Docker

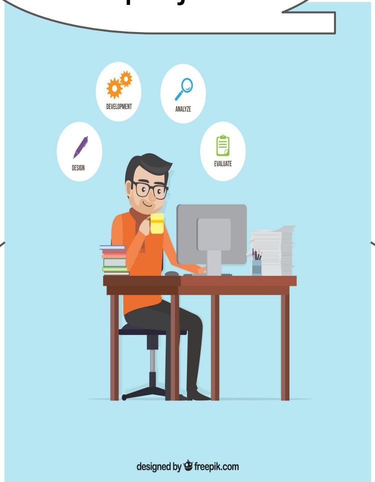
Agenda

1. Why Docker?
2. What is Docker?
3. Architecture of Docker
4. Example
5. Docker-compose
6. Docker commands

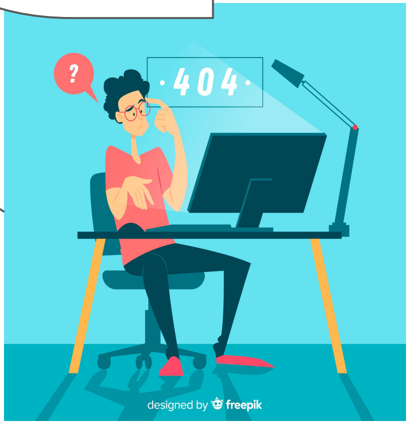
WHY DOCKER ??

It’s working in my system

Test out our new project

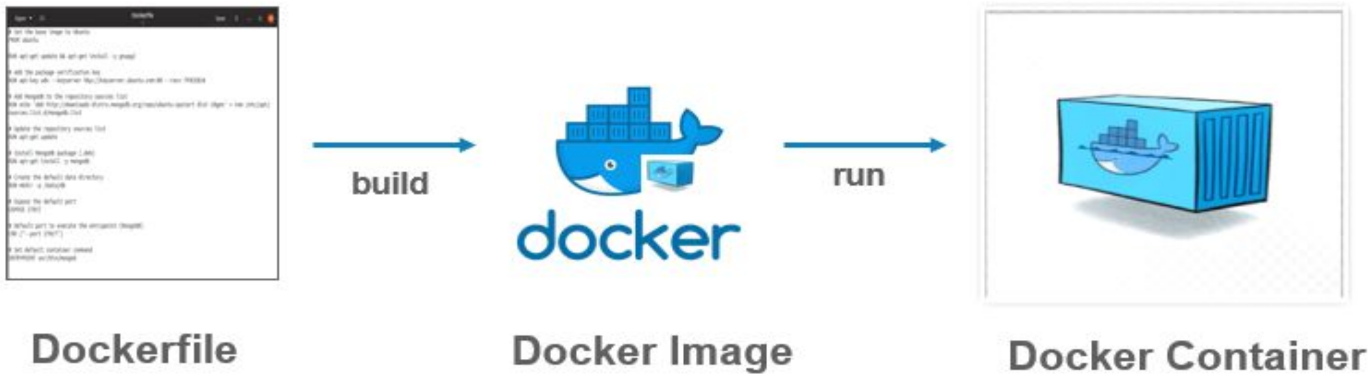


I’m facing an error

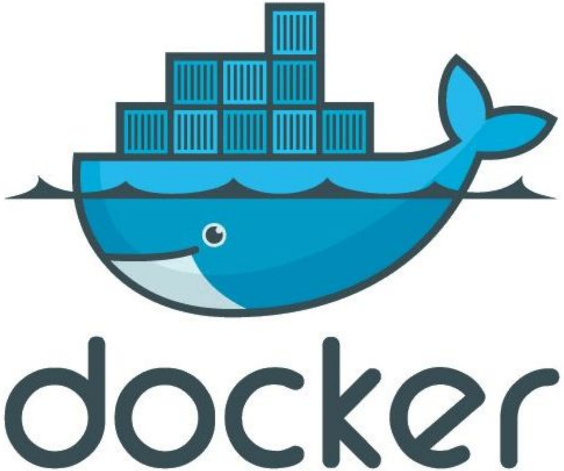


What is Docker?

- Docker is an open platform for developing, shipping, and running applications.
- Docker provides the ability to package and run an application in a loosely isolated environment called a container.

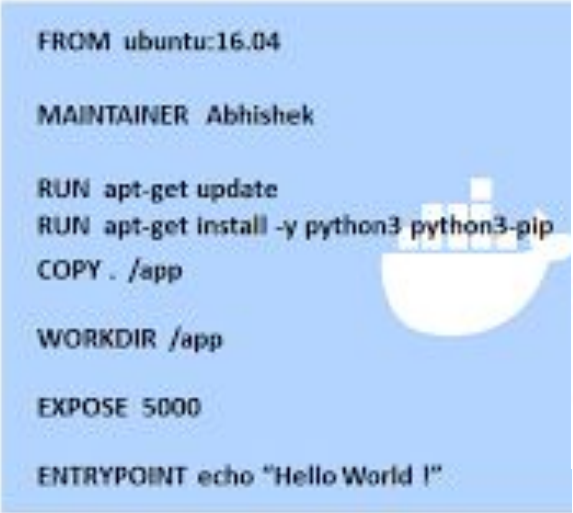


SHIPPING EXAMPLE

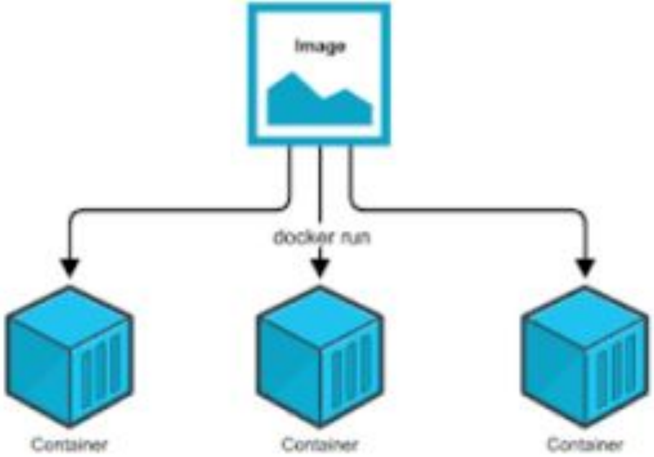


Docker Tools and Terms

- Docker file
- Docker image
- Docker container
- Docker Hub
- Docker daemon
- Docker registry

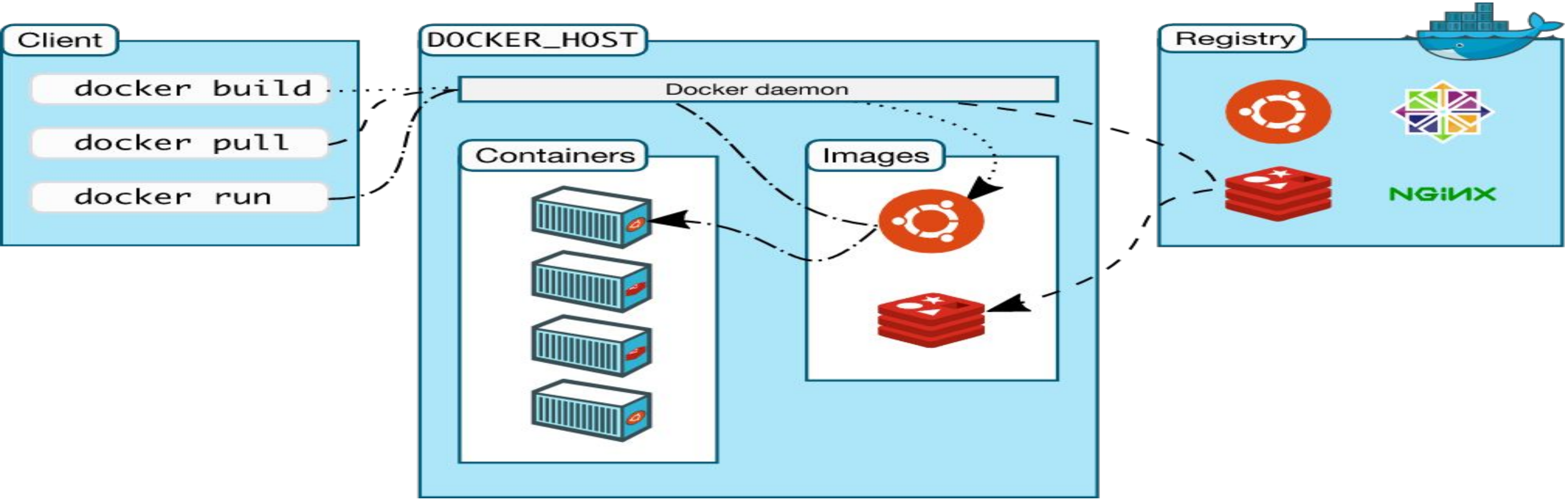


```
FROM  ubuntu:16.04

MAINTAINER  Abhishek

RUN  apt-get update
RUN  apt-get install -y python3 python3-pip
COPY .  /app

WORKDIR  /app

EXPOSE  5000

ENTRYPOINT  echo "Hello World !"
```

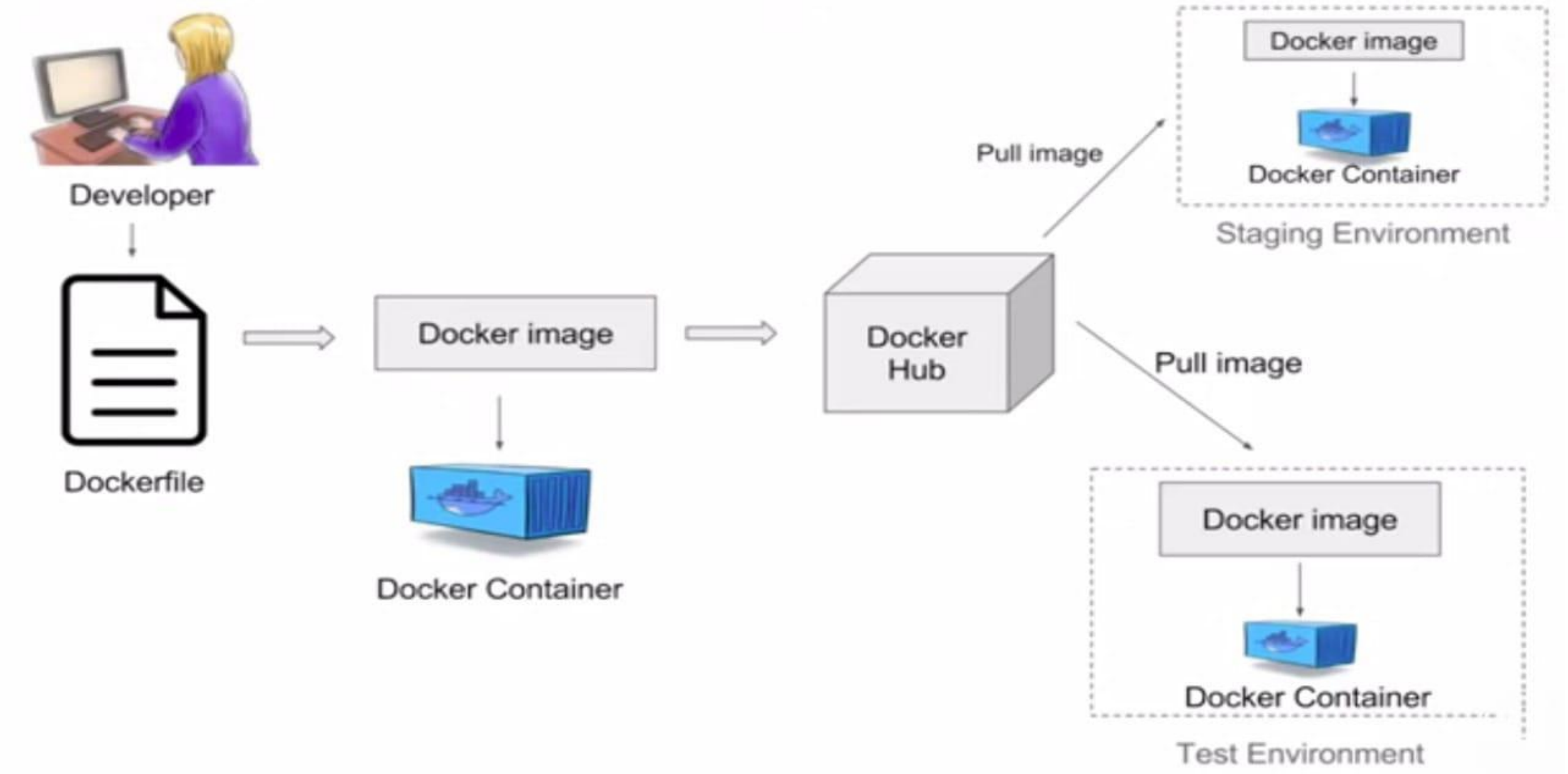


Docker architecture

Build—> Pull—>Run



How it works?


Developer
Dockerfile
Docker image
Docker Container
Docker
Hub
Pull image
Docker image
Docker Container
Staging Environment
Pull image
Docker image
Docker Container
Test Environment

Dockerfile

- A Dockerfile is simply a text-based script of instructions that is used to create a container image
- Automation of Docker image creation
- Instruction sets used in the dockerfile:

  FROM

  RUN

  CMD

  EXPOSE

Steps to create a Dockerfile

Step 1: Create a file named Dockerfile

Step 2: Add instructions in Dockerfile

Step 3: Build Dockerfile and create the image using command :

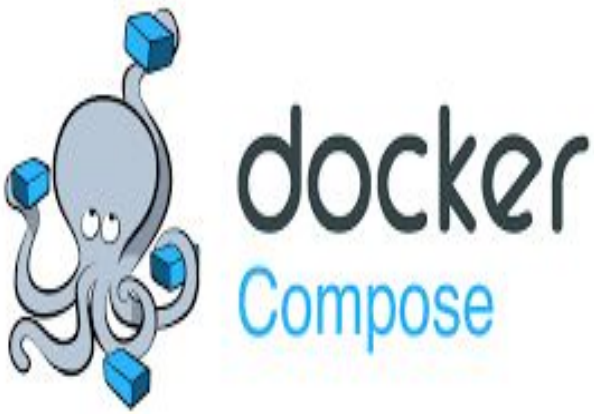
```
docker build -t  <imagename>:<tag> <Directory of Dockerfile>
```

Step 4: Run image to create container

```
docker run --name <container_name> <image_name>
```

Docker-Compose

- Tool for defining and running multiple containers



Three steps

1. Define your app’s environment with a `Dockerfile`.
2. Define the services that make up your app in `docker-compose.yml`
3. Run `docker compose up` and the Docker compose command starts and runs your entire app.

Steps to create docker compose file

Step 1: Install docker compose

Step 2: Create docker compose file at any location on your system

```
docker-compose.yaml
```

Step 3: Check validity of file by command:

```
docker compose config
```

Step 4: Run docker-compose.yaml file by command:

```
docker compose  up -d
```

Step 5: Bring down application by command:

```
docker compose  down
```

CHEAT SHEET

BASIC COMMANDS

Reminder

If You Have Installed Docker and Docker-compose using sudo , then

```
$ sudo <command>
```

CONTAINERS

```
#To Run A Container
$ docker run < image name >

#To Start A Container
$ docker start <container name>

#To Stop A Container
$ docker stop <container name>

# Restart A Container
$ docker restart <container name>

# To list running Containers
$ docker ps

# To Remove A Container
$ docker rm <container name>

# Execute Command Inside Container
$ docker exec <container_name> <command>

# Get Into A Container
$ docker exec -it <container_name> bash

# Get Logs Of A Container
$ docker logs <container_name>
```

DOCKER COMPOSE

```
#To Create & Run Containers
$ docker-compose -f <yaml file> up

#To stop & Remove Containers
$ docker-compose -f <yaml file> down
```

IMAGES

```
#To Pull An image from repository
$ docker pull < image name >

#To List Image Digest
$ docker images

#To Remove Image
$ docker rmi <image name>

# To Build Image From Dockerfile
$ docker build -f < Dockerfile >
```

DOCKER CLEANUP

```
#Delete all stopped containers
$ docker container prune

#Delete All Unused Images
$ docker image prune

#To clear entire docker
$ docker system prune
```

THANK YOU