



# C\_CPI\_15<sup>Q&As</sup>

SAP Certified Development Associate - SAP Integration Suite

**Pass SAP C\_CPI\_15 Exam with 100% Guarantee**

Free Download Real Questions & Answers **PDF** and **VCE** file from:

[https://www.geekcert.com/c\\_cpi\\_15.html](https://www.geekcert.com/c_cpi_15.html)

100% Passing Guarantee  
100% Money Back Assurance

Following Questions and Answers are all new published by SAP Official Exam Center

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers





## QUESTION 1

In your integration flow process, the request reply returns a message with namespaces. What can you do to delete namespaces from this message? Note: There are 2 correct answers to this question.

- A. Set up the request reply to delete the namespaces
- B. Use a Content Modifier
- C. Set up the namespace mapping in the Runtime Configuration.
- D. Use an XSLT mapping.

Correct Answer: BD

A Content Modifier is a step in an integration flow that allows you to modify the message header or body using expressions or constants. You can use a Content Modifier to delete namespaces from a message by using expressions that remove the namespace declarations or prefixes from the XML elements or attributes. For example, you can use the expression `replace($body,"xmlns:ns[0-9]+=\"[^\"]+\"","")` to remove all namespace declarations from the message body. An XSLT mapping is a step in an integration flow that allows you to transform the message body using an XSLT stylesheet. You can use an XSLT mapping to delete namespaces from a message by using XSLT instructions that strip or exclude the namespace information from the output XML document. For example, you can use the instruction to exclude all namespace prefixes from the output document. References: Content Modifier | SAP Help Portal, XSLT Mapping | SAP Help Portal

## QUESTION 2

You configured a content modifier as follows: Action Create. Name: ProductID. Source Type: XPath. Source Value: `//ProductID`. Data Type: java lang string. After testing the content modifier, you receive an error message that contains the

following fragment:

`java.lang.ClassNotFoundException java lang string`. What caused the error message?

- A. Incorrect name
- B. Incorrect source value
- C. Incorrect source type
- D. Incorrect data type

Correct Answer: D

The error message is caused by an incorrect data type in the content modifier configuration. The data type should be `java.lang.String` instead of `java lang string`. The data type defines the type of the value that will be stored in the header or property of the message. It should match one of the supported Java classes, such as `java.lang.String`, `java.lang.Integer`, `java.lang.Boolean`, and so on. If the data type is not valid, a `java.lang.ClassNotFoundException` will be thrown. References: Content Modifier | SAP Help Portal, Getting clear with SAP Integration Suite Content Modifier | SAP Blogs



### QUESTION 3

You want to use an HTTPS endpoint from an integration flow. In which of the following must you use the HTTP client?

- A. API Management capability within SAP Integration Suite
- B. Open Connectors capability within SAP Integration Suite
- C. SAP Integration Suite, advanced event mesh
- D. SAP Business Application Studio

Correct Answer: B

To use an HTTPS endpoint from an integration flow, you can use the Open Connectors capability within SAP Integration Suite. Open Connectors is a feature that allows you to connect to over 160 third-party applications using standardized and normalized APIs. You can use the HTTP client connector to make HTTP or HTTPS requests to any endpoint that supports these protocols. You can configure the HTTP client connector with various parameters, such as URL, method, headers, query parameters, body, and authentication. References: Open Connectors | SAP Help Portal, HTTP Client Connector | SAP Help Portal

---

### QUESTION 4

In which of the following must you call `getBody()` to access the payload in a Groovy script?

- A. camelcontext
- B. property
- C. header
- D. message

Correct Answer: D

To access the payload in a Groovy script, you must use the message object and call the `getBody()` method on it. The message object represents the current message that is being processed in the integration flow. It has methods to get and set the message body, headers, properties, and attachments. The `getBody()` method returns the message body as an object of the specified type, such as `java.lang.String`, `java.io.InputStream`, or `org.w3c.dom.Document`. References: General Scripting Guidelines | SAP Help Portal, Message | SAP Help Portal

---

### QUESTION 5

You want to call an OData v4.0 interface through an adapter in the Cloud Integration capability within SAP Integration Suite. Which adapter can you use?

- A. JMS
- B. ProcessDirect
- C. HTTPS
- D. SOAP



Correct Answer: C

You can use an HTTPS adapter to call an OData v4.0 interface in the Cloud Integration capability within SAP Integration Suite. An HTTPS adapter allows you to send and receive messages over HTTP or HTTPS protocols. You can use an HTTPS adapter to communicate with RESTful services, such as OData v4.0, that support HTTP methods and formats. References: Integration Software | SAP Integration Suite, Modernize Integration with SAP Integration Suite | openSAP

[C\\_CPI\\_15 PDF Dumps](#)

[C\\_CPI\\_15 VCE Dumps](#)

[C\\_CPI\\_15 Exam Questions](#)