Docker:

* Docker is a tool that encapsulates the process of creation of distributable artifact and deploying it to scale it into any environment.
* Benefits of docket workflow.
  + It makes architectural decisions simpler, because all the applications look the same from the hosting machine.
  + Docker wraps up all the requirements in a single file that is defined in a single file.
  + Bundles the application software and the OS requirements in a single package.
  + Same packaged artifact can be tested and delivered across all environments.
  + Unlike VM’s which use a dedicated amount of the resources from the host machine. Docker containers work like a process and talk to the Linux kernel, and use up the resources until the system quota is reached.
* Architecture of docker is simple client/server model, with only one executable serving on both ends depending upon on how the docker command is invoked.
  + Client/Server Model: Docker consists primarily of two parts docker server and docker client and optional third component called as registry which consists of image files.
  + Server does the ongoing work of running and maintaining containers and client is used to tell the server on what needs to be done.
  + Docker daemon can run on any number of servers, clients drive all the communication but docker servers talk directly to the registry whenever instructed by the client.

.

Docker Registry

Docker Client

Docker Server

* Docker is a little different from traditional client server model, instead of having different components for both client and server, it uses the same binary executable for both.