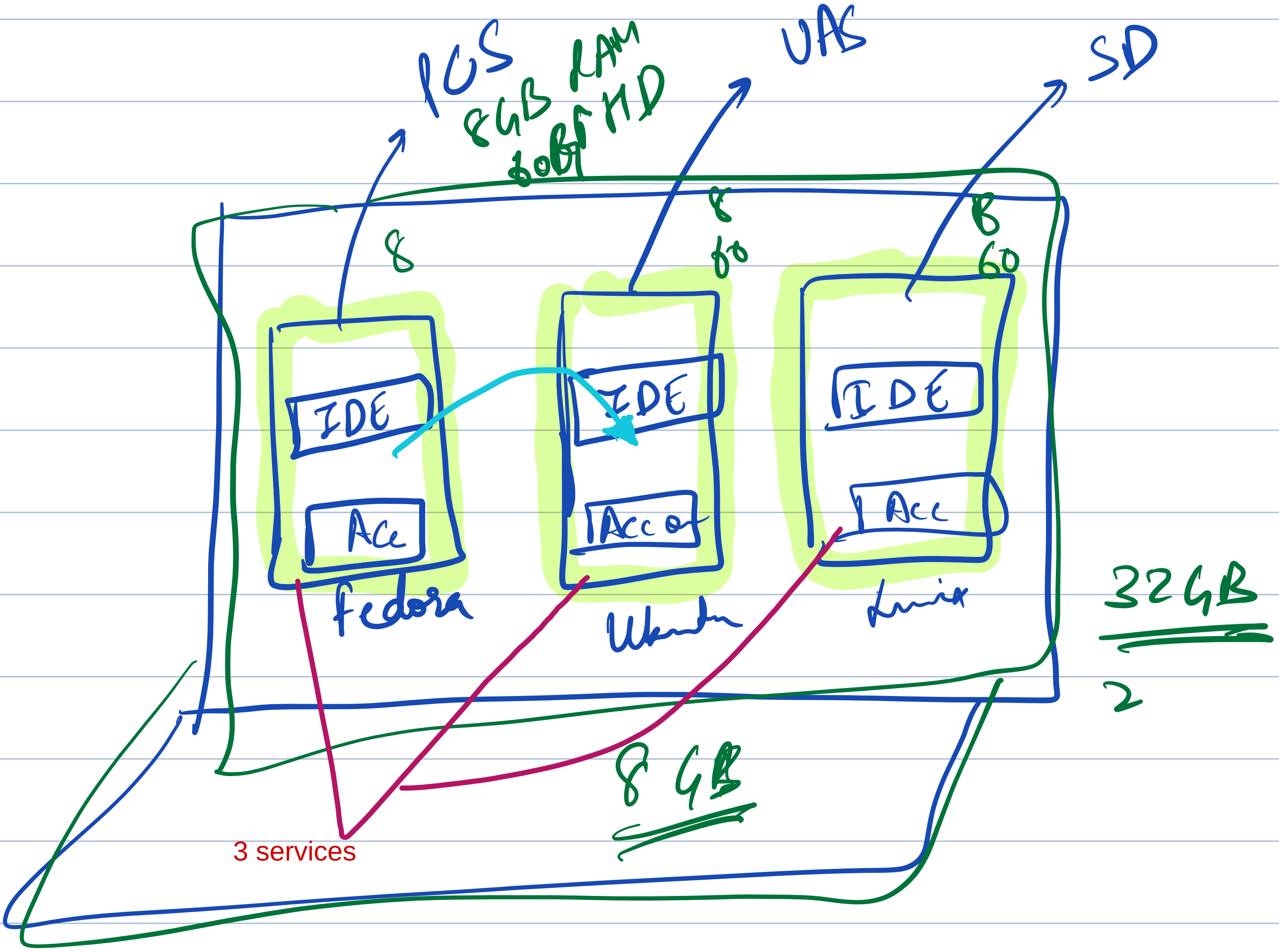
Solution is Virtualization



Out of 2056 GB RAM, it is alloting 8-16 GB RAM



Virtualization is creating small -2 servers out of a big server and each server is capable of running its own resources, own OS, own hardware

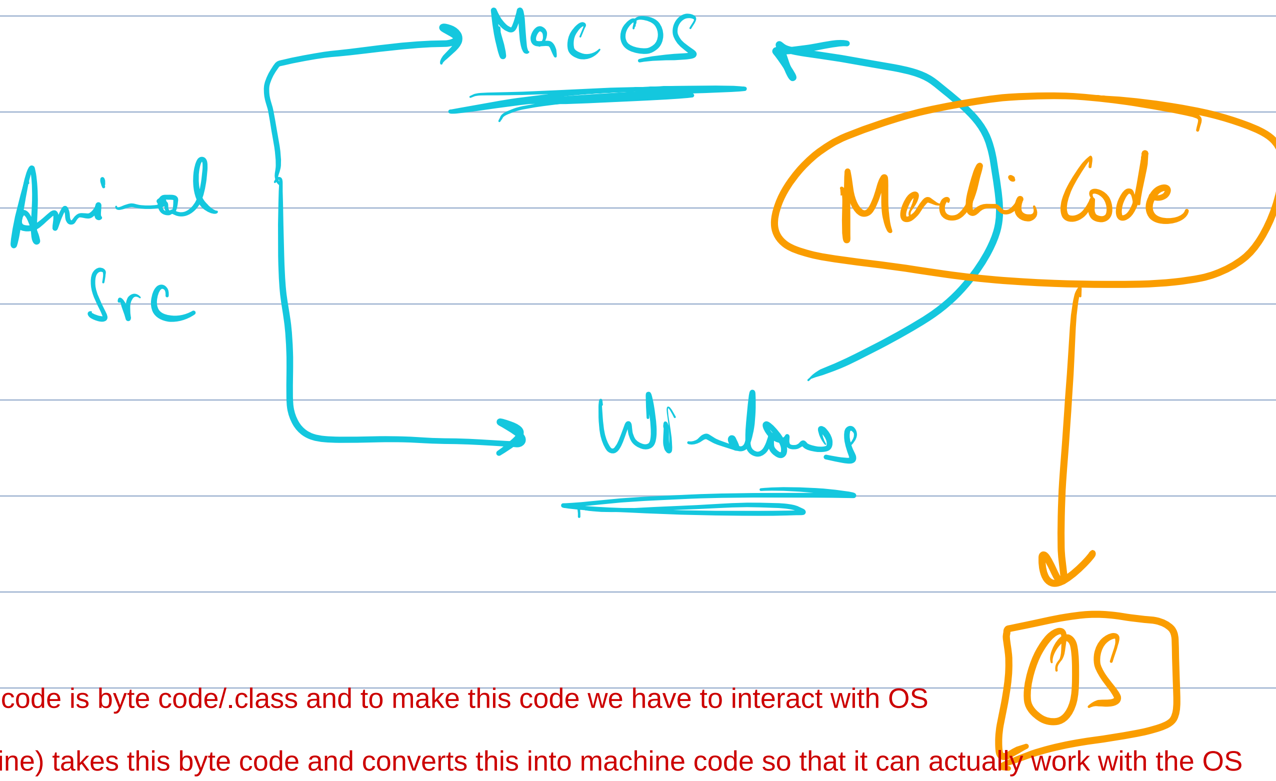


3 services

Hypervisor or VMware can provide virtualization

Experience would be Affected

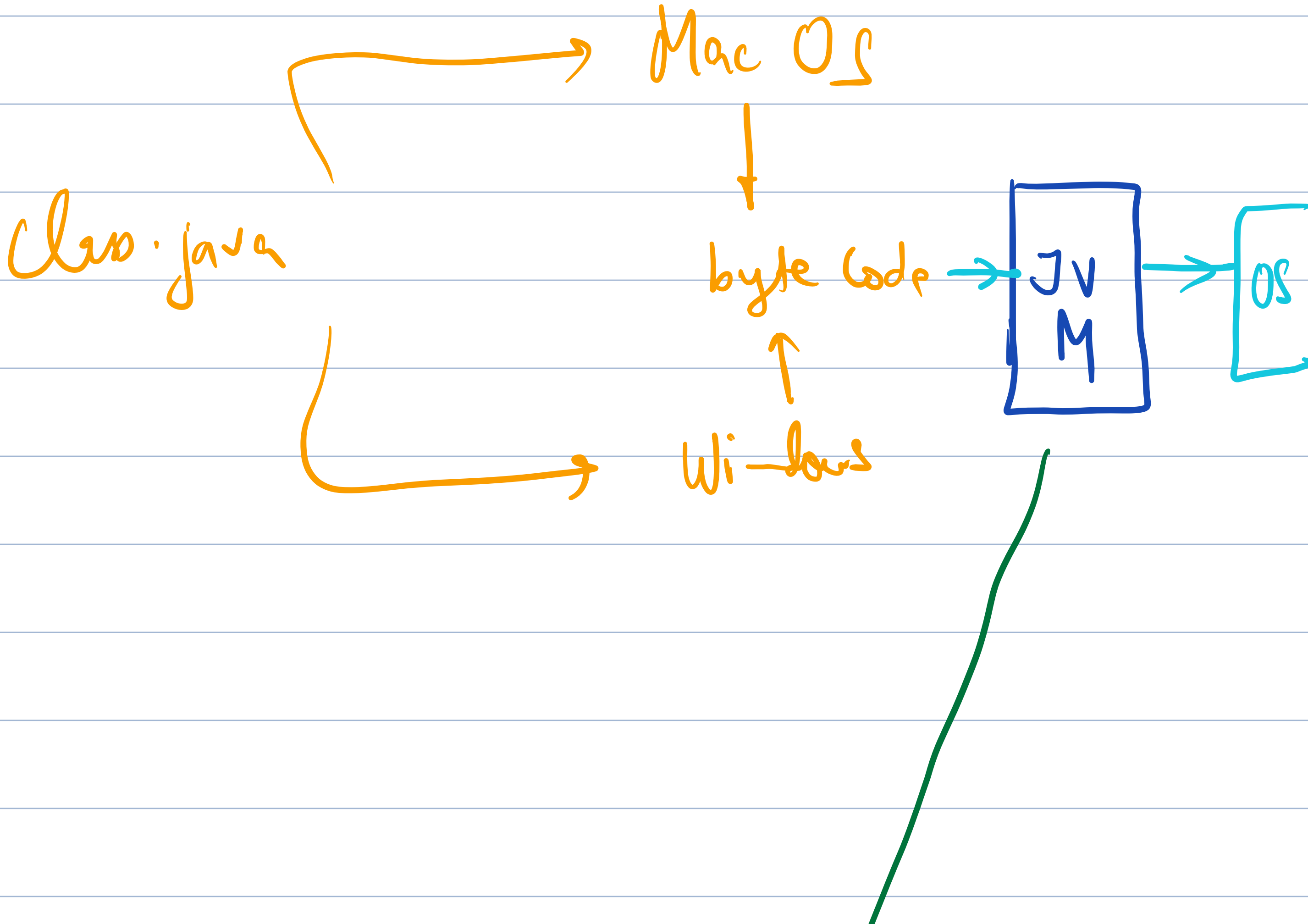
Communication b/w VMs can also be a big challenge



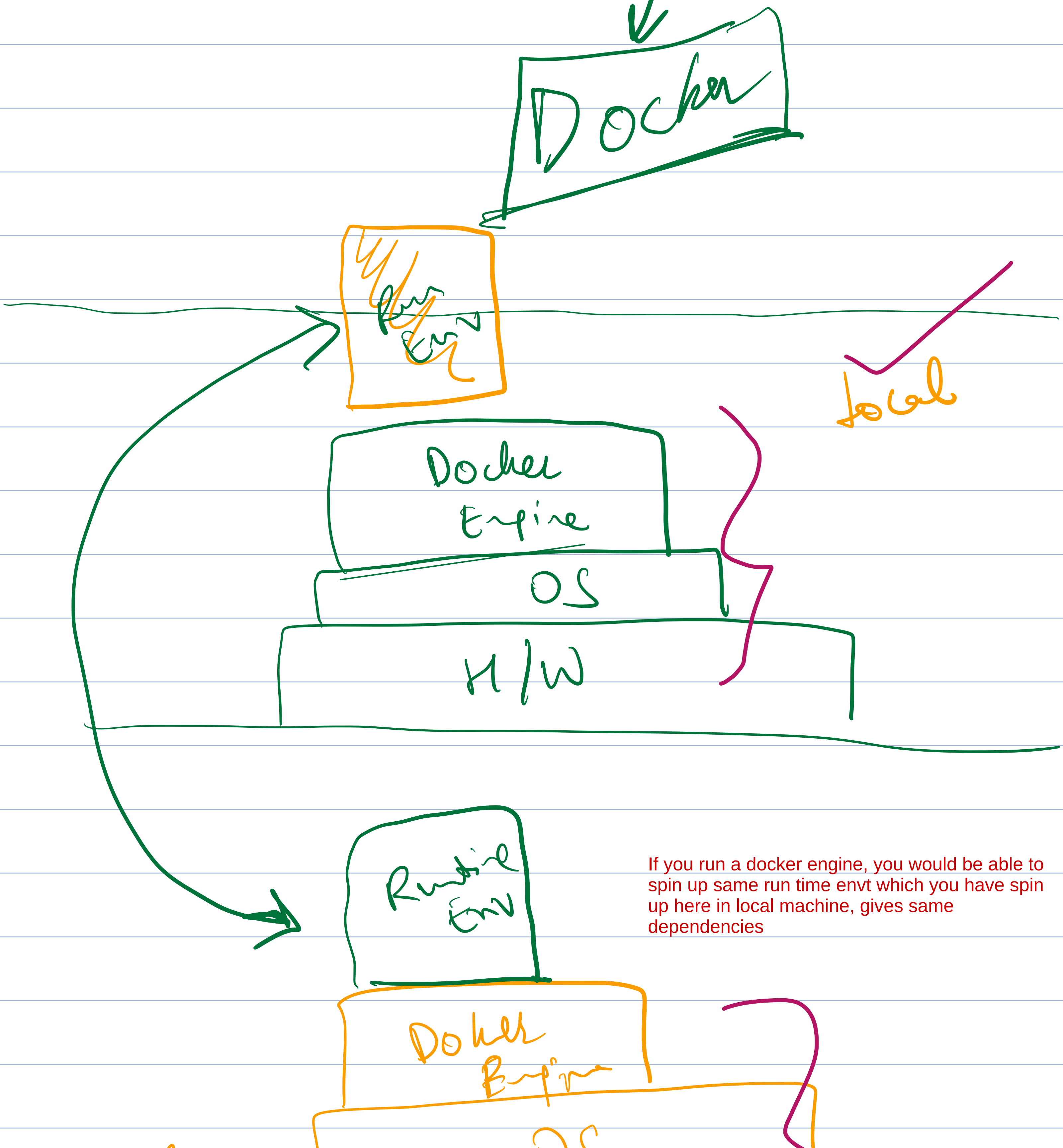
the final end product of our code is byte code/.class and to make this code we have to interact with OS

So JVM (Java Virtual Machine) takes this byte code and converts this into machine code so that it can actually work with the OS

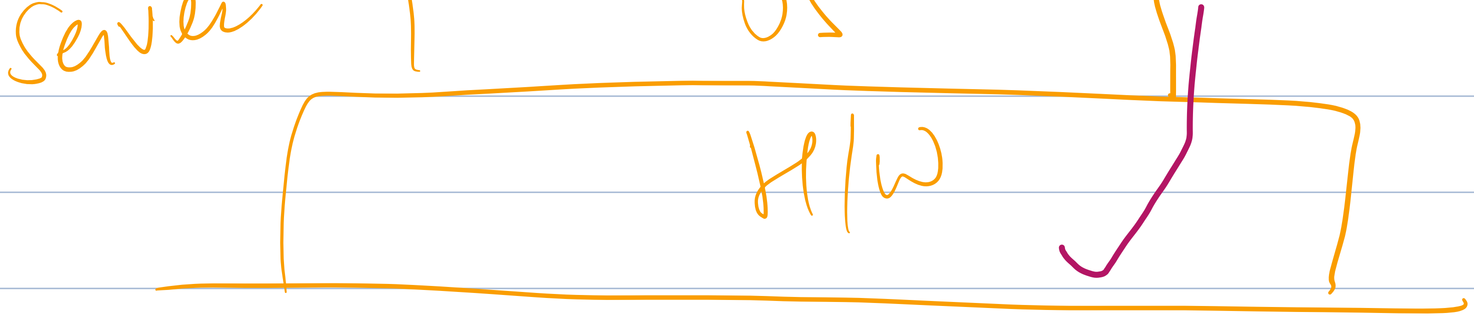
So lets say I have a .class which I created into Windows, if I take it in MAc OS, it will be able to work if I have JVM installed on Mac OS

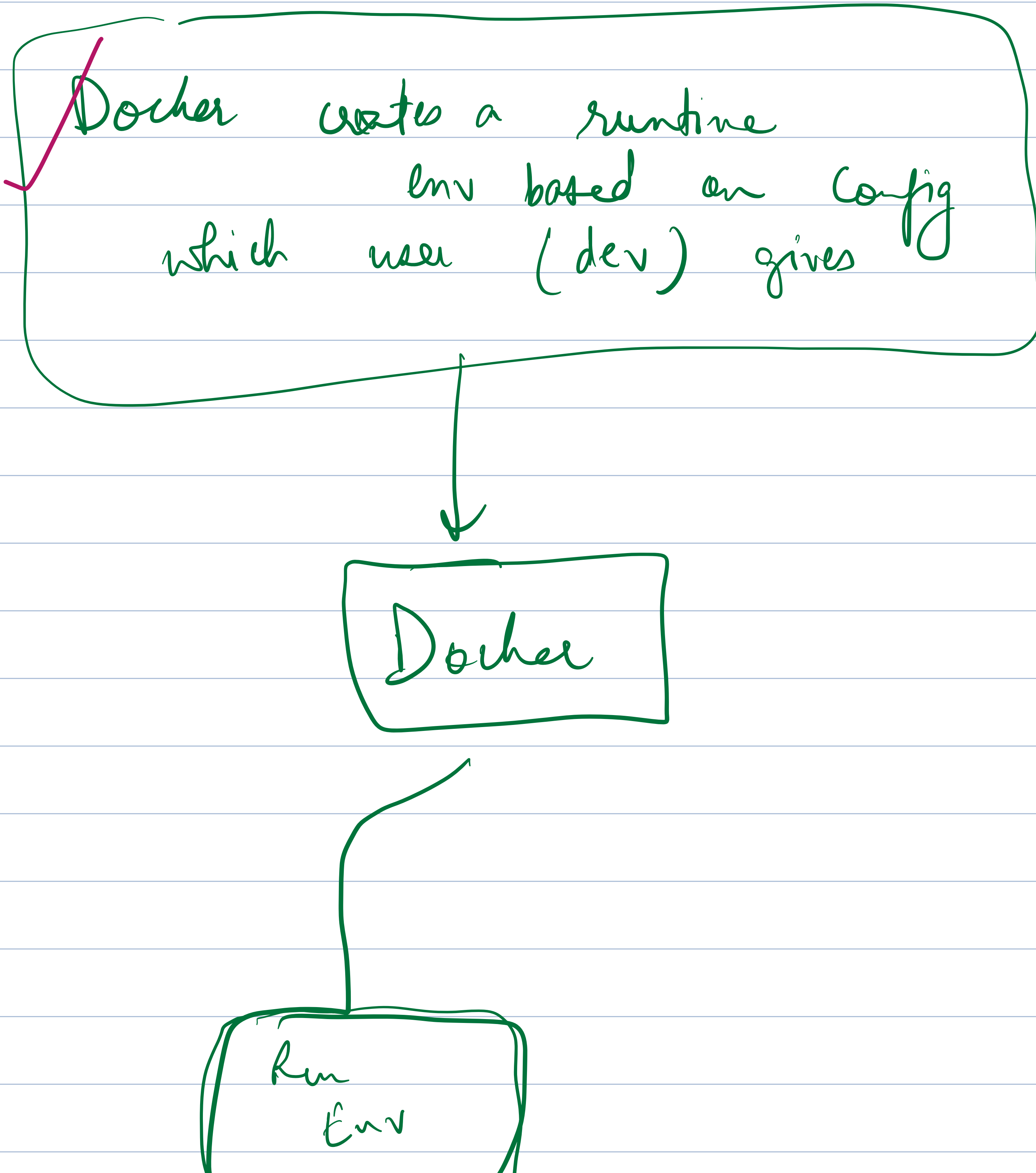


Similar to JVM, Docker also works -> hides the abstraction over the platform whatever the run time is

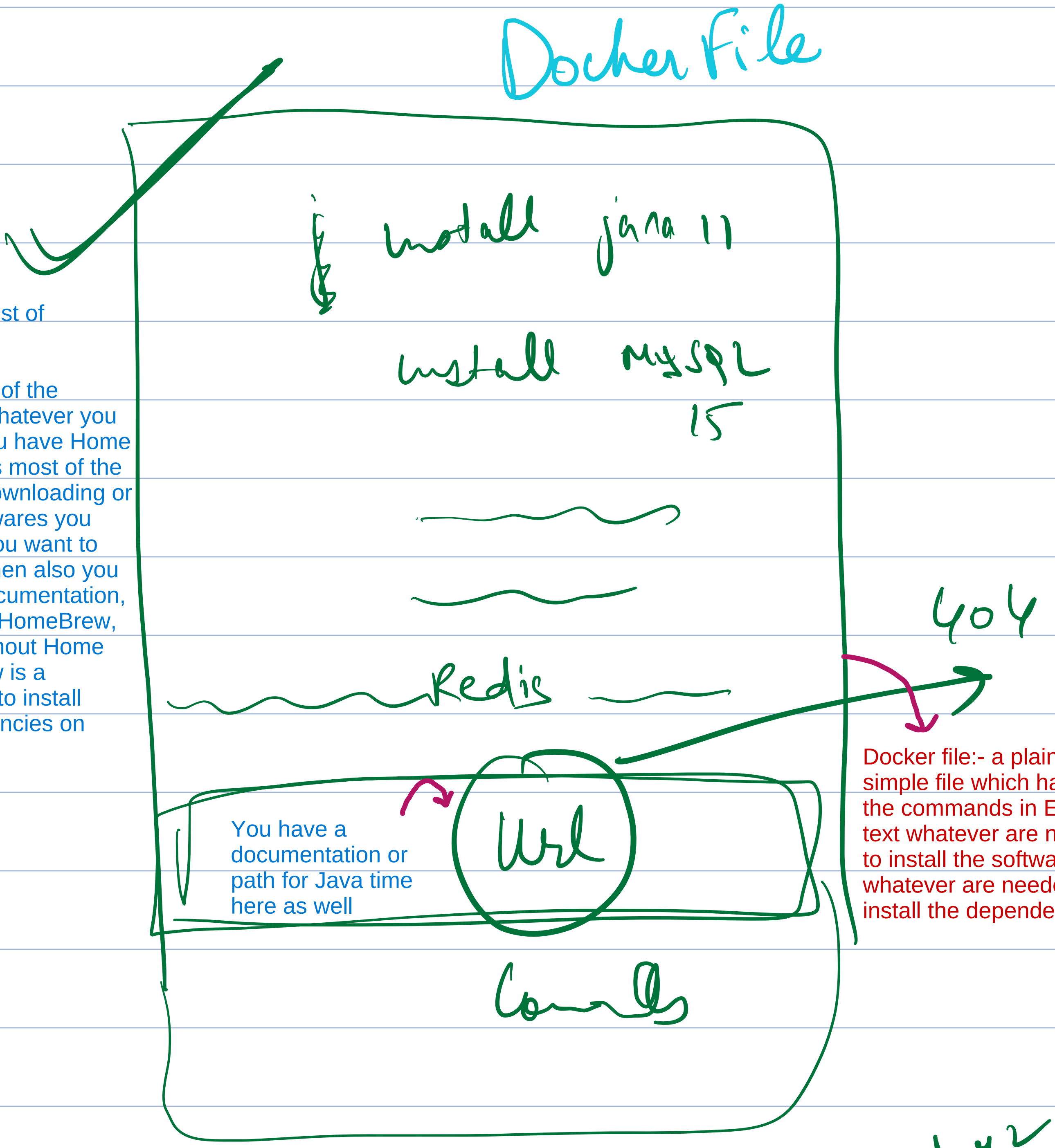


If you run a docker engine, you would be able to spin up same run time envt which you have spin up here in local machine, gives same dependencies





Docker creates a runtime env based on config which user (dev) gives

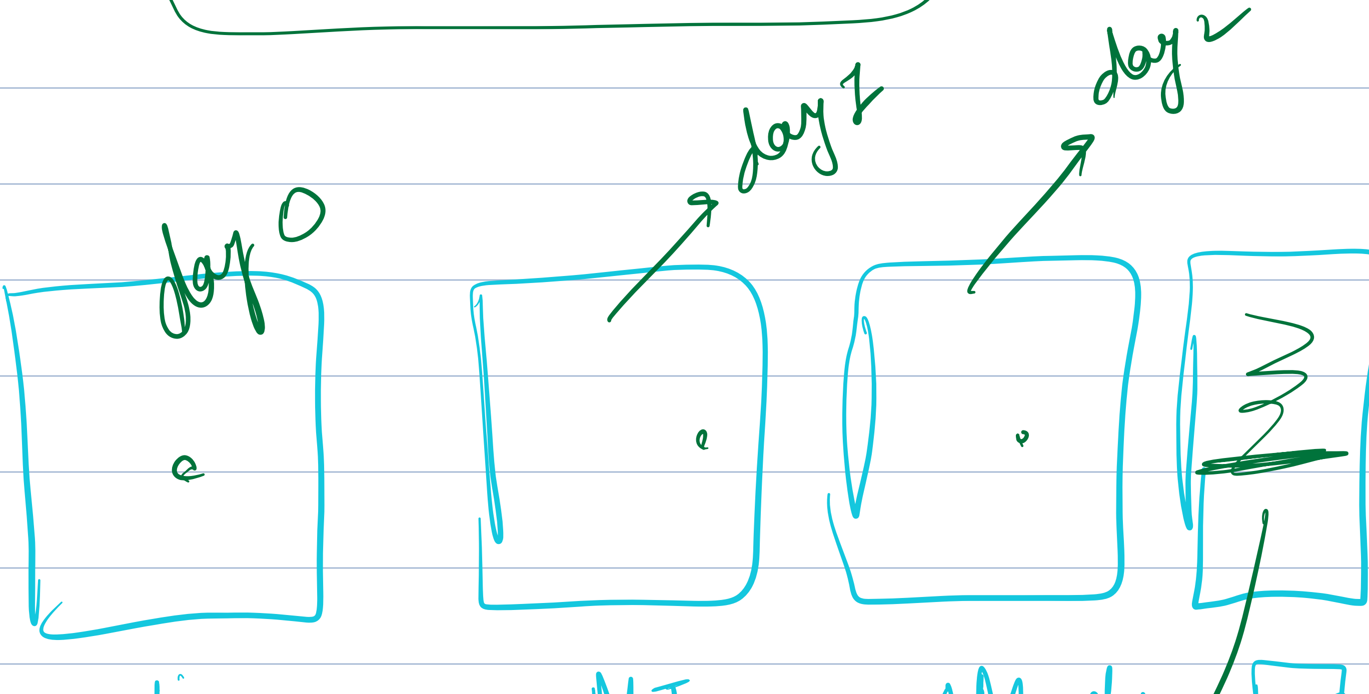


give Docker the list of dependencies.

in Mac OS, most of the dependenices, whatever you want to install you have Home Brew which gives most of the commands for downloading or installing all softwares you need. Lets say you want to install Redis -> then also you get 2 types of documentation, one will say with HomeBrew, other will say without Home Brew. HomeBrew is a standalone utitly to install multiple dependencies on MacOS

You have a documentation or path for Java time here as well

Docker file:- a plain simple file which have all the commands in English text whatever are needed to install the software and whatever are needed to install the dependencies



Lets say we have given copies of docker file to these 4 ppl, and Vinay was able to spun up the runtime envt on Day 0, 2 other person did on Day1 & Day2 respectively -> and lets say the 4th person was not able to do early and did on Day10 and encountered some errors maybe coz the URL in the file, goes not Found 404 on internet coz of which 4th person which not exactly spin up the runtime envt which these 3 ppl were able to and faced some errors

day 10

To avoid this issue, we should provide a compiled version of docker file k/a Docker Image.

Very similar to this analogy -> if I am giving you pom.xml that is Docker file, instead I am giving you jar file which is a compiled version -> that is Docker Image

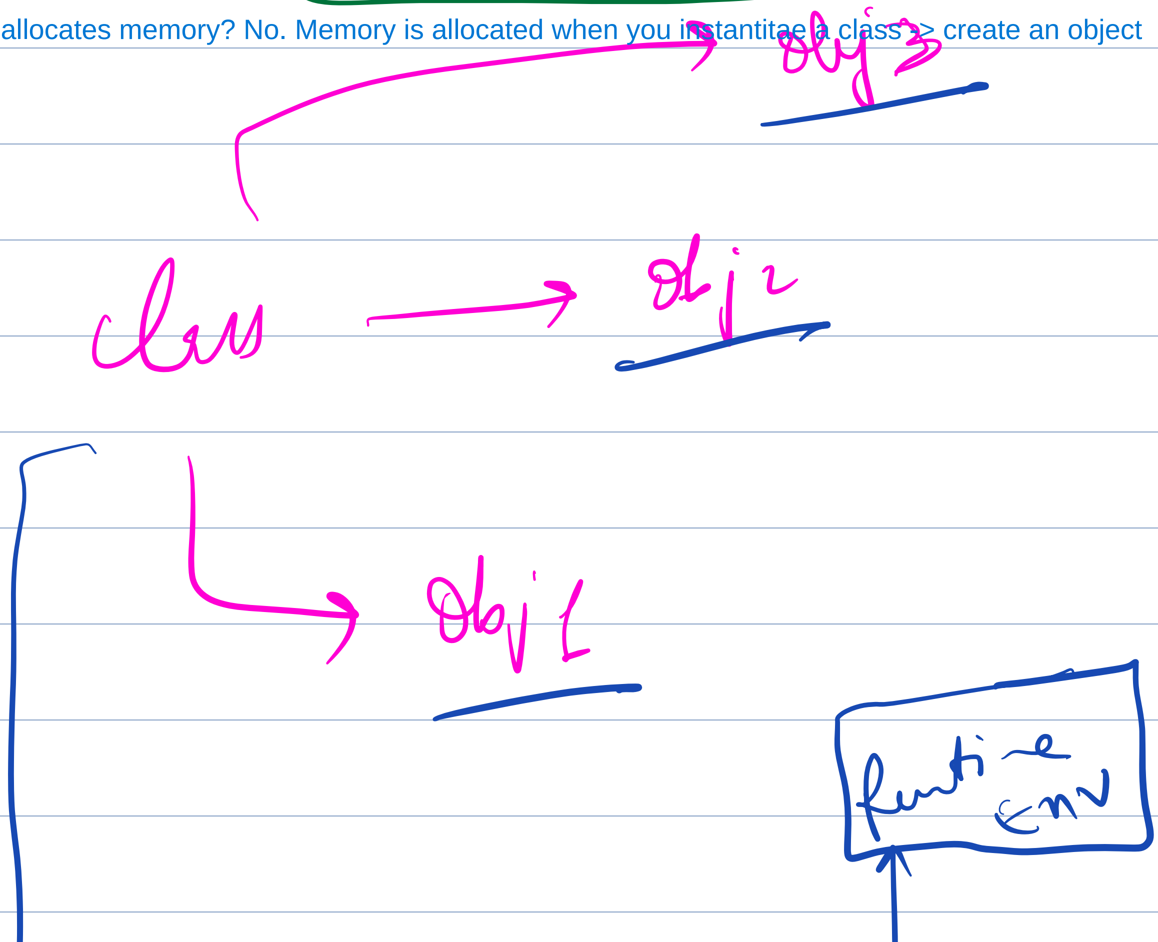
Step 1: From Docker file you need to create Docker Image

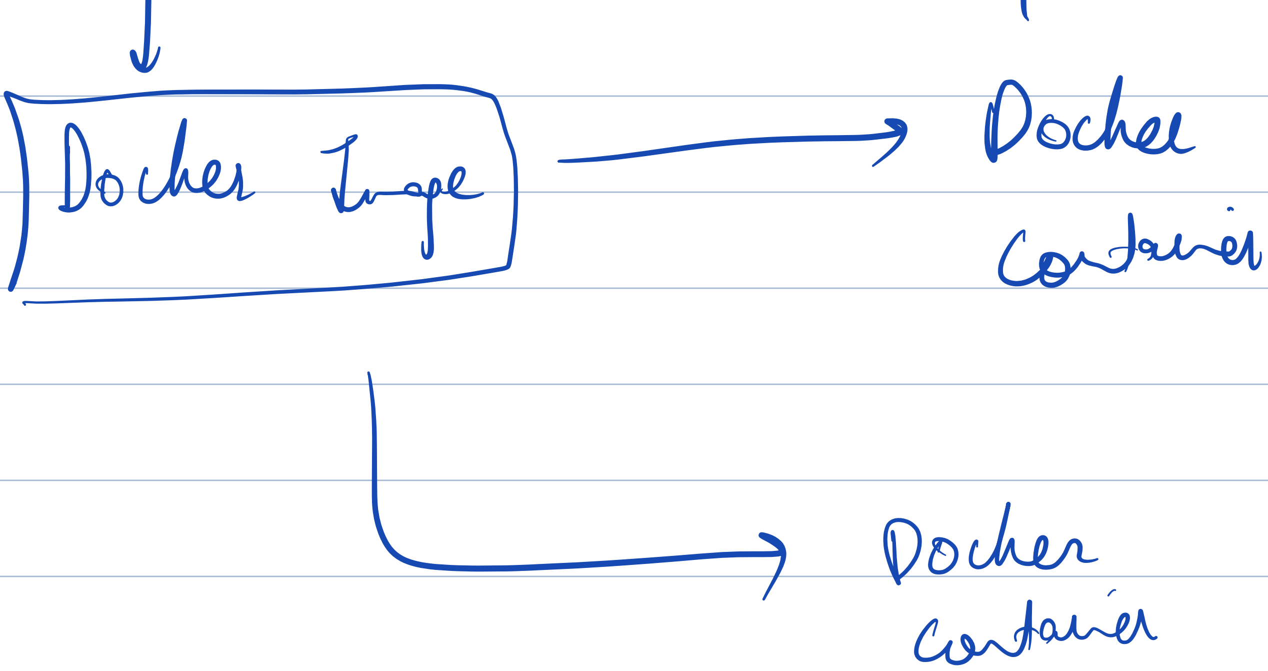
Step 1 → Dockerfile

↓

Docker Image

When we create a class, does it allocates memory? No. Memory is allocated when you instantitae a class -> create an object





Very similar to this analogy, class is Docker Image and objects are Docker container (or RunTime envt). I can create multiple Docker container. One Docker container will be created from One Docker Image but one Docker Image can be used to create multiple Docker containers

What is RunTime envt looks like? When I am running this Product Service or User Service w, we can say it is a running Instance and now I go on Postman and try to make a request on localhost: 8080/products or if I go to browser and enter url : localhost: 8080/user -> I will not be able to get request now on my local machine instead can get req 200 ok on Docker container on which it was running

So Similarly, if I have a runtime envt., if I have my Docker container created from my Docker Image and lets say it is running that means I can simply make a call to that URL: localhost or whatever endpoint my user is running on

docker container will assume like one one independent system? yes and each docker container will have its own port so port confliction will not happen. Each runtime envt, will have its own port, hardware, own OS and is very lighter in comparison to Virtual machine

Coz of Docker runtime envt, you can work on any IDE, all the isseues related to Virtual machine is resolved. VMs consume a lot of space, it is heavy and you ahev to owrite the code on VM as well You will be able perform other work on your machines without any lag. Only when you are running, you can move to Docker engine to create runtime envt. Here we need not to change the service endpoints also. We will be running Docker on EC2 machine