Web Service as a Distributed Application

Chinmay Kunkikar (CK821)

Showing **31** of **31** Questions

Question 1

1/1 Points

Distributed Systems

1. distributed systems which though loosely couple were difficult to write since it involved binary-based communication
2. distributed system that enabled easier way to write distributed applications. It enabled richer data types, was loosely coupled, platform neutral, language neutral. it used text-based communication

DCE/RPC - distributed systems which though loosely couple were difficult to write since it involved binary-based communication
RPC/XML - distributed system that enabled easier way to write distributed applications. It enabled richer data types, was loosely coupled, platform neutral, language neutral. it used text-based communication

Question 2

3/3 Points

RPC/XML based Distributed system

1. means client and server can each run on same or different underlying operating system
2. means client and server could be written in different language and still able to connect
3. means client and server were independent of each other and still could connect to each other at run time

Platform neutral - means client and server can each run on same or different underlying operating system
Language neutral - means client and server could be written in different language and still able to connect
Loosely coupled - means client and server were independent of each other and still could connect to each other at run time

Question 3

1/1 Points

RPC versus Document

1. string, integer, Boolean
2. class object, list, map

Basic data type - string, integer, Boolean
Rich data type - class object, list, map

Question 4

1/1 Points

Web services implement xml/rpc model over http using soap

- (●) A. True
- ( ) B. False

Question 5

1/1 Points

Web Service is a kind of webified application delivered over globally accepted http

- (●) A. True
- ( ) B. False

Question 6

3/3 Points

Advantages of web services

1. Web Services Use Open Industry Standards Like Http, Xml, Json Which Are Used Everywhere. It Can Use The Available Web Servers, Underlying Security, Database Systems With No Need For Any Specific Infrastructure For Its Purpose
2. Web Services Can Interact With Clients Running On Any Operating System Platform And Written In Any Language. Example A Java, Perl, Ruby , C# Clients Can Connect To A Web Service
3. Web Services Are Created Using Smaller Software Parts That Can Come Together To Enable A Fully Functional System. An Inventory Tracking Service Can Work Closely With Order Services.

 OPEN INFRASTRUCTURE - Web Services Use Open Industry Standards Like Http, Xml, Json Which Are Used Everywhere. It Can Use The Available Web Servers, Underlying Security, Database Systems With No Need For Any Specific Infrastructure For Its Purpose
 PLATFORM AND LANGUAGE TRANSPARENCY - Web Services Can Interact With Clients Running On Any Operating System Platform And Written In Any Language. Example A Java, Perl, Ruby , C# Clients Can Connect To A Web Service
 MODULAR DESIGN - Web Services Are Created Using Smaller Software Parts That Can Come Together To Enable A Fully Functional System. An Inventory Tracking Service Can Work Closely With Order Services.

Question 7

1/1 Points

interface and implementation

1. is a class that declares the methods such as a web methods
2. is a class that contains the method implementation of methods defined by the interface

Interface - is a class that declares the methods such as a web methods
Implementation - is a class that contains the method implementation of methods defined by the interface

Question 8

1/1 Points

interface and implementation

1. Interface
2. Implementation

8. SEI (Service Endpoint interface) - Interface
SIB (Service Implementation Bean) - Implementation

Question 9

1/1 Points

Clients access the web service through the service end point interface

- (●) A. True
- ( ) B. False

Question 10

1/1 Points

Fill in blank

JAX – WS ACRONYM STANDS FOR JAVA API FOR XML BASED WEB SERVICES

Question 11

8/8 Points

Annotations

1. marks the class as a service
2. marks the method as a web service operation
3. indicate whether service supports richer data type (DOCUMENT style) or simple data type (RPC style)
4. denotes the Web service operation will receive the request input but not send any response output
5. helps provide custom names to the parameters of the web service operation
6. tie the web service to externally defined handler chain file
7. marks the class as the root element of the XML Document
8. indicates to create XML Scheme type from underlying Java type but it also indicates that its an embedded encapsulated XML in some root XML Document

@WebService - marks the class as a service
@WebMethod - marks the method as a web service operation
@SOAPBinding - indicate whether service supports richer data type (DOCUMENT style) or simple data type (RPC style)
@OneWay - denotes the Web service operation will receive the request input but not send any response output
@WebParam - helps provide custom names to the parameters of the web service operation
@HandlerChain - tie the web service to externally defined handler chain file
@XmlRootElement - marks the class as the root element of the XML Document
@XmlType - indicates to create XML Scheme type from underlying Java type but it also indicates that its an embedded encapsulated XML in some root XML Document

Question 12

1/1 Points

Endpoint class is used to for clients to connect to service. Endpoint publishes the interface via a IP address and a port

- (●) A. True
- ( ) B. False

Question 13

1/1 Points

SOAP structure consists of outer envelope with inner header part and inner body part. It may additionally contain separate attachments

- (●) A. True
- ( ) B. False

Question 14

1/1 Points

If a web service is written in Java, then one can write a Perl client or Ruby client or Java client to connect to it

- (●) A. True
- ( ) B. False

Question 15

4/4 Points

Message Exchange pattern describes how messages are exchanged between client and server in a distributed architecture

1. The client sends a one or more request and the server returns a response for every request
2. The server sends a request and expects the client to give exactly one response only. No further request and response are exchanged. Example sender application asks for subscription confirmation and the receiver responds yes or no to subscription.
3. the client sends a request to server without expecting a response from server. Example client sends a one way request to the server to update the database.
4. The server sends a one time response without any client request. Example The server sends one sms per client.

Request-Response - The client sends a one or more request and the server returns a response for every request
Solicit-Response - The server sends a request and expects the client to give exactly one response only. No further request and response are exchanged. Example sender application asks for subscription confirmation and the receiver responds yes or no to subscription.
One-way - the client sends a request to server without expecting a response from server. Example client sends a one way request to the server to update the database.
Notification - The server sends a one time response without any client request. Example The server sends one sms per client.

Question 16

3/3 Points

Web service uses the underlying infrastructure elements to successfully manage any messaging exchange pattern

1. Web services uses it to describe information to connect to the service and mention the methods and bindings supported by it.
2. Web Services Uses Http Protocol And Is Transport-Neutral
3. Web Services Uses A Shared Typed System Called Xml Schema Type To Share Data All Languages Can Convert To Xml Schema Type And Thereby Enable Web Services To Be Language And Platform Neutral Example. A Java Data Type Named byte Can Be Converted To xsd:byte

SERVICE CONTRACT - Web services uses it to describe information to connect to the service and mention the methods and bindings supported by it.
MESSAGE TRANSPORT - Web Services Uses Http Protocol And Is Transport-Neutral
TYPE SYSTEM - Web Services Uses A Shared Typed System Called Xml Schema Type To Share Data All Languages Can Convert To Xml Schema Type And Thereby Enable Web Services To Be Language And Platform Neutral Example. A Java Data Type Named byte Can Be Converted To xsd:byte

Question 17

1/1 Points

Fill in blank

WSDL ACRONYM STANDS FOR WEB SERVICE DEFINITION LANGUAGE

Question 18

1/1 Points

WSDL file contains the service contract which describes the information to connect a service and the methods supported by it.

- (●) A. True
- ( ) B. False

Question 19

5/5 Points

WSDL structure contains 5 sections

1. DESCRIBES RICH DATA TYPE USING XSD FILES
2. SPECIFIES MESSAGES EXCHANGED AND INDICATES MESSAGE EXCHANGE PATTERN IN DIRECTION PROPERTY
3. SPECIFIES WHETHER SIMPLE DATA TYPE IS USED OR RICH DATA TYPE, TRANSPORT MODE (soap/http), AND WHETHER DATA IS SENT LITERAL(Y) OR ENCODED
4. SPECIFIES OPERATIONS/METHODS/INTERFACE SUPPORTED BY SERVICE AND THE MESSAGE THAT MANAGES IT INPUT/OUTPUT DESCRIBES WHETHER IT'S A MESSAGE INPUT TO THE SERVER OR MESSAGE OUTPUT FROM THE SERVER
5. DESCRIBE ONE OR MORE SERVICE ENDPOINTS THAT SERVICE EXPOSES EG. HTTP://LOCALHOST:9876/TS/ AND THE CORRESPONDING BINDING

TYPES - DESCRIBES RICH DATA TYPE USING XSD FILES
MESSAGE - SPECIFIES MESSAGES EXCHANGED AND INDICATES MESSAGE EXCHANGE PATTERN IN DIRECTION PROPERTY
BINDING - SPECIFIES WHETHER SIMPLE DATA TYPE IS USED OR RICH DATA TYPE, TRANSPORT MODE (soap/http), AND WHETHER DATA IS SENT LITERAL(Y) OR ENCODED
PORTTYPE - SPECIFIES OPERATIONS/METHODS/INTERFACE SUPPORTED BY SERVICE AND THE MESSAGE THAT MANAGES IT INPUT/OUTPUT DESCRIBES WHETHER IT'S A MESSAGE INPUT TO THE SERVER OR MESSAGE OUTPUT FROM THE SERVER
SERVICE - DESCRIBE ONE OR MORE SERVICE ENDPOINTS THAT SERVICE EXPOSES EG. HTTP://LOCALHOST:9876/TS/ AND THE CORRESPONDING BINDING

Question 20

1/1 Points

JAX-WS Provides Java Libraries To Implement Soap Based Web Services In RPC-Style Or DOCUMENT Style

- (●) A. True
- ( ) B. False

Question 21

1/1 Points

rpc versus document

1. SUPPORTS BASIC DATA TYPES LIKE INT, BOOLEAN, CHAR ETC. IN SOAP MESSAGES
2. SUPPORTS RICHER DATA TYPES LIKE LIST OF OBJECTS ETC. IN SOAP MESSAGES

RPC STYLE - SUPPORTS BASIC DATA TYPES LIKE INT, BOOLEAN, CHAR ETC. IN SOAP MESSAGES
DOCUMENT STYLE - SUPPORTS RICHER DATA TYPES LIKE LIST OF OBJECTS ETC. IN SOAP MESSAGES

Question 22

1/1 Points

WSDL and XSD

1. DEFINES THE WEB SERVICE
2. DEFINES THE XML DATA SCHEMA

WSDL - DEFINES THE WEB SERVICE
XSD - DEFINES THE XML DATA SCHEMA

Question 23

1/1 Points

WRAPPED DOCUMENT STYLE ENABLES OPERATION TO BE NAMED WITHIN SOAP MESSAGES THEREBY LOOKING LIKE RPC

- (●) A. True
- ( ) B. False

Question 24

3/3 Points

JAVA Libraries

1. is a JAVA library that supports SOAP Web services
2. JAVA library that supports REST Web Service
3. is a Java library that helps to convert rich data type to XML data type and vice versa

JAX-WS - is a JAVA library that supports SOAP Web services
JAX-RS - JAVA library that supports REST Web Service
JAX-B - is a Java library that helps to convert rich data type to XML data type and vice versa

Question 25

1/1 Points

WSGEN and WSIMPORT utilities

1. Can Be Used To Generate The Wsdl File On Server And Server-Side Helper Class To Publish The Service
2. Can Be Used By Java Client To Create Client-Side Helper Classes With The Help Of WSDL File.

WSGEN UTILITY - Can Be Used To Generate The Wsdl File On Server And Server-Side Helper Class To Publish The Service
WSIMPORT UTILITY - Can Be Used By Java Client To Create Client-Side Helper Classes With The Help Of WSDL File.

Question 26

1/1 Points

WSGEN and WSIMPORT utilities

1. Server Side Java Utility
2. Client Side Java Utility

WSGEN UTILITY - Server Side Java Utility
WSIMPORT UTILITY - Client Side Java Utility

Question 27

1/1 Points

Artifacts are helper Classes that support rich data type document –type bindings to XML Data schema and vice versa. One can create the artifacts using WSGEN and WSIMPORT utilities on server side and client side respectively

- (●) A. True
- ( ) B. False

Question 28

1/1 Points

CODE FIRST versus CONTRACT FIRST approach to write a web service

1. THE CODE FOR THE SERVICE IS WRITTEN FIRST, WSDL (CONTRACT) IS AUTOMATICALLY GENERATED LATER
2. THE WSDL FILE IS CREATED FIRST. THE CODE IS WRITTEN LATER

CODE FIRST APPROACH - THE CODE FOR THE SERVICE IS WRITTEN FIRST, WSDL (CONTRACT) IS AUTOMATICALLY GENERATED LATER
CONTRACT FIRST APPROACH - THE WSDL FILE IS CREATED FIRST. THE CODE IS WRITTEN LATER

Question 29

1/1 Points

CONTRACT FIRST APPROACH - OBEYS THE IMMUTABILITY PRINCIPLE. The IMMUTABILITY PRINCIPLE means the service contract (WSDL file) once published will not change anytime in future

- (●) A. True
- ( ) B. False

Question 30

1/1 Points

CODE FIRST versus CONTRACT FIRST approach to write a web service

1. is service/server side programmer friendly as he/she may not have to worry about WSDL. It will be automatically be generated later as per the implementation.
2. is client side programmer friendly as a tested reliable immutable contract is available upfront.

CODE FIRST - is service/server side programmer friendly as he/she may not have to worry about WSDL. It will be automatically be generated later as per the implementation.
CONTRACT-FIRST - is client side programmer friendly as a tested reliable immutable contract is available upfront.

Question 31

1/1 Points

Asynchronous client versus Synchronous client

1. Passes the Input Parameter to the function, Calls/Invokes A Function And Waits For The Return Result/Values
2. the program passes the Input Parameter to function, Calls/Invokes the Function And Does Not Wait For The Return Result/Values. Some other function will be called to accept the return values as input parameters.

Synchronous function - Passes the Input Parameter to the function, Calls/Invokes A Function And Waits For The Return Result/Values
An Asynchronous function - the program passes the Input Parameter to function, Calls/Invokes the Function And Does Not Wait For The Return Result/Values. Some other function will be called to accept the return values as input parameters.