

[Home](#) > [member-paid](#) > 01: Simple Spring Boot Tutorial in 8 steps

01: Simple Spring Boot Tutorial in 8 steps

Posted on [November 21, 2015](#) by [Arulkumaran Kumaraswamipillai](#)

Assumes that your Java, Maven, and eclipse are set up as per: [Setting up Java, Maven, and eclipse](#)

Step 1: Create a simple maven project using the maven “archetype” plugin.

```
1 mvn archetype:generate -DgroupId=com.mytutorial -
2
```

The above command creates a folder “**simple-spring-boot**”



Step 2: import it into eclipse by choosing **File** → **Import** → **Maven** → Existing Maven projects and select the folder “simple-spring-boot” you had just created with Maven.

600+ Full Stack Java/JEE Interview Q&As ♥Free ♦FAQs

[open all](#) | [close all](#)

✚ Ice Breaker Interview

✚ Core Java Interview C

✚ JEE Interview Q&A (3

✚ Pressed for time? Jav

✚ SQL, XML, UML, JSC

✚ Hadoop & BigData Int

✚ Java Architecture Inte

✚ Scala Interview Q&As

✚ Spring, Hibernate, & I

✚ Spring (18)

✚ Spring boot (4)

└─ ♦ 11 Spring bc

└─ 01: Simple Sp

└─ 02: Simple Sp

└─ 03: Spring box

✚ Spring IO (1)

✚ Spring JavaConl

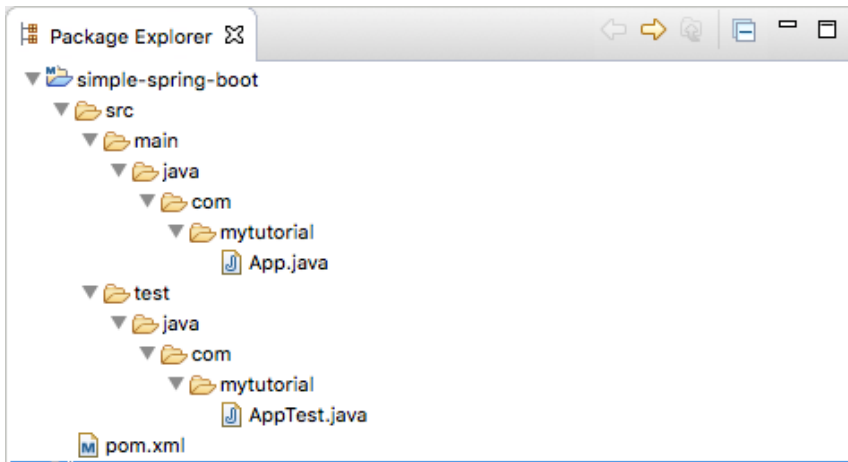
└─ 01: ♥♦ 13 Spring

└─ 01b: ♦ 13 Spring

└─ 02: ► Spring DI

└─ 03: ♥♦ Spring DI

└─ 04 ♦ 17 Spring b



Check “Maven Dependencies”:



Step 3: Add Spring boot dependency “**spring-boot-starter-actuator**”

```

1
2
3 <project xmlns="http://maven.apache.org/POM/4.0.
4     xsi:schemaLocation="http://maven.apache.org/
5     <modelVersion>4.0.0</modelVersion>
6     <groupId>com.mytutorial</groupId>
7     <artifactId>simple-spring-boot</artifactId>
8     <packaging>jar</packaging>
9     <version>1.0-SNAPSHOT</version>
10    <name>simple-spring-boot</name>
11    <url>http://maven.apache.org</url>
12
13    <parent>
14        <groupId>org.springframework.boot</group
15        <artifactId>spring-boot-starter-parent</
16        <version>1.3.0.RELEASE</version>
17    </parent>
18
19    <dependencies>
20        <dependency>
21            <groupId>junit</groupId>
22            <artifactId>junit</artifactId>
23            <scope>test</scope>
24        </dependency>
25
26        <dependency>
27            <groupId>org.springframework.boot</g
28            <artifactId>spring-boot-starter-actu
29        </dependency>
30    </dependencies>
31
32 </project>
33
34

```

- 05: ♦ 9 Spring B
- 06: ♥ Debugging
- 07: Debugging S
- Spring loading p
- ▣ Hibernate (13)
- ▣ AngularJS (2)
- ▣ Git & SVN (6)
- ▣ JMeter (2)
- ▣ JSF (2)
- ▣ Maven (3)
- ▣ Testing & Profiling/Sa
- ▣ Other Interview Q&A 1
- ▣ Free Java Interview

16 Technical Key Areas

open all | close all

- ▣ Best Practice (6)
- ▣ Coding (26)
- ▣ Concurrency (6)
- ▣ Design Concepts (7)
- ▣ Design Patterns (11)
- ▣ Exception Handling (3)
- ▣ Java Debugging (21)
- ▣ Judging Experience In
- ▣ Low Latency (7)
- ▣ Memory Management
- ▣ Performance (13)
- ▣ QoS (8)
- ▣ Scalability (4)
- ▣ SDLC (6)
- ▣ Security (13)
- ▣ Transaction Managen

**80+ step by
step Java
Tutorials**

Note: The “parent” tag and the “dependency” tags having spring boot references.

What if you want to extend a different company level parent project?

You can use the Spring IO BOM (Bill Of Material) config

Note: dependency “spring-boot-dependencies” and the plugin “spring-boot-maven-plugin” with “repackage” goal.

```

1
2 <project xmlns="http://maven.apache.org/POM/4.0.0"
3   xsi:schemaLocation="http://maven.apache.org/
4     <modelVersion>4.0.0</modelVersion>
5     <groupId>com.mytutorial</groupId>
6     <artifactId>simple-spring-boot</artifactId>
7     <packaging>jar</packaging>
8     <version>1.0-SNAPSHOT</version>
9     <name>simple-spring-boot</name>
10    <url>http://maven.apache.org</url>
11
12    <dependencyManagement>
13      <dependencies>
14        <dependency>
15          <!-- Import dependency management
16          <groupId>org.springframework.boot</groupId>
17          <artifactId>spring-boot-dependencies</artifactId>
18          <version>1.3.0.RELEASE</version>
19          <type>pom</type>
20          <scope>import</scope>
21        </dependency>
22      </dependencies>
23    </dependencyManagement>
24
25    <dependencies>
26      <dependency>
27        <groupId>junit</groupId>
28        <artifactId>junit</artifactId>
29        <scope>test</scope>
30      </dependency>
31
32      <dependency>
33        <groupId>org.springframework.boot</groupId>
34        <artifactId>spring-boot-starter-web</artifactId>
35      </dependency>
36    </dependencies>
37
38    <build>
39      <plugins>
40        <plugin>
41          <!-- Build an executable jar -->
42          <groupId>org.apache.maven.plugins</groupId>
43          <artifactId>maven-jar-plugin</artifactId>
44          <configuration>
45            <archive>

```

[open all](#) | [close all](#)

- [Setting up Tutorial \(6\)](#)
- [Tutorial - Diagnosis \(2\)](#)
- [Akka Tutorial \(9\)](#)
- [Core Java Tutorials \(2\)](#)
- [Hadoop & Spark Tutorial \(1\)](#)
- [JEE Tutorials \(19\)](#)
- [Scala Tutorials \(1\)](#)
- [Spring & Hibernate Tutorial \(1\)](#)
- [Hibernate Tutorial \(1\)](#)
- [Spring Basics Tutorial \(1\)](#)
- [Spring batch tutorial \(1\)](#)
- [JPA-Hibernate-Spring \(1\)](#)
- [Spring Boot Tutorial \(1\)](#)
- [01: Simple Spring Boot Tutorial \(1\)](#)
- [02: Simple Spring Boot Tutorial \(1\)](#)
- [03: Spring boot Tutorial \(1\)](#)
- [Spring IO Tutorial \(1\)](#)
- [Spring ORM Tutorial \(1\)](#)
- [Spring other tutorials \(1\)](#)
- [Spring OXM Tutorial \(1\)](#)
- [Tools Tutorials \(19\)](#)
- [Other Tutorials \(45\)](#)

100+ Java pre-interview coding tests

[open all](#) | [close all](#)

- [Can you write code? \(1\)](#)
- [♦ Complete the given code \(1\)](#)
- [Converting from A to B \(1\)](#)
- [Designing your class \(1\)](#)
- [Java Data Structures \(1\)](#)
- [Passing the unit tests \(1\)](#)
- [What is wrong with this code? \(1\)](#)
- [Writing Code Home Assignment \(1\)](#)
- [Written Test Core Java \(1\)](#)
- [Written Test JEE \(1\)](#)

```

46         <manifest>
47             <addClasspath>true</>
48             <classpathPrefix>lib</>
49             <mainClass>com.mytut
50         </manifest>
51     </archive>
52 </configuration>
53 </plugin>
54 <plugin>
55     <groupId>org.springframework.bo
56     <artifactId>spring-boot-maven-pl
57     <version>1.3.0.RELEASE</version>
58     <executions>
59         <execution>
60             <goals>
61                 <goal>repackage</goal>
62             </goals>
63         </execution>
64     </executions>
65 </plugin>
66 </plugins>
67 </build>
68
69 </project>
70

```

How good are your?

open all | close all

-  Career Making Know-
-  Job Hunting & Resum

Step 4: Run the “mvn package” command.

```

1
2 c:/projects/simple-spring-boot> mvn package
3 c:/projects/simple-spring-boot> mvn eclipse:eclip
4

```

Check “Maven Dependencies”: “spring-boot-starter-actuator” has added

May have to click “F5” within eclipse to refresh the projects.

Maven Dependencies

- ▶ junit-3.8.1.jar - /Users/arulkj.m2/repository/junit/junit/3.8.1
- ▶ spring-boot-starter-actuator-1.3.0.RELEASE.jar - /Users/arulkj.m2/repository/org/springframework/boot/spring-boot-starter-actuator/1.3.0.RELEASE
- ▶ spring-boot-starter-1.3.0.RELEASE.jar - /Users/arulkj.m2/repository/org/springframework/boot/spring-boot-starter/1.3.0.RELEASE
- ▶ spring-boot-1.3.0.RELEASE.jar - /Users/arulkj.m2/repository/org/springframework/boot/spring-boot/1.3.0.RELEASE
- ▶ spring-boot-autoconfigure-1.3.0.RELEASE.jar - /Users/arulkj.m2/repository/org/springframework/boot/spring-boot-autoconfigure/1.3.0.RELEASE
- ▶ spring-boot-starter-logging-1.3.0.RELEASE.jar - /Users/arulkj.m2/repository/org/springframework/boot/spring-boot-starter-logging/1.3.0.RELEASE
- ▶ logback-classic-1.1.3.jar - /Users/arulkj.m2/repository/ch/qos/logback/logback-classic/1.1.3
- ▶ logback-core-1.1.3.jar - /Users/arulkj.m2/repository/ch/qos/logback/logback-core/1.1.3
- ▶ slf4j-api-1.7.7.jar - /Users/arulkj.m2/repository/org/slf4j/slf4j-api/1.7.7
- ▶ jcl-over-slf4j-1.7.13.jar - /Users/arulkj.m2/repository/org/slf4j/jcl-over-slf4j/1.7.13
- ▶ jul-to-slf4j-1.7.13.jar - /Users/arulkj.m2/repository/org/slf4j/jul-to-slf4j/1.7.13
- ▶ log4j-over-slf4j-1.7.13.jar - /Users/arulkj.m2/repository/org/slf4j/log4j-over-slf4j/1.7.13
- ▶ spring-core-4.2.3.RELEASE.jar - /Users/arulkj.m2/repository/org/springframework/spring-core/4.2.3.RELEASE
- ▶ snakeyaml-1.16.jar - /Users/arulkj.m2/repository/org/yaml/snakeyaml/1.16
- ▶ spring-boot-actuator-1.3.0.RELEASE.jar - /Users/arulkj.m2/repository/org/springframework/boot/spring-boot-actuator/1.3.0.RELEASE
- ▶ jackson-databind-2.6.3.jar - /Users/arulkj.m2/repository/com/fasterxml/jackson/core/jackson-databind/2.6.3
- ▶ jackson-annotations-2.6.0.jar - /Users/arulkj.m2/repository/com/fasterxml/jackson/core/jackson-annotations/2.6.0
- ▶ jackson-core-2.6.3.jar - /Users/arulkj.m2/repository/com/fasterxml/jackson/core/jackson-core/2.6.3
- ▶ spring-context-4.2.3.RELEASE.jar - /Users/arulkj.m2/repository/org/springframework/spring-context/4.2.3.RELEASE
- ▶ spring-aop-4.2.3.RELEASE.jar - /Users/arulkj.m2/repository/org/springframework/spring-aop/4.2.3.RELEASE
- ▶ aopalliance-1.0.jar - /Users/arulkj.m2/repository/aopalliance/aopalliance/1.0
- ▶ spring-beans-4.2.3.RELEASE.jar - /Users/arulkj.m2/repository/org/springframework/spring-beans/4.2.3.RELEASE
- ▶ spring-expression-4.2.3.RELEASE.jar - /Users/arulkj.m2/repository/org/springframework/spring-expression/4.2.3.RELEASE

Spring-boot brought in all these Spring framework jars

Step 5: Modify the “App.java” as shown below

```
1
2 package com.mytutorial;
3
4 import org.springframework.boot.autoconfigure.EnableAutoConfiguration;
5 import org.springframework.context.annotation.ComponentScan;
6 import org.springframework.context.annotation.Configuration;
7
8 @Configuration
9 @EnableAutoConfiguration
10 @ComponentScan
11
12 public class App
13 {
14     public static void main( String[] args )
15     {
16         System.out.println( "Hello Spring Boot!" );
17     }
18 }
```

Step 6: To run the jar file, you need to create a “**META-INF/MANIFEST.MF**” file. Using the maven jar plugin to generate the “MANIFEST.MF” file.

```
1
2 <project xmlns="http://maven.apache.org/POM/4.0.0"
3     xsi:schemaLocation="http://maven.apache.org/maven-v4_0_0.xsd"
4     <modelVersion>4.0.0</modelVersion>
5     <groupId>com.mytutorial</groupId>
6     <artifactId>simple-spring-boot</artifactId>
7     <packaging>jar</packaging>
8     <version>1.0-SNAPSHOT</version>
9     <name>simple-spring-boot</name>
10    <url>http://maven.apache.org</url>
11
12    <parent>
13        <groupId>org.springframework.boot</groupId>
14        <artifactId>spring-boot-starter-parent</artifactId>
15        <version>1.3.0.RELEASE</version>
16    </parent>
17
18    <dependencies>
19        <dependency>
20            <groupId>junit</groupId>
21            <artifactId>junit</artifactId>
22            <scope>test</scope>
23        </dependency>
24
25        <dependency>
26            <groupId>org.springframework.boot</groupId>
27            <artifactId>spring-boot-starter-actuator</artifactId>
28        </dependency>
29    </dependencies>
30
31    <build>
32        <plugins>
33            <plugin>
34                <!-- Build an executable JAR -->
35                <groupId>org.apache.maven.plugins</groupId>
36                <artifactId>maven-jar-plugin</artifactId>
37                <configuration>
```

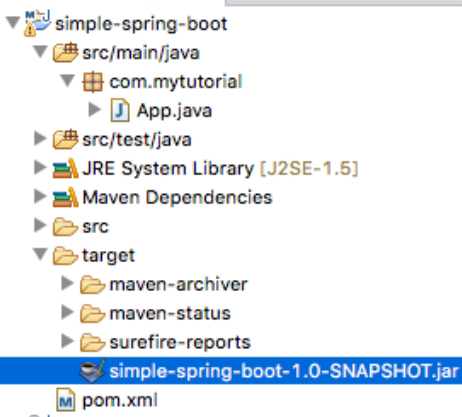
```
38         <archive>
39             <manifest>
40                 <addClasspath>true</
41                 <classpathPrefix>lib
42                 <mainClass>com.mytut
43             </manifest>
44         </archive>
45     </configuration>
46 </plugin>
47 <plugin>
48     <groupId>org.springframework.bo
49     <artifactId>spring-boot-maven-pl
50 </plugin>
51 </plugins>
52 </build>
53
54 </project>
55
56
57
```

Note: “<mainClass>com.mytutorial.App</mainClass>”

Step 7: Ready to build the “jar” now.

```
1
2 c:/projects/simple-spring-boot> mvn package
3
```

The “target” directory will have the jar file “simple-spring-boot-1.0-SNAPSHOT.jar”



Deployable “jar” is built

Step 8: Deploy the very simple Spring boot application & run as shown below

right-click on the jar, and then select “properties” to get the full file path of the jar file.

“c:/projects/simple-spring-boot/target/simple-spring-boot-1.0-SNAPSHOT.jar”

```
1  
2 c:/projects/simple-spring-boot>java -jar c:/proje  
3
```

Output:

Hello Spring Boot!

Conclusion:

If you look at the packaged “simple-spring-boot-1.0-SNAPSHOT.jar” with “jar -tvf simple-spring-boot-1.0-SNAPSHOT.jar”, you will get

```
0 Fri Nov 20 22:03:48 AEDT 2015 META-INF/  
1229 Fri Nov 20 22:03:48 AEDT 2015 META-INF/MANIFEST.MF  
0 Fri Nov 20 22:02:14 AEDT 2015 com/  
0 Fri Nov 20 22:02:14 AEDT 2015 com/mytutorial/  
778 Fri Nov 20 22:02:14 AEDT 2015 com/mytutorial/App.class  
0 Fri Nov 20 22:02:32 AEDT 2015 META-INF/maven/  
0 Fri Nov 20 22:02:32 AEDT 2015 META-INF/maven/com.mytutorial/  
0 Fri Nov 20 22:02:32 AEDT 2015 META-INF/maven/com.mytutorial/simple-spring-boot/  
1685 Fri Nov 20 22:02:10 AEDT 2015 META-INF/maven/com.mytutorial/simple-spring-boot/pom.xml  
125 Fri Nov 20 22:02:32 AEDT 2015 META-INF/maven/com.mytutorial/simple-spring-boot/pom.properties  
0 Fri Nov 20 22:03:48 AEDT 2015 lib/  
2197 Mon Nov 16 02:46:58 AEDT 2015 lib/spring-boot-starter-actuator-1.3.0.RELEASE.jar  
2217 Mon Nov 16 02:43:58 AEDT 2015 lib/spring-boot-starter-1.3.0.RELEASE.jar  
533592 Mon Nov 16 02:34:16 AEDT 2015 lib/spring-boot-1.3.0.RELEASE.jar  
748933 Mon Nov 16 02:48:32 AEDT 2015 lib/spring-boot-autoconfigure-1.3.0.RELEASE.jar  
2252 Mon Nov 16 02:43:58 AEDT 2015 lib/spring-boot-starter-logging-1.3.0.RELEASE.jar  
288928 Tue Mar 24 12:06:48 AEDT 2015 lib/logback-classic-1.1.3.jar  
455841 Tue Mar 24 12:05:28 AEDT 2015 lib/logback-core-1.1.3.jar  
33888 Tue Nov 18 21:14:04 AEDT 2015 lib/slf4j-api-1.7.13.jar  
16438 Tue Nov 18 21:19:10 AEDT 2015 lib/jcl-over-slf4j-1.7.13.jar  
4595 Tue Nov 18 21:19:54 AEDT 2015 lib/jul-to-slf4j-1.7.13.jar  
24137 Tue Nov 18 21:19:36 AEDT 2015 lib/slf4j-over-slf4j-1.7.13.jar  
1077165 Sun Nov 15 08:03:48 AEDT 2015 lib/spring-core-4.2.3.RELEASE.jar  
269579 Tue Aug 18 22:07:32 AEST 2015 lib/snakeyaml-1.16.jar  
444861 Mon Nov 16 02:43:48 AEDT 2015 lib/spring-boot-actuator-1.3.0.RELEASE.jar  
1178001 Mon Oct 12 11:31:24 AEDT 2015 lib/jackson-databind-2.6.3.jar  
46989 Mon Oct 12 09:49:58 AEDT 2015 lib/jackson-annotations-2.6.3.jar  
258875 Mon Oct 12 18:07:44 AEDT 2015 lib/jackson-core-2.6.3.jar  
1088916 Sun Nov 15 08:04:16 AEDT 2015 lib/spring-context-4.2.3.RELEASE.jar  
368915 Sun Nov 15 08:04:00 AEDT 2015 lib/spring-aop-4.2.3.RELEASE.jar  
4467 Fri Mar 19 14:57:02 AEDT 2004 lib/aopalliance-1.0.jar  
738168 Sun Nov 15 08:03:54 AEDT 2015 lib/spring-beans-4.2.3.RELEASE.jar  
262382 Sun Nov 15 08:04:02 AEDT 2015 lib/spring-expression-4.2.3.RELEASE.jar  
0 Fri Nov 20 22:03:48 AEDT 2015 org/  
0 Fri Nov 20 22:03:48 AEDT 2015 org/springframework/  
0 Fri Nov 20 22:03:48 AEDT 2015 org/springframework/boot/  
0 Fri Nov 20 22:03:48 AEDT 2015 org/springframework/boot/loader/  
1257 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/LaunchedURLClassLoader$JavaLockProvider.class  
1688 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/PropertiesLauncher$ArchiveEntryFilter.class  
1964 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/PropertiesLauncher$PrefixMatchingArchiveFilter.class  
1165 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/ExecutableArchiveLauncher$1.class  
18488 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/PropertiesLauncher.class  
1588 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/LaunchedURLClassLoader$ResourceEnumeration.class  
0 Fri Nov 20 22:03:48 AEDT 2015 org/springframework/boot/loader/data/  
1531 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/data/ByteArrayRandomAccessData.class  
3526 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/data/RandomAccessDataFilesDataInputStream.class  
351 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/data/RandomAccessData.class  
2231 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/data/RandomAccessDataFilesFilePool.class  
3398 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/data/RandomAccessDataFile.class  
1341 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/data/RandomAccessDataResourceAccess.class  
0 Fri Nov 20 22:03:48 AEDT 2015 org/springframework/boot/loader/util/  
4947 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/util/SystemPropertyUtils.class  
3788 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/util/AsciiBytes.class  
2126 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/LaunchedURLClassLoader$1.class  
3373 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/InputArgumentsJavaAgentDetector.class  
5749 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/Launcher.class  
6298 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/LaunchedURLClassLoader.class  
2128 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/JarLauncher.class  
0 Fri Nov 20 22:03:48 AEDT 2015 org/springframework/boot/loader/jar/  
314 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/jar/JarEntryFilter.class  
6561 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/jar/JarURLConnection.class  
2365 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/jar/JarEntry.class  
2161 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/jar/Bytes.class  
2943 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/jar/CentralDirectoryEndRecord.class  
1563 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/jar/JarFile$2.class  
1629 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/jar/ZipInflaterInputStream.class  
16414 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/jar/JarFile.class  
1382 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/jar/JarFile$1.class  
672 Mon Nov 16 02:31:16 AEDT 2015 org/springframework/boot/loader/jar/JarURLConnection$1.class
```

UBER jar with Spring spring-boot-maven-plugin

Java does not provide any standard way to load nested jar files, so we use “uber” jars. An uber jar simply packages all classes, from all jars, into a single archive. The following plugin creates an uber jar.

```
1
2         <plugin>
3             <groupId>org.springframework.boot
4             <artifactId>spring-boot-maven-plu
5             <version>1.3.0.RELEASE</version>
6         </plugin>
7
```

In the next tutorial I will expand this to a simple “RESTful” web service. [Simple Spring Boot Restful Web Service Tutorial](#)

Popular Member Posts

♦ 11 Spring boot interview questions & answers

904 views

♦ Q11-Q23: Top 50+ Core on Java OOP Interview Questions & Answers

816 views

001A: ♦ 7+ Java integration styles & patterns interview questions & answers

427 views

18 Java scenarios based interview Questions and Answers

407 views

♦ 7 Java debugging interview questions & answers

321 views

01b: ♦ 13 Spring basics Q8 – Q13 interview questions & answers

310 views

01: ♦ 15 Ice breaker questions asked 90% of the time in Java job interviews with hints

302 views

♦ 10 ERD (Entity-Relationship Diagrams) Interview Questions and Answers

301 views

♦ Q24-Q36: Top 50+ Core on Java classes, interfaces and generics interview questions & answers

251 views

001B: ♦ Java architecture & design concepts

interview questions & answers

209 views

0

Like

Share

↑

↓

reddit

0

G+1

Share

0

Bio

Latest Posts

**Arulkumaran Kumaraswamipillai**

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via [Amazon.com](https://www.amazon.com) in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site. **945+** paid members. [join my LinkedIn Group](#). [Reviews](#)

**About Arulkumaran Kumaraswamipillai**

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via [Amazon.com](https://www.amazon.com) in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site. **945+** paid members. [join my LinkedIn Group](#). [Reviews](#)

[◀ ♥ Converting a BST Tree to a List in Java](#)[02: Simple Spring Boot Restful Web Service Tutorial ▶](#)**Posted in** member-paid, Spring boot, Spring Boot Tutorial

Empowers you to open more doors, and fast-track

Technical Know Hows

☀ [Java generics in no time](#) ☀ [Top 6 tips to transforming your thinking from OOP to FP](#) ☀ [How does a HashMap internally work? What is a hashing function?](#)
☀ [10+ Java String class interview Q&As](#) ☀ [Java auto un/boxing benefits & caveats](#) ☀ [Top 11 slacknesses that can come back and bite you as an experienced Java developer or architect](#)

Non-Technical Know Hows

☀ [6 Aspects that can motivate you to fast-track your career & go places](#) ☀ [Are you reinventing yourself as a Java developer?](#) ☀ [8 tips to safeguard your Java career against offshoring](#) ☀ [My top 5 career mistakes](#)

Prepare to succeed

☀ [Turn readers of your Java CV go from “Blah blah” to “Wow”?](#) ☀ [How to prepare for Java job interviews?](#) ☀ [16 Technical Key Areas](#) ☀ [How to choose from multiple Java job offers?](#)

Select Category ▼

© Disclaimer

The contents in this Java-Success are copy righted. The author has the right to correct or enhance the current content without any prior notice.

These are general advice only, and one needs to take his/her own circumstances into consideration. The author will not be held liable for any damages caused or alleged to be caused either directly or indirectly by these materials and resources. Any trademarked names or labels used in this blog remain the property of their respective trademark owners. No guarantees are made regarding the accuracy or usefulness of content, though I do make an effort to be accurate. Links to external sites do not imply endorsement of the linked-to sites.