

# Docker

## What is Docker?

Docker is a tool that performs operating-system-level virtualization, commonly known as "containerization." It enables the creation of isolated environments called containers, in which applications can be deployed easily and efficiently. Unlike traditional virtual machines, containers share the host system's kernel but operate independently, making them lightweight and fast.

The name "Docker" is derived from shipping containers, emphasizing the concept of packaging and moving software consistently across different environments. Docker leverages a union file system and a layered file system to manage the contents of containers.

## Important Differences

- **Container:** A container is an isolated environment similar to a virtual machine but more lightweight, sharing the host OS's kernel.
- **Docker:** Docker is the tool used to create and manage these containers. It simplifies the process of deploying and running applications in a consistent environment.

## Differences Between VMWare & VirtualBox

- **Image:** Docker uses images that contain the necessary OS components and software, whereas VMWare and VirtualBox create full virtual machines with a complete guest OS.

## Docker Benefits

- **Containerization (OS-Level Virtualization):** Containers eliminate the need for a guest OS, making them more efficient.
- **No Pre-Allocation of RAM:** Resources are allocated dynamically as needed.
- **Environment Replication:** Easily replicate the same environment across different systems.
- **Cost-Efficiency:** Containers are less resource-intensive, reducing costs.
- **Lightweight:** Containers are typically only a few megabytes in size.
- **Fast Startup:** Containers can be started quickly, reducing deployment times.
- **Versatility:** Containers can run on physical, virtual, or cloud environments.
- **Reusability:** Docker images can be reused across different projects.
- **Quick Setup:** Machines can be created in less time compared to traditional virtual machines.

## Docker Components

- **Docker Image:** A lightweight, minimalistic OS image combined with the necessary software. It serves as a blueprint for creating containers.
- **Docker Container:** A running instance of a Docker image, functioning as an isolated environment.

- **Dockerfile:** A text file containing instructions to build a Docker image.
- **Docker Hub/Registry:** A repository that stores Docker images, which can be accessed publicly.
- **Docker Daemon:** The background service that manages Docker containers and images.

## Ways to Create Docker Images

1. **Pull from Docker Hub:** Use pre-existing images available in the Docker Hub.
2. **Create from Existing Docker Containers:** Generate an image from a running container.
3. **Build from a Dockerfile:** Write a Dockerfile to define the steps for creating a custom image.