Title: Spring Boot War deployed to Tomcat

Post Body:

I am trying to deploy a Spring Boot app to Tomcat, because I want to deploy to AWS. I created a WAR file, but it does not seem to run on Tomcat, even though it is visible.

Details:
0. Here is my app:

```
@Configuration @ComponentScan @EnableAutoConfiguration public class App {      public static void main(String[] args) {
```

application.properties has following:

```
server.port=${port:7777}
```

After reading a number of [pages](#) and [question-answers](#) I added following to my POM:

http://maven.apache.org/xsd/maven-4.0.0.xsd'> 4.0.0

```
<groupId>com.niewlabs</groupId> <artifactId>highlighter</artifactId> <version>1.0-SNAPSHOT</version>  <packaging>war</packaging
```

I ran'mvn package'and got WAR file (size 250Mb), which I put into 'webapps' folder.

3. I started Tomcat and am able to see my app listed, in my case '/highlighter-1.0-SNAPSHOT'.
4. Clicking on the link for the app results in 'Status 404' page.
5. When I run Spring Boot app just by itself, without container it runs on localhost:7777, but there is nothing there when I run it in Tomcat.

Update: There is another [reference](#). Not sure how useful it is.

Accepted Answer:

This guide explains in detail how to deploy Spring Boot app on Tomcat:
http://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/#howto-create-a-deployable-war-file

Essentially I needed to add following class:

```
public class WebInitializer extends SpringBootServletInitializer {        @Override      protected SpringApplicationBuilder con
```

Also I added following property to POM:

```
<properties>                <start-class>mypackage.App</start-class> </properties>
```

Highest Rated Answer:

Hey make sure to do this changes to the pom.xml

```
<packaging>war</packaging>
```

in the dependencies section make sure to indicated the tomcat is provided so you dont need the embeded tomcat plugin.

```
<dependency>            <groupId>org.springframework.boot</groupId>            <artifactId>spring-boot-starter-tomcat</artifactI
```

This is the whole pom.xml

```
<?xml version='1.0' encoding='UTF-8'?> <project xmlns='http://maven.apache.org/POM/4.0.0' xmlns:xsi='http://www.w3.org/2001/XM
```

And the Application class should be like this

Application.java

```
package com.example;  import org.springframework.boot.SpringApplication; import org.springframework.boot.autoconfigure.SpringB
```

And you can add a controller for testing MyController.java

```
package com.example;  import org.springframework.stereotype.Controller; import org.springframework.web.bind.annotation.Request
```

Then you can run the project in a tomcat 8 version and access the controller like this

http://localhost:8080/demo/hi

If for some reason you are not able to add the project to tomcat do a right click in the project and then go to the Build Path->configure build path->Project Faces

make sure only this 3 are selected

Dynamic web Module 3.1 Java 1.8 Javascript 1.0