

Title: Deploying Spring WAR to Tomcat-based docker

Post Body:

I have followed through Spring's [Building a RESTful Web Service](#) tutorial and created a dummy webapp (with 'Build with Maven' instructions). I build and package the WAR. Then I run it with this command:

```
java -jar ./target/Dummy-1.0-SNAPSHOT.war
```

I can see the dummy JSON endpoint at <http://localhost:8080/greeting/>.

Now I want to containerize the app with Docker so I can further test it without the needs to install Tomcat to system space. This is the `Dockerfile` I created:

```
FROM tomcat:7-jre8-alpine # copy the WAR bundle to tomcat COPY /target/Dummy-1.0-SNAPSHOT.war /usr/local/tomcat/webapps/app.war
```

I build and run the docker binding to <http://localhost:8080>. I can see the Tomcat welcome page on '<http://localhost:8080>'. But I couldn't see my app on neither:

- <http://localhost:8080/app/>
- <http://localhost:8080/app/greeting/>
- <http://localhost:8080/greeting/>

How should I track down the issue? What could be the problem?

Update 1: The Tomcat admin interface screenshot

Accepted Answer:

The `Application.java` file in [the example](#) looks like this:

```
package hello; import org.springframework.boot.SpringApplication; import org.springframework.boot.autoconfigure.SpringBootApplication;
```

This is a valid SpringBoot application, but NOT a deployable application to Tomcat. To make it deployable, you can can:

1. `redefineApplication` to extend `SpringBootServletInitializer` from Spring framework web support; then

override the `configure` method:

```
package hello; import org.springframework.boot.SpringApplication; import org.springframework.boot.autoconfigure.SpringBootApplication;
```

No need to change the `pom.xml` file (or any other configurations).

After rebuilding the `dockerfile` and run it with proper port binding, the greeting example endpoint will be available through: <http://localhost:8080/app/greeting/>

References

1. [Spring Boot War deployed to Tomcat](#)
2. [Spring Boot Reference Guide: Create a deployable war file](#)

Highest Rated Answer: None