

	is a text document that contains all the commands you would normally execute manually in order to build a Docker image. Docker can build images automatically by reading the instructions from a Dockerfile.
	is the primary way that many Docker users interact with Docker. When you use commands such as docker run, the client sends these commands to dockerd, which carries them out. The docker command uses the Docker API. The Docker client can communicate with more than one daemon.
Docker Command Line Interface (CLI)	The Docker client provides a command line interface (CLI) that allows you to issue build, run, and stop application commands to a Docker daemon.
Docker daemon (dockerd)	creates and manages Docker objects, such as images, containers, networks, and volumes.
Docker Hub	is the world's easiest way to create, manage, and deliver your team's container applications.
Docker localhost	Docker provides a host network which lets containers share your host's networking stack. This approach means that a localhost in a container resolves to the physical host, instead of the container itself.
Docker remote host	A remote Docker host is a machine, inside or outside our local network which is running a Docker Engine and has ports exposed for querying the Engine API.
Docker networks	help isolate container communications.
Docker plugins	such as a storage plugin, provides the ability to connect external storage platforms.
Docker storage	uses volumes and bind mounts to persist data even after a running container is stopped.
LXC	Linux Containers is a OS-level virtualization technology that allows creation and running of multiple isolated Linux virtual environments (VE) on a single control host.
IBM Cloud Container Registry	stores and distributes container images in a fully managed private registry.
Image	An immutable file that contains the source code, libraries, and dependencies that are necessary for an application to run. Images are templates or blueprints for a container.
Immutability	Images are read-only; if you change an image, you create a new image.
Microservices	are a cloud-native architectural approach in which a single application contains many loosely coupled and independently deployable smaller components or services.
Namespace	A Linux namespace is a Linux kernel feature that isolates and virtualizes system resources. Processes which are restricted to a namespace can only interact with resources or processes that are part of the same namespace. Namespaces are an important part of Docker's isolation model. Namespaces exist for each type of resource, including networking, storage, processes, hostname control and others.
Operating System Virtualization	OS-level virtualization is an operating system paradigm in which the kernel allows the existence of multiple isolated user space instances, called containers, zones, virtual private servers, partitions, virtual environments, virtual kernels, or jails.
Private Registry	Restricts access to images so that only authorized users can view and use them.
REST API	A REST API (also known as RESTful API) is an application programming interface (API or web API) that conforms to the constraints of REST architectural style and allows for interaction with RESTful web services.
Registry	is a hosted service containing repositories of images which responds to the Registry API.
Repository	is a set of Docker images. A repository can be shared by pushing it to a registry server. The different images in the repository can be labelled using tags.
Server Virtualization	Server virtualization is the process of dividing a physical server into multiple unique and isolated virtual servers by means of a software application. Each virtual server can run its own operating systems independently.
Serverless	is a cloud-native development model that allows developers to build and run applications without having to manage servers.
Tag	A tag is a label applied to a Docker image in a repository. Tags are how various images in a repository are distinguished from each other.

