bora shiva

[Company name]  [Company address]

DOCKER

* **Docker** is a tool designed to make it easier to create, deploy, and run applications by using containers. **Containers** allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package
* **Docker** is a container management.
* **Container** having binary and library files, it will use the kernel of host OS
* **Container** having required files remaining files we get from kernel OS
* **Containers** easy to scale(easy start, easy stop, east delete)
* **Containers** is self sustained(all necessary dependencies downloaded automatically)
* **Virtualization** means run multiple OS on single physical system.
* **Hypervisor** is a program that would enable you to host several different machine on a single machine hardware.
* Hypervisor is a software component
* It works on host machine
* Enable virtualization
* Distributes host hardware resources
* Provides is isolation
* Guest cannot affect host or other guest
* **Docker** **Image** is a source of container
* **Image** is a layer by layer architecture.
* **Dockerfile** is used to create images.
* **Dockerfile :**

FROM java:latest

MAINTAINER shiva shivardy06@gmail.com

LABEL env=production

ENV apparea /var/jenkins\_home

Run mkdir -p $apparea

ADD ./jenkins.war $apparea

WORKDIR $apparea

CMD ["java","-jar","jenkins.war"]

* **To run a docker file**: docker build –t Jenkins . (dot declared current path of docker file) (Jenkins is a image name)
* **Create a container through image:**

Docker run –p 8080:8080 jenkins /bin/bash

* Note: if u want to remove image first stop the container done by image.
* **Stop image & stop all images:**

Docker stop container\_id

Docker stop $(docker images –a –q)

* **Remove image & remove all images:**

Docker rm container\_id

Docker rmi $(docker images –a –q)

* Details of container

Docker inspect container\_id

* **Stop container & stop containers:**

Docker stop container\_id

Docker stop $(docker ps –a –q)

* **Remove container & remove containers:**

Docker rm container\_id

Docker rm $(docker ps –a –q)

* **Open a container:**

Docker exec -ti docker\_id /bin/bash