

In this tutorial, we will try to deploy our Spring Boot Application to tomcat server. Here I have used

- ➔ JDK 1.8+
- ➔ Apache Tomcat 8.5.11
- ➔ Spring Boot 1.5.3 Release
- ➔ MySQL Database (Optional)

In previous tutorial we have seen How to configure tomcat server in Windows.

At first change your application's **POM.xml** file.

1. Packaging jar ➔ war

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

2. Add following dependency to your pom file.

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

3. You can define tomcat version in your application POM file's **<properties>** tag (Optional).

```
<tomcat.version>8.5.11</tomcat.version>
```

After that, we have to change our **MyProjectApplication.java** file. Source code will look like this.

```
package com.technovalley21.MyForum;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.boot.builder.SpringApplicationBuilder;
import org.springframework.boot.web.support.SpringBootServletInitializer;

@SpringBootApplication
public class MyForumApplication extends SpringBootServletInitializer {

    @Override
    protected SpringApplicationBuilder configure(SpringApplicationBuilder application)
    {
        return application.sources(MyForumApplication.class);
    }

    public static void main(String[] args) {
        SpringApplication.run(MyForumApplication.class, args);
    }
}
```

* Application configuration is finished. Now we have to build our final **WAR** file. We will use CLI to do this.

1. Open **Command Prompt** then go to your project directory via **cd**.
2. Clean project. Run following command. **mvn clean package**
3. Build WAR file. **mvn package**
4. Find your application war file to your spring-application ➔ target folder.