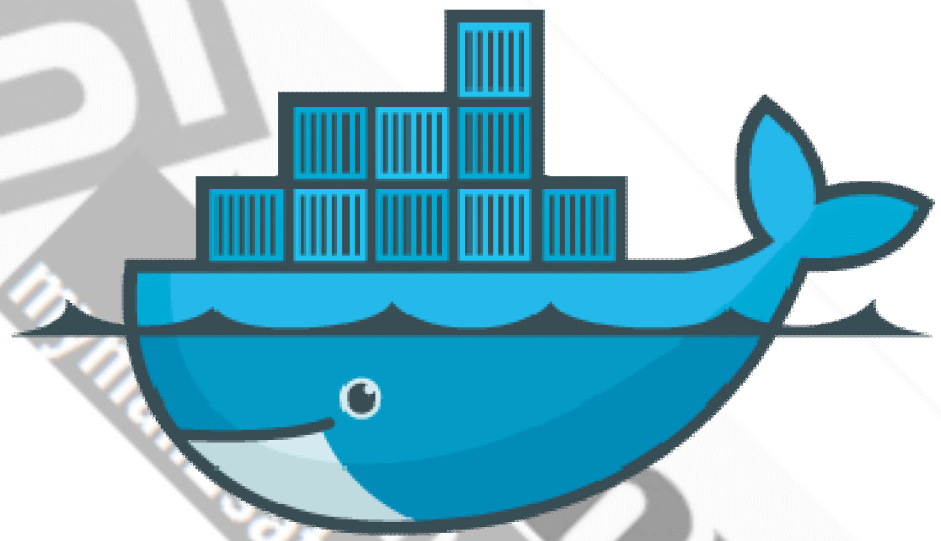

docker

Docker Volumes

- In Docker, you have a separate volume that can shared across containers. These are known as **data volumes**. Some of the features of data volume are –
- They are initialized when the container is created.
- They can be shared and also reused amongst many containers.
- Any changes to the volume itself can be made directly.
- They exist even after the container is deleted.

 # docker run -it -v /root/files:/files ubuntu

Dockerfile

- Create images automatically using a build script: «Dockerfile»
- Can be versioned in a version control system like Git or SVN, along with all dependencies
- Docker Hub can automatically build images based on dockerfiles on Github

Dockerfile Example

```
#vi  Dockerfile
    FROM ubuntu
    MAINTAINER satish
    RUN apt-get update
    RUN echo “sathya tech”
    CMD ["echo","Hello world"]
# docker build -t hello:latest .
# docker run  hello
```

Command	Description
ADD	Copies a file from the host system onto the container
COPY	Copies a file from the host system onto the container
CMD	The command that runs when the container starts
ENV	Sets an environment variable in the new container
EXPOSE	Opens a port for linked containers
FROM	The base image to use in the build. This is mandatory and must be the first command in the file.
MAINTAINER	An optional value for the maintainer of the script
RUN	Executes a command and save the result as a new layer
USER	Sets the default user within the container
VOLUME	Creates a shared volume that can be shared among containers or by the host machine
WORKDIR	Set the default working directory for the container

Example-1:

```
FROM ubuntu
MAINTAINER satish
RUN apt-get update && apt-get install -y wget
RUN apt-get install -y apache2
COPY index.html /var/www/html/index.html
ENV NAME=JAVA_MACHINE
VOLUME  myvol
EXPOSE 80
CMD ["service","apache2","start"]
```

```
#docker build  -t  myapache:latest   .
#docker run -itd  myapache  /bin/bash
#cd myvol
```

Example-2:

```
FROM tomcat:8.0
MAINTAINER satish
RUN apt-get update && apt-get install -y curl
RUN apt-get install -y git-core
COPY index.html /usr/local/tomcat/webapps/ROOT/index.html
ADD https://tomcat.apache.org/tomcat-7.0-
  doc/appdev/sample/sample.war
  /usr/local/tomcat/webapps/sample.war
USER root
WORKDIR /usr/local/tomcat/webapps
EXPOSE 8080
CMD ["catalina.sh","run"]
```

```
#docker build  -t  mytom   .
#docker run -itd  –p 8080:8080 mytom
```

Example-3:

```
FROM ubuntu
MAINTAINER satish
RUN apt-get update && apt-get install -y nginx
RUN echo 'Our first Docker image for Nginx' >
    /usr/share/nginx/html/index.html
CMD [‘/etc/init.d/nginx’,’start’]
EXPOSE 80
```