

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 › 11: ♥ JAX-WS how to create a SOAP web service in Java?

11: ♥ JAX-WS how to create a SOAP web service in Java?

Posted on June 18, 2015 by Arulkumaran Kumaraswamipillai



Step 1: Create the Java Maven project:

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

Step 2: Need the libraries/frameworks

This example shows Apache CXF library as JAX-WS implementation with Spring and deployed to JBoss server 5.1.1.

```
1 <project xmlns="http://maven.apache.org/POM/4.0.0"
2   xsi:schemaLocation="http://maven.apache.org/
3   <modelVersion>4.0.0</modelVersion>
4   <groupId>com.mytutorial</groupId>
5   <artifactId>simpleSoapWeb</artifactId>
```

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)

```

6   <packaging>war</packaging>
7   <version>1.0-SNAPSHOT</version>
8   <name>simpleSoapWeb Maven Webapp</name>
9   <url>http://maven.apache.org</url>
10  <properties>
11      <cxf.version>2.7.9</cxf.version>
12      <javax-servlet.version>3.0.1</javax-serv
13      <spring.version>3.1.0.RELEASE</spring.ve
14  </properties>
15
16  <dependencies>
17      <!-- JAX-WS -->
18      <!-- apache cxf -->
19      <dependency>
20          <groupId>org.apache.cxf</groupId>
21          <artifactId>cxf-rt-frontend-jaxws</a
22          <version>${cxf.version}</version>
23      </dependency>
24
25      <dependency>
26          <groupId>org.apache.cxf</groupId>
27          <artifactId>cxf-rt-transports-http</
28          <version>${cxf.version}</version>
29      </dependency>
30
31      <!-- Servlet API -->
32      <dependency>
33          <groupId>javax.servlet</groupId>
34          <artifactId>javax.servlet-api</artif
35          <scope>provided</scope>
36          <version>${javax-servlet.version}</v
37      </dependency>
38
39      <!-- Spring -->
40      <dependency>
41          <groupId>org.springframework</groupI
42          <artifactId>spring-web</artifactId>
43          <scope>compile</scope>
44          <version>${spring.version}</version>
45      </dependency>
46
47      <dependency>
48          <groupId>org.slf4j</groupId>
49          <artifactId>slf4j-api</artifactId>
50          <version>1.7.0</version>
51      </dependency>
52  </dependencies>
53  <build>
54      <finalName>simpleSoapWeb</finalName>
55  </build>
56 </project>
57

```

- [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](#)

Step 3: Web service interface with JAX-WS annotations

Folder: src/main/java

```

1 package com.mytutorial;
2
3 import javax.jws.WebMethod;

```

80+ step by step Java Tutorials

open all | close all

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

```

4 import javax.jws.WebService;
5
6 @WebService
7 public interface SimpleSoapWeb {
8
9     @WebMethod
10    public String sayHello(String name);
11 }

```

Step 4: Implementation of Web service interface

```

1 package com.mytutorial;
2
3 import javax.jws.WebService;
4
5 @WebService(endpointInterface = "com.mytutorial.
6 public class SimpleSoapWebImpl implements Simple
7
8     public String sayHello(String name) {
9         String result = "Hello " + name;
10        return result;
11    }
12 }
13

```

Step 5: 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/sch
3     xmlns:jaxws="http://cxf.apache.org/jaxws" xm
4     xsi:schemaLocation="http://www.springframework
5     http://www.springframework.org/schema/beans/
6     http://cxf.apache.org/jaxrs
7     http://cxf.apache.org/schemas/jaxrs.xsd
8     http://cxf.apache.org/jaxws
9     http://cxf.apache.org/schemas/jaxws.xsd">
10
11     <jaxws:endpoint id="simpleSoapService" implem
12
13 </beans>

```

Note down the “address”.

Step 6: web.xml deployment descriptor

to bootstrap Spring and initialize servlet, mapping, etc.

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

```

1 <!DOCTYPE web-app PUBLIC

```

- [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? \(](#)
- [♦ 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](#)

```

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     <listener>
13         <listener-class>
14             org.springframework.web.context.Cont
15         </listener-class>
16     </listener>
17
18     <servlet>
19         <servlet-name>CXFServlet</servlet-name>
20         <display-name>CXF Servlet</display-name>
21         <servlet-class>org.apache.cxf.transport.
22         <load-on-startup>2</load-on-startup>
23     </servlet>
24
25     <servlet-mapping>
26         <servlet-name>CXFServlet</servlet-name>
27         <url-pattern>/*</url-pattern>
28     </servlet-mapping>
29 </web-app>
30
31

```

Build the war and deploy to say JBoss application server.

Step 6: JBoss deployment descriptor

Folder: src/main/webapp/WEB-INF/jboss-web.xml

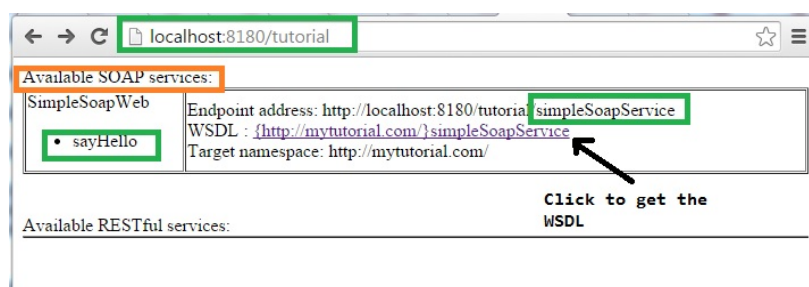
```

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

```

The URL = <http://localhost:8080/tutorial/>

tutorial: defined in jboss-web.xml (application context)



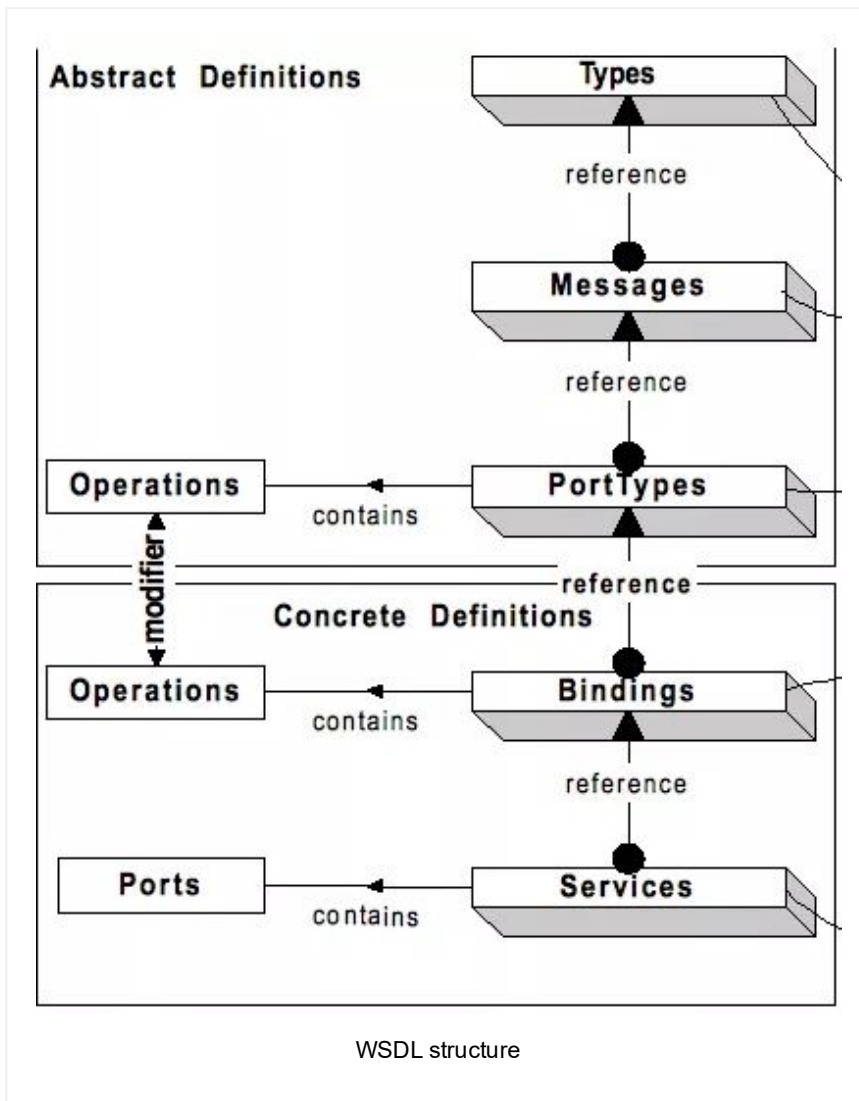
To get the WSDL

URL: <http://localhost:8180/tutorial/simpleSoapService?wsdl>

```
1 <?xml version='1.0' encoding='UTF-8'?><wsdl:defi
2 <wsdl:types>
3 <xs:schema xmlns:xs="http://www.w3.org/2001/
4 <xs:element name="sayHello" type="tns:sayH
5 <xs:element name="sayHelloResponse" type="
6
7 <xs:complexType name="sayHello">
8 <xs:sequence>
9 <xs:element minOccurs="0" name="arg
10 </xs:sequence>
11 </xs:complexType>
12
13 <xs:complexType name="sayHelloResponse">
14 <xs:sequence>
15 <xs:element minOccurs="0" name="retur
16 </xs:sequence>
17 </xs:complexType>
18 </xs:schema>
19 </wsdl:types>
20
21 <wsdl:message name="sayHelloResponse">
22 <wsdl:part element="tns:sayHelloResponse" na
23 </wsdl:part>
24 </wsdl:message>
25 <wsdl:message name="sayHello">
26 <wsdl:part element="tns:sayHello" name="para
27 </wsdl:part>
28 </wsdl:message>
29 <wsdl:portType name="SimpleSoapWeb">
30 <wsdl:operation name="sayHello">
31 <wsdl:input message="tns:sayHello" name="s
32 </wsdl:input>
33 <wsdl:output message="tns:sayHelloResponse
34 </wsdl:output>
35 </wsdl:operation>
36 </wsdl:portType>
37
38 <wsdl:binding name="simpleSoapServiceSoapBindi
39 <soap:binding style="document" transport="ht
40 <wsdl:operation name="sayHello">
41 <soap:operation soapAction="" style="docum
42 <wsdl:input name="sayHello">
43 <soap:body use="literal"/>
44 </wsdl:input>
45 <wsdl:output name="sayHelloResponse">
46 <soap:body use="literal"/>
47 </wsdl:output>
48 </wsdl:operation>
49 </wsdl:binding>
50
51 <wsdl:service name="simpleSoapService">
52 <wsdl:port binding="tns:simpleSoapServiceSoa
53 <soap:address location="http://localhost:8
```

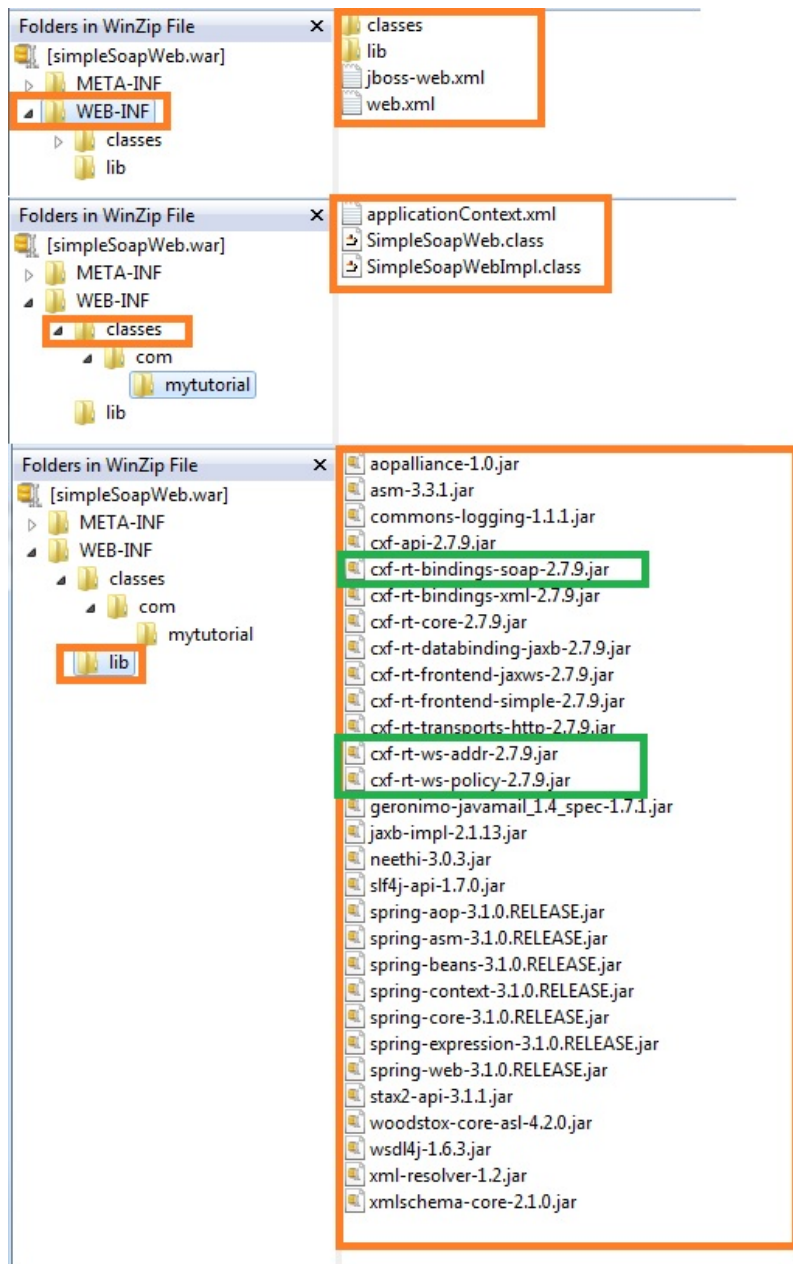
```
54 </wsdl:port>
55 </wsdl:service>
56 </wsdl:definitions>
```

WSDL structure



Project Structure





Deployable war file

Popular Posts

◆ 11 Spring boot interview questions & answers

825 views

◆ 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](#)

◀ ♥ Debugging Java version issues

◆ Singleton design pattern & 5 key follow up Q&A ▶

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.