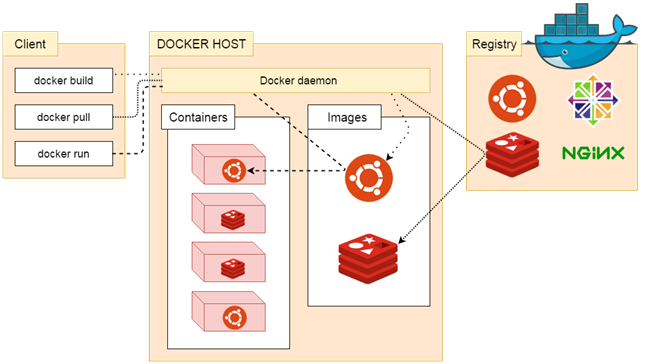
Whenever we have to install software, we have to take care of a lot of things. There are a lot of **different versions of software available for different operating systems** and their different versions.

 Docker runs containers which **contain the software plus the other additional things** that the software needs to run. So, this means you just use a ‘docker run’ command with the name of the image that you want to install,

for example - the ‘docker run imagename’ command, the request is received by the docker daemon. The daemon will search for the **image** locally, and if found, it will run it as a container. Think of an image as an executive file. If the image is not found locally, the daemon will search for it in a registry and then run it as a container.

1. Docker is a software which provides **centralized platform** to execute your application.

It wraps software components into a complete standardized unit which contains everything require to run.



2. What are the three components of Docker Architecture

* Client:  Docker provides Command Line Interface (CLI) tools to client to interact with Docker daemon.
* Docker-Host: It contains Containers, Images, and Docker daemon.
* Registry: It is **global** repository of images. You can access and use these images to run your application in Docker environment.

**Daemon**:It is a process which is used to listen for Docker API requests. It also manages Docker objects like: images, container, network etc. A daemon can also communicate with other daemons to manage Docker services.Continue…..<https://www.javatpoint.com/docker-architecture>

What is Docker with Example:- <https://www.quora.com/What-is-Docker-Please-explain-it-in-simple-terms>

**docker login**

This command is used to login to the docker hub repository

| Command | Description |
| --- | --- |
| [docker image build](https://docs.docker.com/engine/reference/commandline/image_build/) | Build an image from a Dockerfile |
| [docker image history](https://docs.docker.com/engine/reference/commandline/image_history/) | Show the history of an image |
| [docker image import](https://docs.docker.com/engine/reference/commandline/image_import/) | Import the contents from a tarball to create a filesystem image |
| [docker image inspect](https://docs.docker.com/engine/reference/commandline/image_inspect/) | Display detailed information on one or more images |
| [docker image load](https://docs.docker.com/engine/reference/commandline/image_load/) | Load an image from a tar archive or STDIN |
| [docker image ls](https://docs.docker.com/engine/reference/commandline/image_ls/)  **docker ps**  **docker ps –a**  **docker images** | List images   This command is used to list the running containers   show all the running and exited containers  lists all the locally stored docker images |
| [docker image prune](https://docs.docker.com/engine/reference/commandline/image_prune/) | Remove unused images |
| [docker image pull](https://docs.docker.com/engine/reference/commandline/image_pull/) | Pull an **image or a repository** from a registry |
| [docker image push](https://docs.docker.com/engine/reference/commandline/image_push/) | Push an **image or a repository** to a registry |
| [docker image rm](https://docs.docker.com/engine/reference/commandline/image_rm/) | Remove one or more images |
| [docker image save](https://docs.docker.com/engine/reference/commandline/image_save/) | Save one or more images to a tar archive (streamed to STDOUT by default) |
| [docker image tag](https://docs.docker.com/engine/reference/commandline/image_tag/) | Create a tag TARGET\_IMAGE that refers to SOURCE\_IMAGE |