Web Services - Interview Questions

Dear readers, these **Web Services Interview Questions** have been designed specially to get you acquainted with the nature of questions you may encounter during your interview for the subject of **Web Services**. As per my experience good interviewers hardly plan to ask any particular question during your interview, normally questions start with some basic concept of the subject and later they continue based on further discussion and what you answer −

What are web services?

Web services are open standard (XML, SOAP, HTTP etc.) based Web applications that interact with other web applications for the purpose of exchanging data. Web Services can convert your existing applications into Web-applications.

What are the features of web services?

Following are the features of Web service −

> It is available over the Internet or private (intranet) networks.

> It uses a standardized XML messaging system.

> It is not tied to any one operating system or programming language.

> It is self-describing via a common XML grammar.

> It is discoverable via a simple find mechanism.

What the components of a Web Service?

The basic web services platform is XML + HTTP. All the standard web services work using the following components −

SOAP (Simple Object Access Protocol)

UDDI (Universal Description, Discovery and Integration)

WSDL (Web Services Description Language)

How Does a Web Service Work?

A web service enables communication among various applications by using open standards such as HTML, XML, WSDL, and SOAP.

You can build a Java-based web service on Solaris that is accessible from your Visual Basic program that runs on Windows.

You can also use C# to build new web services on Windows that can be invoked from your web application that is based on JavaServer Pages (JSP) and runs on Linux.

What is the purpose of XML in a web service?

A web services takes the help of XML to tag the data, format the data.

What is the purpose of SOAP in a web service?

A web service takes the help of SOAP to transfer a message.

What is the purpose of WSDL in a web service?

A web service takes the help of WSDL to describe the availability of service.

What are the benefits of Web Services?

Following are the benefits of using web services −

Exposing the Existing Function on the network − Web services allows you to expose the functionality of your existing code over the network. Once it is exposed on the network, other application can use the functionality of your program.

Interoperability − Web services allow various applications to talk to each other and share data and services among themselves.

Standardized Protocol − Web services use standardized industry standard protocol for the communication. All the four layers (Service Transport, XML Messaging, Service Description, and Service Discovery layers) use well-defined protocols in the web services protocol stack.

Low Cost of Communication − Web services use SOAP over HTTP protocol, so you can use your existing low-cost internet for implementing web services.

What do you mean by Interoperability of Web Services?

Web services allow various applications to talk to each other and share data and services among themselves. Other applications can also use the web services. For example, a VB or .NET application can talk to Java web services and vice versa. Web services are used to make the application platform and technology independent.

What do you mean by loosely coupled architecture of Web services?

A consumer of a web service is not tied to that web service directly. The web service interface can change over time without compromising the client's ability to interact with the service. A tightly coupled system implies that the client and server logic are closely tied to one another, implying that if one interface changes, the other must be updated. Adopting a loosely coupled architecture tends to make software systems more manageable and allows simpler integration between different systems.

Do Web services supports Remote Procedure Calls(RPCs)?

Web services allow clients to invoke procedures, functions, and methods on remote objects using an XML-based protocol. Remote procedures expose input and output parameters that a web service must support.

Component development through Enterprise JavaBeans (EJBs) and .NET Components has increasingly become a part of architectures and enterprise deployments over the past couple of years. Both technologies are distributed and accessible through a variety of RPC mechanisms.

A web service supports RPC by providing services of its own, equivalent to those of a traditional component, or by translating incoming invocations into an invocation of an EJB or a .NET component.

What are the behavioral characteristics of web services?

Web services have the following special behavioral characteristics –

XML-Based – Web Services uses XML at data representation and data transportation layers.

Loosely Coupled – A consumer of a web service is not tied to that web service directly.

Coarse-Grained – Businesses and the interfaces that they expose should be coarse-grained. Web services technology provides a natural way of defining coarse-grained services that access the right amount of business logic.

Ability to be Synchronous or Asynchronous − Asynchronous clients retrieve their result at a later point in time, while synchronous clients receive their result when the service has completed. Asynchronous capability is a key factor in enabling loosely coupled systems.

Supports Remote Procedure Calls(RPCs) − A web service supports RPC by providing services of its own, equivalent to those of a traditional component, or by translating incoming invocations into an invocation of an EJB or a .NET component.

Supports Document Exchange − Web services support the transparent exchange of documents to facilitate business integration.

What are the benefits of having XML based WEB services?

Using XML eliminates any networking, operating system, or platform binding. So Web Services based applications are highly interoperable application at their core level.

What is the benefit of a Web services being loosely coupled?

The web service interface can change over time without compromising the client's ability to interact with the service. A tightly coupled system implies that the client and server logic are closely tied to one another, implying that if one interface changes, the other must be updated. Adopting a loosely coupled architecture tends to make software systems more manageable and allows simpler integration between different systems.

What is Synchronicity?

Synchronicity refers to the binding of the client to the execution of the service. In synchronous invocations, the client blocks and waits for the service to complete its operation before continuing. Asynchronous operations allow a client to invoke a service and then execute other functions.

What are the core Roles in Web Service architecture?

There are three major roles within the web service architecture −

Service Provider

Service Requestor

Service Registry

What is the purpose of Service Provider in Web Service architecture?

This is the provider of the web service. The service provider implements the service and makes it available on the Internet.

What is the purpose of Service Requestor in Web Service architecture? ▾

This is any consumer of the web service. The requestor utilizes an existing web service by opening a network connection and sending an XML request.

What is the purpose of Service Registry in Web Service architecture? ▾

This is a logically centralized directory of services. The registry provides a central place where developers can publish new services or find existing ones. It therefore serves as a centralized clearing house for companies and their services.

What are the core layers in Web Service Protocol Stack? ▾

The Web Service Protocol Stack is still evolving, but currently has four main layers −

> Service Transport
>
> XML Messaging
>
> Service Description
>
> Service Discovery

What is the purpose of Service Transport layer in Web Service Protocol Stack? ▾

This layer is responsible for transporting messages between applications. Currently, this layer includes Hyper Text Transport Protocol (HTTP), Simple Mail Transfer Protocol (SMTP), File Transfer Protocol (FTP), and newer protocols such as Blocks Extensible Exchange Protocol (BEEP).

What is the purpose of XML Messaging layer in Web Service Protocol Stack? ▾

This 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.

What is the purpose of Service Description layer in Web Service Protocol Stack? ▾

A. This layer is responsible for describing the public interface to a specific web service. Currently, service description is handled via the Web Service Description Language (WSDL).

What is the purpose of Service Discovery layer in Web Service Protocol Stack? ▾

This 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).

What HTTP stands for?

HTTP stands for Hyper Text Transfer Protocol.

What is HTTP?

Currently, HTTP is the most popular option for service transport. HTTP is simple, stable, and widely deployed. Furthermore, most firewalls allow HTTP traffic. This allows XML-RPC or SOAP messages to masquerade as HTTP messages.

What BEEP stands for?

BEEP stands for Blocks Extensible Exchange Protocol.

What is BEEP?

This is a promising alternative to HTTP. BEEP is a new Internet Engineering Task Force (IETF) framework for building new protocols. BEEP is layered directly on TCP and includes a number of built-in features, including an initial handshake protocol, authentication, security, and error handling. Using BEEP, one can create new protocols for a variety of applications, including instant messaging, file transfer, content syndication, and network management.

What is XML-RPC?

XML-RPC is a simple protocol that uses XML messages to perform RPCs.

How request is sent in XML-RPC?

Requests are encoded in XML and sent via HTTP POST.

How response is sent in XML-RPC?

XML responses are embedded in the body of the HTTP response.

What are the features of XML-RPC?

Following are the features of XML-RPC −

XML-RPC is a simple protocol that uses XML messages to perform RPCs.

Requests are encoded in XML and sent via HTTP POST.

XML responses are embedded in the body of the HTTP response.

XML-RPC is platform-independent.

XML-RPC allows diverse applications to communicate.

A Java client can speak XML-RPC to a Perl server.

XML-RPC is the easiest way to get started with web services.

What SOAP stands for?

SOAP stands for Simple Access Object Protocol.

What is SOAP?

SOAP is an XML-based protocol for exchanging information between computers.

What are the features of SOAP?

Following are the features of SOAP −

SOAP is a communication protocol.

SOAP is for communication between applications.

SOAP is a format for sending messages.

SOAP is designed to communicate via Internet.

SOAP is platform independent.

SOAP is language independent.

SOAP is simple and extensible.

SOAP allows you to get around firewalls.

SOAP will be developed as a W3C standard.

Is SOAP platform independent?

Yes!

What WSDL stands for?

WSDL stands for Web Services Description Language.

What is WSDL?

WSDL is an XML-based language for describing web services and how to access them.

What are the features of WSDL?

Following are the features of WSDL −

WSDL was developed jointly by Microsoft and IBM.

WSDL is an XML based protocol for information exchange in decentralized and distributed environments.

WSDL is the standard format for describing a web service.

WSDL definition describes how to access a web service and what operations it will perform.

WSDL is a language for describing how to interface with XML-based services.

WSDL is an integral part of UDDI, an XML-based worldwide business registry.

WSDL is the language that UDDI uses.

WSDL is pronounced as 'wiz-dull' and spelled out as 'W-S-D-L'.

What UDDI stands for?

UDDI stands for Universal Description, Discovery, and Integration.

What is UDDI?

UDDI is an XML-based standard for describing, publishing, and finding web services.

What are the features of UDDI?

Following are the features of UDDI −

UDDI is a specification for a distributed registry of web services.

UDDI is platform independent, open framework.

UDDI can communicate via SOAP, CORBA, and Java RMI Protocol.

UDDI uses WSDL to describe interfaces to web services.

UDDI is seen with SOAP and WSDL as one of the three foundation standards of web services.

UDDI is an open industry initiative enabling businesses to discover each other and define how they interact over the Internet.

What are the primary security issues with web services?

There are three specific security issues with web services –

> Confidentiality
>
> Authentication
>
> Network Security

Which component of Web service describes interfaces to web services?

UDDI describes interfaces to web services.

Which language UDDI uses?

WSDL is the language that UDDI uses.

Is XML-RPC is platform-dependent?

No! XML-RPC is platform-independent.

If a client sends an XML request to a server, can we ensure that the communication remains confidential?

Yes! As XML-RPC and SOAP run primarily on top of HTTP and HTTP has support for Secure Socketes Layer (SSL). Communication can be encrypted via SSL.

If a client connects to a web service, how do we identify the user? Is the user authorized to use the service?

The following options can be considered but there is no clear consensus on a strong authentication scheme.

HTTP includes built-in support for Basic and Digest authentication, and services can therefore be protected in much the same manner as HTML documents are currently protected.

SOAP Digital Signature (SOAP-DSIG) leverages public key cryptography to digitally sign SOAP messages. It enables the client or server to validate the identity of the other party. Check it at http://www.w3.org/TR/SOAP-dsig.

The Organization for the Advancement of Structured Information Standards (OASIS) is working on the Security Assertion Markup Language (SAML).

What do you mean by Web services manageability?

Web services manageability is defined as a set of capabilities for discovering the existence, availability, health, performance, usage, as well as the control and configuration of a web service within the web services architecture. As web services become pervasive and critical to business operations, the task of managing and implementing them is imperative to the success of business operations.

How to handle Network security threats in Web services? ▾

There are two possible solutions −

Filter out all HTTP POST requests that set their content type to text/xml.

Another alternative is to filter the SOAPAction HTTP header attribute.

What is Next ?

Further you can go through your past assignments you have done with the subject and make sure you are able to speak confidently on them. If you are fresher then interviewer does not expect you will answer very complex questions, rather you have to make your basics concepts very strong.

Second it really doesn't matter much if you could not answer few questions but it matters that whatever you answered, you must have answered with confidence. So just feel confident during your interview. We at tutorialspoint wish you best luck to have a good interviewer and all the very best for your future endeavor. Cheers :-)

Enter email for newsletter   go