edureka!

What is Docker?

Do you know the following?

1. Why We Need Docker?

2. What Is Docker?

3. Docker Example

4. Docker Case-Study

5. Docker Components

6. Hands-On

?

Why We Need Docker?
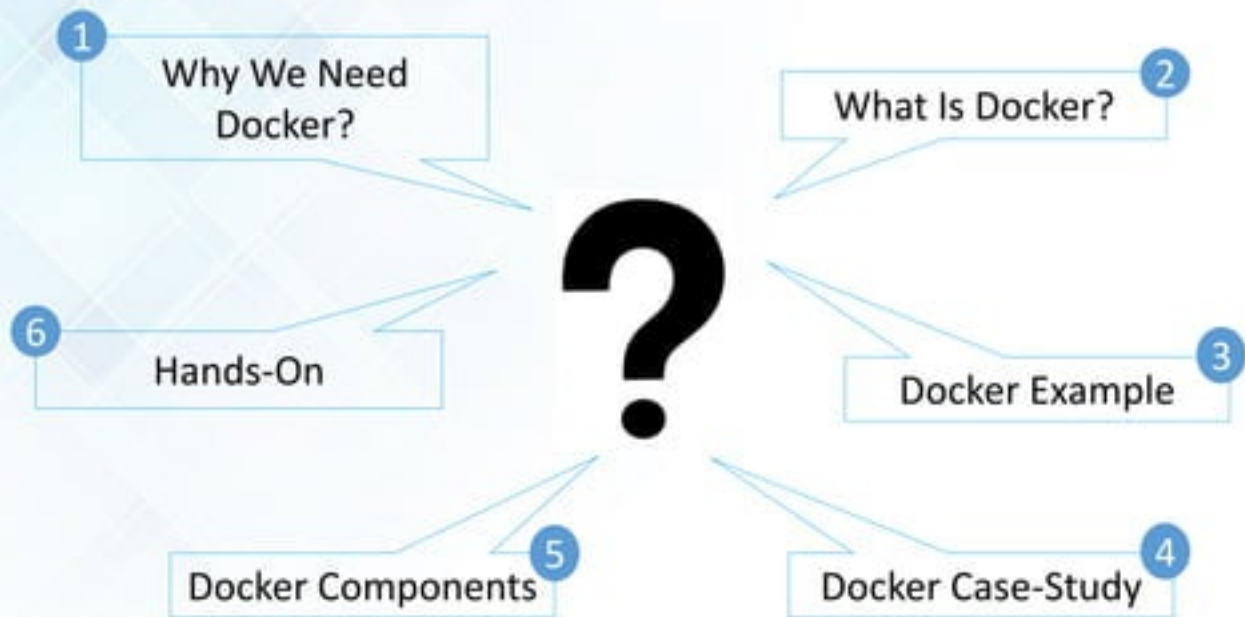
Problems Before Docker

An application works in developer's laptop but not in testing or production. This is due to difference in computing environment between Dev, Test and Prod.
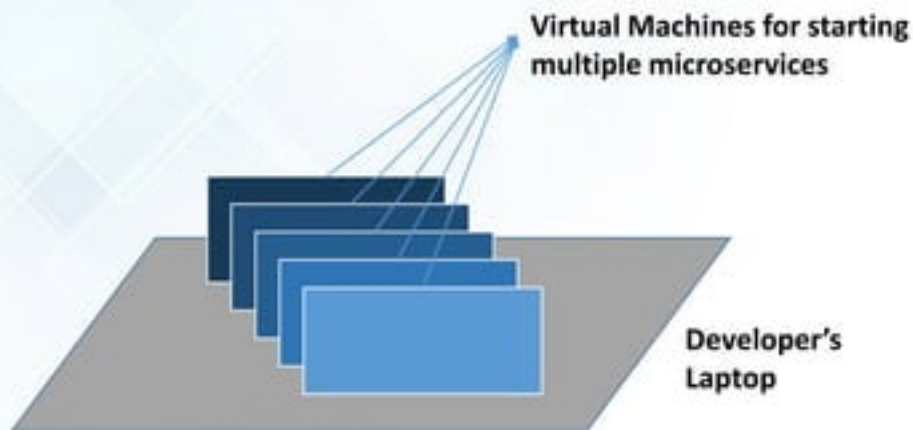
Code works fine in my laptop

There is some problem with the code
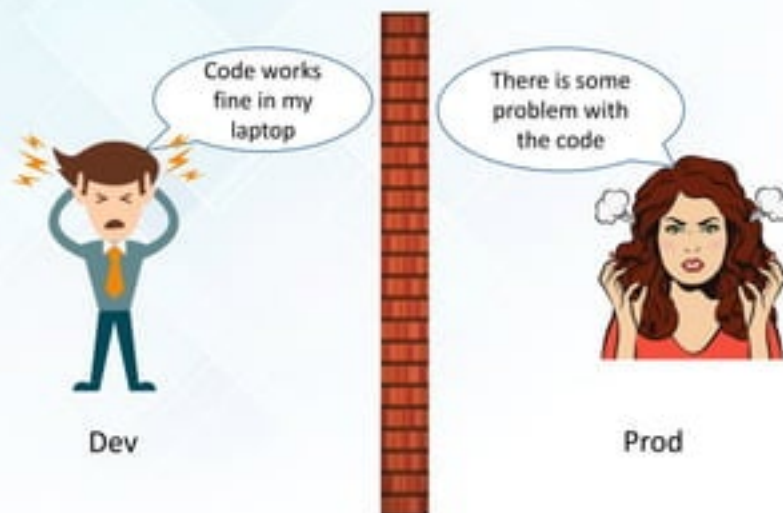
In Dev there can be a software that is upgraded and in Prod the old version of software might be present

Dev

Prod

Problems Before Docker

The idea behind microservices is that some types of applications become easier to build and maintain when they are broken down into smaller, composable pieces which work together. Each component is developed separately, and the application is then simply the sum of its constituent components.


For example imagine an online shop with separate microservices for user-accounts, product-catalog order-processing and shopping carts

Problems Before Docker

Developing an application requires starting several of microservices in one machine. So if you are starting five of those services you require five VMs on that machine.

Virtual Machines for starting multiple microservices

Developer's Laptop
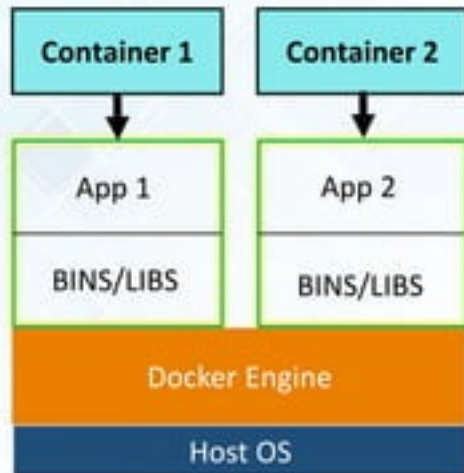
How Docker Solves These Problems

You can run several microservices in the same VM by running various Docker containers for each microservice.
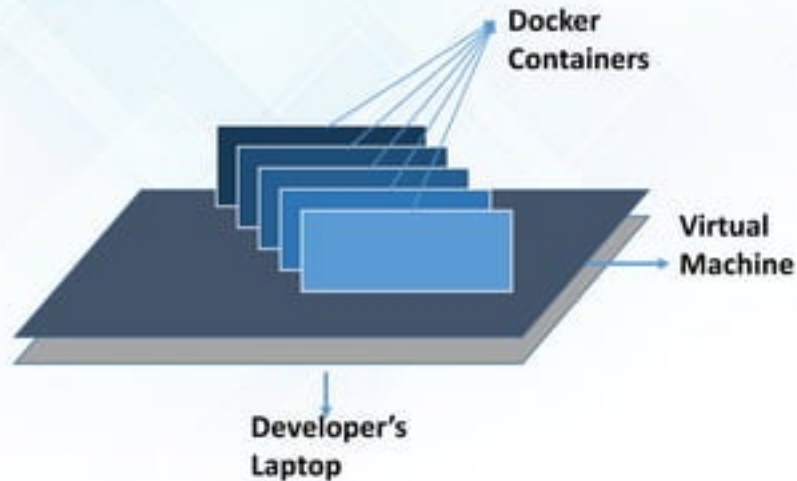
Provides a consistent computing environment throughout the whole SDLC.

Docker Containers

Virtual Machine

Developer's Laptop
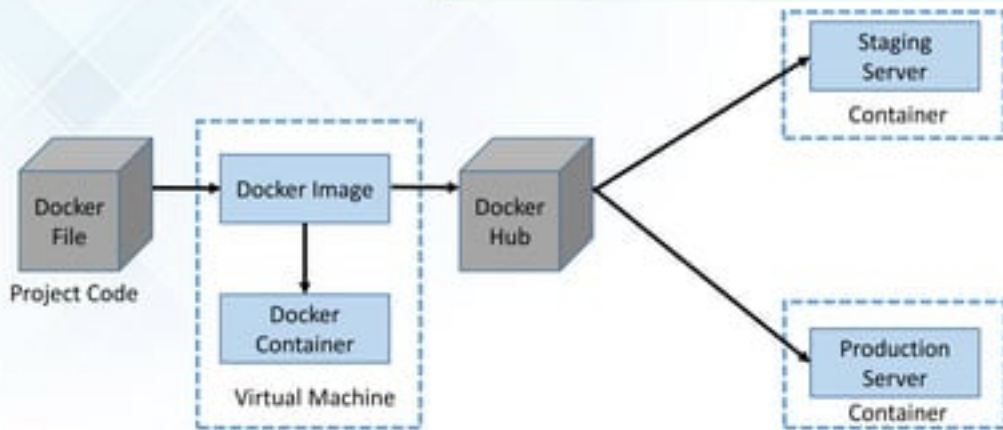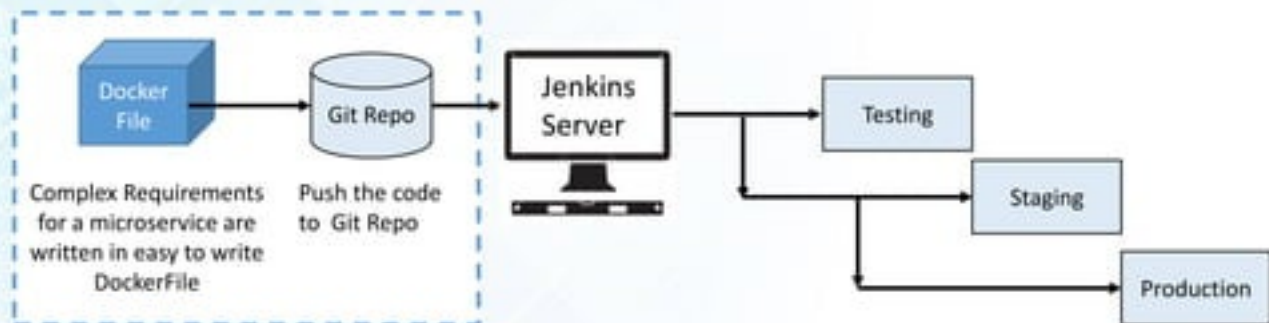
Software
Development
Life
Cycle

What Is Docker?

- Docker is a tool designed to make it easier to create, deploy, and run applications by using containers.
- Docker containers are lightweight alternatives to Virtual Machines and it uses the host OS.
- You don't have to pre-allocate any RAM in containers.

Container 1	Container 2

App 1	App 2
BINS/LIBS	BINS/LIBS

Docker Engine

Host OS

Docker In A Nutshell

- Docker file builds a Docker image and that image contains all the project's code
- You can run that image to create as many docker containers as you want
- Then this Image can be uploaded on Docker hub, from Docker hub any one can pull the image and build a container

Docker Example

Docker File — Complex Requirements for a microservice are written in easy to write DockerFile

Git Repo — Push the code to Git Repo

Jenkins Server

Testing

Staging

Production

- Create complex requirements for a microservice within an easy-to-write Dockerfile.
- Push the code up to the Git Repo.

- CI server pull it down and build the exact environment that will be used in production to run the test suite without needing to configure the CI server at all.
- Deploy it out to a staging environment for testers.
- Roll exactly what you had in development, testing, and staging into production

Docker Case-Study Indiana University

Problem Statement:

1

Custom Script → VM

Custom Script → VM

Custom Script → VM

Applications are deployed in the VMs using custom scripts

2

- Their environment was optimized for their legacy Java-based applications.

edureka!
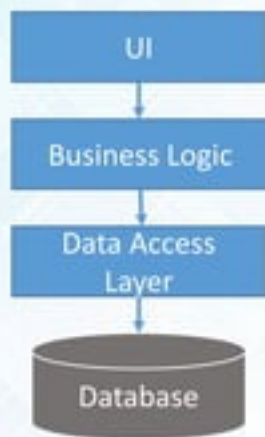
Problem Statement:

Security was needed for student's data such as SSNs and student health data.

4

Docker Case-Study Indiana University

Solution: Docker Data Center (DDC)

Management
Universal Control Plane

Security
LDAP

Registry Services
Trusted Registry

Orchestration
Swarm

Container Runtime
CS Engine

Sorce: https://www.docker.com/customers/indiana-university-delivers-state-art-it-115000-students-docker-datacenter

Docker Case-Study Indiana University
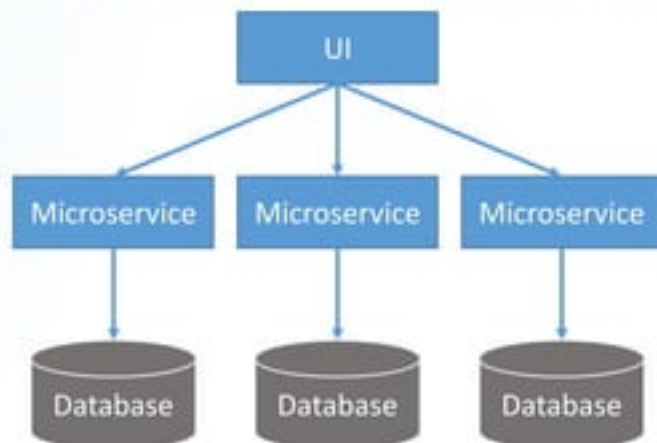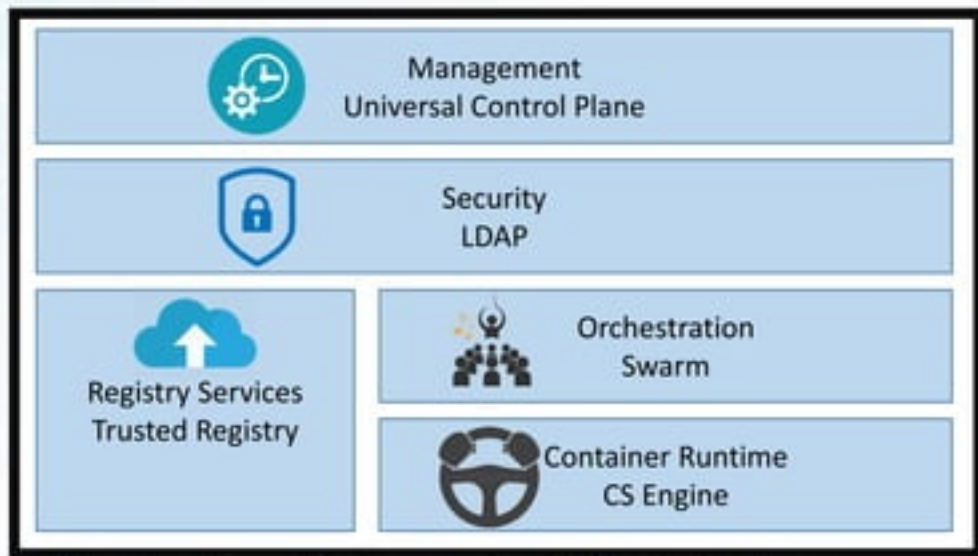
Solution: Docker Data Center (DDC)

Helps in managing whole cluster from a single place. Services are deployed using UCP web UI, using Docker images that are stored in DTR.

Docker Trusted Registry → **UCP Web UI** → Host A / Host B / Host C

It stores the Docker images

IT ops teams leverages Universal Control Plane to provision Docker installed software on hosts, and then deploy their applications without having to do a bunch of manual steps to set up all their infrastructure.
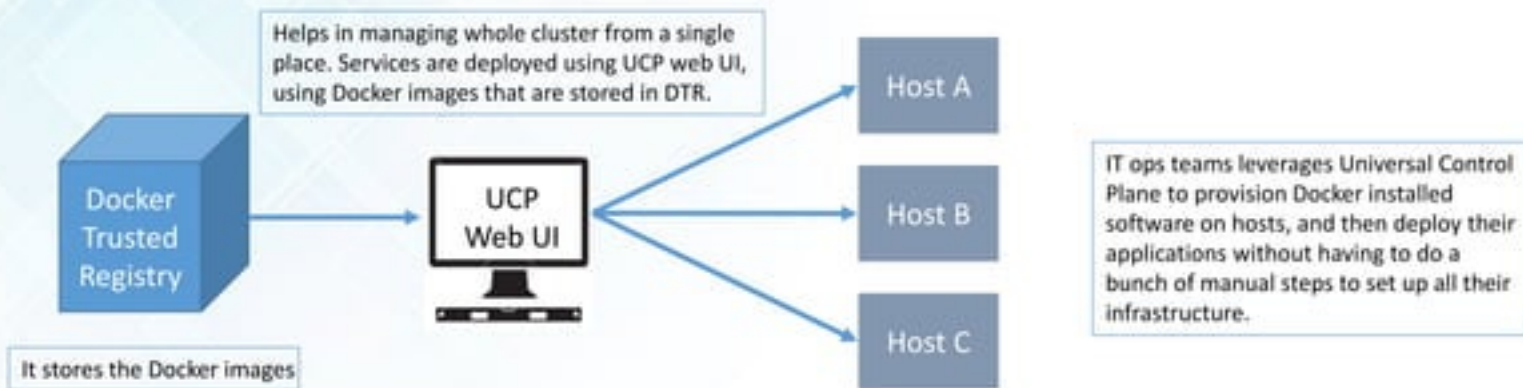
Sorce: https://www.docker.com/customers/indiana-university-delivers-state-art-it-115000-students-docker-datacenter

Docker Registry

- Docker Registry is a storage component for Docker Images
- We can store the Images in either Public / Private repositories
- Docker Hub is Docker's very own cloud repository

Why Use Docker Registries?

- Control where your images are being stored
- Integrate image storage with your in-house development workflow

Docker Images & Containers

Docker Images

- Read Only Template Used To Create Containers
- Built By Docker Users
- Stored In Docker Hub Or Your Local Registry

Docker Images & Containers
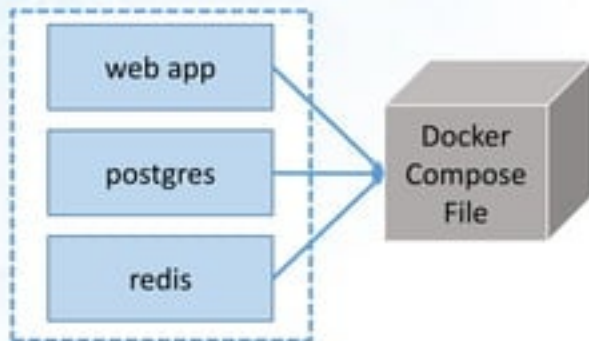
run

Docker Images

Docker Containers

- Read Only Template Used To Create Containers
- Built By Docker Users
- Stored In Docker Hub Or Your Local Registry

- Isolated Application Platform
- Contains Everything Needed To Run The Application
- Built From One Or More Images

Docker Compose

Docker Compose makes it easier to configure and run applications made up of multiple containers. For the example: imagine being able to define three containers—one running a web app, another running postgres, and a third running redis—all in one YAML file and then running those three connected containers with a single command.

Containers

web app

postgres

redis

Docker Compose File

You can run these three containers with a single command

edureka!

Thank You

For more information please visit our website
www.edureka.co