Java67

Java Programming tutorials and Interview Questions, book and course recommendations from Udemy, Pluarlsight etc

Searc

Home | core java | thread | java 8 | array | coding | string | sql | books | j2ee | oop | collections | data structure

Top 10 RESTful Web Service Interview Questions for Java Developers

REST is an architectural style of developing web services which have become immensely popular in the last couple of years and consequently gained a lot of importance in core Java and Java EE interviews. If you are a Java web developer then you are most likely see a couple of questions from web services every time you go for a Java web developer interview. One of the most frequent one is difference between REST and SOAP web services, which I have recently answered there, but there are lot many other good questions I have collected from friends and my readers, which I have not yet published. In this article, I am sharing those questions, mainly based on REST-style web services for your practice and preparation.

Some of them are easy to answer and you will find them either in my blog or by doing Google but a couple of them is really interesting and challenging and required real solid experience in Java web service domain to answer. I leave it to you guys for now and will probably update this post with answers in near future.

If you don't find answer of any REST interview question then I suggest you to take a look at these two resources, Java Programming Interview Exposed and RESTful Web Services, you will most likely find your answer there.

RESTful Web Services Interview Questions

Here is my list of RESTful web services questions for senior Java developers who have a couple of years of experience developing both SOAP and REST based web services. This is actually the second part of my series of Java web services based question, in the earlier article, I have shared some SOAP web services based questions. If you have not looked already, you may want to take a look.

Question 1 : What is REST?
Answer : REST is an architectural style of developing web services which take advantage of ubiquity of HTTP protocol and leverages HTTP method to define actions. REST stands for *REpresntational State Transfer*.

Question 2 : What is RESTFul Web Service?
Answer : There are two popular way to develop web services, using SOAP (Simple Object Access Protocol) which is XML based way to expose web services and second REST based web services which uses HTTP protocol. Web services developed using REST style is also known as RESTful Web Services.

Question 3 : What is HTTP Basic Authentication and how it works?

Question 4 : Can you tell me which API can be used to develop RESTFul web service in Java?
Answer : There are many framework and libraries out there which helps to develop RESTful web services in Java including JAX-RS which is standard way to develop REST web services. Jersey is one of the popular implementation of JAX-RS which also offers more than specification recommends. Then

you also have RESTEasy, RESTlet and Apache CFX. If you like Scala then you can also use Play framework to develop RESTful web services.

Question 5 : How do you configure RESTFul web service?

Question 6 : How you apply security in RESTFul web services?

Question 7 : Have you used securing RESTful APIs with HTTP Basic Authentication

Question 8 : How you maintain session in RESTful services?

Question 9 : Have you used Jersey API to develop RESTful services in Java?
Answer : Jersey is one of the most popular framework and API to develop REST based web services in Java. Since many organization uses Jersey they check if candidate has used it before or not. It's simple to answer, say Yes if you have really used and No, if you have not. In case of No, you should also mention which framework you have used for developing RESTful web services e.g. Apache CFX, Play or Restlet.

Question 10 : How you test RESTful web services?

Question 11 : What is WADL in RESTFul?

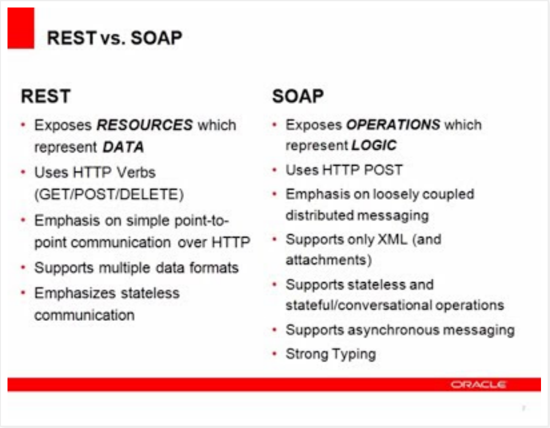
Question 12 : What do you understand by payload in RESTFul?
Answer : Payload means data which passed inside request body also payload is not request parameters. So only you can do payload in POST and not in GET and DELTE method

Question 13 : Can you do payload in GET method?
Answer : No, payload can only be passed using POST method.

Question 14 : Can you do payload in HTTP DELETE?
Answer : This is again similar to previous REST interview question, answer is No. You can only pass payload using HTTP POST method.

Question 15 : How much maximum pay load you could do in POST method?
Answer : If you remember difference between GET and POST request then you know that unlike GET which passes data on URL and thus limited by maximum URL length, POST has no such limit. So, theoretically you can pass unlimited data as payload to POST method but you need to take practical things into account e.g. sending POST with large payload will consume more bandwidth, take more time and present performance challenge to your server.

Question 16 : What is difference between SOAP and RESTFul web services?
Answer : There are many difference between these two style of web services e.g. SOAP take more bandwidth because of heavy weight XML based protocol but REST takes less bandwidth because of popular use of JSON as message protocol and leveraging HTTP method to define action. This also means that REST is faster than SOAP based web services. You can derive many differences between SOAP and RESTful with the fact that its HTTP based e.g. REST URLs can be cached or bookmarked. Here are few more differences between them :



REST vs. SOAP

REST

- Exposes ***RESOURCES*** which represent ***DATA***
- Uses HTTP Verbs (GET/POST/DELETE)
- Emphasis on simple point-to-point communication over HTTP
- Supports multiple data formats
- Emphasizes stateless communication

SOAP

- Exposes ***OPERATIONS*** which represent ***LOGIC***
- Uses HTTP POST
- Emphasis on loosely coupled distributed messaging
- Supports only XML (and attachments)
- Supports stateless and stateful/conversational operations
- Supports asynchronous messaging
- Strong Typing

Question 17 : If you have to develop web services which one you will choose SOAP OR RESTful and why?
Answer : You can answer this question based upon your experience but the key here is if you know difference between them than you can answer this question in more detail. For example, its easy to develop RESTful web services than SOAP based web services but later comes with some in-built security features.

Question 18 : What framework you had used to develop RESTFul services?
Answer : This is really experience based question. If you have used Jersey to develop RESTFul web services then answer as Jersey but expect some follow-up question on Jersey. Similarly if you have used Apache CFX or Restlet then answer them accordingly.

That's all in this list of some good **RESTful web service interview questions for Java developers**. Though this list is meant for Java developer, you can use this questions to check any candidate's knowledge on REST style web services independent of programming language because REST doesn't say that you need to implement web service in Java only. Since it take advantage of ubiquitous HTTP protocol you can build backed with any web technology stack e.g. `Java`, `.NET` or any other.

Further Learning
REST Java Web Services
REST API Automation testing from scratch-(REST Assured java)
RESTFul Services in Java using Jersey By Bryan Hansen

P.S. - If you are looking for online training to learn how to develop RESTful Web Services in Java using Spring framework, I suggest you joining Eugen Paraschiv's REST with Spring course. The course has various options depending upon your experience level and how much you want to learn e.g. beginner's class, intermediate class, and master class. You can join the one which suits you better, though I suggest joining the master class if you are serious about becoming an expert Java REST developer.

Posted by Javin Paul G+

Labels: interview questions, REST tutorials, Web Service

21 comments:

mohan rathour January 6, 2016 at 6:05 AM

Question 3 : What is HTTP Basic Authentication and how it works?

The server sends back a header stating it requires authentication for a given realm. The user provides the username and password, which the browser concatenates (username + ":" + password), and base64 encodes. This encoded string is then sent using a "Authorization"-header on each request from the browser. Because the credentials are only encoded, not encrypted, this is highly insecure unless it is sent over https.