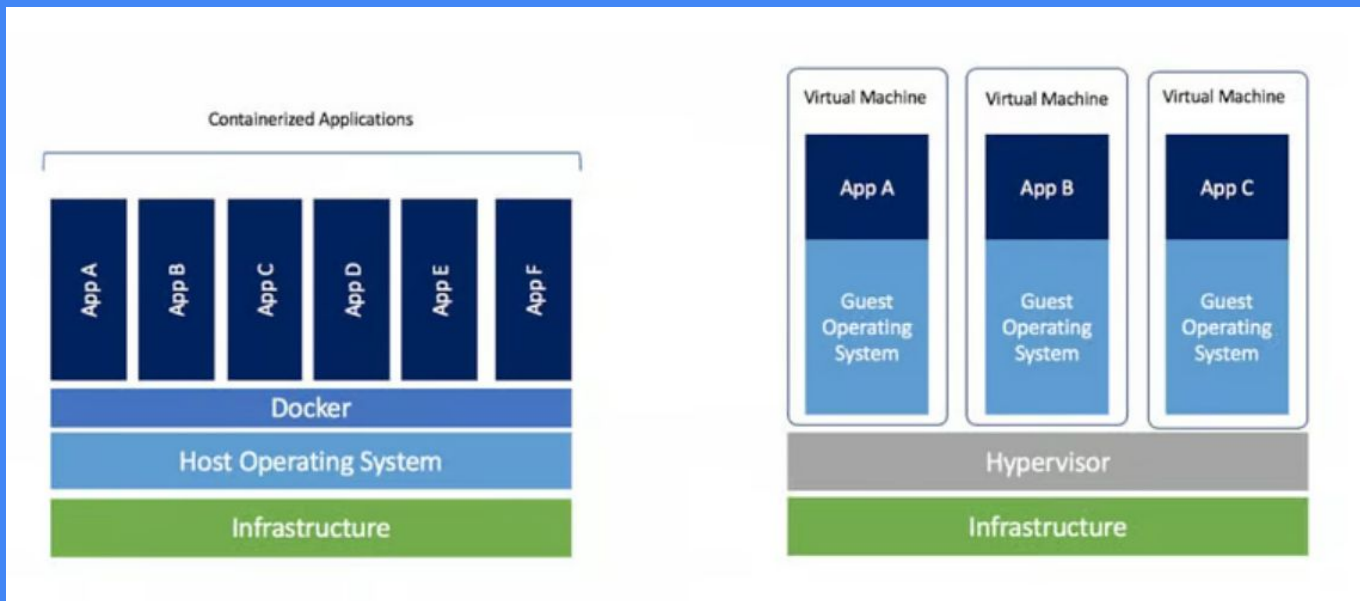
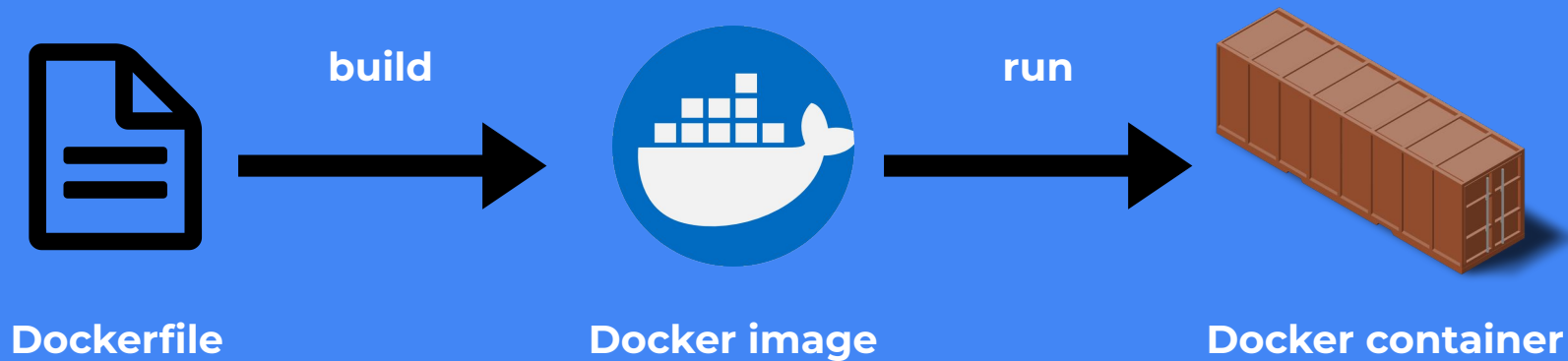




Introduction to docker






A Dockerfile is a text file that contains a series of instructions for building a Docker image. In a Dockerfile, you specify the base image, define any dependencies or packages that need to be installed, copy files into the image, and configure the container environment. Dockerfile instructions are executed in order, and each instruction creates a new layer in the image.

```
FROM python:3.10.9
WORKDIR /api
COPY ./ /api

RUN ls

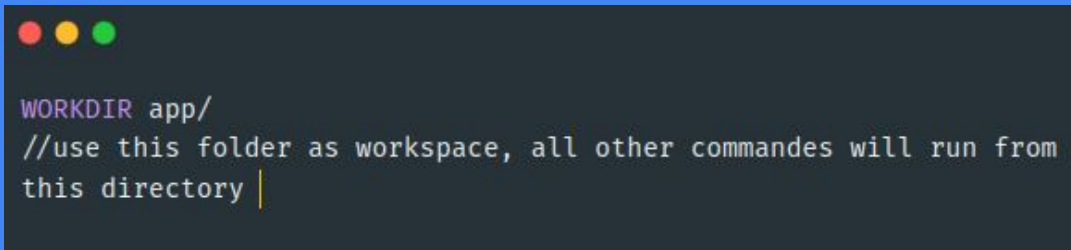
RUN pip install --upgrade pip
RUN pip install -r requirements.txt
RUN pip install protobuf==3.20.*
RUN pip list
CMD ["uvicorn", "main:app", "--host", "0.0.0.0", "--port", "8000"]
EXPOSE 9876
    return go(f, seed, [])
}
```

The **FROM** instruction in a Dockerfile is used to specify the base image that will be used for building the Docker image. The base image serves as the starting point for building the new image, and it contains the basic operating system and any other dependencies that are required for the application or service that will run inside the container.



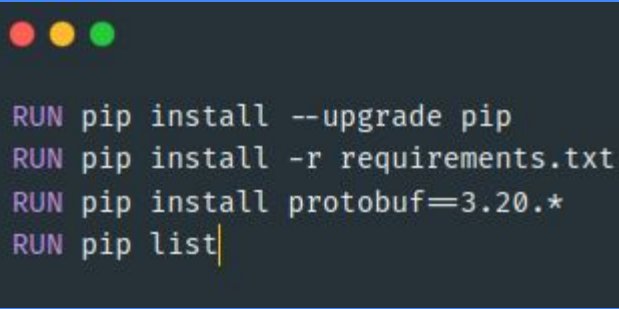
```
FROM python:3.10.9
//use python image version 3.10.9 as base image
```

The **WORKDIR** instruction in a Dockerfile is used to set the working directory for any subsequent instructions in the Dockerfile. This means that any commands that are executed after the **WORKDIR** instruction will be run from the specified directory. For example, if you want to set the working directory to `/app` in your Dockerfile, you can use the following **WORKDIR** instruction:



```
WORKDIR app/  
//use this folder as workspace, all other commandes will run from  
this directory |
```

The **RUN** instruction in a Dockerfile is used to execute a command during the image building process. The command is executed in a new layer on top of the current image and the resulting changes are committed to the new image. For example, if you need to install some packages or dependencies in your Docker image, you might use the following RUN instruction in your Dockerfile:



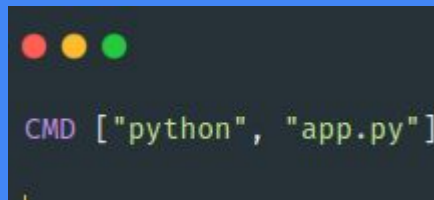
```
RUN pip install --upgrade pip
RUN pip install -r requirements.txt
RUN pip install protobuf=3.20.*
RUN pip list
```

The **COPY** instruction in a Dockerfile is used to copy files or directories from the host system into the Docker image being built. This allows you to include application code, configuration files, or other assets in the image that are required for the application or service to run. For example, if you have an application that consists of a single Python file named `app.py`, you might use the following **COPY** instruction in your Dockerfile to copy the file into the Docker image:



```
COPY app.py /app/
```


The **CMD** instruction in a Dockerfile is used to specify the default command that should be executed when a Docker container is started from the image. This command is typically the main executable for the application or service that is running in the container. For example, if you are building a Docker image for a Python web application that is using the Flask framework, you might use the following CMD instruction in your Dockerfile:



```
CMD ["python", "app.py"]
```

Command	Description
Docker build	Builds a Docker image from a Dockerfile
Docker run	Runs a Docker container from an image
Docker ps	Lists running Docker containers
Docker images	Lists Docker images
Docker stop	Stops a running Docker container
Docker rm	Removes a stopped Docker container

Host

Port 8000

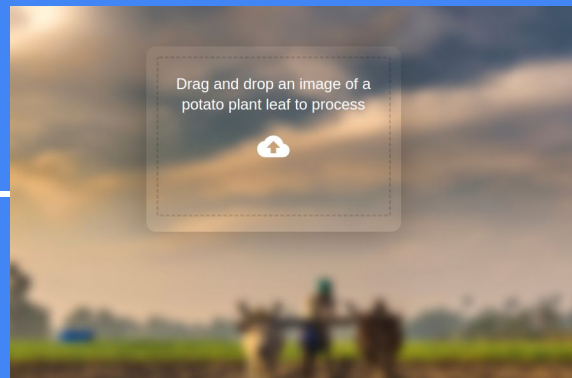
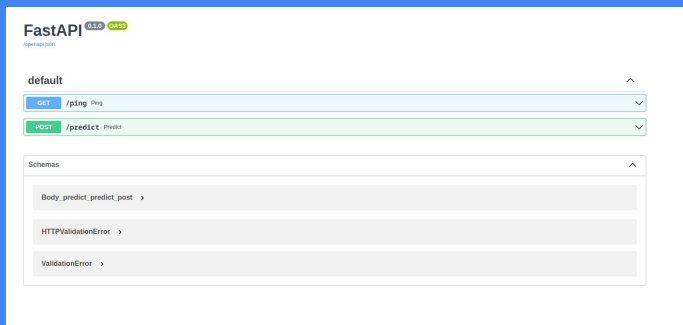
API

ML model

Port 3000

Frontend





Host

Port 8000

API

ML model

Port 3000

Frontend



Host

Port 8000



Port 3000

