

We have been seeing multiple articles on how to run applications inside the Docker container. Similarly in this article we will see how we can run a tomcat server in a Docker container. We will also expose ports so that we can access the tomcat application.

Search for the Docker container with tomcat server by using the command "docker search tomcat" which will give the details below,

```
[root@vx111a testTomcat]# docker search tomcat
```

INDEX	NAME	DESCRIPTION	STARS
docker.io	docker.io/tomcat	Apache Tomcat is an open source implementa...	409
docker.io	docker.io/consol/tomcat-7.0	Tomcat 7.0.57, 8080, "admin/admin"	14
docker.io	docker.io/consol/tomcat-8.0	Tomcat 8.0.15, 8080, "admin/admin"	13
docker.io	docker.io/dordoka/tomcat	Ubuntu 14.04, Oracle JDK 8 and Tomcat 8 ba...	10
docker.io	docker.io/consol/tomcat-6.0	Tomcat 6.0.43, 8080, "admin/admin"	6
docker.io	docker.io/consol/tomcat-5.5	Tomcat 5.5.36, 8080, "admin/admin"	4

Now once we got the tomcat image to be downloaded and used, we can then write the DockerFile to this purpose.

Create a DockerFile with the below contents

```
[root@vx111a testTomcat]# cat Dockerfile
FROM tomcat
MAINTAINER tomcat/ub jagadish
COPY ./myApp.war /usr/local/tomcat/webapps/
CMD ["catalina.sh","start"]
```

As we can see that we are going to use the "tomcat" image for our purpose. In the second line we have the MAINTAINER details. The third line will copy the application myApp.war from the current location to the /usr/local/tomcat/webapps/ location inside the container. The last step is to run the command cataline.sh by passing the argument Start to that.

Lets build the docker using the DockerFile created as,

```
[root@vx111a testTomcat]# docker build -t jagadish/tomcat .
```

Sending build context to Docker daemon 4.096 kB

Step 0 : FROM tomcat

Trying to pull repository [docker.io/library/tomcat](https://hub.docker.com/r/docker.io/library/tomcat) ... latest: Pulling from library/tomcat

6d1ae97ee388: Pull complete

8b9a99209d5c: Pull complete

2e05a52ffd47: Pull complete

9fdaeae348bb: Pull complete

67d05086af43: Pull complete

2e9d1ec89d66: Pull complete

1afb0d51eee0: Pull complete

5cb24a57fa37: Pull complete

110c2f290b04: Pull complete

966dcd51a14f: Pull complete

8a57ce404f1b: Pull complete

e1b97b980d07: Pull complete

548f21c48132: Pull complete

3e93be06ad38: Pull complete

3e2882dd7e87: Pull complete

4ef5a14c7b39: Pull complete

fca011d2612a: Pull complete

119ddf0db1a7: Pull complete

1b8329afb263: Pull complete

Digest: sha256:6880839ca278600ea2a853455dd73c8ec8db9c0860d4aafc4a2b8b4d23dcdd85

**Status: Downloaded newer image for [docker.io/tomcat:latest](https://hub.docker.com/r/docker.io/tomcat)**

---> 1b8329afb263

**Step 1 : MAINTAINER tomcat/ub jagadish**

```
---> Running in e0588ccfdb59
```

```
---> c5829c11b42b
```

```
Removing intermediate container e0588ccfdb59
```

**Step 2 : COPY ./myApp.war /usr/local/tomcat/webapps/**

```
---> b3765f3df7c3
```

```
Removing intermediate container 091028d853b9
```

**Step 3 : CMD catalina.sh start**

```
---> Running in 0aba2a5f35a4
```

```
---> ca997a0f848e
```

```
Removing intermediate container 0aba2a5f35a4
```

```
Successfully built ca997a0f848e
```

Now once the Tomcat Image for docker is downloaded and configured, we can see the image using the "docker images" as,

```
[root@vx111a testTomcat]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	VIRTUAL SIZE
jagadish/tomcat	latest	ca997a0f848e	6 minutes ago	350 MB

Now once we have the Tomcat Container downloaded and available, we can run the Docker container using the above image as,

```
[root@vx111a testTomcat]# docker run -t -t -p 8080:8080 jagadish/tomcat  
/usr/local/tomcat/bin/catalina.sh run
```

```
Using CATALINA_BASE:  /usr/local/tomcat
```

```
Using CATALINA_HOME:  /usr/local/tomcat
```

```
Using CATALINA_TMPDIR: /usr/local/tomcat/temp
```

```
Using JRE_HOME:      /usr
```

```
Using CLASSPATH:     /usr/local/tomcat/bin/bootstrap.jar:/usr/local/tomcat/bin/tomcat-juli.jar
```

```
*****
```

```
08-Jan-2016 13:51:26.925 INFO [main] org.apache.coyote.AbstractProtocol.start Starting  
ProtocolHandler ["ajp-nio-8009"]
```

```
08-Jan-2016 13:51:26.925 INFO [main] org.apache.catalina.startup.Catalina.start Server startup  
in 547 ms
```

We ran the Cataline.sh command by passing the "run" argument to it. Once we ran the command we will also see the catalina.sh logs shown on the screen.

We also made sure that the ports 8080 is exposed on the tomcat docker container that we are running. Now we can access the tomcat container using the <http://localhost:8080> which will show you the tomcat container login page.