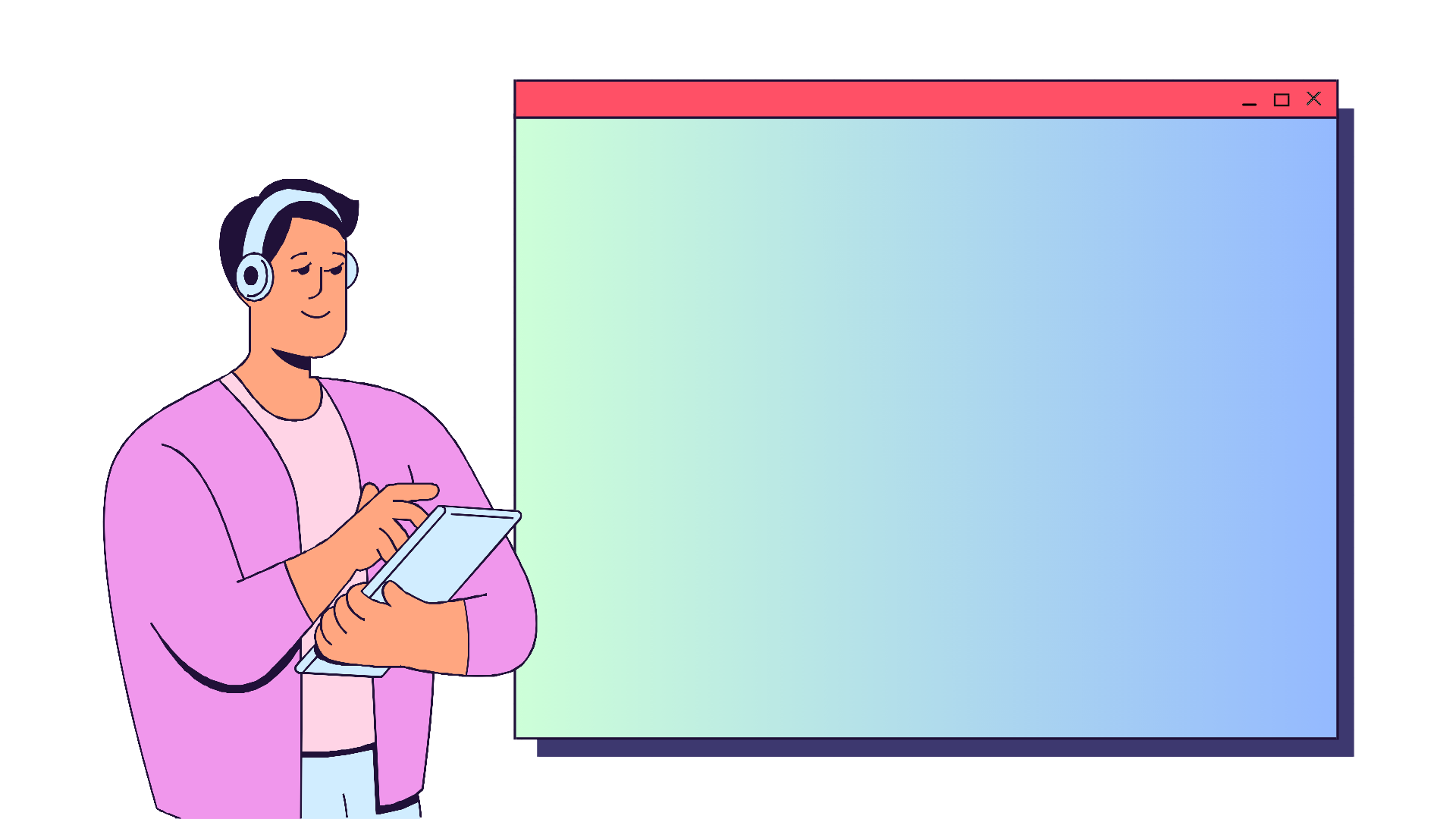
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Docker Installation on Mac, Windows & Linux Creating Demo Project on Node and Python Creating DockerFile

Creating Docker Image Running Containers Pre-defined Images DockerHub

Docker Volumes and Network Docker Compose



Overview

**What is Docker and Why?**

**What are Containers?**

**How Container Works?**

What is a Docker?



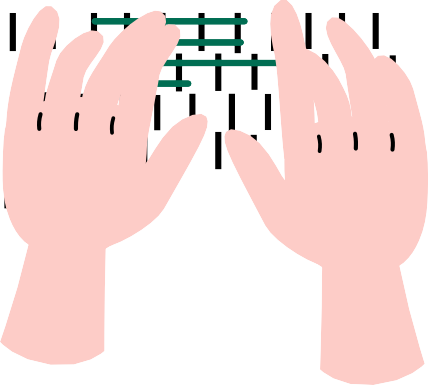
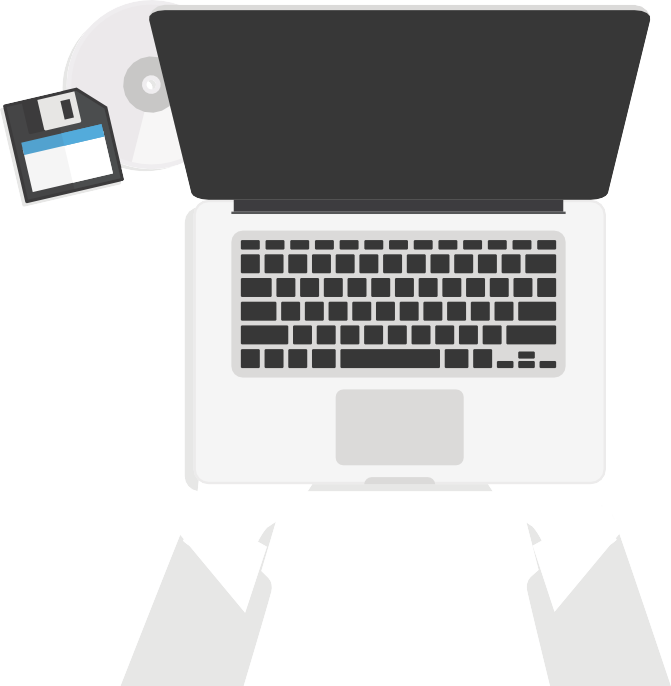
What is a Docker?

 Docker is a containerization platform for developing, packaging, shipping, and running applications.

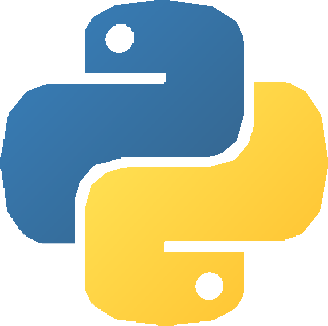
 It provides the ability to run an application in an isolated environment called a container.

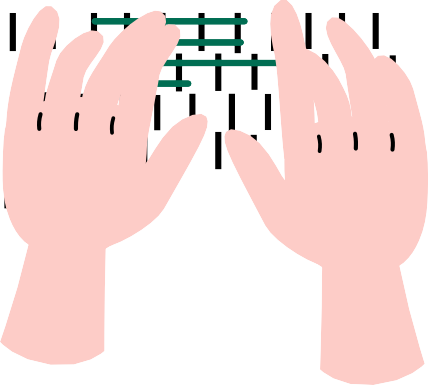
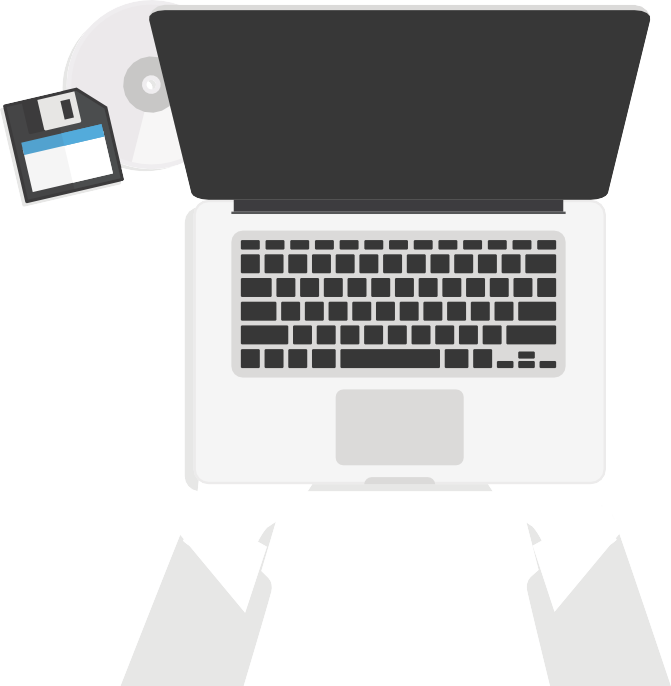
 Makes deployment and development efficient.

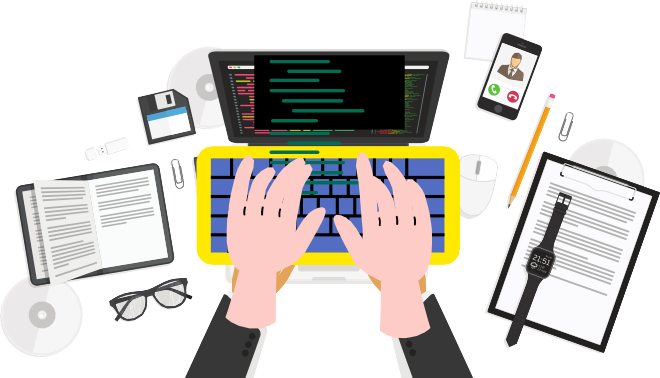
Why do we need Docker?

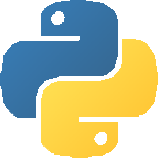


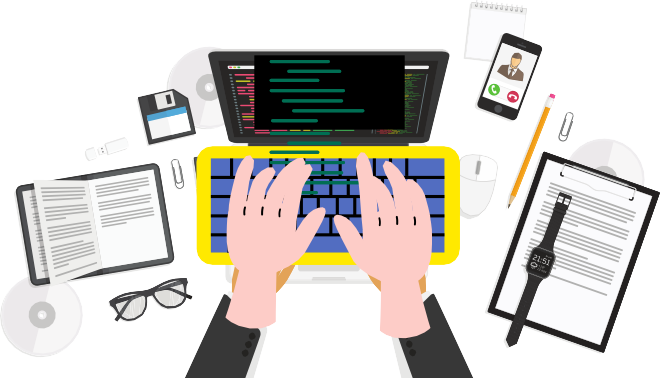
Developer

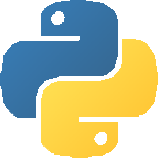


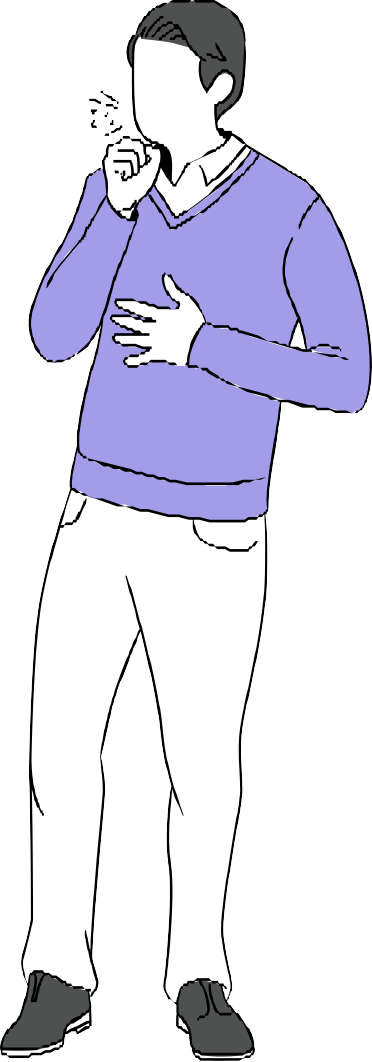




Working Successfully



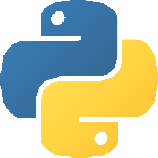
Working Successfully

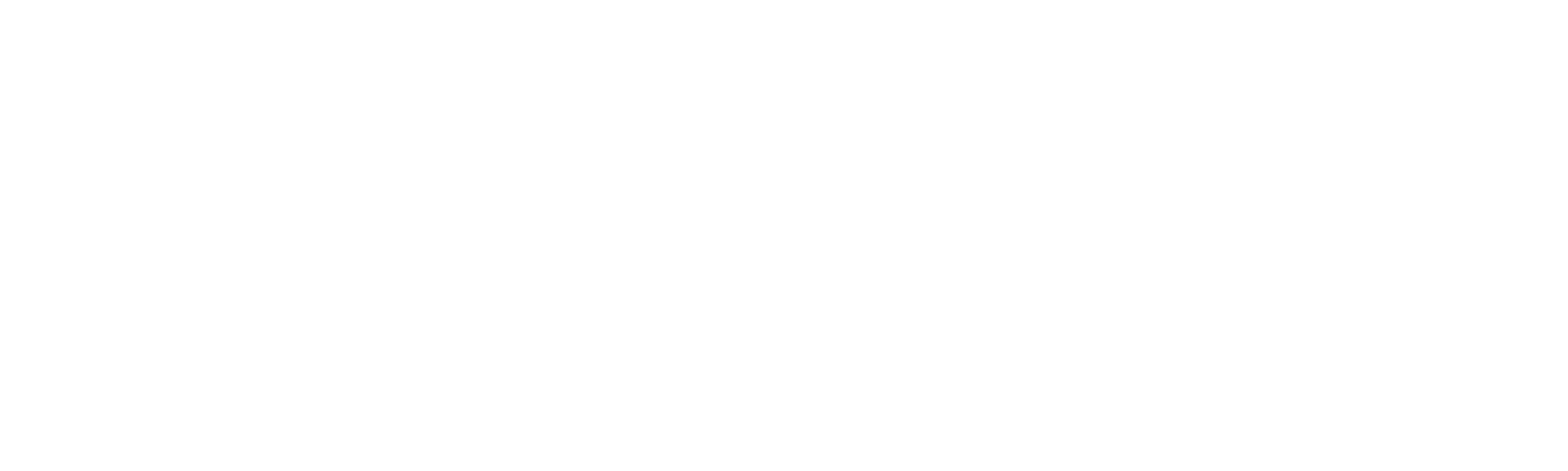


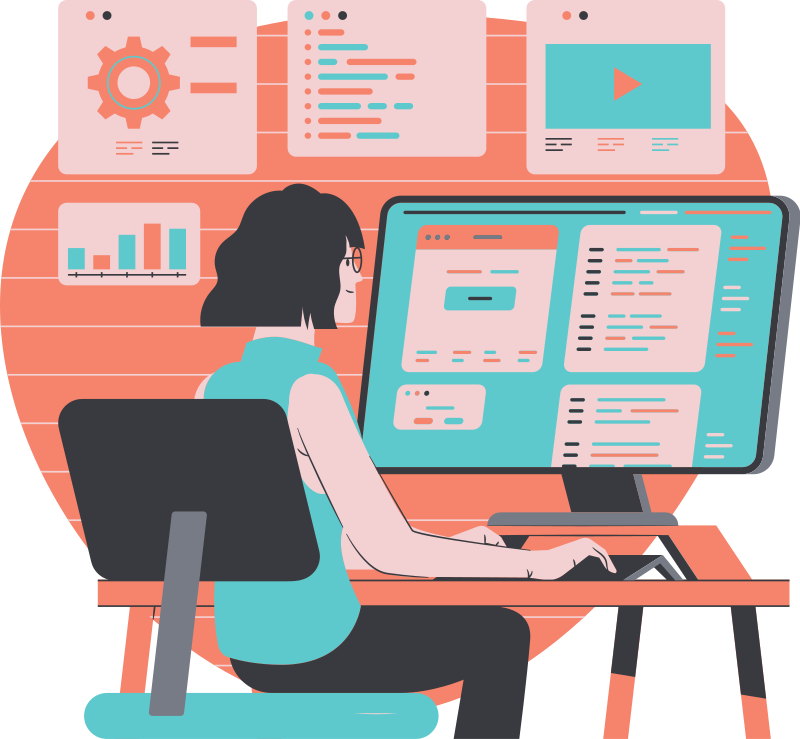
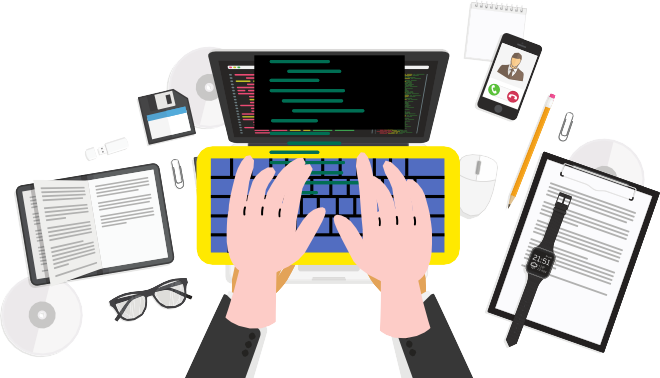
Let me test

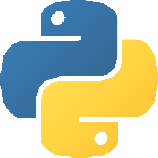
on my machine

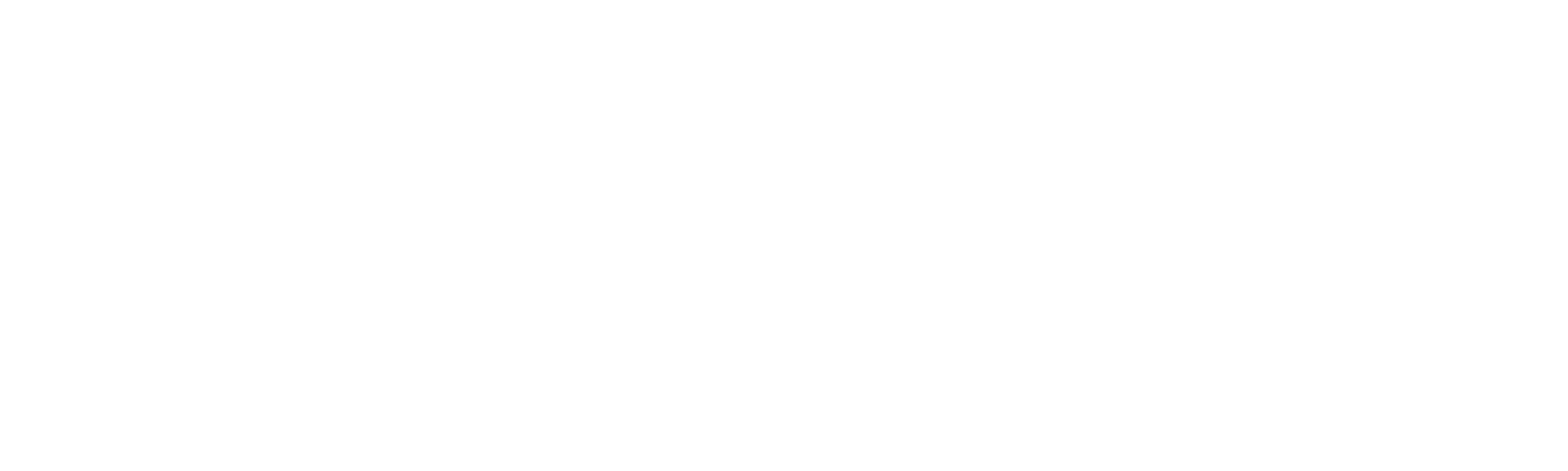
Tester

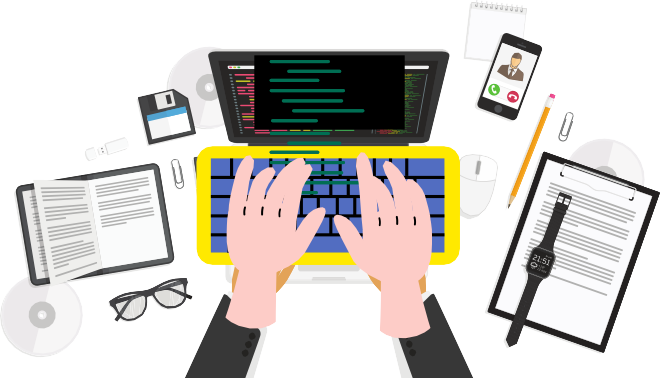


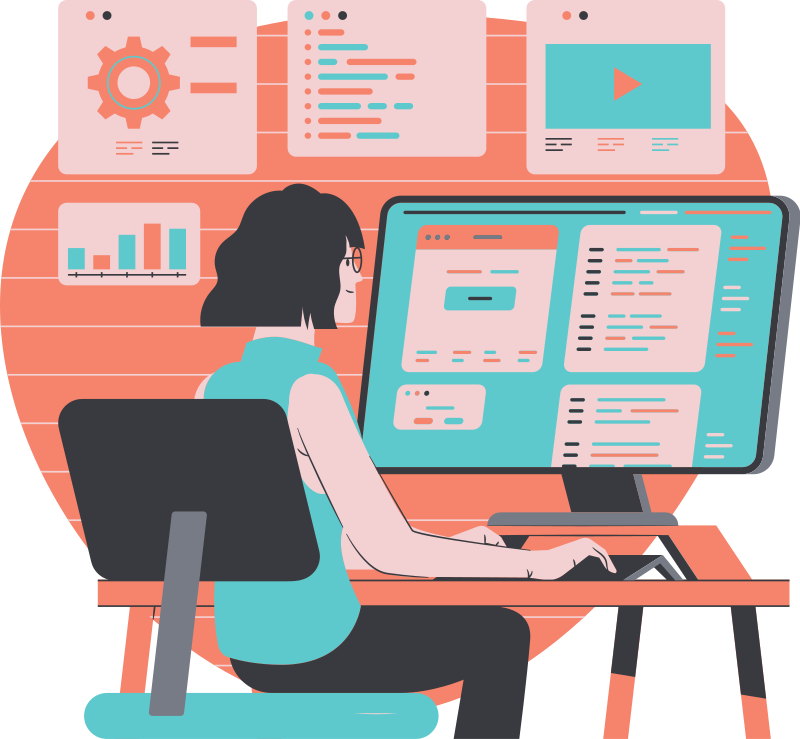


Developer Tester

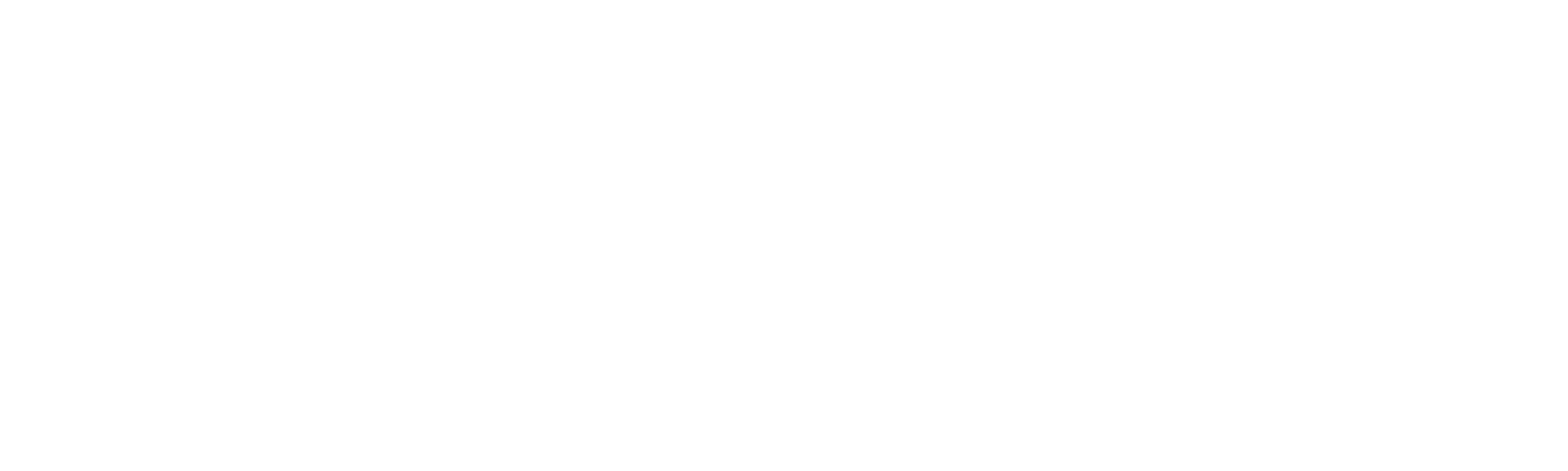


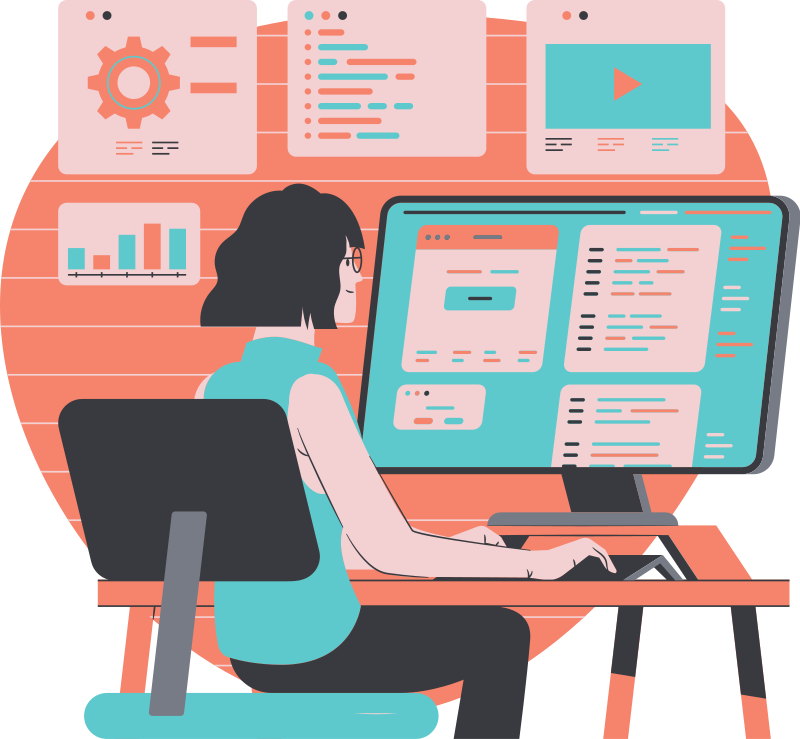


Developer Tester



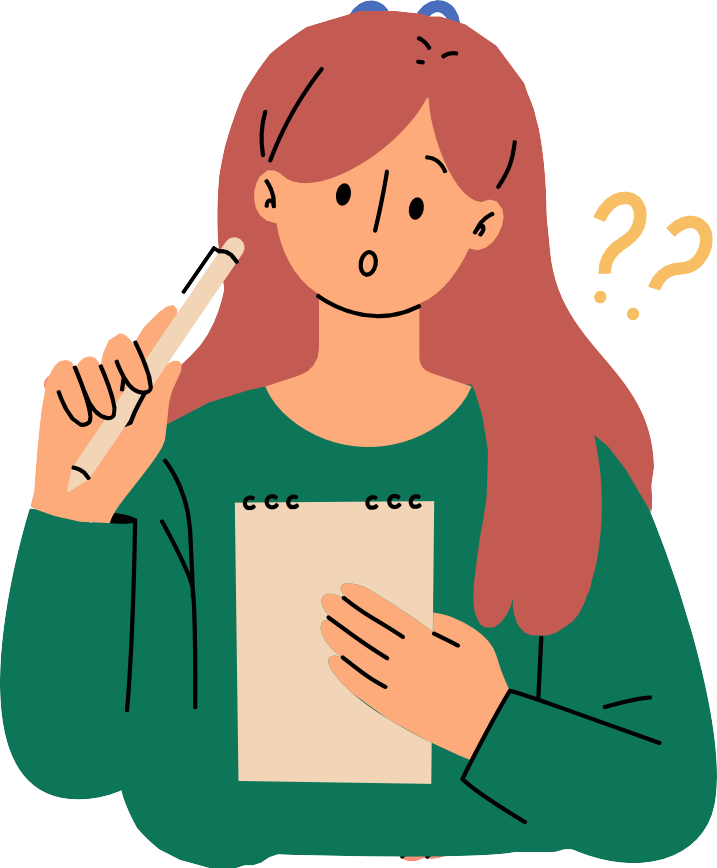


Developer



But it is working on my machine

Tester

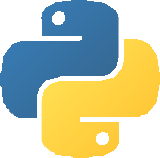
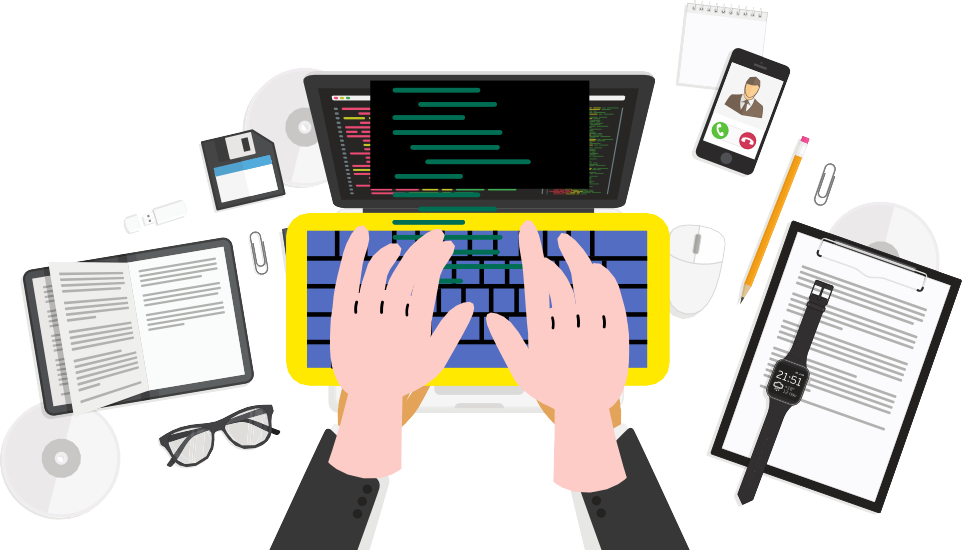


What is the solution now?



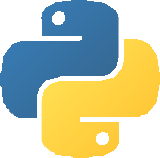
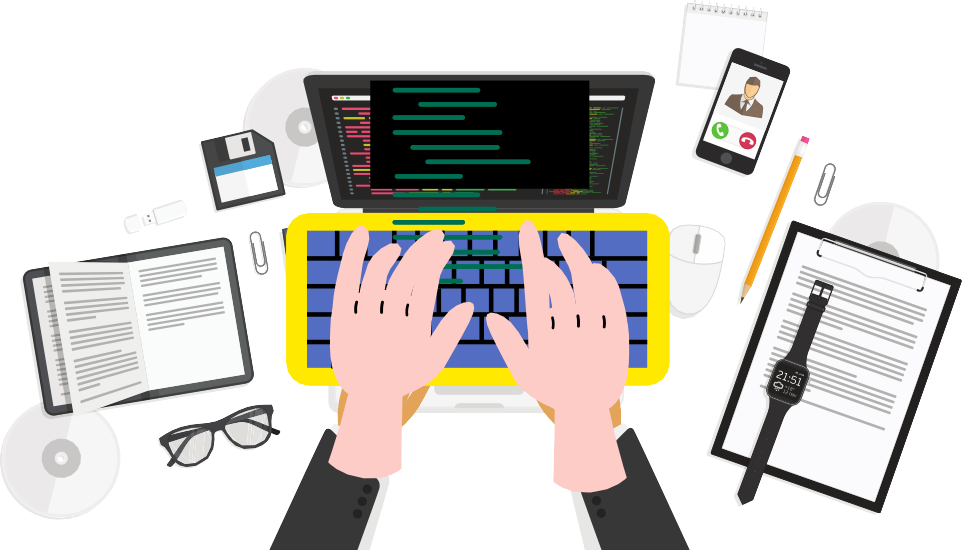
No worries We have

Docker

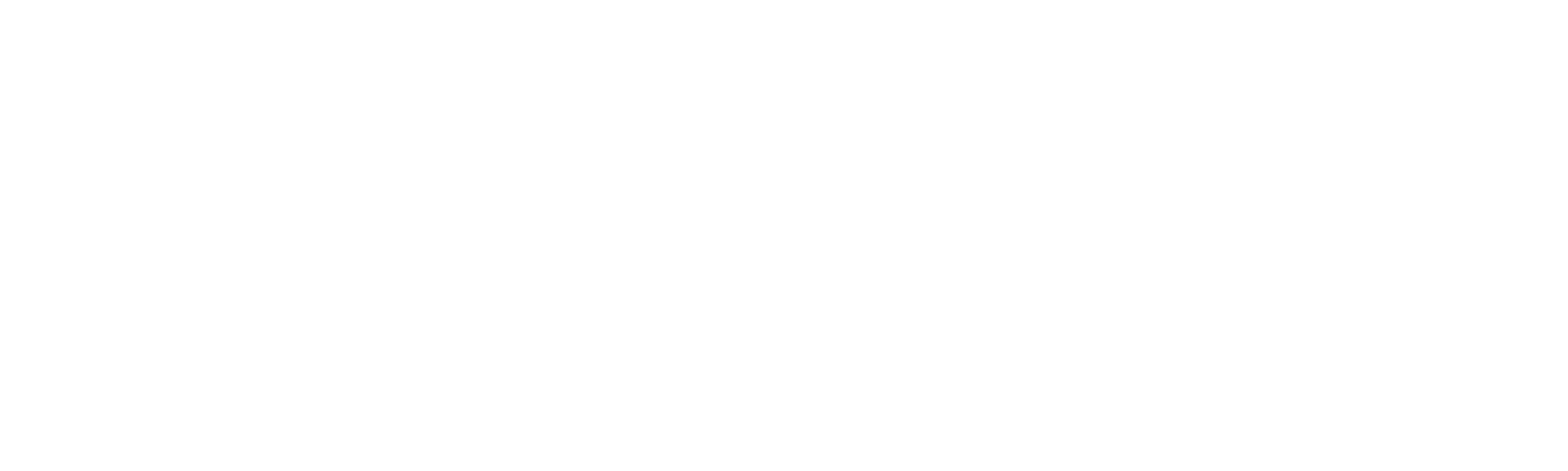


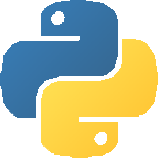
## Developer

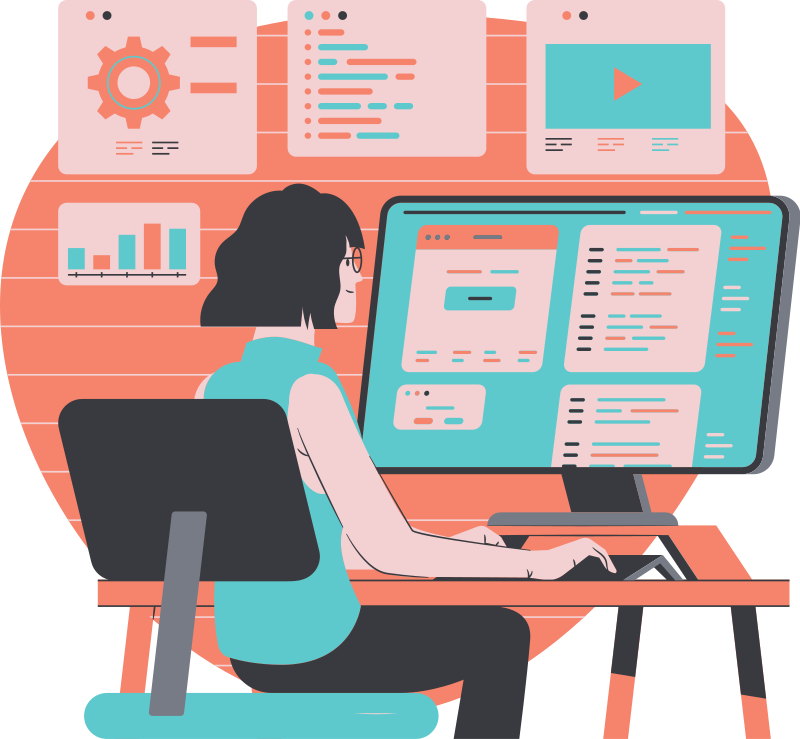
Packaging Docker Image

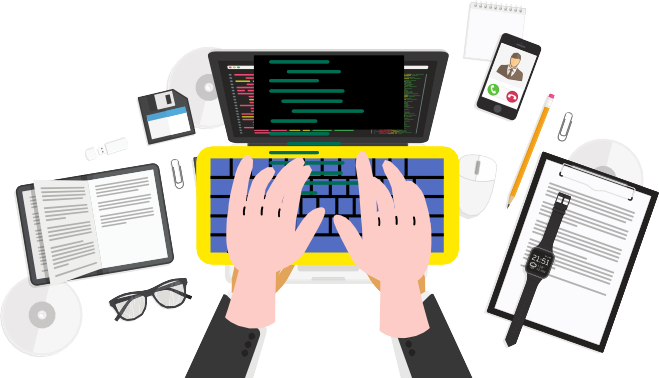


## Developer

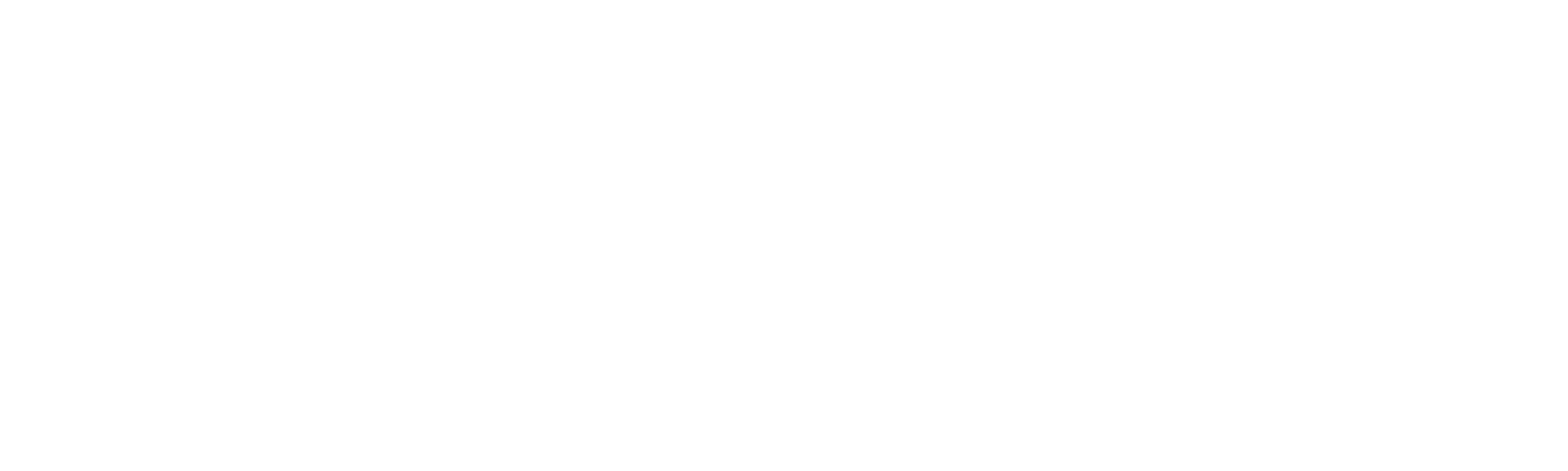


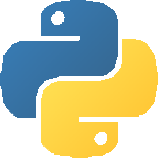


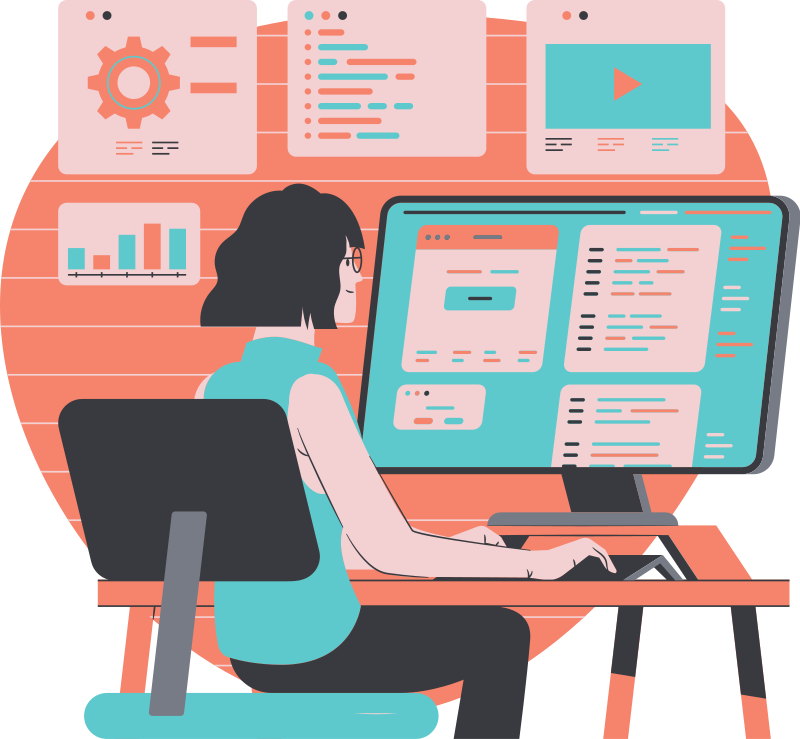


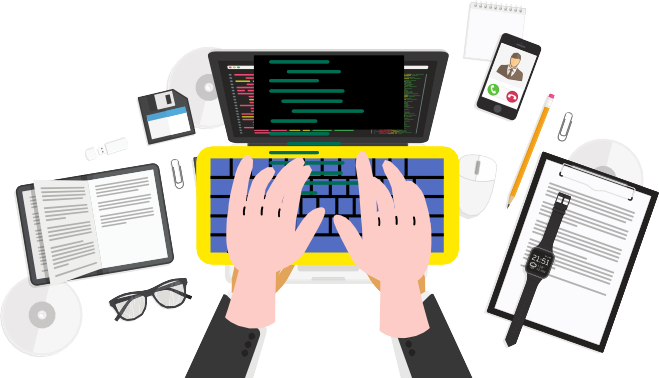
Developer

Tester

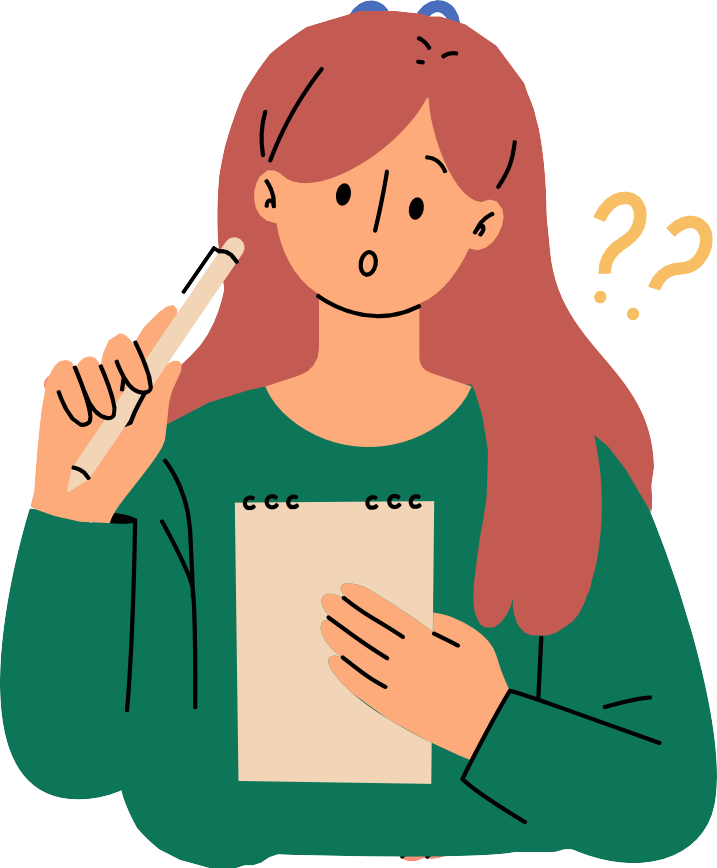






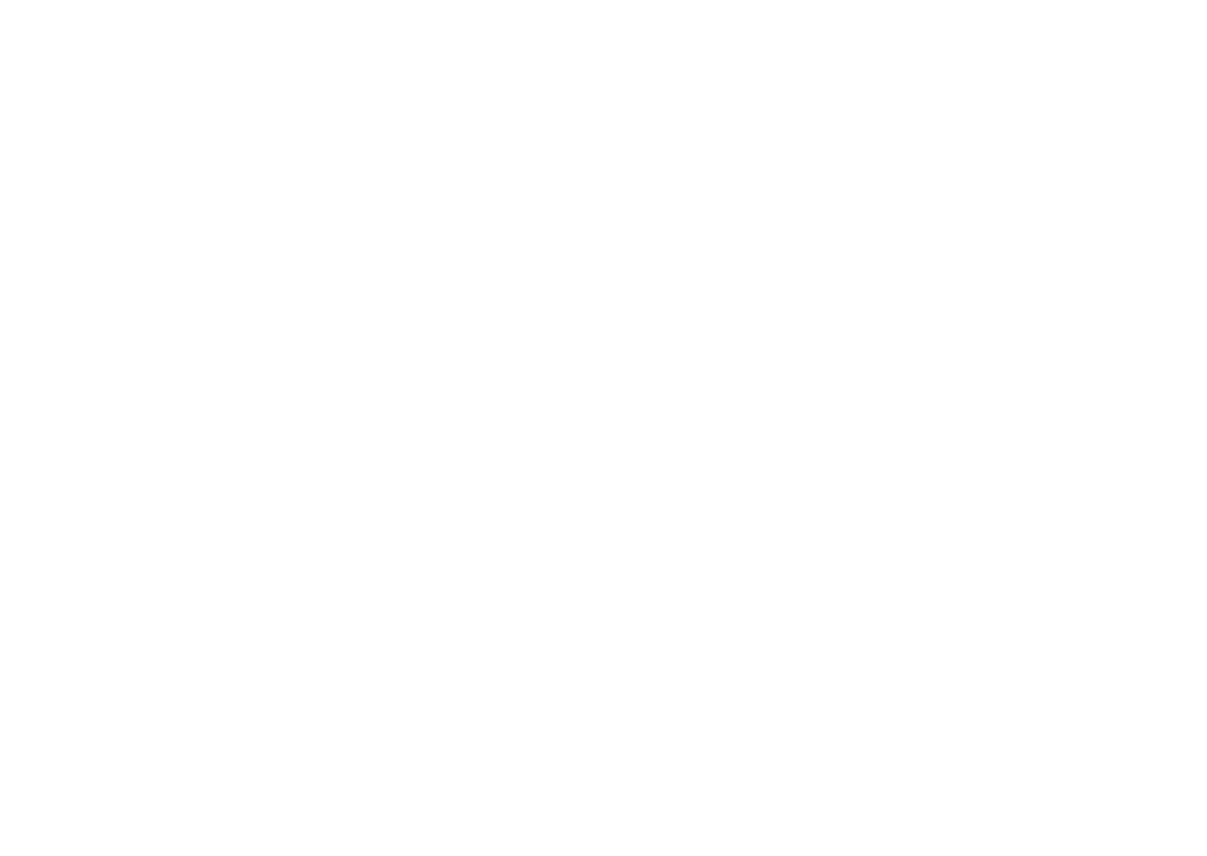
Developer

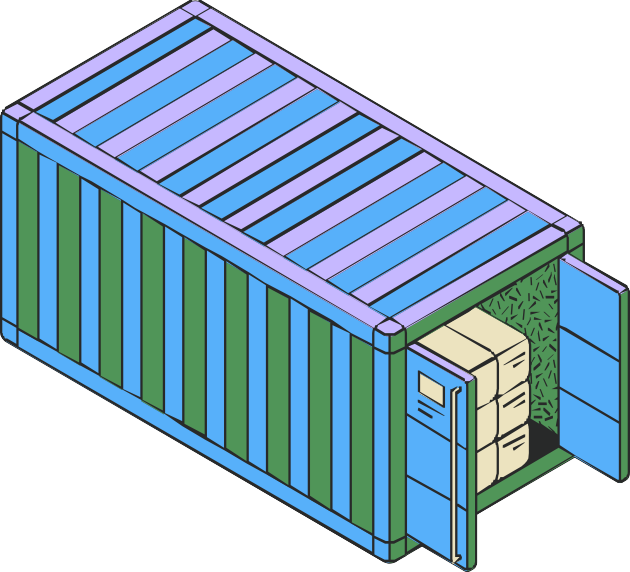
Tester



How did it work?

What is a Container?



What is a Container?

 A way to package an application with all the necessary dependencies and configuration.

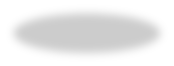
 It can be easily shared

 Makes deployment and development efficient.

Container

Lib, Dependencies, Tools

APP1



Hardware

Operating System

Docker Engine

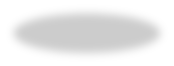
Container Container

Lib, Dependencies, Tools

APP1

Lib, Dependencies, Tools

APP2



Hardware

Operating System

Docker Engine

Container Container Container

Lib, Dependencies, Tools

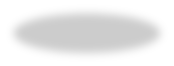
APP1

Lib, Dependencies, Tools

APP2

Lib, Dependencies, Tools

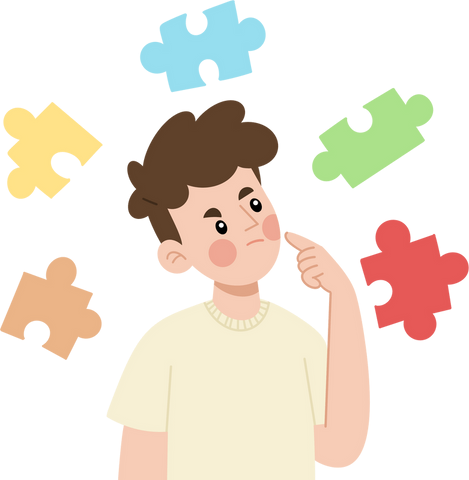
APP3

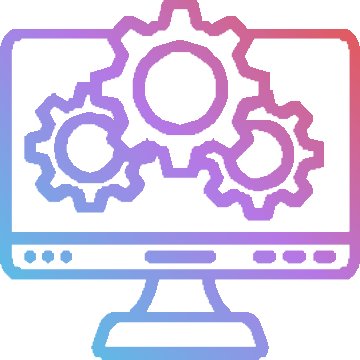
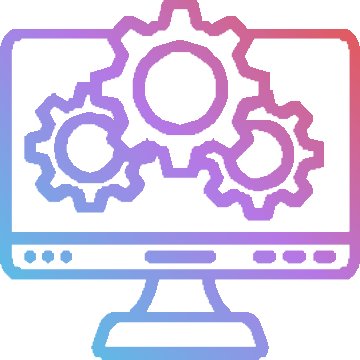


Hardware

Operating System

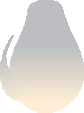
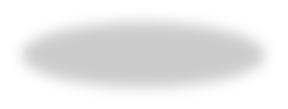
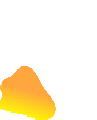
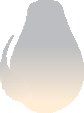
Docker Engine

App1 App2



V14 V16

App1



Virtualization



V14

### HOST OS



App2

V16



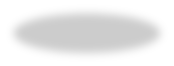
Container Container

V14

APP1

V16

APP2

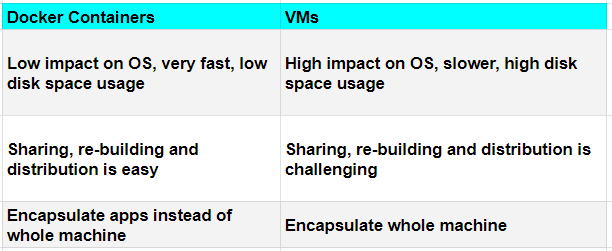


Hardware

Operating System

Docker Engine

Docker vs VMs



Main components of Docker

DockerFile Docker Image Docker Container Docker Registry



Instance of an Image

Container

It is a simple text file with instructions to build an image.

Container

DockerFile

Image

Container

Single File with all the dep and lib to run the

program

Docker Registry

A Docker registry is a central repository for storing and distributing Docker images.

Docker Hub

Or Private Registry

MyAPP v1.0

DockerFile

Image

Docker Hub

Or Private Registry

MyAPP v1.0



MyAPP v1.0

DockerFile

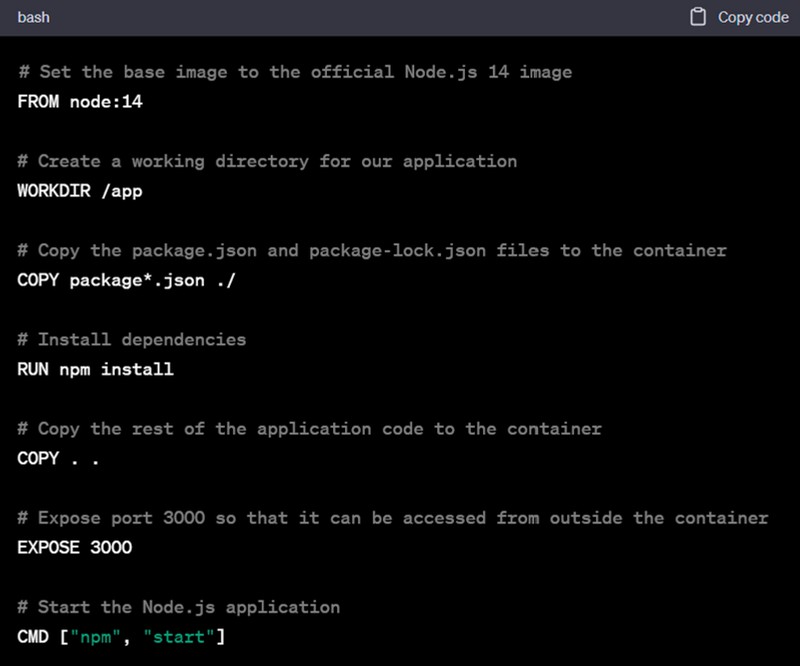
Image

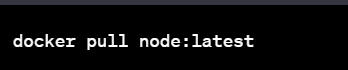
Conclusion:

 Docker is a powerful technology that allows developers to create, package, and deploy applications in containers.

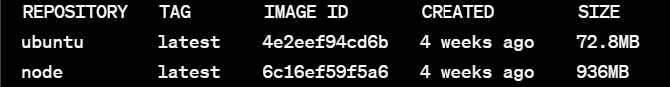
 It provides a consistent environment for development, testing, and deployment, and it's compatible with any platform that supports Docker.

 By using Docker, developers can focus on building great applications instead of worrying about infrastructure and compatibility issues.

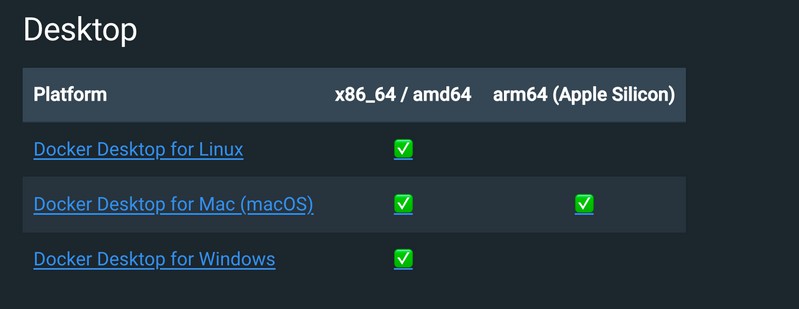








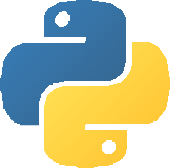
# Install Docker Engine



Docker Volumes

### Container

servers.txt



myapp.py

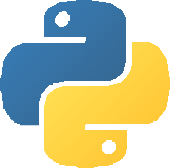
servers.txt



Docker Volumes (bind mount)

### Container

servers.txt

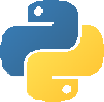
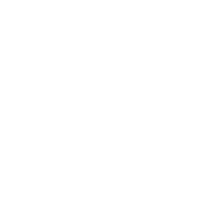
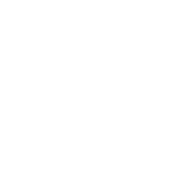
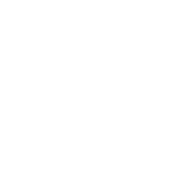
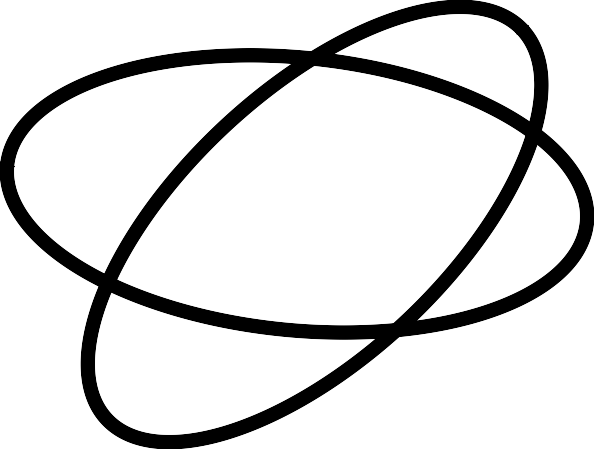
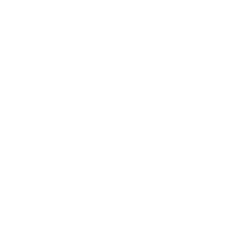
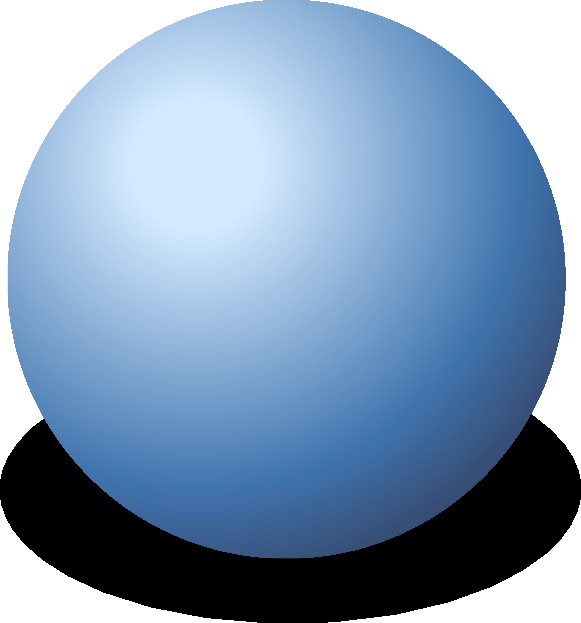


myapp.py

servers.txt

# Communication From/To Containers

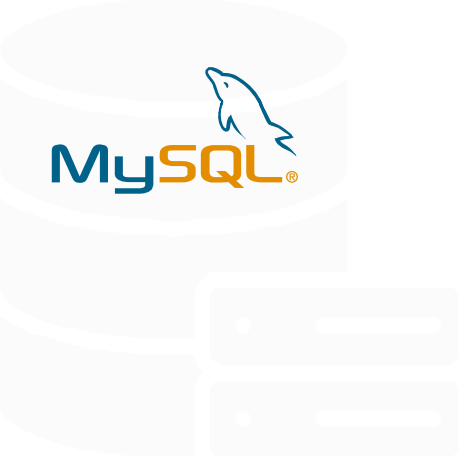
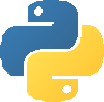
Container



servers.txt

myapp.py

Container

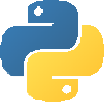


servers.txt

myapp.py

Local Machine

Container Container

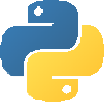


MySQL

myapp.py

# Docker Network

Network: my-net



Container

Container

MySQL

myapp.py

--network my-net

# Docker Compose

Configuration file to manage multiple containers running on same machine..



Please Like & SUBSCRIBE