docker

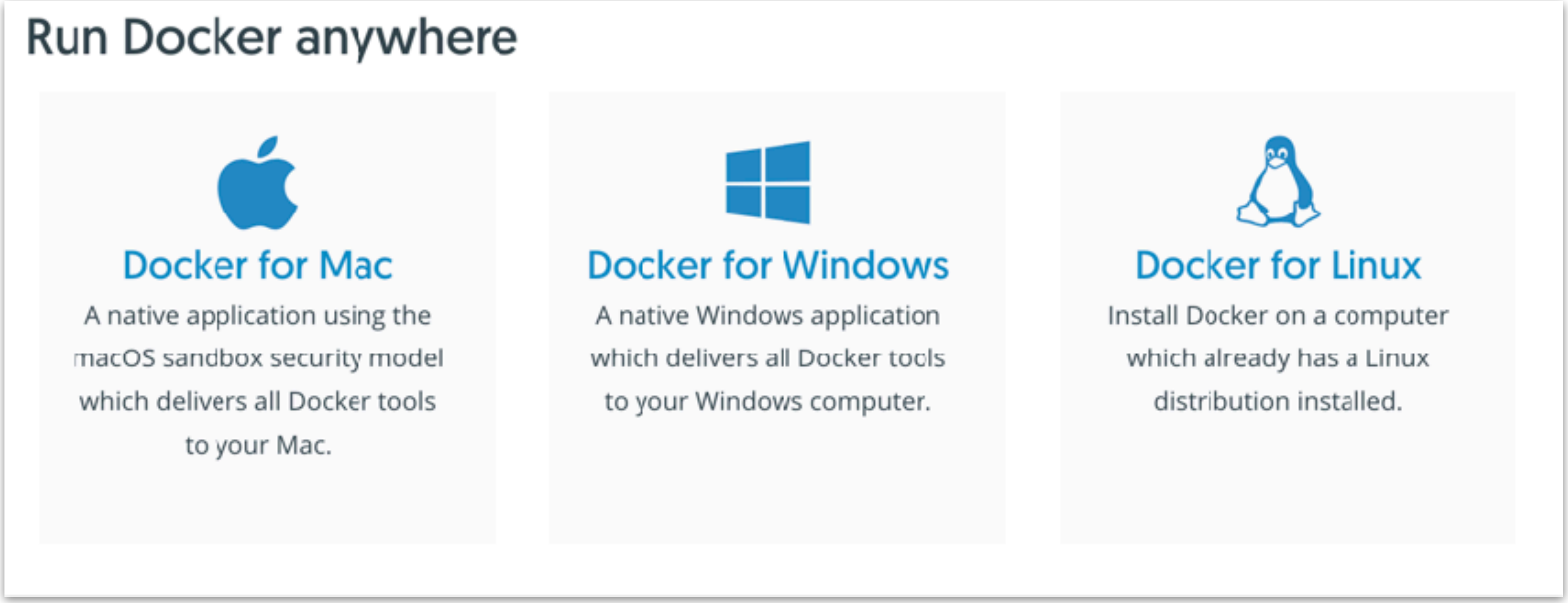
Lucy Owen

Overview

- General intro
- Basics of building an image
- Pulling an image from docker hub
- Demo:
    - Jupyter notebook/lab
    - PyCharm
- Help getting set up

How do I get started?

- Install Docker - https://docs.docker.com/



Run Docker anywhere

Docker for Mac

A native application using the macOS sandbox security model which delivers all Docker tools to your Mac.

Docker for Windows

A native Windows application which delivers all Docker tools to your Windows computer.

Docker for Linux

Install Docker on a computer which already has a Linux distribution installed.

What is Docker?

- A tool designed to make it easier to create, deploy, and run applications by using containers.

- A container is a virtual environment that allows a developer to package an application with all of the necessary ingredients (requirements) for it to run, and bundles it up into one portable package.

- Allows you to create computing environments that can be replicated on most modern computers.

- Allows a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package.

Why use it?

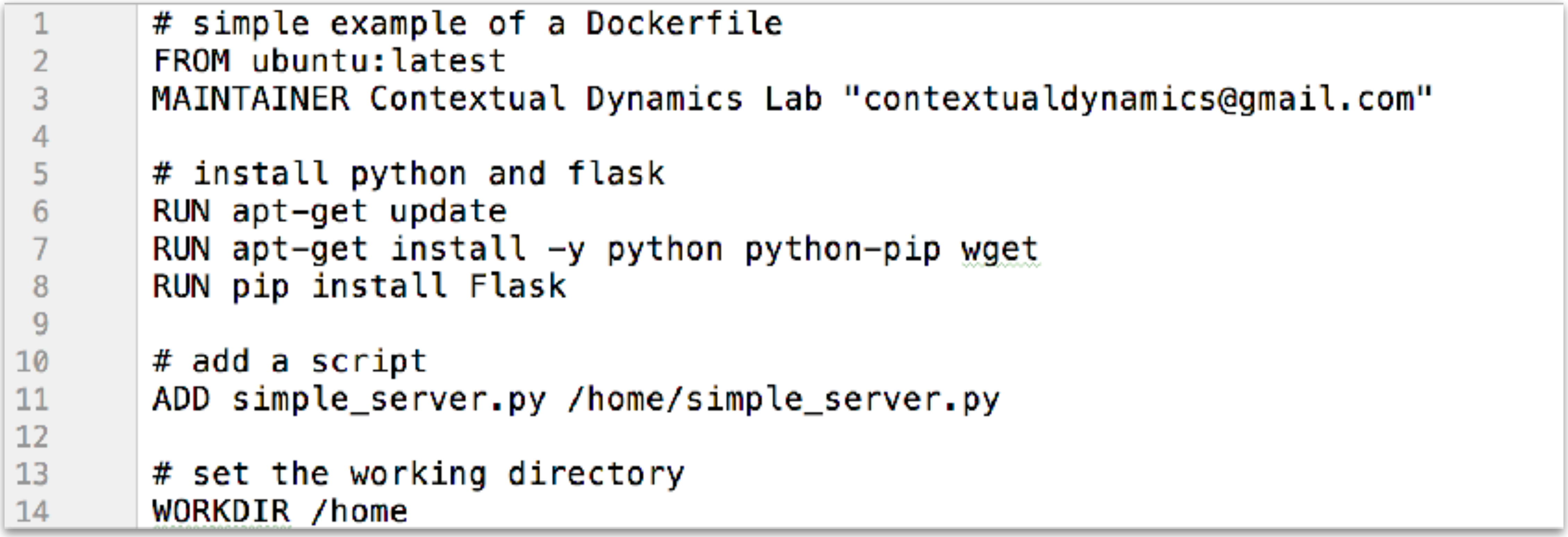
- A sharable computing environment so that everyone can have access to the same tools.
- An easily replicable experiment that can be run from anywhere.
- An analysis pipeline with all of the necessary code to replicate figures and statistics from a published paper.
- An application that runs Jupyter server so that you can access data and software without installing anything locally.

To build your own:

- Follow the tutorial outlined in the README on the GitHub page
    - 1) Make a Dockerfile
    - 2) Build the image
    - 3) Run the container

What is a Dockerfile?

- A set of instructions for building a docker image.



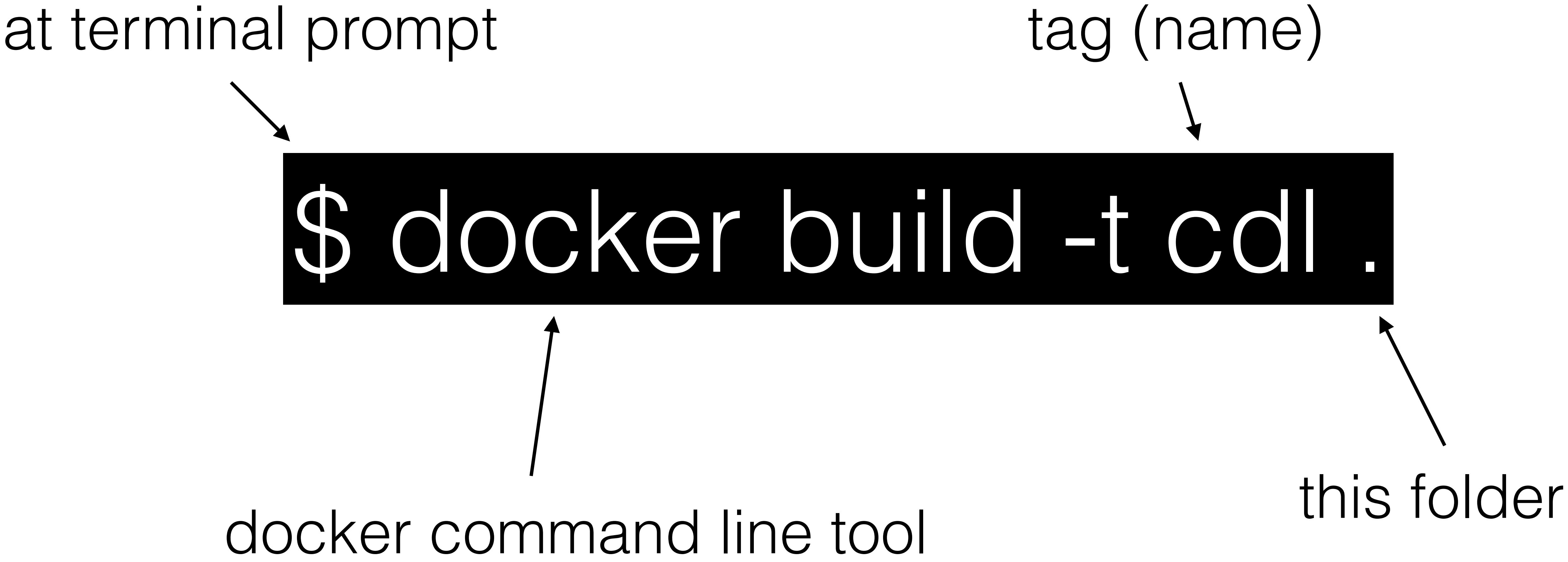
```
# simple example of a Dockerfile
FROM ubuntu:latest
MAINTAINER Contextual Dynamics Lab "contextualdynamics@gmail.com"

# install python and flask
RUN apt-get update
RUN apt-get install -y python python-pip wget
RUN pip install Flask

# add a script
ADD simple_server.py /home/simple_server.py

# set the working directory
WORKDIR /home
```

How do I build an image?



at terminal prompt

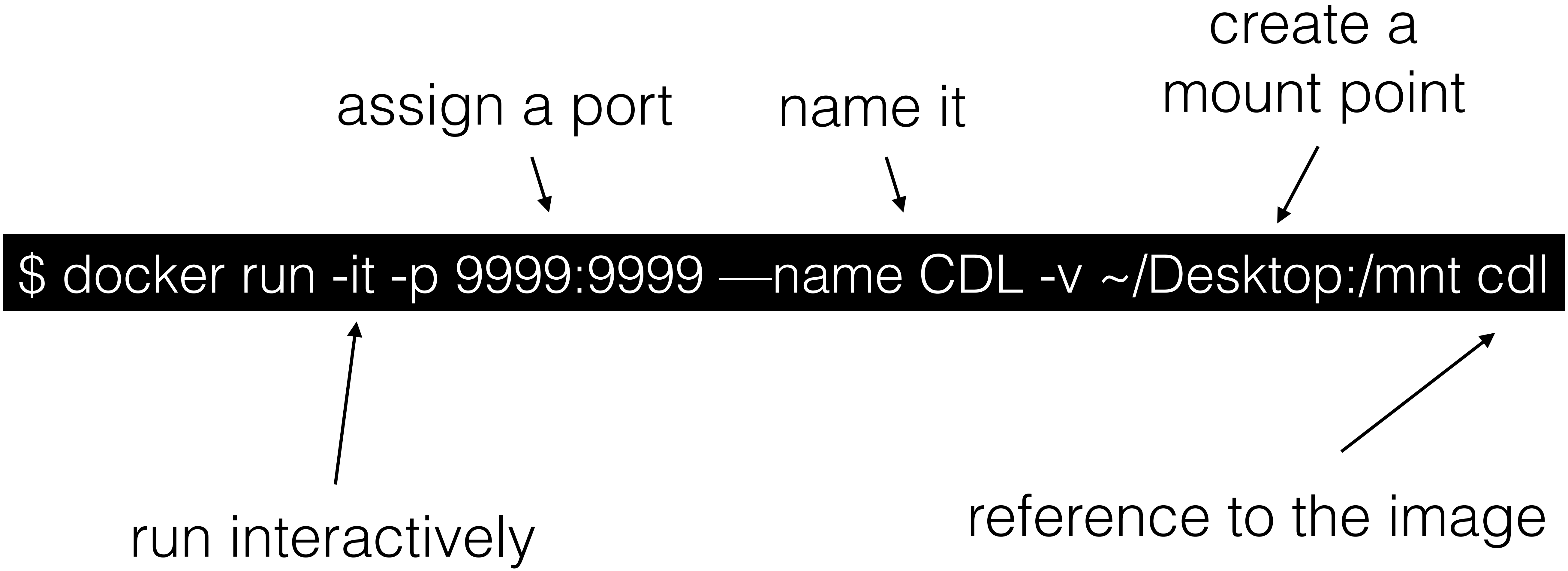
tag (name)

```
$ docker build -t cdl .
```

docker command line tool

this folder

How do I run it?



assign a port

name it

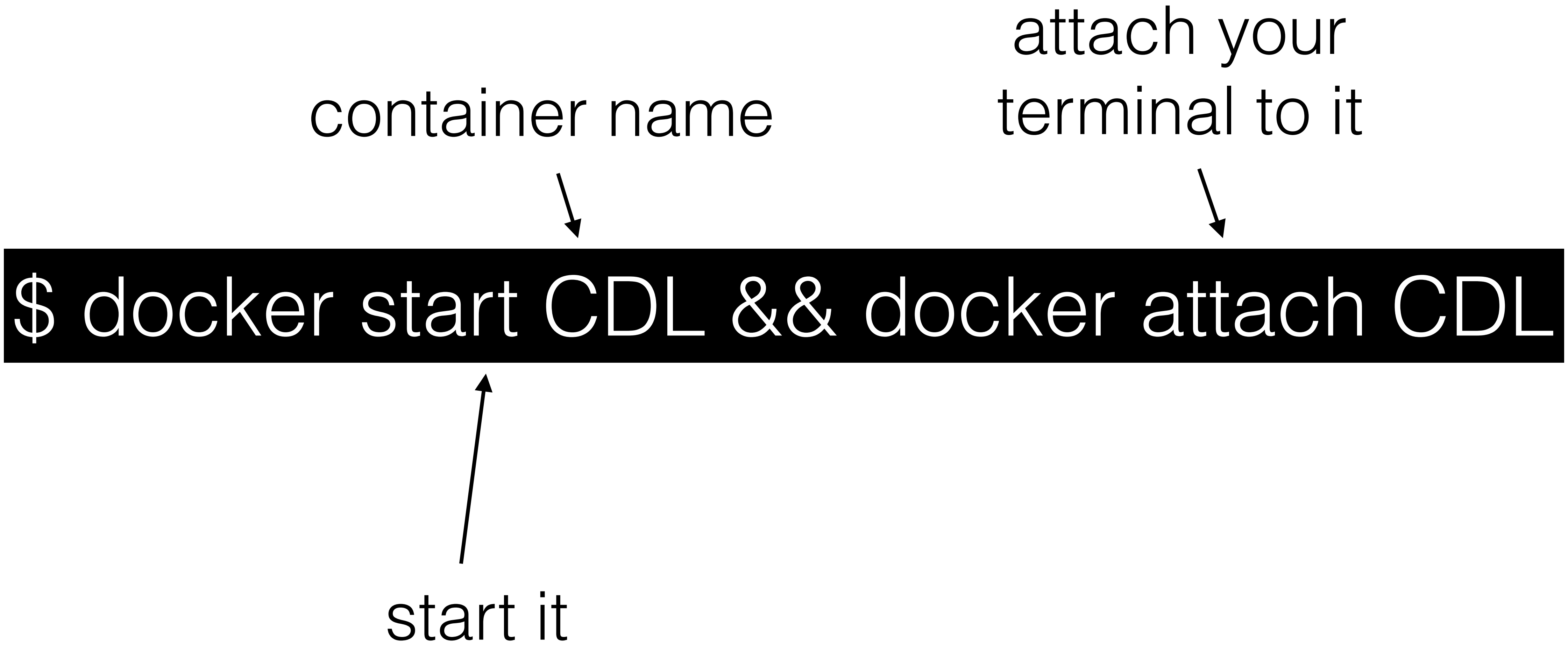
create a
mount point

```
$ docker run -it -p 9999:9999 —name CDL -v ~/Desktop:/mnt cdl
```

run interactively

reference to the image

How do I open it again?



container name

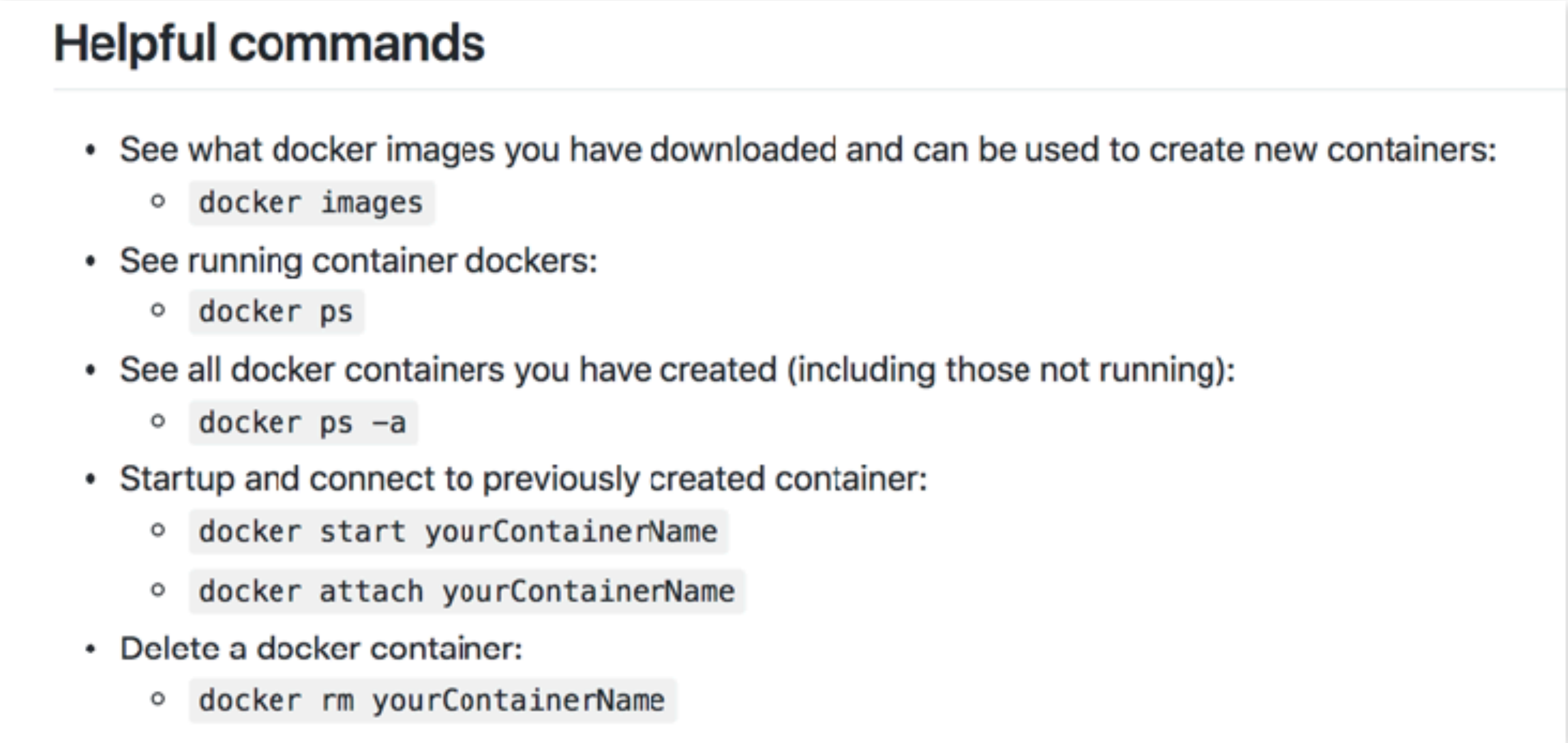
attach your terminal to it

```
$ docker start CDL && docker attach CDL
```

start it

Other useful commands:

- Check out README for docker tutorial on GitHub



Helpful commands

- See what docker images you have downloaded and can be used to create new containers:
  - `docker images`
- See running container dockers:
  - `docker ps`
- See all docker containers you have created (including those not running):
  - `docker ps -a`
- Startup and connect to previously created container:
  - `docker start yourContainerName`
  - `docker attach yourContainerName`
- Delete a docker container:
  - `docker rm yourContainerName`

Setup for MIND

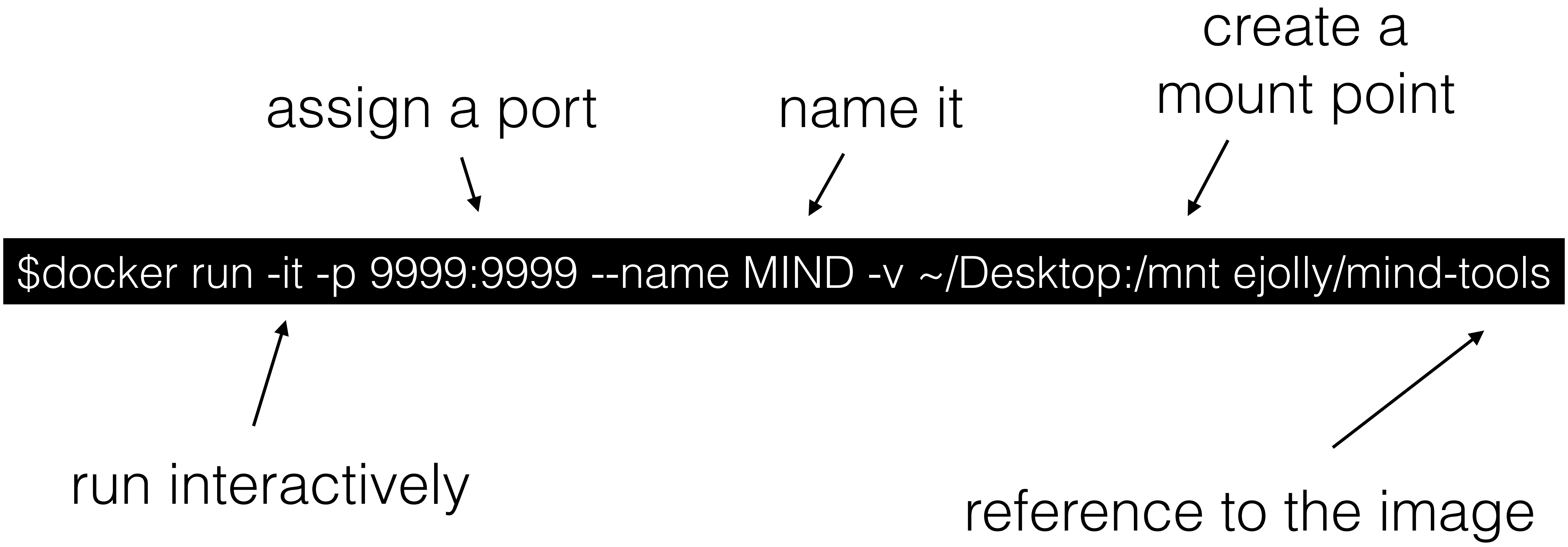
- Launch Docker and adjust the preferences to allocate sufficient resources (e.g. > 4GB RAM)

- Download the Docker image for MIND from docker hub (this may take a while)

```
$ docker pull ejolly/mind-tools
```

Create a container

- Use the downloaded image to create a new container for the workshop



create a mount point

assign a port

name it

```
$docker run -it -p 9999:9999 --name MIND -v ~/Desktop:/mnt ejolly/mind-tools
```

run interactively

reference to the image

Demos:

- Once setup, you can open it again with this:

```
$ docker start MIND && docker attach MIND
```

- When you see `root@` , you’re in the container
- Open a jupyter notebook session by typing: `jp`
- Open a jupyter lab session by typing: `jl`
- Navigate to this is your web browser: localhost:9999

Demos:

- PyCharm Docker integration
  - IDE - integrated development environment
  - Meant for developing programs and/or building software in python
- If you’re interested, check ou
- t the PyCharm tutorial!

Summary

- Docker is a very useful tool to scientists because it allows us to share computing environments.

- Sharable computing env = easily replicable experiments, analysis pipelines, figures, etc.

- Please follow the tutorial outlined in the README to learn how to build a simple Docker image.