

DOCKER

IA FRAMEWORKS

CODE DEVELOPMENT FOR DATA SCIENTIST

OBJECTIVE:

- Go further than data exploration and training model.
- Run code on real dataset with more computation power.
- Discover tool that you will need to know if you need to deploy model.

A THREE PARTS [LAB](#).

- Write **python script**.
- Run code on **Google's Virtual Machine**.
- Run code on **Google's Virtual Machine** within **Docker container**.

CODE DEVELOPMENT FOR DATA SCIENTIST

OBJECTIVE:

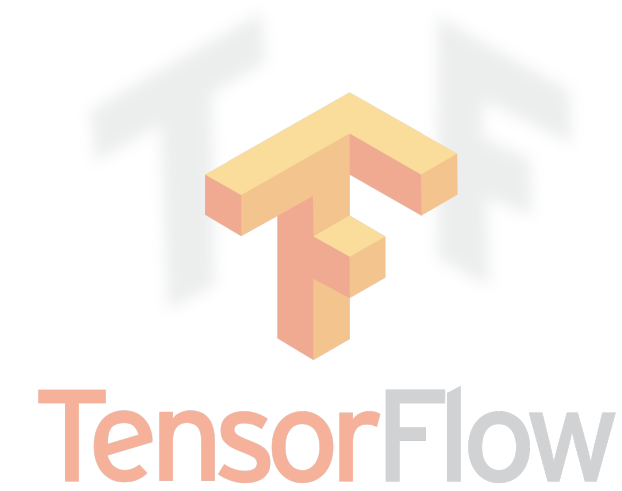
- Go further than data exploration and training model.
- Run code on real dataset with more computation power.
- Discover tool that you will need to know if you need to deploy model.

A THREE PARTS [LAB](#).

- Write **python script**.
- Run code on **Google's Virtual Machine**.
- —> Run code on **Google's Virtual Machine** within **Docker container** <—.

GOOGLE CLOUD PLATFORM

ML Python Libraries



Python Environment



Viz' Python Libraries



Framework & Tool



TABLE OF CONTENTS

Introduction

Application

INTRODUCTION

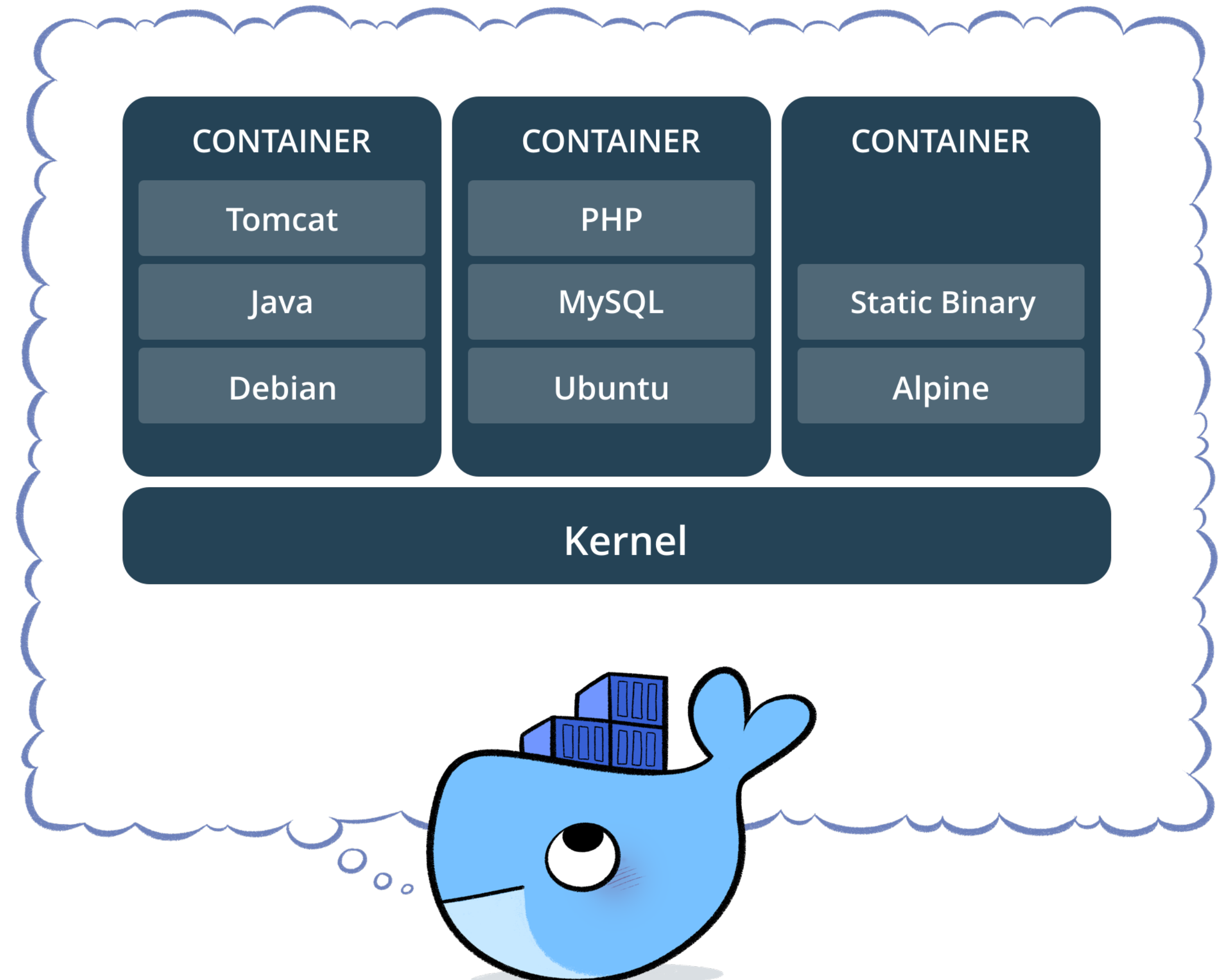
WHAT IS DOCKER ?

As for python environment, docker enables to create environment isolated from each other called **container**.

This environment are built at an **OS-level**, which means you make run a windows program on unix and *vice versa*.

It enables to easily recreate environment for application whatever the dependencies are.

Facilitate development and test.



WHY SHOULD I USE DOCKER AS A DATA SCIENTIST ?

- **SHARING YOUR WORK** within a docker container ensure that anyone can recreate it whatever the devices he is using. More people can have access to your work.
- **USE THE WORK OF OTHERS.** Docker is widely used today to share their works. Being comfortable with it ensure you easy access to it.
- **EASE USE OF COMPUTATIONAL POWER.** Developing within a docker container ensure you to easily move your code to devices with more computational power.
- **EASE USE OF DEPLOYMENT.** Developing within a docker container ensure you to easily move your model to be use in production environment.

DOCKER TERMINOLOGY

- **DOCKERFILE**: Recipe for creating an image
- **IMAGE**: It's like a turned-off VM which contains the tools you want. Ex: Ubuntu + Tensorflow with Nvidia drives and a Jupyter server.
- **CONTAINER**: is an instantiation of an image. You can have multiple copies of the same image running.
- **DOCKERHUB / IMAGE REGISTRY**: Place where Organisation or individual can post public or private docker images for ease collaboration and sharing.
<https://hub.docker.com>

APPLICATION

LAB - OBJECTIVES

- Write a **Dockerfile** based on the official *Tensorflow Dockerfile* available on *DockerHub*.
- Use the **Dockerfile** to build a Docker **image**.
- Launch **containers** with different options from the built **image**.
- Run training and prediction on cats vs dogs (with code developed on previous lab) on a google cloud instance.

DOCKERFILE

- The Dockerfile we use on this lab is based on the official [Tensorflow gpu image](#).
- Below is the dockerfile we'll use.

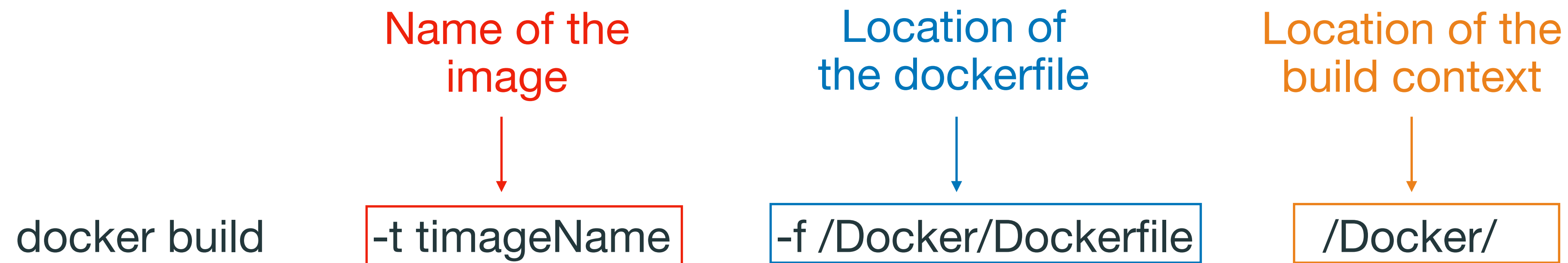
```
FROM tensorflow/tensorflow:2.1.0-gpu-py3
ENV TZ=Europe/Paris
RUN ln -snf /usr/share/zoneinfo/$TZ /etc/localtime && echo $TZ > /etc/timezone
RUN apt-get update && apt-get install -y python-opencv python-tk unzip pv
RUN pip install h5py scikit-image tqdm
```

- The **FROM** instruction sets the based image. A valid docker file starts with a from instruction.
- The **ENV** instruction sets environment variable (PYTHONPATH, etc.)
- The **RUN** instruction will execute any commands.

Official documentation of [dockerfile](#).

BUILD IMAGE

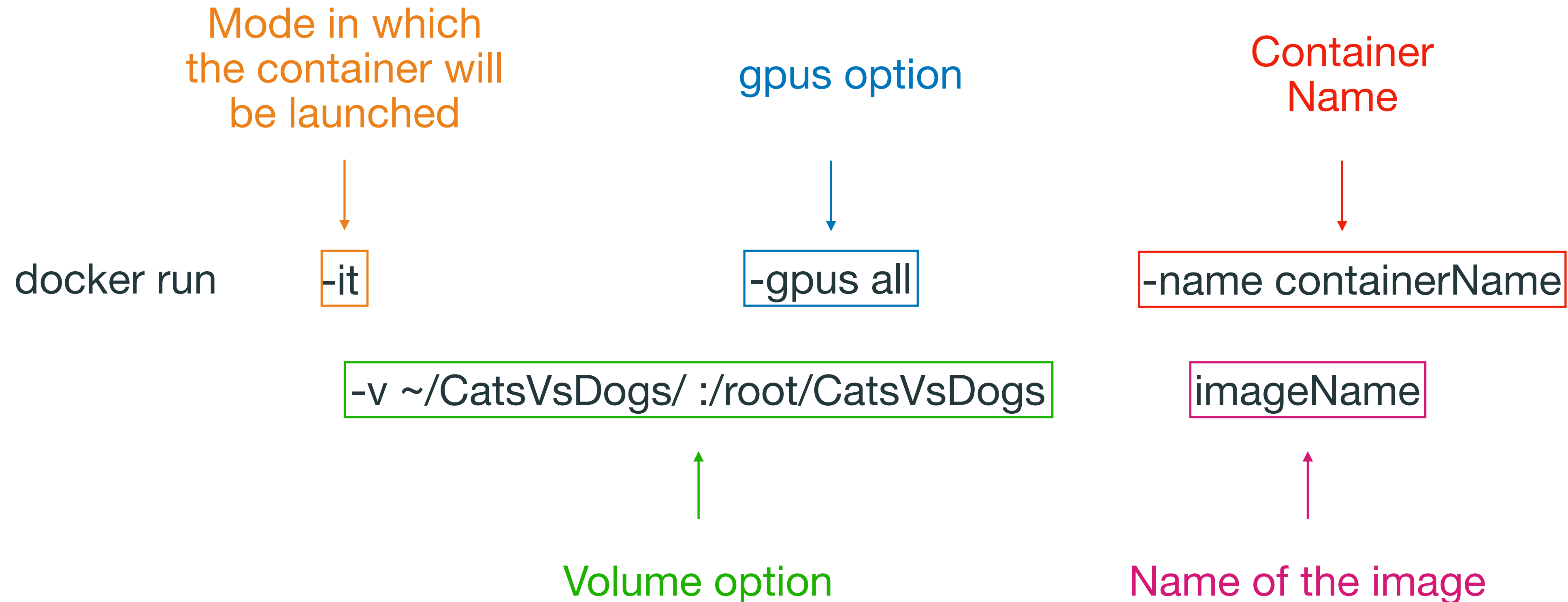
Run the build command in in order to build a docker **image**



The **build context** is the location of the folder to which the ADD statement will reference. All the files required by the **dockerfile** to build the image are located here.

RUN CONTAINER

Run the run command in order to run your **container**.



Mode of the **container** could be:

- it : interactive mode
- dt : detached mode

MOUNTED VOLUME

The -v option allows you to use some data you have in your machine within a container.

declare
option



-v

Local directory that
will be mounted
inside the container



~/CatsVsDogs/

directory on your
container where the
local directory will be
mounted



: /root/CatsVsDogs

OTHER UTILS COMMAND

- *sudo docker images ls -a*: list all existing images.
- *sudo docker containers ls -a*: list all existing containers.
- *sudo docker start/stop/rm container_name*: start, stop or remove a container.
- *sudo docker exec container_name `command`*: Execute the `command` within a container.

You are not allowed to run sudo command on your INSA's session.

- *Run codes on **GCP VM instances** (with gcloud ssh connection).*

Once your image is build on the instances:

- *Run same pipeline code with some adjustment.*
- *gcloud compute ssh —container ContainerName*