



Getting Started with **ROS:Docked**

Prerak Patidar | Saravan Kumar

But first, What is ROS?

The Robot Operating System (ROS) is a set of software libraries and tools that help you build robot applications.



HUMBLE
HAWKSBILL

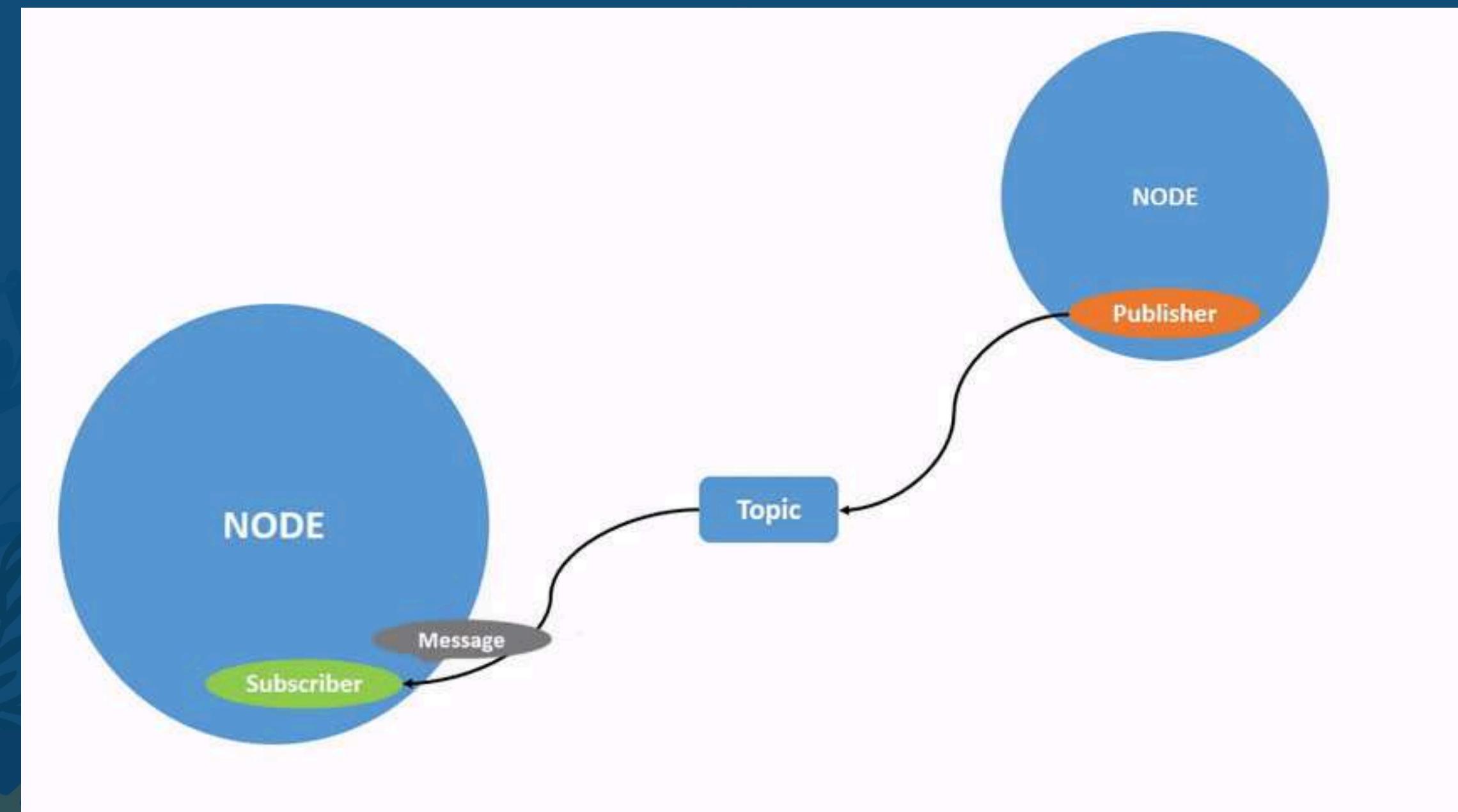


How does it work?

ROS works by sending and receiving messages. A ROS package consists of many components.

- Subscriber
- Node : An executable file
- Publisher
- ROS Master : A central message Hub
- Topic

How does it work?



Why ROS?



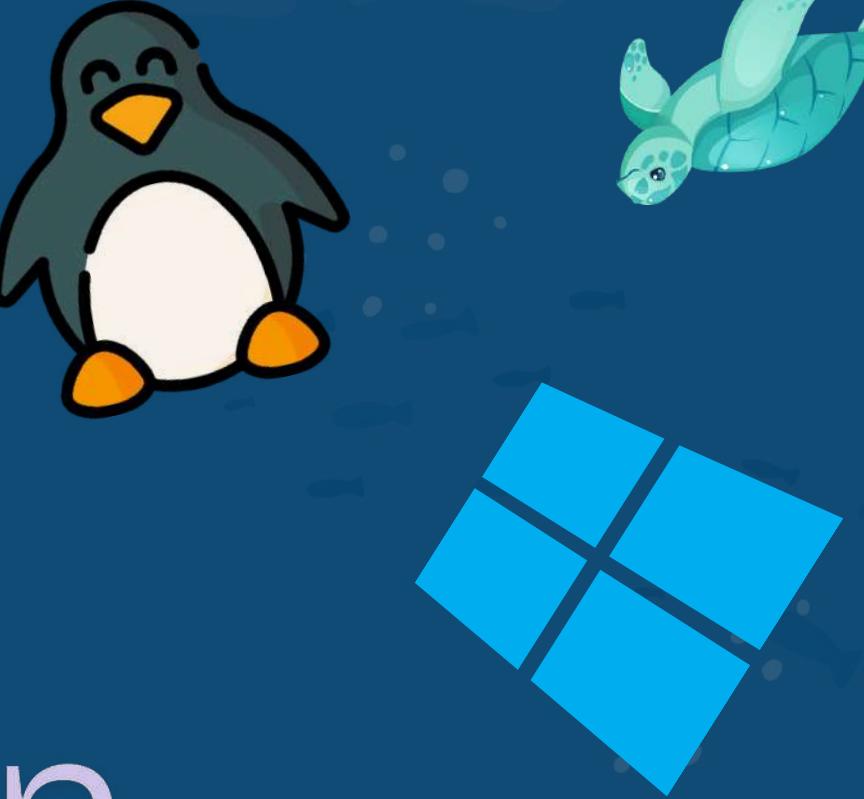
Open Source

Global Community

Multi-platform

Versatility

But wait, is it really an
Operating System?



Nah.

- ROS isn't an operating system in the same way that **Microsoft Windows** or **Linux** is.
- Instead, it's more like a framework or set of tools that runs on top of an existing operating system, typically **LINUX**.

BUT DO YOU USE IT ??? :)



Lets Install ROS



ROS

HUMBLE
HAWKSBILL



Installation



```
locale # verify settings
```

Setup Sources

You will need to add the ROS 2 apt repository to your system.

First ensure that the Ubuntu Universe repository is enabled.

```
sudo apt install software-properties-common  
sudo add-apt-repository universe
```

Now add the ROS 2 GPG key with apt.

```
sudo apt update && sudo apt install curl -y  
sudo curl -sSL https://raw.githubusercontent.com/ros/rosdistro/master/ros.key -o /usr/share/keyrings/ros-archive-keyring.gpg
```

Then add the repository to your sources list.

```
echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/ros-archive-keyring.gpg] http://packages.ros.org/ros2/ubuntu $(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/ros2.list && sudo apt update
```

Install ROS 2 packages

Update your apt repository caches after setting up the repositories.

```
sudo apt update
```

ROS 2 packages are built on frequently updated Ubuntu systems. It is always recommended that you ensure your system is up to date before installing new packages.

```
sudo apt upgrade
```

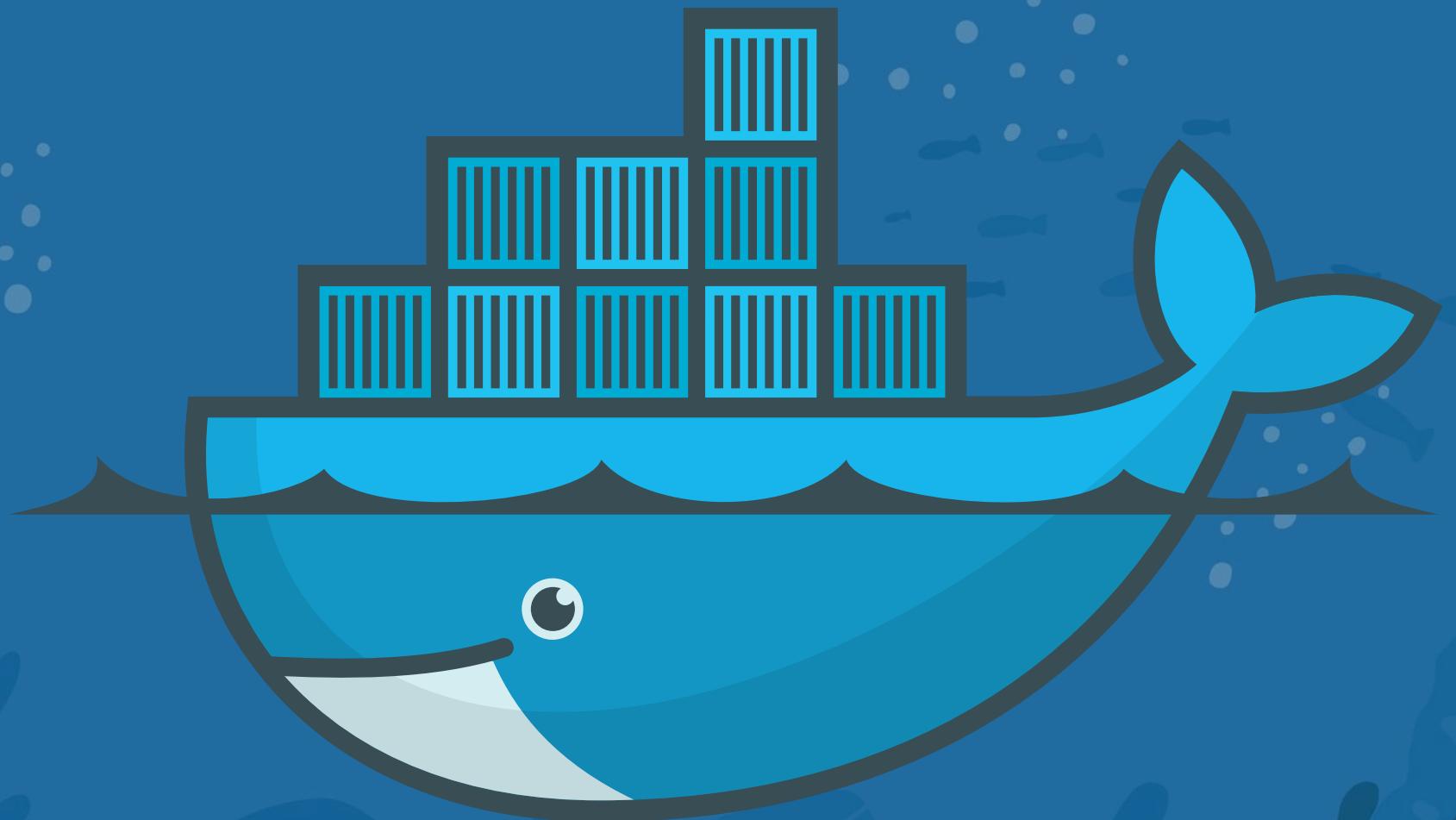
⚠ Warning

Due to early updates in Ubuntu 22.04 it is important that `systemd` and `udev`-related packages are updated before installing ROS 2. The installation of ROS 2's dependencies on a freshly installed system without upgrading can trigger the removal of critical system packages.





Lets start with
What is Docker?



Lets take a scenario

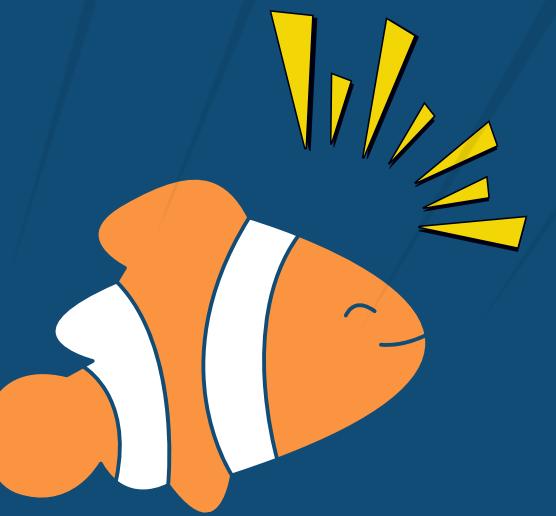
Submitting your assignment



Your code doesn't Work It Works in my machine



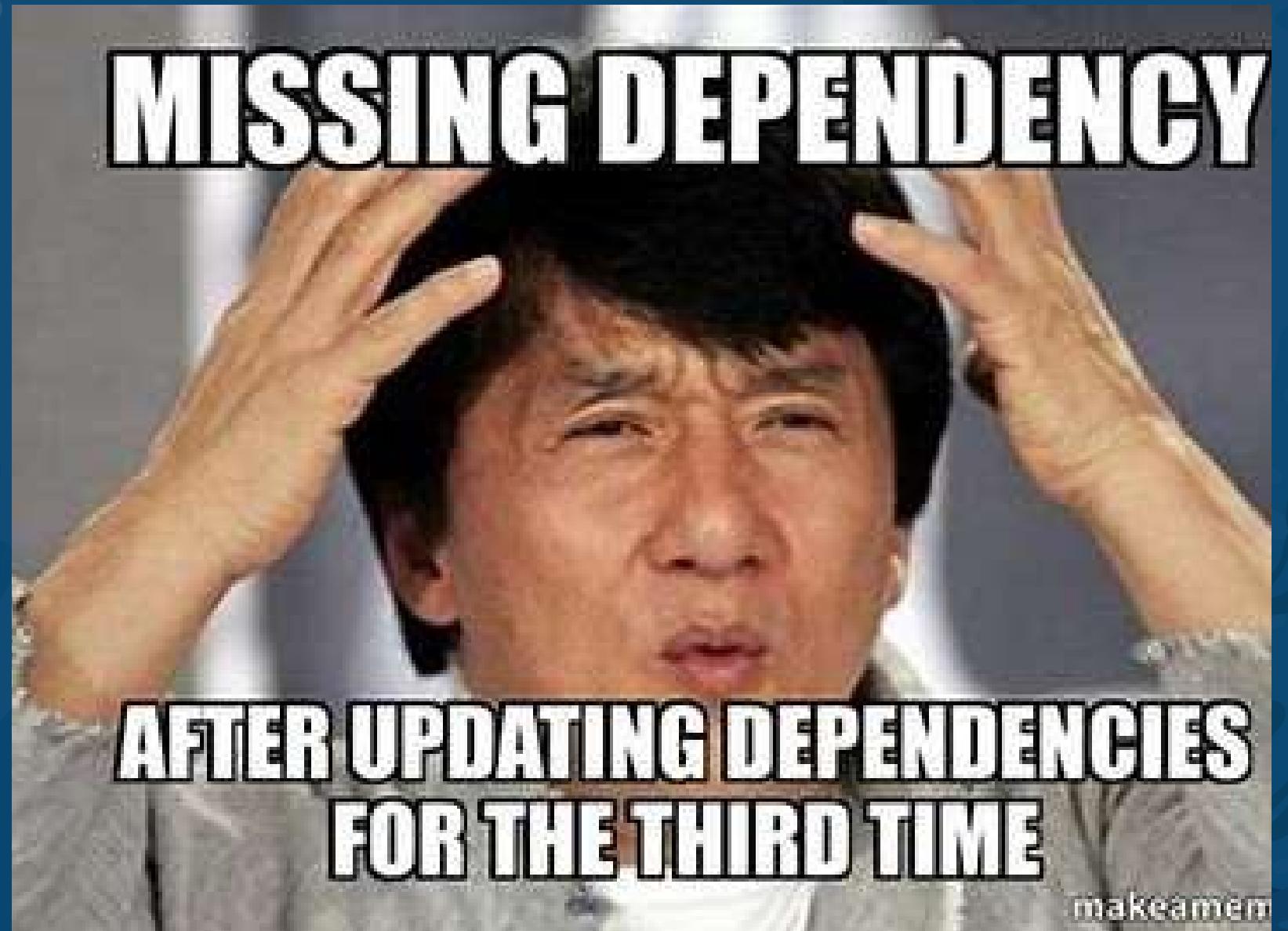
What could be its
Reason ??





What could be the Reasons??

Dependencies





What could be the Reasons??

Dependencies

version

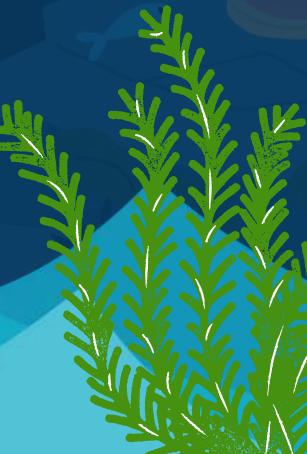


3.13 ✗



3.12 ✓

How many of you have attended image processing workshop last week ??





What could be the Reasons??

Dependencies

version

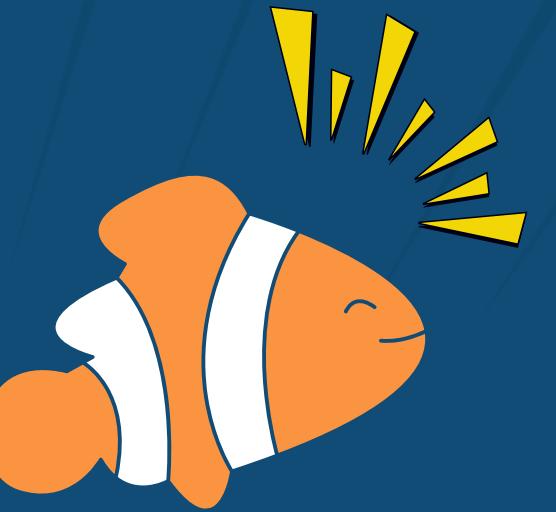
Libraries

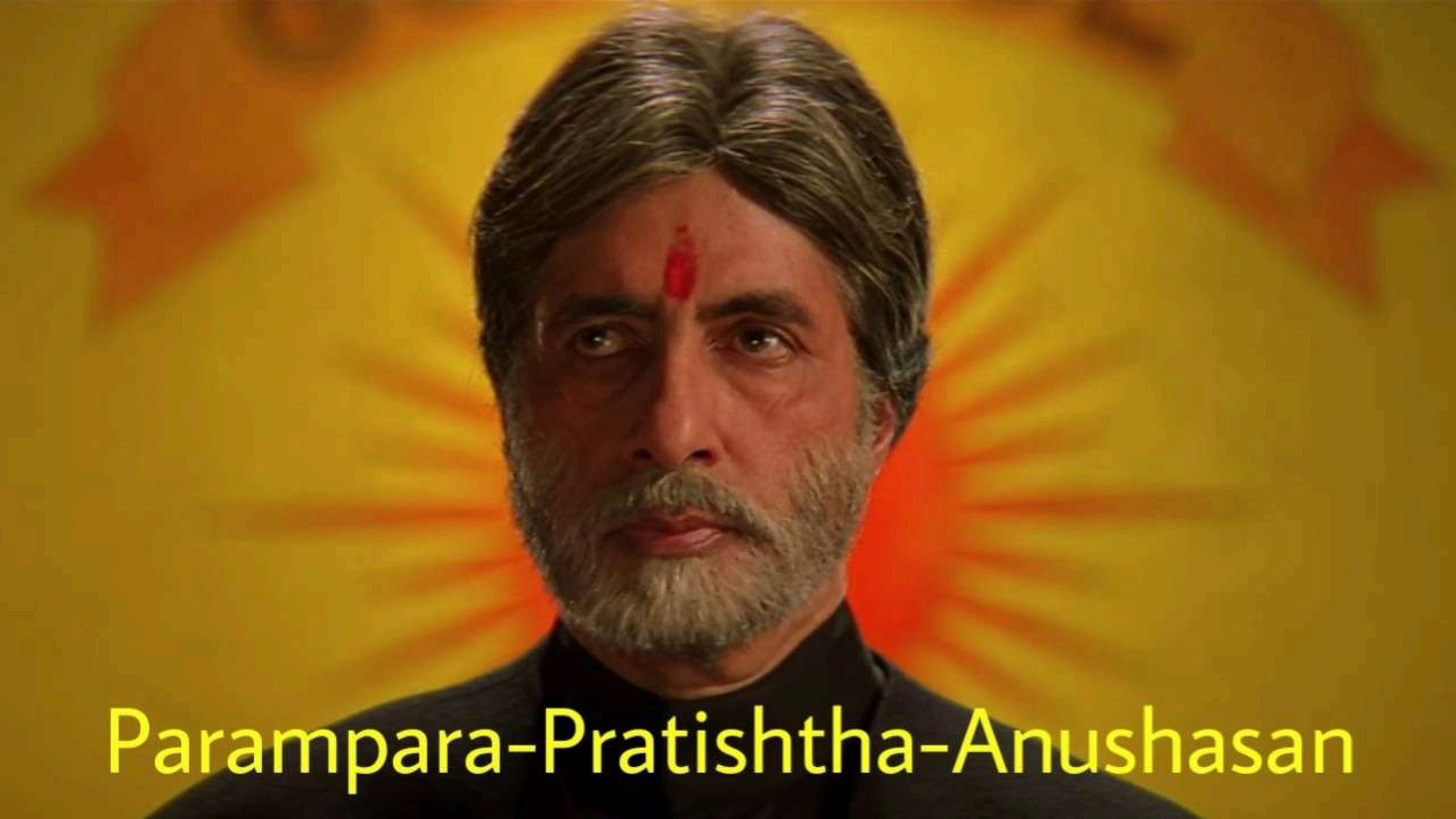


Write
10 lines
if code

Import 5
libraries
and write 8
lines of code

What could be its
Solution ??





Parampara-Pratishtha-Anushasan

Let's first see how things work
on our machine!

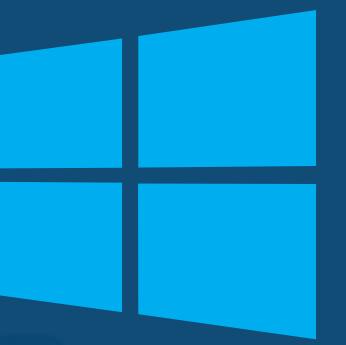




Application



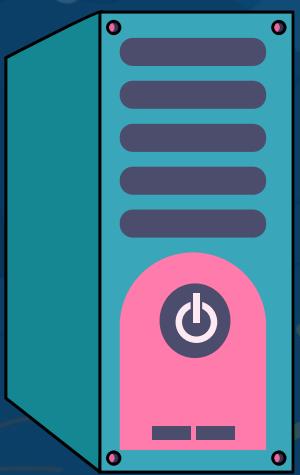
KERNEL



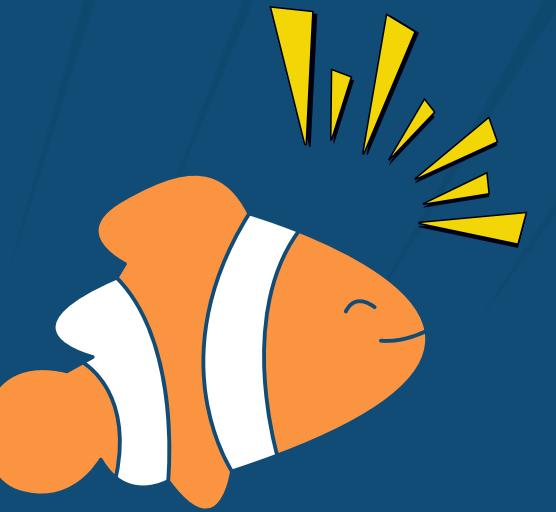
CPU

RAM

Devices

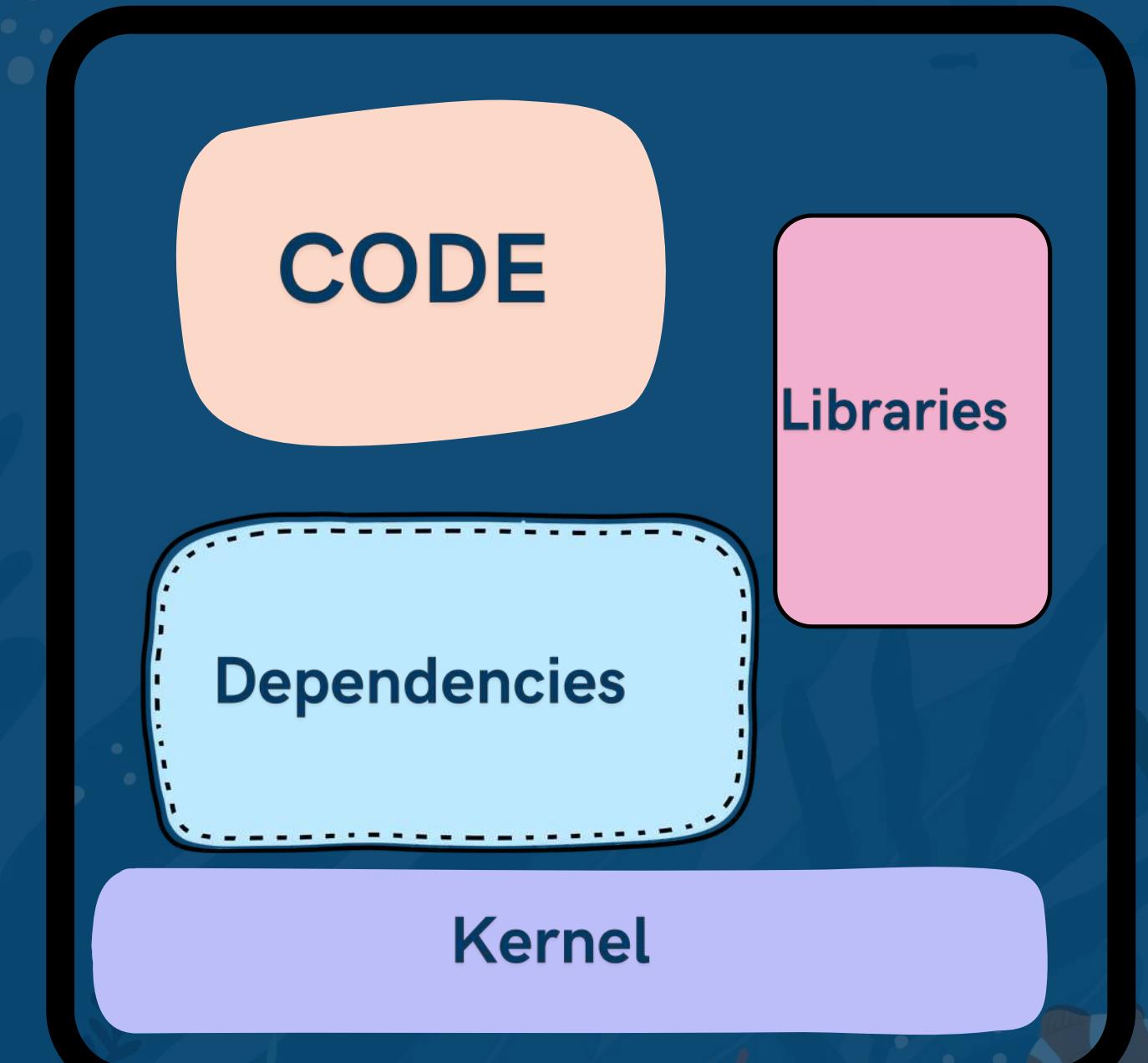


What could be its
Solution ??



Solution

Instead of
giving only
code





ISOLATION

SHIPPING

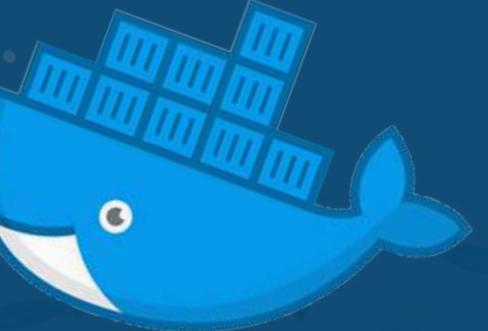
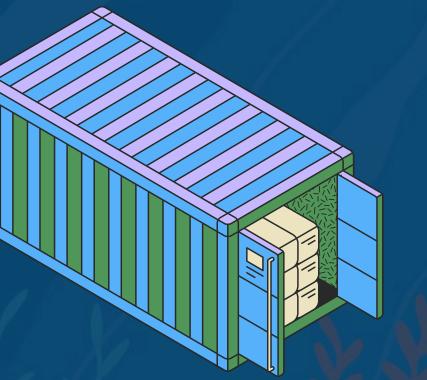
ISOLATION



**Virtual
machine**



Docker

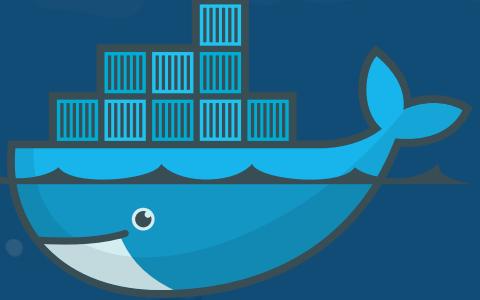
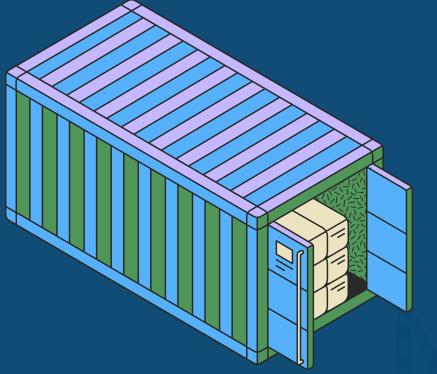


Virtual Machine

A virtual machine (VM) is a **software-based simulation** of a physical computer that operates within another computer's operating system.



Docker



Docker is a tool that helps you run applications in **containers**.

Think of a container like a small box that holds everything an app needs to work, so it can run anywhere without problems.

CODE

Libraries

Dependencies

MINI COMPUTER

**Application level Simulation using
HOST KERNEL**

VM vs Docker

Hardware

Hardware

VM vs Docker

Host Kernel

Hardware

Host Kernel

Hardware

VM vs Docker

Hypervisor

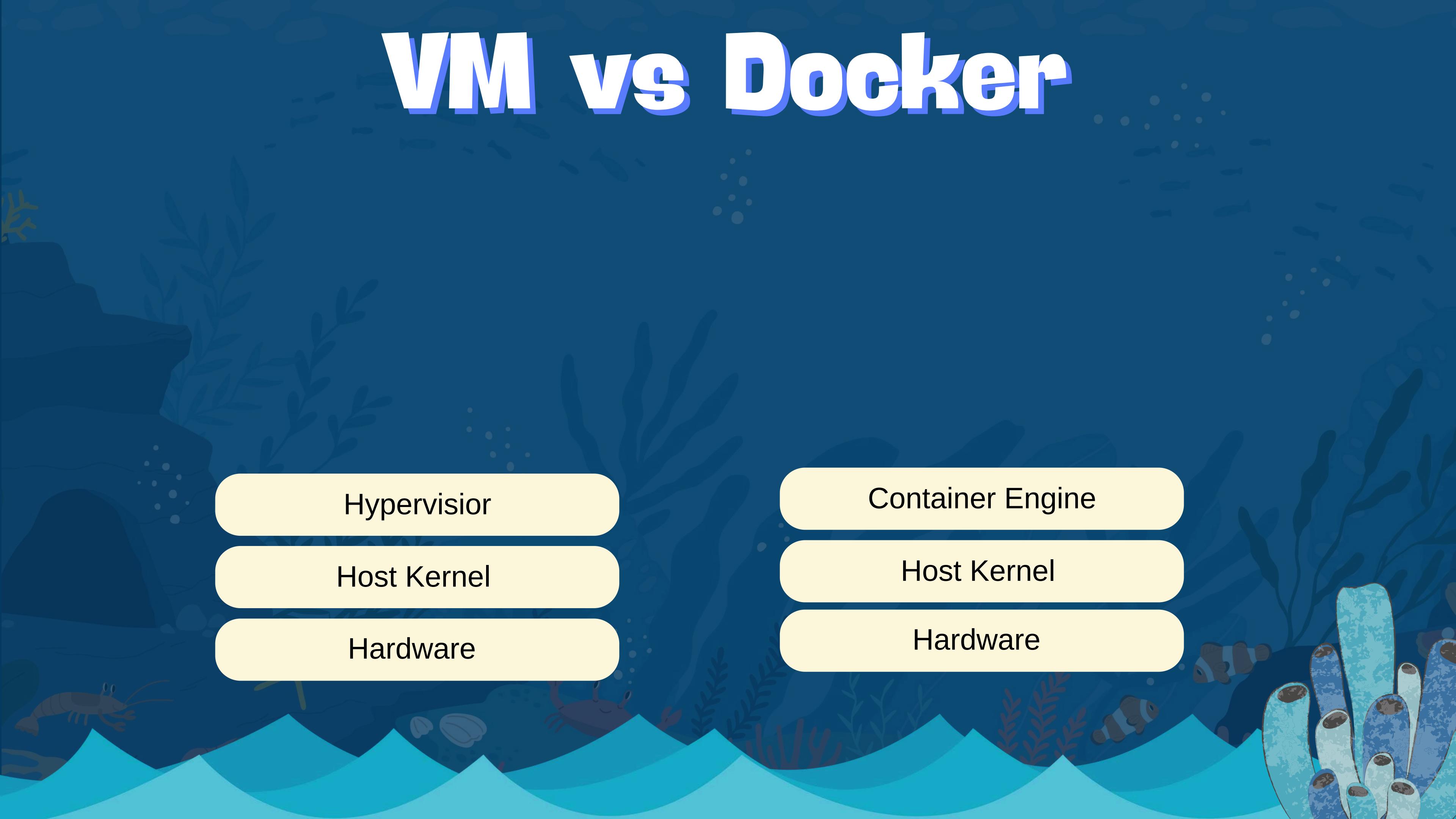
Host Kernel

Hardware

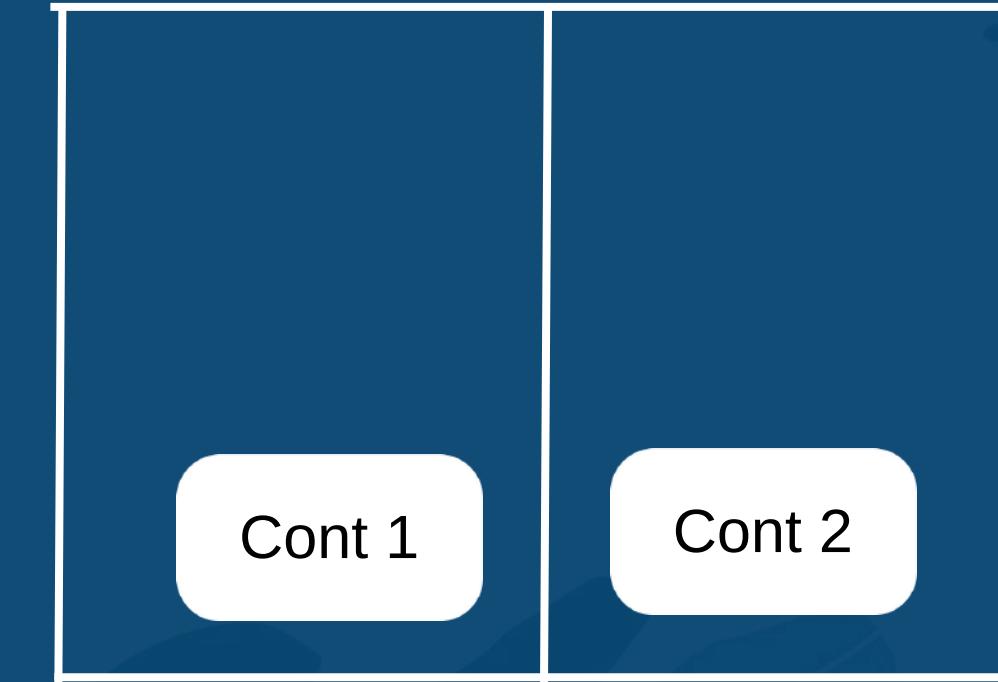
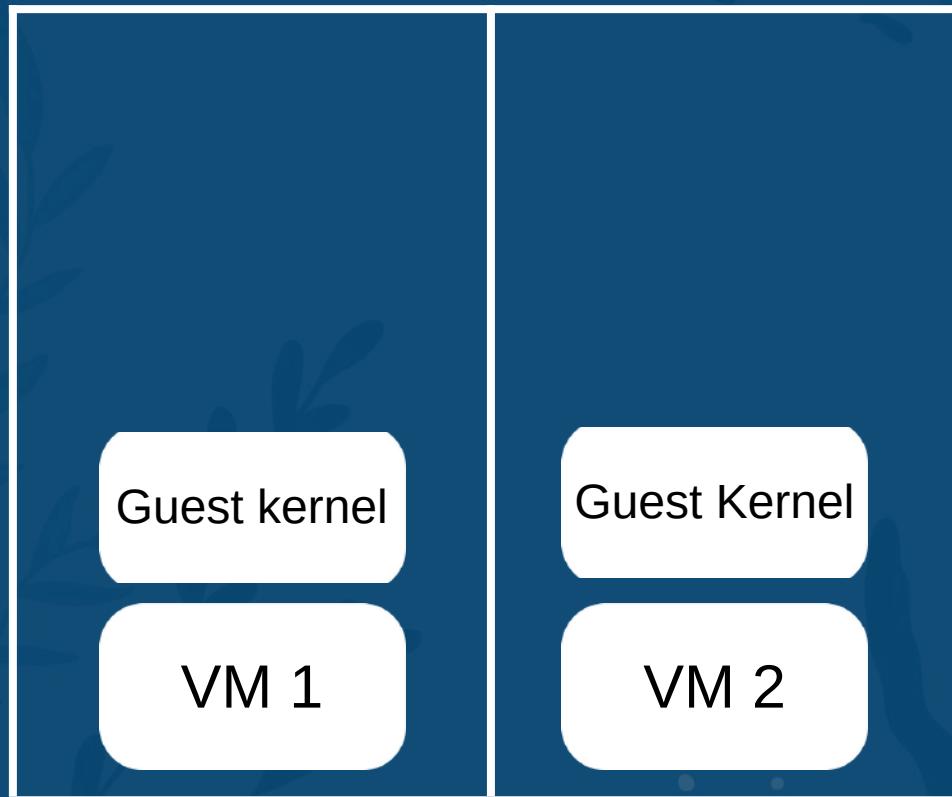
Container Engine

Host Kernel

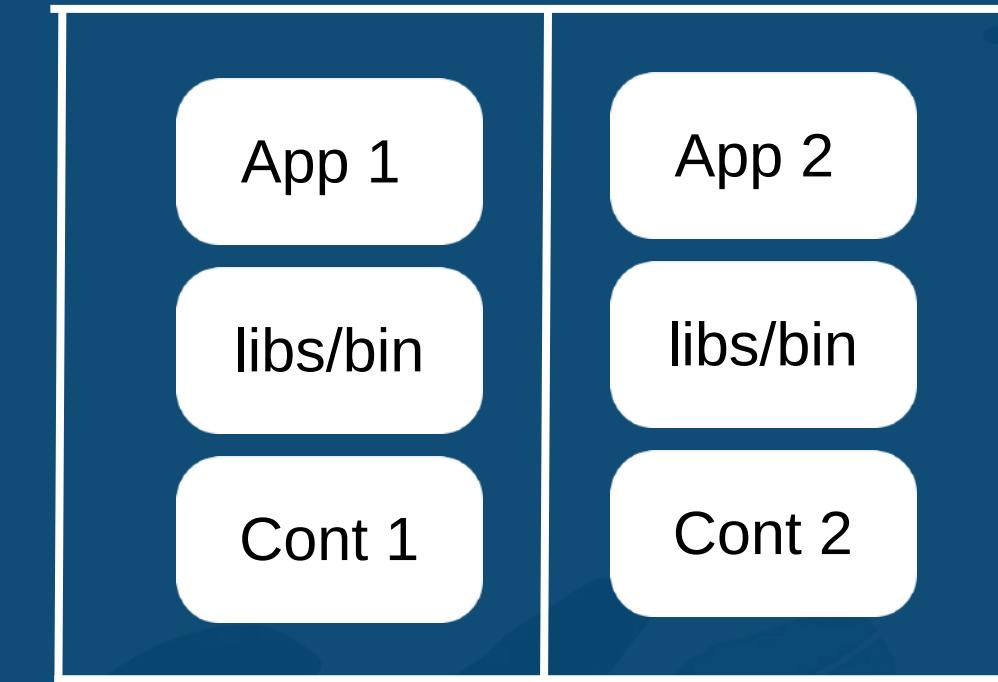
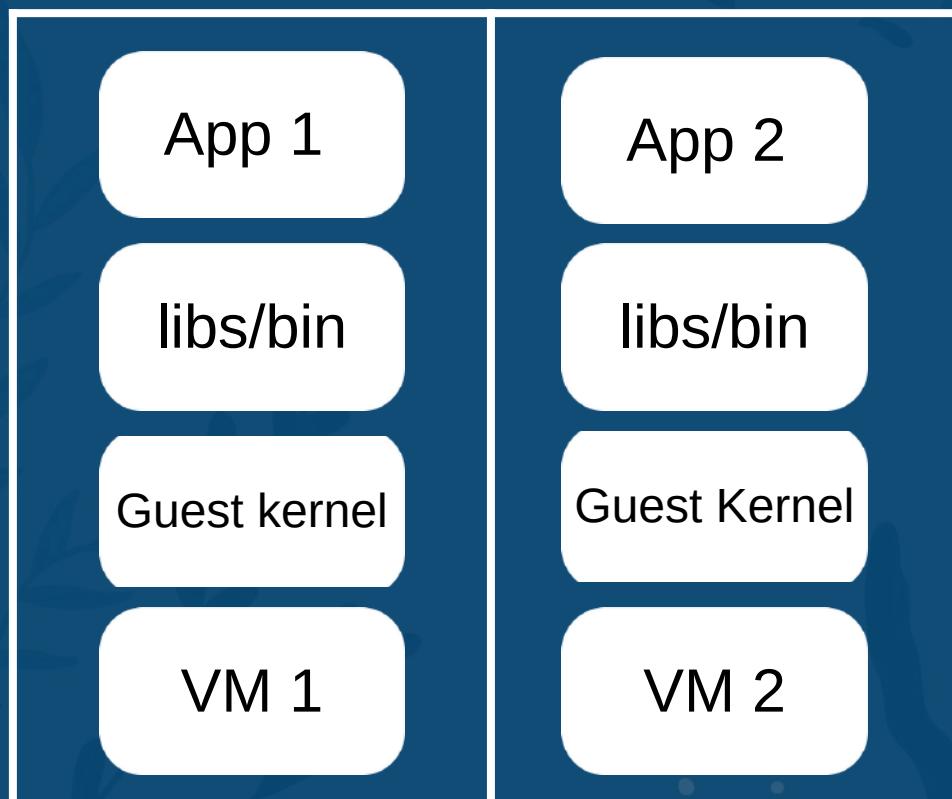
Hardware



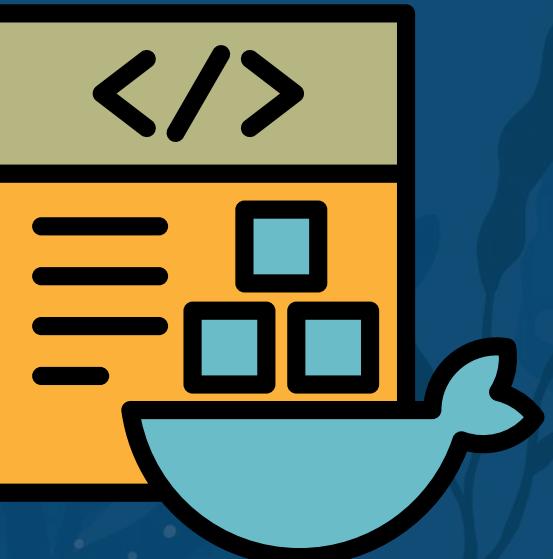
VM vs Docker



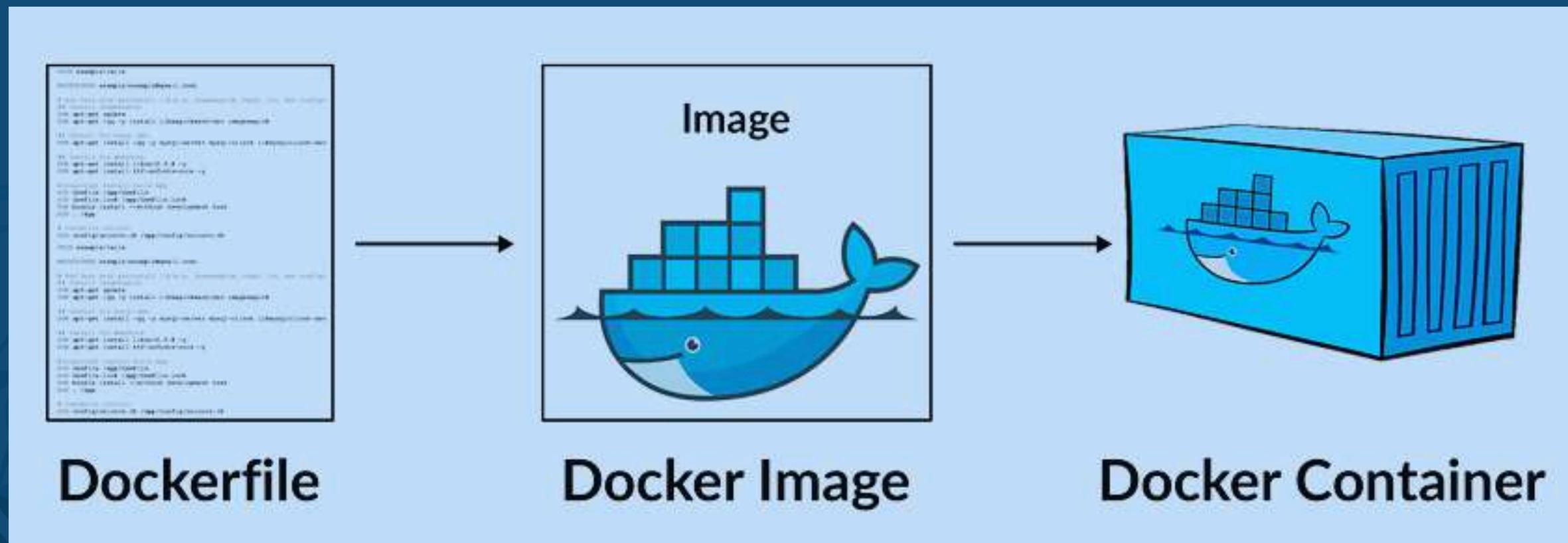
VM vs Docker



So Now we Know what is
docker and where we can use ,
Now lets see how Docker
works on our machine!



Working



Docker File

A **text document** that **includes all the commands** a user can execute in the **command line** to create an **image**.

```
1 # A simple wget container
2
3 FROM alpine
4
5 LABEL VERSION=0.1 \
6      AUTHOR=LMCS \
7      EMAIL=lmcsdeveloper@gmail.com
8
9 RUN apk update \
10   && apk add wget \
11   && rm -rf /var/cache/apk/*
12
13 WORKDIR /root/
14
15 ENTRYPOINT [ "wget" ]
16
17 CMD [ "--help" ]
18
```



Image

A self-contained package
with everything needed to
run an application (code,
libraries, configurations).

Template for
Containers



Container



A lightweight runtime instance of the image that runs the application in isolation

ADVANTAGE



Portability

Scalability

Isolation



Wanna know More on Docker



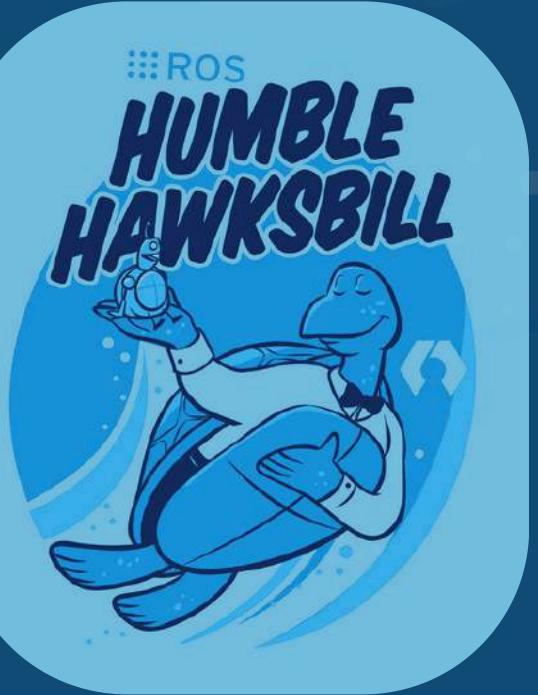
Managing more than one Docker container.

Docker Compose And Kubernetes

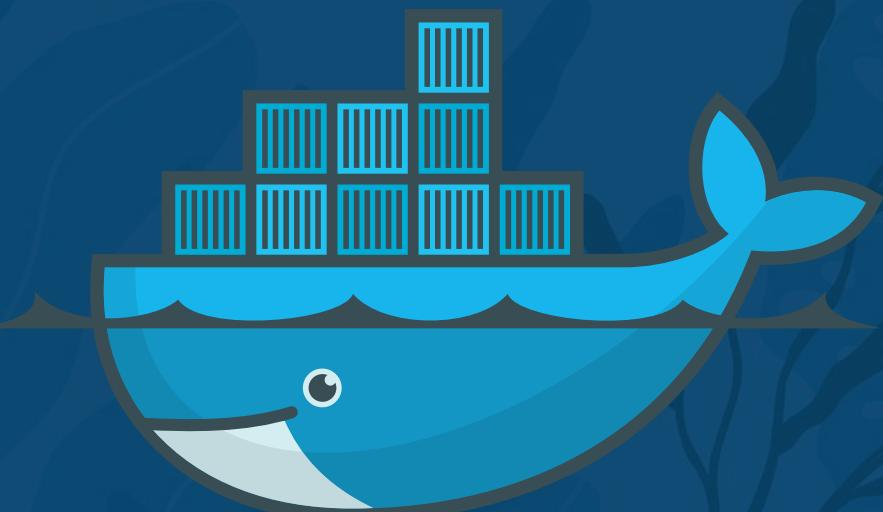
Simplified container management on a local system

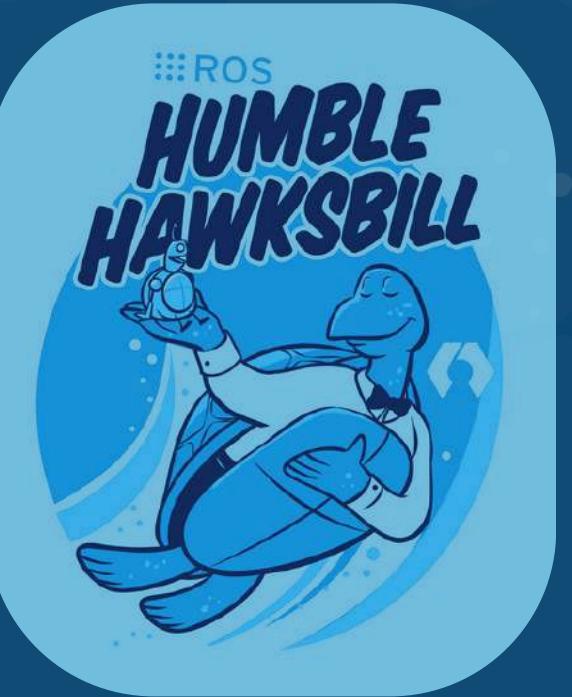
Advanced managing of containers in cloud level , allocates the resource for optimization



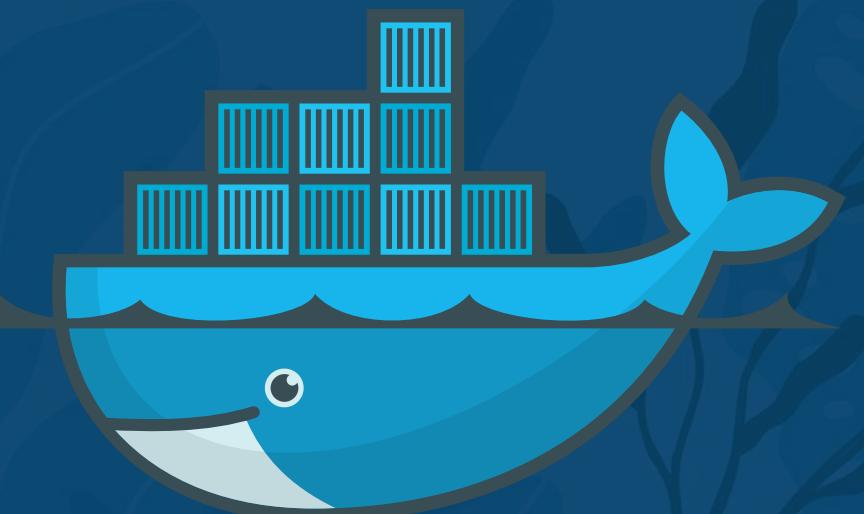


**But Wait ,How come
ROS and Docker
are related ?**

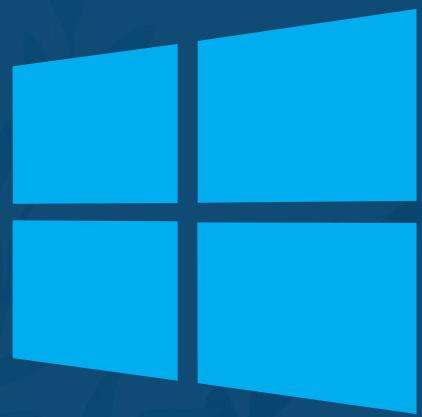




We will Install ROS inside a container



Github



<https://tinyurl.com/35yhtdp7>

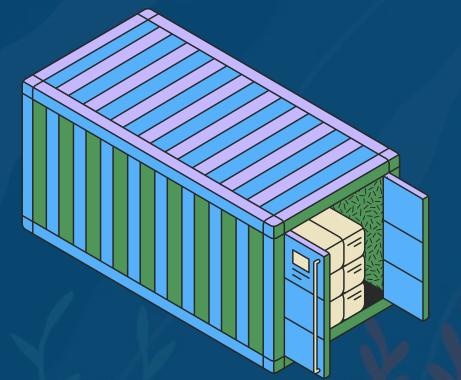


<https://tinyurl.com/y2tfrmk4>

Installation

Ubuntu

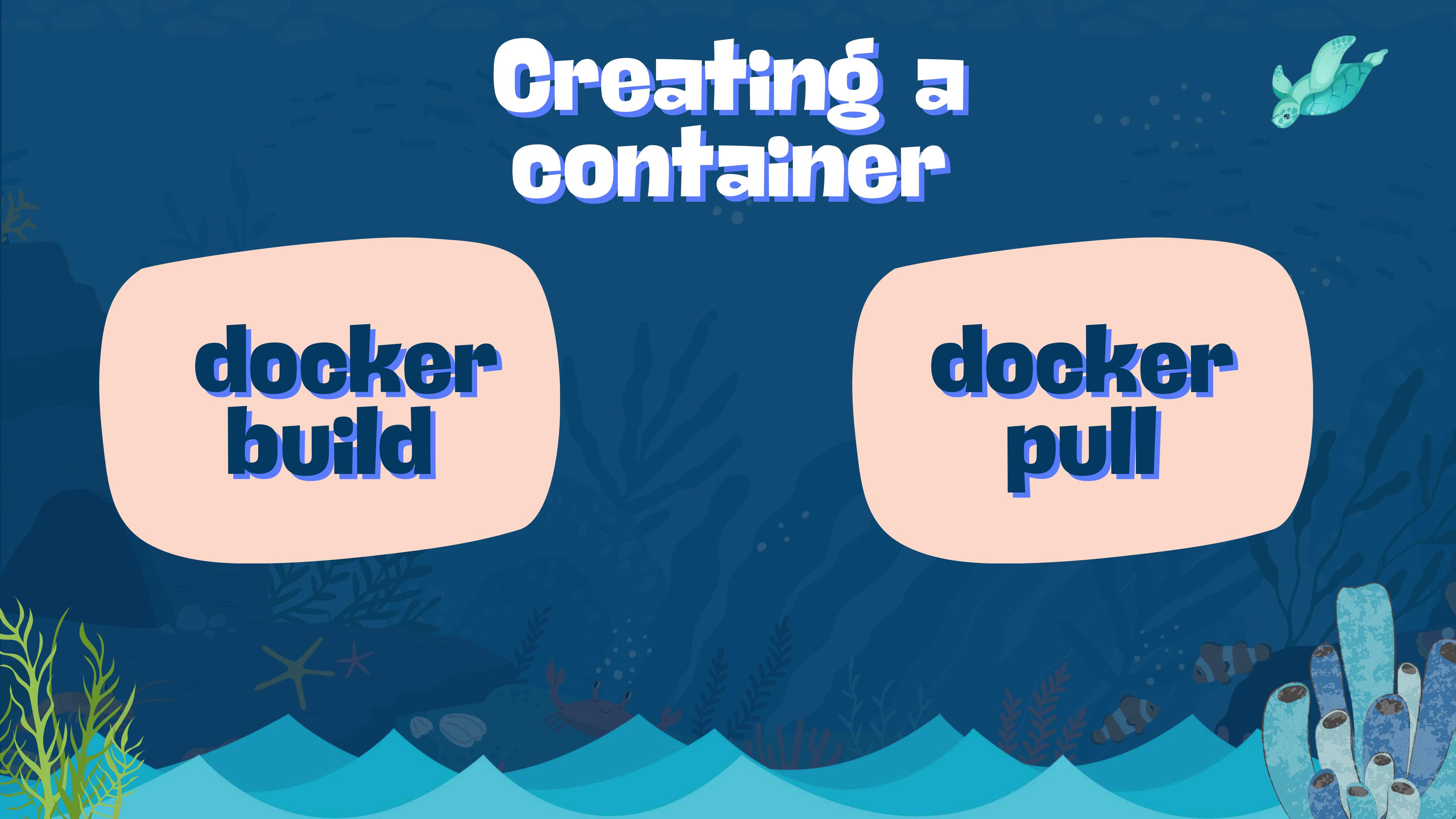
Docker



Creating a container

**docker
build**

**docker
pull**



Mentos Life

