

Industrial strength Java/JEE Career Companion to open more doors

search here ...

Go

Home

Java FAQs

600+ Java Q&As

Career

Tutorials

Member

Why?

Can u Debug?

Java 8 ready?

Top X

Productivity Tools

Judging Experience?

[Home](#) > [Interview](#) > [JEE Interview Q&A](#) > [WebService](#) > 03: ♥ JAX-RS how to create a RESTful Web Service in Java?

03: ♥ JAX-RS how to create a RESTful Web Service in Java?

Posted on [June 17, 2015](#) by [Arulkumaran Kumaraswamipillai](#)



Tweet



3

Share

Q. What are the high level steps in developing a RESTful web service in Java?

A. You can see the complete tutorial in the tutorials section. This post is to quickly explain the major steps at the job interview. Knowing all the basics is of no use if you can't write a web service.

Step 1: Create a Maven project structure:

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

Step 2: Need the libraries/frameworks

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](#)

[JEE Overview \(2\)](#)

[Web basics \(8\)](#)

[WebService \(11\)](#)

[01: ♥♦ 40+ Java](#)

[02: ♦ 6 Java RE](#)

[03: ♥ JAX-RS hc](#)

[04: 5 JAXB inter](#)

[05: RESTful We](#)

[06: RESTful Wel](#)

[07: HATEOAS R](#)

[08: REST constr](#)

[09: 11 SOAP We](#)

[10: SOAP Web](#)

[11: ♥ JAX-WS hc](#)

[JPA \(2\)](#)

[JTA \(1\)](#)

[JDBC \(4\)](#)

[JMS \(5\)](#)

This example shows RESTEasy as JAX-RS implementation from JBoss.

```

1  <project xmlns="http://maven.apache.org/POM/4.0.
2      xsi:schemaLocation="http://maven.apache.org/
3      <modelVersion>4.0.0</modelVersion>
4      <groupId>com.mytutorial</groupId>
5      <artifactId>simpleRestWeb</artifactId>
6      <packaging>war</packaging>
7      <version>1.0-SNAPSHOT</version>
8      <name>simpleRestWeb Maven Webapp</name>
9      <url>http://maven.apache.org</url>
10
11     <properties>
12         <resteasy.version>2.3.6.Final</resteasy.
13         <javax-servlet.version>3.0.1</javax-serv
14         <spring.version>3.1.0.RELEASE</spring.ve
15     </properties>
16
17     <dependencies>
18         <dependency>
19             <groupId>junit</groupId>
20             <artifactId>junit</artifactId>
21             <version>3.8.1</version>
22             <scope>test</scope>
23         </dependency>
24
25         <!-- JAX-RS RESTEasy implementation-->
26         <dependency>
27             <groupId>org.jboss.resteasy</groupId>
28             <artifactId>jaxrs-api</artifactId>
29             <!-- <scope>provided</scope> -->
30             <version>${resteasy.version}</versio
31         </dependency>
32         <dependency>
33             <groupId>org.jboss.resteasy</groupId>
34             <artifactId>resteasy-jaxrs</artifact
35             <!-- <scope>provided</scope> -->
36             <version>${resteasy.version}</versio
37         </dependency>
38         <dependency>
39             <groupId>org.jboss.resteasy</groupId>
40             <artifactId>resteasy-jaxb-provider</
41             <!-- <scope>provided</scope> -->
42             <version>${resteasy.version}</versio
43         </dependency>
44         <dependency>
45             <groupId>org.jboss.resteasy</groupId>
46             <artifactId>resteasy-spring</artifac
47             <scope>runtime</scope>
48             <version>${resteasy.version}</versio
49         </dependency>
50
51         <!-- Servlet API -->
52         <dependency>
53             <groupId>javax.servlet</groupId>
54             <artifactId>javax.servlet-api</artif
55             <scope>provided</scope>
56             <version>${javax-servlet.version}</v
57         </dependency>
58
59         <!-- Spring -->
60         <dependency>

```

- [JMX \(3\)](#)
- [JNDI and LDAP \(1\)](#)
- [Pressed for time? Jav](#)
- [SQL, XML, UML, JSC](#)
- [Hadoop & BigData Int](#)
- [Java Architecture Inte](#)
- [Scala Interview Q&As](#)
- [Spring, Hibernate, & I](#)
- [Testing & Profiling/Sa](#)
- [Other Interview Q&A I](#)
- [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

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

- [Setting up Tutorial \(6\)](#)

```

61         <groupId>org.springframework</groupId>
62         <artifactId>spring-web</artifactId>
63         <scope>compile</scope>
64         <version>${spring.version}</version>
65     </dependency>
66 </dependencies>
67 <build>
68     <finalName>simpleRestWeb</finalName>
69 </build>
70 </project>
71

```

Step 3: Web service interface with JAX-RS annotations

Folder: src/main/java

```

1  package com.mytutorial;
2
3  import javax.ws.rs.GET;
4  import javax.ws.rs.Path;
5  import javax.ws.rs.PathParam;
6
7  @Path("/myapp")
8  public interface SimpleRestWeb {
9
10     @GET
11     @Path("/name/{name}")
12     public String sayHello(@PathParam("name") String name);
13 }

```

Step 4: Implementation of Web service interface

```

1  package com.mytutorial;
2
3  public class SimpleRestWebImpl implements SimpleRestWeb {
4
5      @Override
6      public String sayHello(String name) {
7          String result = "Hello " + name;
8          return result;
9      }
10 }
11

```

Step 4: Spring XML config as IoC.

Folder: src/main/resources/com/mytutorial

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <beans xmlns="http://www.springframework.org/schema/beans"
3        xmlns:aop="http://www.springframework.org/schema/aop"

```

- [Tutorial - Diagnosis \(2\)](#)
- [Akka Tutorial \(9\)](#)
- [Core Java Tutorials \(2\)](#)
- [Hadoop & Spark Tuto](#)
- [JEE Tutorials \(19\)](#)
- [Scala Tutorials \(1\)](#)
- [Spring & Hibernate Ti](#)
- [Tools Tutorials \(19\)](#)
- [Other Tutorials \(45\)](#)

100+ Java pre-interview coding tests

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

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

How good are your?

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

- [Career Making Know-](#)
- [Job Hunting & Resum](#)

```

4     xmlns:xsi="http://www.w3.org/2001/XMLSchema-
5     xsi:schemaLocation="http://www.springframewo
6                           http://www.springframewo
7
8     <bean id="simpleRestWeb" class="com.mytutor
9 </beans>
10

```

Step 5: web.xml deployment descriptor

to bootstrap Spring and initialize servlet, mapping, etc.

Folder: src/main/webapp/WEB-INF.

```

1 <!DOCTYPE web-app PUBLIC
2   "-//Sun Microsystems, Inc.//DTD Web Application
3   "http://java.sun.com/dtd/web-app_2_3.dtd" >
4
5 <web-app>
6   <display-name>Archetype Created Web Applicat
7
8   <context-param>
9     <param-name>contextConfigLocation</param
10    <param-value>classpath:/com/mytutorial/a
11  </context-param>
12
13  <context-param>
14    <param-name>resteasy.use.deployment.sens
15    <param-value>false</param-value>
16  </context-param>
17
18  <context-param>
19    <param-name>resteasy.servlet.mapping.pre
20    <param-value>/rest</param-value>
21  </context-param>
22
23  <listener>
24    <listener-class>org.jboss.resteasy.plugi
25  </listener>
26  <listener>
27    <listener-class>org.jboss.resteasy.plugi
28  </listener>
29
30  <servlet>
31    <servlet-name>resteasy-simple</servlet-n
32    <servlet-class>org.jboss.resteasy.plugin
33  </servlet>
34
35  <servlet-mapping>
36    <servlet-name>resteasy-simple</servlet-n
37    <url-pattern>/rest/*</url-pattern>
38  </servlet-mapping>
39 </web-app>
40

```

Build the war and deploy to say JBoss application server.

Step 6: JBoss deployment descriptor

```
1 <jboss-web>
2   <context-root>tutorial</context-root>
3 </jboss-web>
4
```

5/8

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

766 views

18 Java scenarios based interview Questions and Answers

400 views

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

388 views

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

295 views

♦ 7 Java debugging interview questions & answers

293 views

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

285 views

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

279 views

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

239 views

001B: ♦ Java architecture & design concepts interview questions & answers

201 views

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 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](#) 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](#)

◀ 08: REST constraints (i.e. design rules) interview Q&A

♥ Debugging Java version issues ▶

Posted in WebService

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.