



docker

# Reasons You Are Using Docker?



**Solomon Hykes**



**Sebastien Pahl**

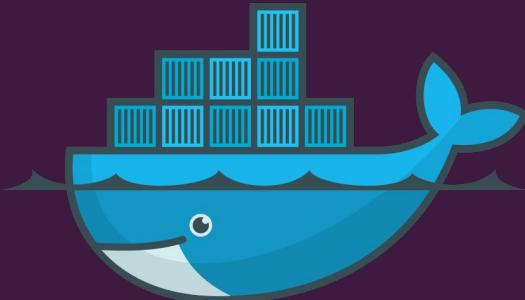


**Kamel Founadi** (Not with  
Docker Inc. Now)

Docker, Inc. is an American technology company.

**Solomon, Sebastien, Kamel, and the whole Docker team till today.....**

# What is Docker?



A standardized unit of Software i.e. A package of code and dependencies to run that particular code. (Node JS Code and Node JS Runtime)

Docker is a **container** technology: A tool for creating and managing containers.

Containers in Software Development?

The container always yields the exact same application and execution behavior no matter where you execute that container.....

# In Layman Terms? Please



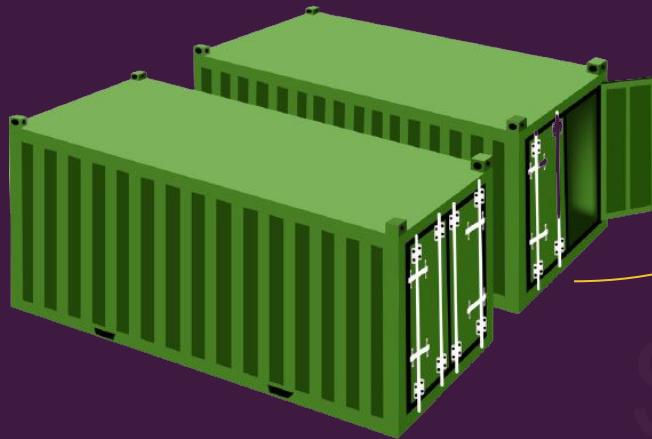
A container is similar to a picnic basket.  
A picnic basket contains all the things  
needed to go on a picnic and use it  
anywhere you want anytime!!

- Contains plates to eat.
- Eating materials like fruits, foods, etc.
- Just take the picnic basket to your picnic destination. **Open it. Serve it. Eat it.**

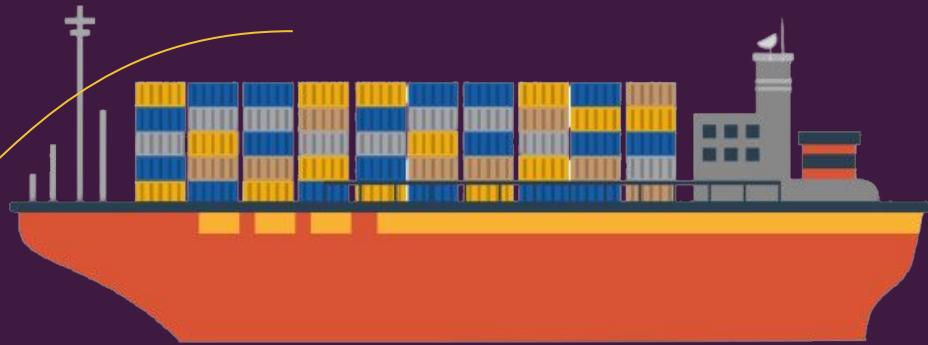
You can share the picnic basket to your friends/relatives as well and they can use it the same way you do/did on your picnic. Same as picnic basket, you give container to a fellow developer, and he/she will just use it anywhere he/she wants anytime!!



# One More?



Containers



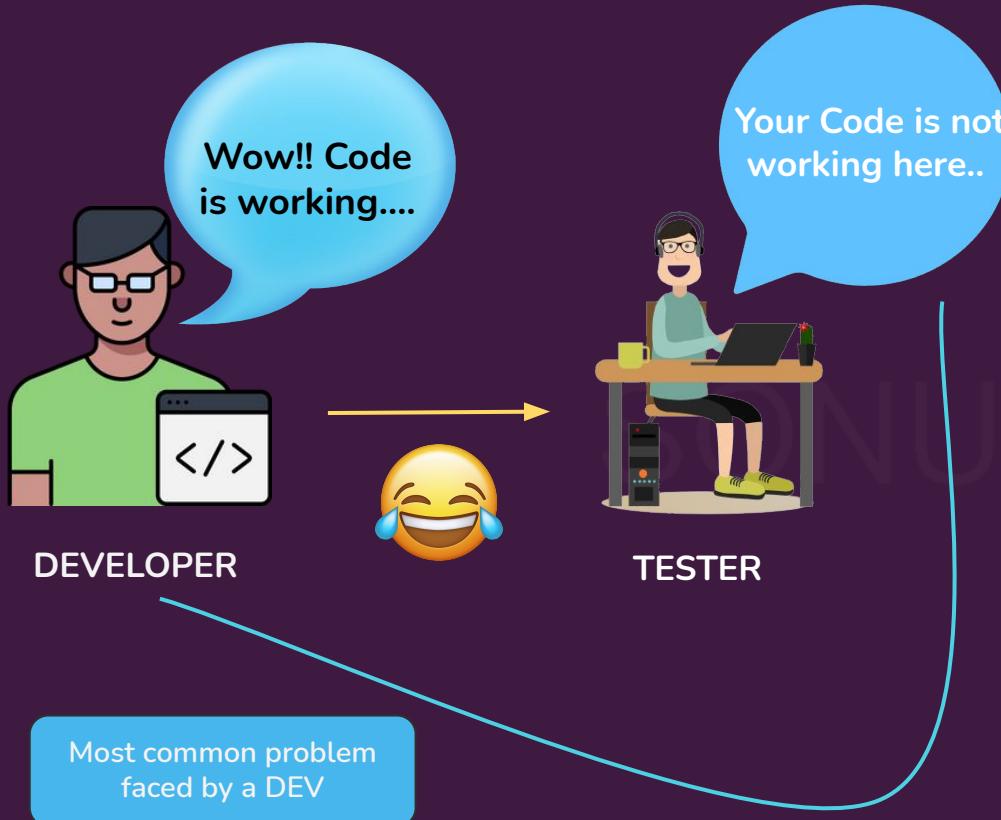
Containers on the Ship

If you have ever seen/visited the Docks then you must be familiar with the above scenes. The concepts of Docker are quite similar to the above scenarios.

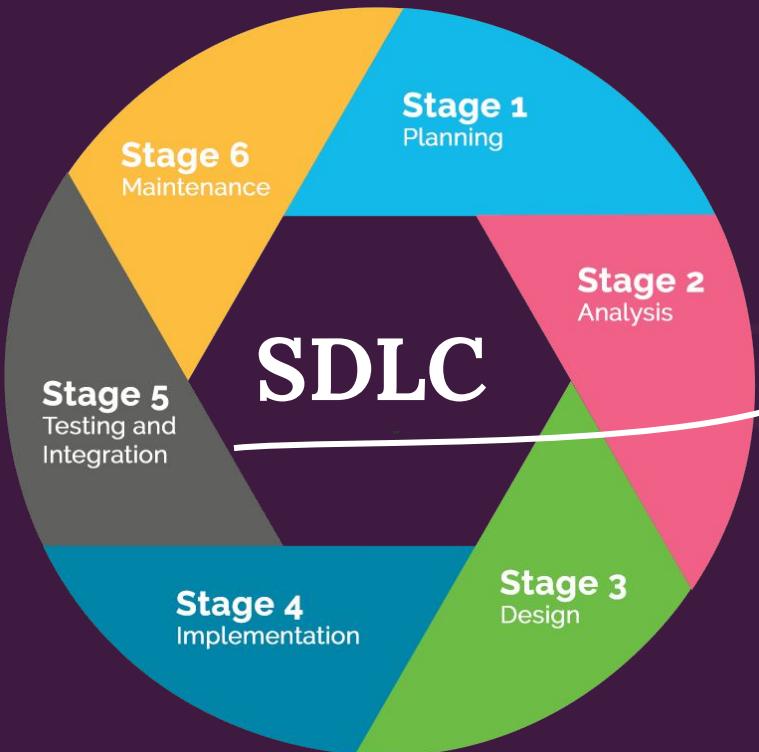


Solomon might have visited Docks a lot 😂

# Why Docker? Ever faced this?



# Docker on Software Level



The most common use of Docker is in the “**Integration/Deployment/Production**” of the SDLC but **not limited to**.....



BUILD

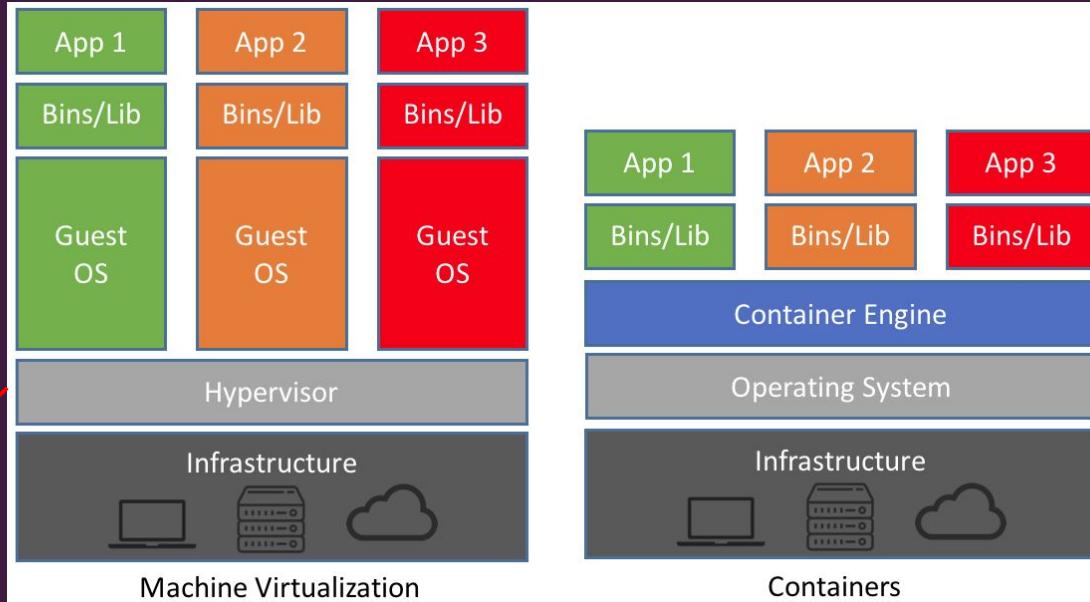


SHIP



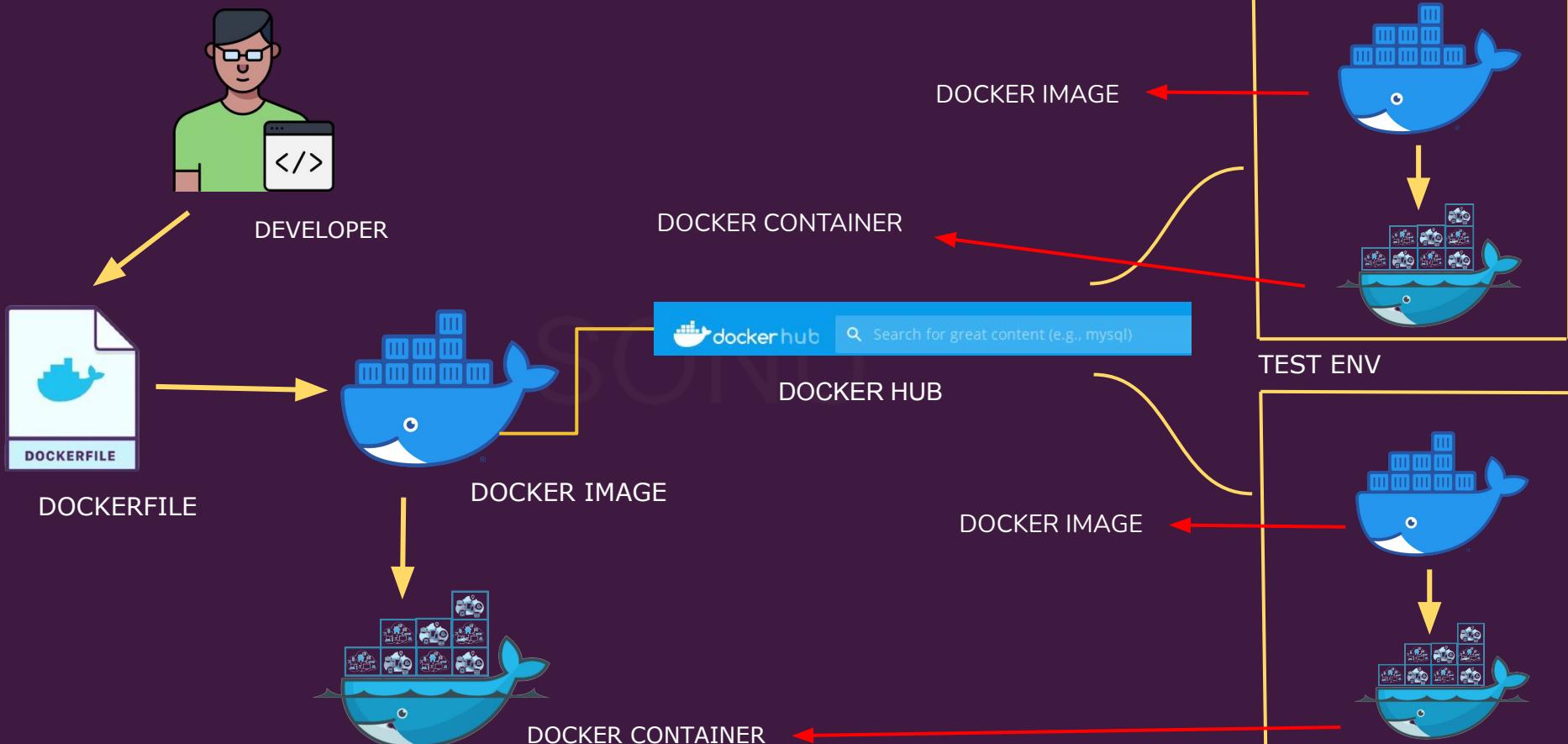
RUN

# Docker on Hardware Level



Virtual machines basically **virtualizes the hardware** where as Docker **virtualizes the Operating System**.

# How Docker Works?



# Some Hands-On... Enough of Theory!!

Let's create a Docker File, Image, and then build & run that Image.

**Install Docker Engine**

Estimated reading time: 5 minutes

**Supported platforms**

Docker Engine is available on a variety of [Linux platforms](#), [macOS](#) and [Windows 10](#) through Docker Desktop, and as a [static binary installation](#). Find your preferred operating system below.

**Desktop**

Platform	x86_64 / amd64
Docker Desktop for Mac (macOS)	•
Docker Desktop for Windows	•

**Server**

Docker provides [.deb](#) and [.rpm](#) packages from the following Linux distributions and architectures:

Platform	x86_64 / amd64	ARM	ARM64 / AARCH64
CentOS	•		•
Debian	•	•	•
Fedor	•		•
Raspbian		•	•
Ubuntu	•	•	•

**Other Linux distributions**

**Note**

While the instructions below may work, Docker does not test or verify installation on derivatives.

You will need:

1. Docker installed on your system.
2. A text editor (Notepad, VIM, etc) /IDE (Sublime, VS Code, Atom, etc)
3. Time

**Docker Documentation (Must) : <https://docs.docker.com/>**

You might mess with the Docker installation in both Windows and Linux. In Windows, You need to deal with Hypervisor in BIOS and in Linux, You need to create groups. **Read documentation closely.**

# Building the first Docker Image

```
sonu@sonu-GF63-Thin-9SC:~$ docker --version  
Docker version 20.10.2, build 2291f61  
sonu@sonu-GF63-Thin-9SC:~$
```

To check the Version of Docker

```
sonu@sonu-GF63-Thin-9SC:~$ sudo docker images  
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE  
picoziali       latest   00a89f3a5cf3  23 hours ago  975MB  
picozi/picoziali latest   00a89f3a5cf3  23 hours ago  975MB  
ubuntu          latest   f63181f19b2f  4 weeks ago   72.9MB  
python          3.8.5-slim 6cf621cb1327  5 months ago  113MB  
sonu@sonu-GF63-Thin-9SC:~$
```

To see if we have any Docker image in the system (You can see that I have few images already)

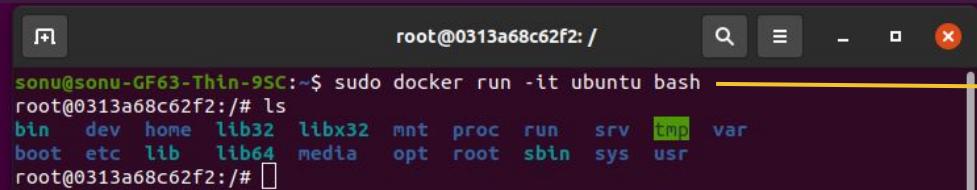
Let's pull an Image from Docker Hub  
(Ubuntu in this case)

[https://hub.docker.com/\\_/ubuntu](https://hub.docker.com/_/ubuntu)

The screenshot shows the Docker Hub interface. A search bar at the top contains the text "ubuntu". Below it, a card for the "ubuntu" image is displayed, labeled "Docker Official Images". It features a red icon with a white Ubuntu logo, the text "ubuntu ☆", and "Docker Official Images". A brief description states: "Ubuntu is a Debian-based Linux operating system based on free software." Below the card, there are tabs for "Container", "Linux", "IBM Z", "PowerPC 64 LE", "x86-64", "ARM", "386", "ARM 64", "Base Images", and "Operating Systems". A "1B+" badge is visible. At the bottom, there are links for "Description", "Reviews", and "Tags".

The screenshot shows a terminal window with a dark background. It displays the command "docker pull ubuntu" in white text. Above the command, there is a placeholder text "Copy and paste to pull this Image". Below the command, there is a link "View Available Tags".

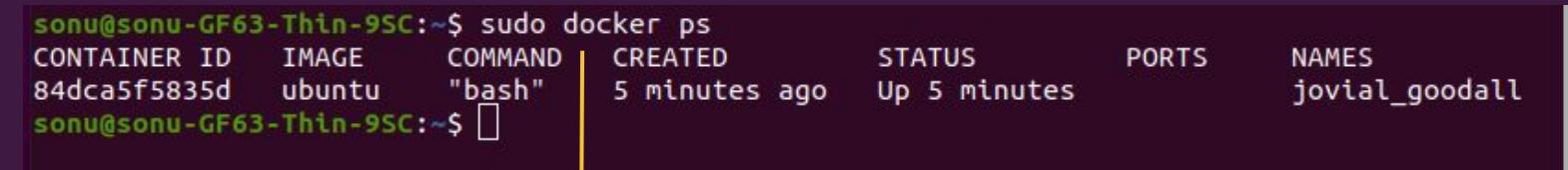
# Building the first Docker Image



```
sonu@sonu-GF63-Thin-9SC:~$ sudo docker run -it ubuntu bash
```

The terminal shows the command `sudo docker run -it ubuntu bash` being run. The output shows the root prompt in a new container with directory `/`, followed by a listing of directory contents (`bin dev home lib32 libx32 mnt proc run srv tmp var boot etc lib lib64 media opt root sbin sys usr`). A yellow arrow points from the command in the first terminal to the explanatory text on the right.

To run the ubuntu image that we pulled from Docker Hub. We are using `-it` to use it in an interactive way.



```
sonu@sonu-GF63-Thin-9SC:~$ sudo docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
84dca5f5835d	ubuntu	"bash"	5 minutes ago	Up 5 minutes		jovial_goodall

```
sonu@sonu-GF63-Thin-9SC:~$
```

A yellow arrow points from the command in the second terminal to the explanatory text below.

Open a new toolbox/terminal and run the command to check the images which are in running mode.

# Building the first Docker Image

Creating the Dockerfile



```
sonu@sonu-GF63-Thin-9SC:~$ cd Project
sonu@sonu-GF63-Thin-9SC:~/Project$ mkdir dockerfolder
sonu@sonu-GF63-Thin-9SC:~/Project$ cd dockerfolder
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ touch Dockerfile
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ ls
Dockerfile
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ 
```

In Dockerfile, We enlist all the steps/instructions to build an Image. We run the docker build command to build an image using the Dockerfile.

```
FROM ubuntu
RUN apt-get update
CMD ["echo", "Our Docker Container is up and running...."]
```

```
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ vim Dockerfile
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ cat Dockerfile
FROM ubuntu
RUN apt-get update
CMD ["echo", "Our Docker Container is up and running...."]

sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ 
```

We are giving instructions to update Ubuntu Packages in our Dockerfile.

# Run Docker Image

```
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ ls
Dockerfile
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ sudo docker build -t firstdockering:1.0.0 .
Sending build context to Docker daemon 2.048kB
Step 1/3 : FROM ubuntu
--> f63181f19b2f
Step 2/3 : RUN apt-get update
--> Running in 99e3999488b3
Get:1 http://archive.ubuntu.com/ubuntu focal InRelease [265 kB]
Get:2 http://security.ubuntu.com/ubuntu focal-security InRelease [109 kB]
Get:3 http://security.ubuntu.com/ubuntu focal-security/universe amd64 Packages [670 kB]
Get:4 http://archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:5 http://archive.ubuntu.com/ubuntu focal-backports InRelease [101 kB]
Get:6 http://archive.ubuntu.com/ubuntu focal/multiverse amd64 Packages [177 kB]
Get:7 http://archive.ubuntu.com/ubuntu focal/main amd64 Packages [1275 kB]
Get:8 http://security.ubuntu.com/ubuntu focal-security/restricted amd64 Packages [177 kB]
Get:9 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 Packages [21.6 kB]
Get:10 http://archive.ubuntu.com/ubuntu focal/restricted amd64 Packages [33.4 kB]
Get:11 http://archive.ubuntu.com/ubuntu focal/universe amd64 Packages [11.3 MB]
Get:12 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [650 kB]
Get:13 http://archive.ubuntu.com/ubuntu focal-updates/restricted amd64 Packages [209 kB]
Get:14 http://archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 Packages [29.6 kB]
Get:15 http://archive.ubuntu.com/ubuntu focal-updates/universe amd64 Packages [933 kB]
Get:16 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [1059 kB]
Get:17 http://archive.ubuntu.com/ubuntu focal-backports/universe amd64 Packages [4301 B]
Fetched 17.2 MB in 1min 34s (183 kB/s)
Reading package lists...
Removing intermediate container 99e3999488b3
--> 3866b8c12202
Step 3/3 : CMD ["echo", "Our Docker Container is up and running...."]
--> Running in 73a8c4687635
Removing intermediate container 73a8c4687635
--> a48b943f00fe
Successfully built a48b943f00fe
Successfully tagged firstdockering:1.0.0
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$
```

1. docker build command to build our image.
2. firstdockering is the name of our image and 1.0.0 is the tag.

You can see that Ubuntu packages got updated and we successfully build our first image.

To check our Docker Images

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
firstdockering	1.0.0	a48b943f00fe	5 minutes ago	100MB
picozai	latest	00a89f3a5cf3	24 hours ago	975MB
picozai/picozai	latest	00a89f3a5cf3	24 hours ago	975MB
ubuntu	latest	f63181f19b2f	4 weeks ago	72.9MB
python	3.8.5-slim	6cf621cb1327	5 months ago	113MB

Voila!! We build our first docker Image.

# Run Docker Image and Push on Docker Hub

```
sonu@sonu-GF63-Thin-9SC:~$ sudo docker run a48b943f00fe
Our Docker Container is up and running....
sonu@sonu-GF63-Thin-9SC:~$ █
```

→ Pass the ID of image with docker run command to run the Docker container

Go to Docker Hub account and create a repository

Docker Hub

Explore Repositories Organizations Get Help picozi

Repositories Create

Using 0 of 1 private repositories. [Get more](#)

Create Repository

picozi updatepkgs

Pro tip

You can push a new image to this repository using the CLI

```
docker tag local-image:tagname new-repo:tagname
docker push new-repo:tagname
```

Make sure to change tagname with your desired image repository tag.

Visibility

Using 0 of 1 private repositories. [Get more](#)

Public Public repositories appear in Docker Hub search results

Private Only you can view private repositories

Build Settings (optional)

Autobuild triggers a new build with every git push to your source code repository. [Learn More](#).

Please re-link a GitHub or Bitbucket account

We've updated how Docker Hub connects to GitHub and Bitbucket. You'll need to re-link a GitHub or Bitbucket account to create new automated builds. [Learn More](#)

Disconnected Disconnected

[Cancel](#) [Create](#) [Create & Build](#)

```
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ sudo docker login
Authenticating with existing credentials...
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ sudo docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
firstdockingimg 1.0.0 a48b943f00fe 20 minutes ago 100MB
picozi/picozil latest 00a89f3a5cf3 24 hours ago 975MB
picozial latest 00a89f3a5cf3 24 hours ago 975MB
ubuntu latest f63181f19b2f 4 weeks ago 72.9MB
python 3.8.5-slim 6cf621cb1327 5 months ago 113MB
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ sudo docker tag firstdockingimg picozi/updatepkgs:1.0
Error response from daemon: No such image: firstdockingimg:latest
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ sudo docker tag firstdockingimg:1.0.0 picozi/updatepkgs:1.0
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ sudo docker push picozi/updatepkgs:1.0
The push refers to repository [docker.io/picozi/updatepkgs]
d948ea639603: Pushed
02473af360b: Mounted from library/ubuntu
dbf2c0f42a39: Mounted from library/ubuntu
9f32931c9d28: Mounted from library/ubuntu
1.0: digest: sha256:6b08aa39fabfff81ad1113fc46222a81dcc20bc44917faeafe8034600f1f2d18 size: 1155
sonu@sonu-GF63-Thin-9SC:~/Project/dockerfolder$ █
```

Here you go, You pushed your Docker Image to the Docker Hub!!

# Share the link to anyone who wants to use!!

Docker hub Search for great content (e.g., mysql)

Explore Repositories Organizations Get Help picozi Fingerprint

Repositories picozi / updatepkgs Using 0 of 1 private repositories. Get more

General Tags Collaborators Webhooks Settings

picozi / updatepkgs An image to run the Ubuntu's update command Last pushed: 4 minutes ago

Docker commands To push a new tag to this repository. docker push picozi/updatepkgs:tagname Public View

Tags and Scans VULNERABILITY SCANNING - DISABLED Enable This repository contains 1 tag(s). TAG OS PULLED PUSHED 1.0 4 minutes ago 4 minutes ago See all

Recent builds Link a source provider and run a build to see build results here.

Readme Repository description is empty. Click here to edit.



The End.....Bye!!  
Connect: <https://www.linkedin.com/in/sonucr7/>