Register | Login | Logout | Contact Us

Java-Success.com

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 › Pressed for time? Java/JEE Interview FAQs › FAQ Java Web Services Interview Q&A Essentials › 09: 11 SOAP Web service interview questions and answers

09: 11 SOAP Web service interview questions and answers

Posted on August 20, 2014 by Arulkumaran Kumaraswamipillai

1 Like Share Tweet 0 G+1 Share

SOAP Web Services interview Questions & Answers Links:

6 Java RESTful Web services Interview | 5 JAXB interview Questions & Answers | Java Web Services interview Questions & Answers

Q1. What are the different approaches to developing a SOAP based Web service?

A1. There are 2 approaches.

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

open all | close all

⊞ Ice Breaker Interview
⊞ Core Java Interview (
⊟ JEE Interview Q&A (3
  ⊞ JEE Overview (2)
  ⊞ Web basics (8)
  ⊟ WebService (11)
    01: ♥♦ 40+ Java
    02: ♦ 6 Java RES
    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 S
    11: ♥ JAX-WS hc
  ⊞ JPA (2)
  ⊞ JTA (1)
  ⊞ JDBC (4)
  ⊞ JMS (5)

1) The **contract-first approach**, where you define the contract first with XSD and WSDL and the generate the Java classes from the contract.
2) The **contract-last approach** where you define the Java classes first and then generate the contract, which is the WSDL file from the Java classes.

Note: The WSDL describes all operations that the service provides, locations of the endpoints (i.e.e where the services can be invoked), and simple and complex elements that can be passed in requests and responses.

Q2. What are the pros and cons of each approach, and which approach would you prefer?
A2.

Contract-first Web service

PROS:

a) Clients are decoupled from the server, hence the implementation logic can be revised on the server without affecting the clients.
b) Developers can work simultaneously on client and server side based on the contract both agreed on.
c) You have full control over how the request and response messages are constructed — for example, should "status" go as an element or as an attribute? The contract clearly defines it. You can change OXM (i.e. Object to XML Mapping) libraries without having to worry if the "status" would be generated as "attribute" instead of an element. Potentially, even Web service frameworks and tool kits can be changed as well from say Apache Axis to Apache CXF, etc

CONS:

a) More upfront work is involved in setting up the XSDs and WSDLs. There are tools like XML Spy, Oxygen XML, etc to make things easier. The object models need to be written as well.

⊞ 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 f
⊞ ▶ Free Java Interviev

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 I
⊞ Low Latency (7)
⊞ Memory Managemen
⊞ 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)

b) Developers need to learn XSDs and WSDLs in addition to just knowing Java.

Contract-last Web service

PROS:

a) Developers don't have to learn anything related to XSDs, WSDLs, and SOAP. The services are created quickly by exposing the existing service logic with frameworks/tool sets. For example, via IDE based wizards, etc.
b) The learning curve and development time can be smaller compared to the Contract-first Web service.

CONS:

a) The development time can be shorter to initially develop it, but what about the on going maintenance and extension time if the contract changes or new elements need to be added? In this approach, since the clients and servers are more tightly coupled, the future changes may break the client contract and affect all clients or require the services to be properly versioned and managed.
b) In this approach, The XML payloads cannot be controlled. This means changing your OXM libraries could cause something that used to be an element to become an attribute with the change of the OXM.

So, which approach will you choose? The best practice is to use "contract-first" as the contract-last can be more fragile. You will have to decide what is most appropriate based on your requirements, tool sets you use, etc.

Q3. A web service protocol stack from bottom to top consists of

a) HTTP, SOAP, description language, UDDI
b) SMTP, XML messaging, WSDL, Service discovery
c) HTTP, XML messaging, WSDL, UDDI
d) HTTP, XML-RPC, WSDL, UDDI
e) HTTP, WSDL, SOAP, UDDI

⊞ Tutorial - Diagnosis (2
⊞ Akka Tutorial (9)
⊞ Core Java Tutorials (2
⊞ Hadoop & Spark Tuto
⊞ JEE Tutorials (19)
⊞ Scala Tutorials (1)
⊞ Spring & HIbernate Tu
⊞ 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

A3. The answer is **a**,**b**,**c**, and **d**. The e is not right because of the order.

This is an evolving standard, but the basic Web service protocol stack is (aka web service components) comprised of

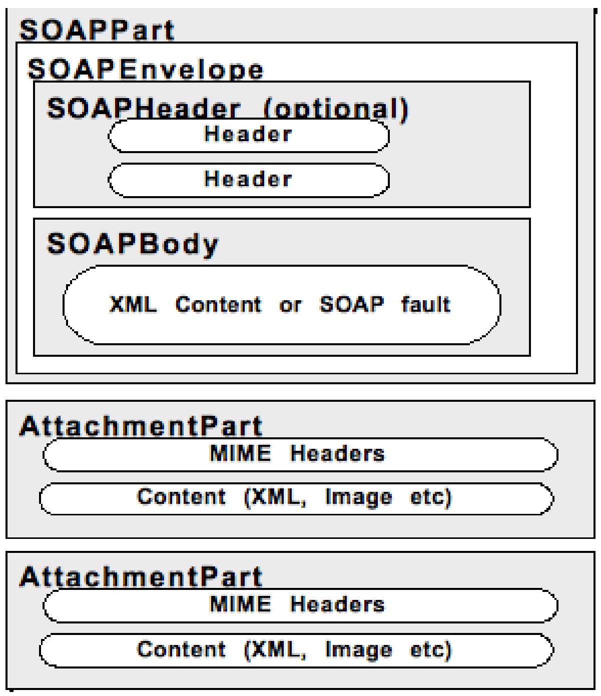
Service transport is the lowest layer in the stack, and is responsible for transporting messages between applications. Currently, this layer includes hypertext transfer protocol (HTTP), Simple Mail Transfer Protocol (SMTP), file transfer protocol (FTP), and newer protocols, such as Blocks Extensible Exchange Protocol (BEEP).

XML messaging layer is responsible for encoding messages in a common XML format so that messages can be understood at either end. Currently, this layer includes XML-RPC and SOAP.

Service description layer responsible for describing the public interface to a specific web service. Currently, service description is handled via the Web Service Description Language (WSDL or WADL[for RESTful]).

Service discovery layer is responsible for centralizing services into a common registry, and providing easy publish/find functionality. Currently, service discovery is handled via Universal Description, Discovery, and Integration (UDDI).

SOAP stands for Simple Object Access Protocol. It is an XML based lightweight protocol, which allows software components and application components to communicate, mostly using HTTP (can use SMTP etc). SOAP sits on top of the HTTP protocol. SOAP is nothing but XML message based document with predefined format. SOAP is designed to communicate via the Internet in a platform and language neutral manner and allows you to get around firewalls as well.



SOAP Message

SOAP Request:

```

POST /Price HTTP/1.1
Host: www.mysite.com
Content-Type: application/soap+xml; charset=utf-
Content-Length: 300

<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envel
soap:encodingStyle="http://www.w3.org/2001/12/so
  <soap:Body>
     <m:GetPrice xmlns:m="http://www.mysite.com/
        <m:Item>PlasmaTV</m:Item>
     </m:GetPrice>
  </soap:Body>
</soap:Envelope>


```

SOAP Response:

```

HTTP/1.1 200 OK
Content-Type: application/soap; charset=utf-8
Content-Length: 200

<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envel
```

```
soap:encodingStyle="http://www.w3.org/2001/12/so
  <soap:Body>
      <m:GetPriceResponse
                  xmlns:m="http://www.mysite.com
            <m:Price>3500.00</m:Price>
      </m:GetPriceResponse>
   </soap:Body>
</soap:Envelope>

```

Q4. What are the key roles played in a Web service

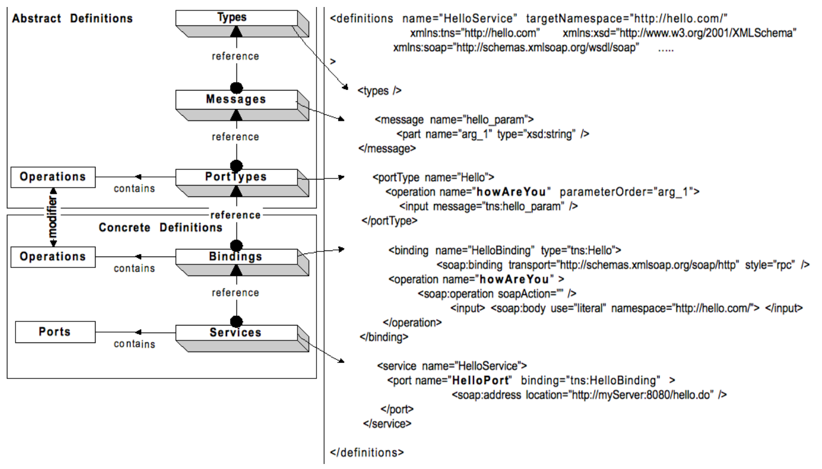
a) Service provider, Service requester, Service registry
b) producer, consumer, Service registry
c) producer, consumer
d) publish, bind, find

A4. . The answer is **a**,**b**,**c**, and **d**.

Q5. Which of the following are in the WSDL document structure?

a) types, message, port type, binding
b) input, output, binding, exception
c) types, input, output, exception
d) input, output, operations, binding

A5. The answer is **a**.



```
<definitions  name="HelloService"  targetNamespace="http://hello.com/"
              xmlns:tns="http://hello.com"      xmlns:xsd="http://www.w3.org/2001/XMLSchema"
           xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap"     .....
>

    <types />

       <message  name="hello_param">
           <part  name="arg_1"  type="xsd:string"  />
    </message>

      <portType  name="Hello">
        <operation name="howAreYou"  parameterOrder="arg_1">
          <input  message="tns:hello_param"  />
     </portType>

         <binding  name="HelloBinding"  type="tns:Hello">
                 <soap:binding  transport="http://schemas.xmlsoap.org/soap/http"  style="rpc"  />
         <operation name="howAreYou" >
              <soap:operation soapAction="" />
                   <input> <soap:body  use="literal"  namespace="http://hello.com/"> </input>
        </operation>
    </binding>

       <service  name="HelloService">
         <port name="HelloPort"  binding="tns:HelloBinding"  >
                    <soap:address  location="http://myServer:8080/hello.do"  />
        </port>
     </service>

</definitions>
```

WSDL

types: The data types used by the web service.

message: The input and output messages used by the web service.

port type: The operations performed by the web service. This is analogous to a class in Java programming. This also defines the input/output via messages. This is the most important part of the wsdl.

binding: The communication protocol used by the web service.

Q6. What are the different types of operations available in a WSDL?

a) One-way
b) Request-Response
c) Solicit-Response
d) Notification or fire and forget

A6. The answer is **a**, **b**, **c**, and **d**. Supports all 4 types of operations.

One-way: The operation (or endpoint) receives a message, but will not return a response.
Request-Response: The most common one. The operation (or endpoint) receives a request message and responds with a response message.
Solicit-Response: The operation (or endpoint) sends a request message and receives a correlated response message.
Notification or fire and forget: The operation (or endpoint) sends a request message, but will not wait for a response.

Q7. What is the difference between Request-Response and Solicit-Response?
A7. . Solicit-Response is a **push operation** like Notification, but waits for a response. Request-Response is a **pull operation**.

The only way to tell the difference between a request-response operation and a solicit-response operation is the ordering of the input and output elements. In request-response, the input child element comes first. In solicit-response, the output child element comes first.

Q8. Which of the following are true?

a) SOAP is a protocol and REST is a concept without any defined spec at all
b) SOAP is a XML-based message protocol, while REST is an architectural style
c) You can send SOAP envelopes in a REST application.
d) The REST verbs are “get”, “put”, “post” and “delete” and the nouns are identified by URLs.
e) SOAP allows many different verbs to be applied to many different nouns.

A8. The answer is **a**,**b**,**c**,**d**, and **e**. All are true.

a, b, and c are true because they state the fact that SOAP is an XML based message protocol and REST is a concept or architectural style.

d is true because RESTful url define the noun via the urls like

http://localhost:8080/accounting-services/1.0/forecasting/account/123/transaction/567
http://localhost:8080/accounting-services/1.0/forecasting/account/123/transactions/search?txn-date=20120201
http://localhost:8080/accounting-services/1.0/forecasting/account/123/transaction

e is true because you use different functions in SOAP port-type definition

Q9. Though both RESTful web series and SOAP web service can operate cross platform, they are architecturally different to each other. Which of the following statements are correct?

a) REST is more simple and easy to use than SOAP, hence currently more popular.
b) REST uses HTTP protocol for producing or consuming web services while SOAP uses XML.
c) REST is lightweight as compared to SOAP and preferred choice in mobile devices and PDA's.
d) REST supports different format like text, JSON and XML while SOAP only support XML.
e) REST web services call can be cached to improve performance.
f) SOAP provides more comprehensive security and transaction management.

A9. All are correct.

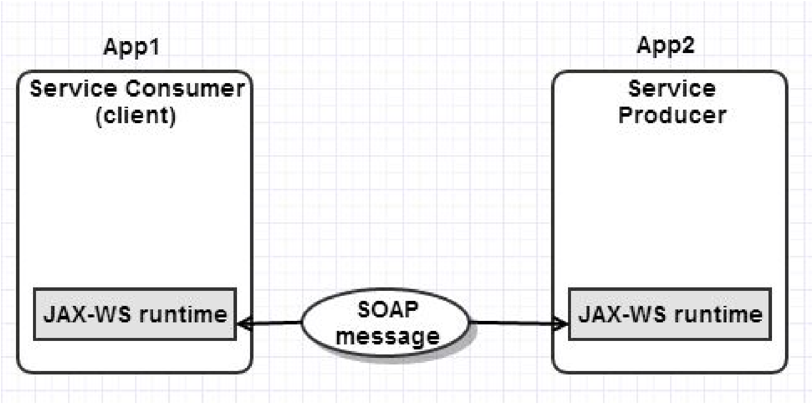
Q10. Which of the following statements are correct?

a) JAX-WS is an API for SOAP based web service.
b) JAX-RS is an API for RESTFul web service.
c) SOAP invokes services by calling RPC method, REST just simply calls services via URL path.
d) Apache CXF framework only supports JAX-WS
e) Jersey and RESTEasy are reference implementations of JAX-RS.

A10. a, b, c, and e are correct. d is incorrect because Apache CXF supports both JAX-WS and JAX-RS.

Q11. What is JAX-WS and why does it replace JAX-RPC?
A11.**JAX-WS** stands for Java Apifor Xml Web Services, which contains a set of APIs for creating web services in XML format (SOAP). JAX-WS provides many annotation to simplify the development and deployment for both web service consumers (i.e. clients) and web service providers (i.e. endpoints).

SOAP has been there for a while now, and its history goes like — firstly there was SOAP. But SOAP only described what the messages looked like. Then there was WSDL. But WSDL didn't tell you how to write web services in Java™. Then along came JAX-RPC 1.0. Since the industry was using message-oriented web services, “RPC” was removed from the name and replaced with “WS”, so JAX-RPC became JAX-WS. JAX-WS defines a standard Java-to-WSDL mapping, which determines which Java method gets invoked and how that SOAP message is mapped to the method’s parameters.



JAX-WS

JAX-WS uses a number of annotations like @WebService and @WebMethod.

```java

import javax.jws.WebMethod;
import javax.jws.WebService;

@WebService
public interface Greeting {
   @WebMethod String sayHello(String name);
}

```

```java

import javax.jws.WebService;

@WebService(endpointInterface = "Greeting")
public class GreetingImpl implements Greeting {

    @Override
    public String sayHello(String name) {
        return "Hello: " + name;
    }
}

```

```java

import javax.xml.ws.Endpoint;

```

```
public class WSProvider {
    public static void main(String[] args) {
        Endpoint.publish("http://localhost:8080/W
    }
}

```

The WSDL can be found at http://localhost:8080/WS/Greeting?wsdl, and this is contract last approach.

To create the consumer, you can use wsimport on the client project folder

```

wsimport -s . http://localhost:8080/WS/Greeting?w

```

Which creates a few classes.

```

public class WSConsumer {
    public static void main(String[] args){
        GreetingImplService service = new Greetin
        Greeting greeting = service.getGreetingIm
        System.out.println(greeting.sayHello("Joh
    }
}

```

Q. What is the difference between JAX-WS and JAX-RS?
A. JAX-WS represents SOAP and JAX-RS represents REST

SOAP Web Services interview Questions & Answers Links:

6 Java RESTful Web services Interview | 5 JAXB interview Questions & Answers | Java Web Services interview Questions & Answers

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**

‹ 01: ♥♦ 40+ Java Web Services Architecture & Basics Interview Q&A – Q01 – Q12

02: ♦ 6 Java RESTful Web services Interview Q&A you must know ›

Posted in FAQ Java Web Services Interview Q&A Essentials, member-paid, WebService

Tags: Java/JEE FAQs, JEE FAQs

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.

↑