

Docker - Dockerfile - Image - Containers

# Topics

- What is Docker File
- How to create Dockerfile
- How to build images from docker file
- Basic Commands
- Build dockerfile, image and container for tomcat server
- FROM → from where the image is created
- RUN → for executing certain commands while building image
- CMD → executes in command line while creating container

## Links

<https://docs.docker.com/engine/reference/builder/>

## Building Basic Dockerfile - Image - running Container

#mkdir Dockerfiles

#cd Dockerfiles

#touch Dockerfile

#vim Dockerfile

press i for insert mode

```
root@ip-172-31-93-205:/dockerimages# cat Dockerfile
#basic image
FROM ubuntu

MAINTAINER rajesh <rajeshdeveloper99@gmail.com>

RUN apt-get update

CMD ["echo","Hello World"]
root@ip-172-31-93-205:/dockerimages#
```

#docker images → to list all images

#docker build <location of dockerfile> → building docker image if you are in diff dir

#docker build . → building docker image if you are in same dir

#docker build -t <image name>:tag . → building docker image with user defined name

Image name should be always small letters and it needs at least one arg

#docker run <image id>

# Tomcat Docker

```
FROM tomcat:8.0

MAINTAINER rajesh

RUN apt-get update && apt-get install -y curl

RUN apt-get install -y git-core

ADD https://s3.ap-south-1.amazonaws.com/vijay-rajesh/tomcat-users.xml /usr/local/tomcat/conf/tomcat-users.xml

ADD https://s3.ap-south-1.amazonaws.com/vijay-rajesh/context.xml /usr/local/tomcat/webapps/manager/META-INF/

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 .` → for building image

`#docker run -itd -p 8080:8080 mytom` → for running container

# Docker - Compose

- what / why - Docker Compose
- How to install
- How to create docker compose file
- how to use docker compose file to create services
- basic commands

Links

<https://github.com/docker/compose/releases>

## Docker compose

tool for defining & running multi-container docker applications

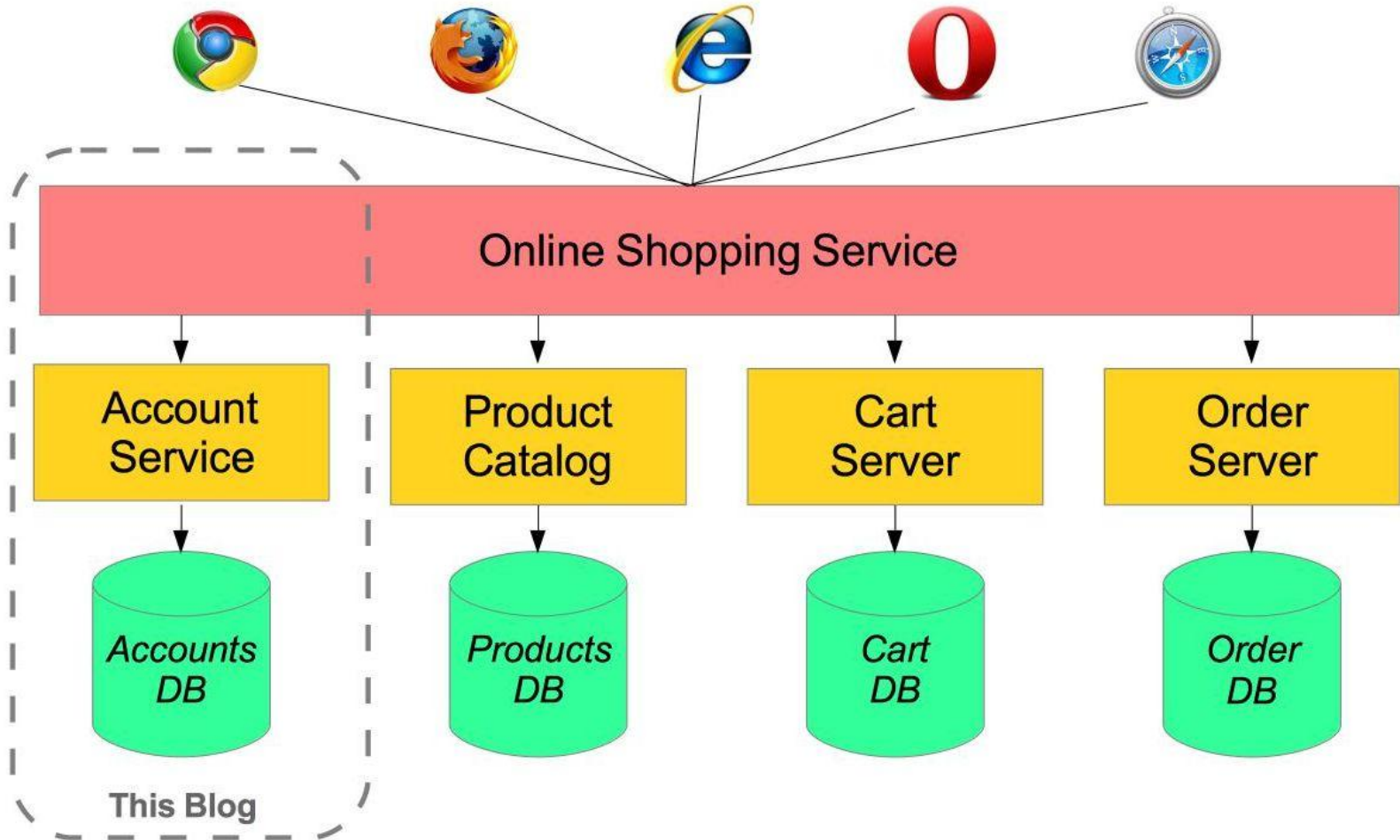
use yaml files to configure application service (docker-compose.yml)

can start all services with a single command : `docker compose up`

can stop all services with a single command : `docker compose down`

can scale up selected services when required





#docker-compose -v → to check docker compose version

```
curl -L
```

```
https://github.com/docker/compose/releases/download/1.24.0-rc1/docker-compo  
se-`uname -s`-`uname -m` -o /usr/local/bin/docker-compose  
chmod +x /usr/local/bin/docker-compose
```

```
#docker-compose -v
```

Create docker image and docker-compose file in same location

# Docker File

```
FROM tomcat:8.0

MAINTAINER rajesh

RUN apt-get update && apt-get install -y curl

RUN apt-get install -y git-core

ADD https://s3.ap-south-1.amazonaws.com/vijay-rajesh/tomcat-users.xml /usr/local/tomcat/conf/tomcat-users.xml

ADD https://s3.ap-south-1.amazonaws.com/vijay-rajesh/context.xml /usr/local/tomcat/webapps/manager/META-INF/

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"]
```

After creating docker image -- write yml file

```
version: '2'

services:
  mytom:
    build: .
    container_name: development
    ports:
      - 8080:8080
    environment:
      - spring.profiles.active=development
```

~

~

~

~

#docker-compose config → to check compose file written correctly

#docker-compose up -d

# Running two services using docker compose

Check docker-engine installed or not → `#docker -v`

Check docker-compose installed or not → `#docker-compose -v` (if not installed refer above slides for installing)

Create a dir → `#mkdir dockerfile`

Change to dockerfile dir → `cd dockerfile`

Create a dockerfile for tomcat → `#vi Dockerfile`

(write dockerfile content and save & quit)

# Docker file Content

FROM tomcat:8.0

MAINTAINER rajesh

RUN apt-get update && apt-get install -y curl

RUN apt-get install -y git-core

ADD <https://s3.ap-south-1.amazonaws.com/vijay-rajesh/tomcat-users.xml>

/usr/local/tomcat/conf/tomcat-users.xml

ADD <https://s3.ap-south-1.amazonaws.com/vijay-rajesh/context.xml>

/usr/local/tomcat/webapps/manager/META-INF/

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 . → for building image

#docker run -itd -p 8080:8080 mytom → for running container

### **Create a docker compose file**

#vi docker-compose.yml

After writing content script and saving

#docker-compose config

#docker-compose up -d → to run docker compose

#docker-compose down → to down docker compose

Check in browser both are running or not



# Docker-compose.yml

version: "3"

services:

mytom:

build: .

container\_name: development

ports:

- 8080:8080

environment:

- spring.profiles.active=development

web:

image: nginx

ports:

- 9090:80/tcp