SOAP Based Web Services

Chinmay Kunkikar (CK821)

Showing 28 of 28 Questions

Question 1

3/3 Points

SOAP Structure

- [x] A. 1. The SOAP structure consists of SOAP Envelope and Optional Soap Attachments
- [x] B. The SOAP Envelope consists of Optional SOAP Header and mandatory SOAP Body
- [ ] C. The Soap Attachments are part of SOAP XML format
- [x] D. The SOAP attachments may not be in SOAP XML format

Question 2

1/1 Points

When the message is sent from a sender to a receiver, multiple handler nodes can intercept the message, inspect the message, manipulate the message and forward it to the next actor in the chain.

- (•) A. True
- ( ) B. False

Question 3

1/1 Points

As a recommended practice, SOAP Body received via original sender should never be modified by non-terminal handler nodes except for validation of data. But the handlers can modify the SOAP header to include using information such as userid, IP address, timestamp etc.

- (•) A. True
- ( ) B. False

Question 4

1/1 Points

In case, any node handler finds an error in SOAP message, a fault (eg. an exception) can be raised. The message will not be forwarded to the next actor in the chain. Instead The fault will be sent back to original sender. The SOAP message is as good as not processed, not received.

- (•) A. True
- ( ) B. False

Question 5

1/1 Points

The SOAP fault is sent as a SOAP XML structure with the order of XML tags within the body being

<SOAP-ENV:Fault>

<faultcode>

<faultstring>

<faultactor>

<detail>

- (•) A. True
- ( ) B. False

Question 6

1/1 Points

JWS provides the base "Handler" class to implement Handlers.

- (•) A. True
- ( ) B. False

Question 7

1/1 Points

Handler types

1. processes only SOAP messages

2. processes SOAP as well as non-SOAP messages

SOAPHandler class - processes only SOAP messages
LogicalHandler class - processes SOAP as well as non-SOAP messages

Question 8

3/3 Points

SoapHandler class

- [x] A. supports SOAP protocol
- [ ] B. is protocol neutral
- [x] C. can access SOAP Header, SOAP Body
- [ ] D. can access only the SOAP Body
- [x] E. supports 4 methods handleMessage(), handleFault(), close(), getHeaders()
- [ ] F. supports 3 methods handleMessage(), handleFault(), close()

Question 9

3/3 Points

LogicalHandler class

- [ ] A. supports SOAP protocol only
- [x] B. is protocol neutral
- [ ] C. can access SOAP Header, SOAP Body
- [x] D. can access only the SOAP Body
- [ ] E. supports 4 methods handleMessage(), handleFault(), close(), getHeaders()
- [x] F. supports 3 methods handleMessage(), handleFault(), close()

Question 10

3/3 Points

Handler execution priorities

1. Multiple Handlers can be configured using a Handler Chain XML Configuration file. The Handlers will execute in top to bottom order as mentioned in the Handler Chain Configuration file.
2. If there are Logical Handler and SOAP Handler in the handler-chain configuration file on client side, the Logical Handler is called before SOAP Handler If there are Logical Handler and SOAP Handler in the handler-chain configuration file on server side, the Soap Handler is called before LogicalHandler.
3. If there are getHeaders, handleMessage, handleFault and close implemented by handlers on client side, the getHeaders always gets called first as per order of SOAPHandler configuration from top to bottom order and close functins gets called last from bottom to top order as per the order of configuration and order of execution file.

Order of Configuration - Multiple Handlers can be configured using a Handler Chain XML Configuration file. The Handlers will execute in top to bottom order as mentioned in the Handler Chain Configuration file.
Order of Execution - If there are Logical Handler and SOAP Handler in the handler-chain configuration file on client side, the Logical Handler is called before SOAP Handler If there are Logical Handler and SOAP Handler in the handler-chain configuration file on server side, the Soap Handler is called before LogicalHandler.
Order of Handler Methods - If there are getHeaders, handleMessage, handleFault and close implemented by handlers on client side, the getHeaders always gets called first as per order of SOAPHandler configuration from top to bottom order and close functins gets called last from bottom to top order as per the order of configuration and order of execution file.

Question 11

1/1 Points

The Order of Handler Methods takes preference over Order of Execution, which in turns takes preference over Order of Configuration file.

- (●) A. True
- ( ) B. False

Question 12

3/3 Points

Handler execution priorities

1. emphasizes on order of single handler type such as SoapHandler
2. emphasize on order between two different types of handlers such SoapHandler and LogicalHandler
3. emphasizes on the order in which 4 handler methods will be called.

Order of Configuration - emphasizes on order of single handler type such as SoapHandler
Order of Execution - emphasize on order between two different types of handlers such SoapHandler and LogicalHandler
Order of Handler methods - emphasizes on the order in which 4 handler methods will be called.

Question 13

1/1 Points

Handlers can be added dynamically in the program at run time instead of using the Handler Chain Configuration file, by executing the method of web service named setHandlerResolver()

- (●) A. True
- ( ) B. False

Question 14

1/1 Points

SOAP Faults can be generated using two ways, implement the getFaultInfo() method of a class derived from 'Exception' class or using SOAPFault class.

- (●) A. True
- ( ) B. False

Question 15

1/1 Points

Payload

- [x] A. refers to SOAP Body size only
- [ ] B. refers to SOAP Envelope which includes header and body
- [ ] C. refers to SOAP envelope and SOAP attachments

Question 16

1/1 Points

A bigger payload size is good in a web service.

- ( ) A. True
- (●) B. False

Question 17

1/1 Points

MessageContext object in Java represents the SOAP message which in turn contains the transport headers and follows the Context Architecture.

- (●) A. True
- ( ) B. False

Question 18

1/1 Points

In a context architecture all requests are intercepted by container, which converts the requests into appropriate context object and passes the object reference to any programs (such as SIB , Node Handlers) needing it.

- (●) A. True
- ( ) B. False

Question 19

1/1 Points

Transport headers can be fetched using the MessageContext object in the form of Java class type Map<String, Object>.

The Map class has two columns, the first one contains the unique key and second column contains the value to the key

- (●) A. True
- ( ) B. False

Question 20

1/1 Points

There are two types of MessageContext classes. SOAPMessageContext which contains SOAP messages and LogicalMessageContext which contains Non-SOAP messages passed via other protocols such as SMTP, FTP, JMS

- (●) A. True
- ( ) B. False

Question 21

1/1 Points

When binary data is sent as part of SOAP Body, the payload can be very big as binary files such as images are very big in size.

- (●) A. True
- ( ) B. False

Question 22

1/1 Points

The main problem of sending binary data in SOAP Body is that the binary characters may be misinterpreted by SOAP services as some special XML characters and thereby give undesirable output.

- (●) A. True
- ( ) B. False

Question 23

1/1 Points

Binary data should preferably be sent outside SOAP Body, as SOAP attachments.

- (●) A. True
- ( ) B. False

Question 24

1/1 Points

Base64 encoding technique resolves the problem of misinterpreting binary data by grouping every 6-bits binary into 6-bits text character representation. It is a binary to text encoding.

- (●) A. True
- ( ) B. False

Question 25

1/1 Points

The problem of Base64 encoding is the data size increases by three times when binary is encoded into text format. This problem is called Data Bloat problem.

- (●) A. True
- ( ) B. False

Question 26

3/3 Points

SOAP attachment techniques

1. technique is difficult to implement with Document-style SOAP binding and hence is not preferred
2. was introduced by Microsoft and IBM as a light binary protocol to send binary data to support its old Visual Basic applications but never found wide acceptance in non-Microsoft platforms
3. MTOM encodes and optimizes binary to binary. The binary data size is optimized and hence eliminates the Base64 data bloat problem too.

SwA (SOAP WITH ATTACHMENTS) - technique is difficult to implement with Document-style SOAP binding and hence is not preferred
DIME (DIRECT INTERNET MESSAGING ENCAPSULATION) - was introduced by Microsoft and IBM as a light binary protocol to send binary data to support its old Visual Basic applications but never found wide acceptance in non-Microsoft platforms
MTOM (MESSAGE TRANSMISSION OPTIMIZATION MECHANISM) - MTOM encodes and optimizes binary to binary. The binary data size is optimized and hence eliminates the Base64 data bloat problem too.

Question 27

1/1 Points

MTOM can be considerd as a combination of

XOP (XML-Binary Optimized Packaging) protocol,

SOAP Attachment and

MIME (MULTIPURPOSE INTERNET MAIL EXTENSIONS)

- (●) A. True
- ( ) B. False

Question 28

1/1 Points

In MTOM,

step 1: the binary data is optimized,

step 2: shifted to SOAP attachment section outside the body,

step 3: a link xml tag called <xop:Include> is created inside the SOAP body to point to appropriate section in SOAP attachment.

It's the most preferred optimized technique to send binary data

- (●) A. True
- ( ) B. False