

XOS-T.

**1 of** In an application network, if the implementation but not the interface of a product API changes, what needs to be done to the other  
60. APIs that consume the product API?

- A.  Nothing needs to be changed in the other APIs or their associated applications
- B.  The applications associated with the other APIs must be recoded
- C.  The other APIs must be updated to consume the updated product API
- D.  The applications associated with the other APIs must be restarted



Mark this item for later review.

## MCD - Level 1 (Mule 4)

Time Remaining: 1:59:50

- 2 of** To avoid hard-coding values, a flow uses some property placeholders and the corresponding values are stored in a configuration file.
- 60.**

Where does the configuration file's location need to be specified in the Mule application?

- A. A global element the configuration file's location need to be specified as a global element
- B. The mule-artifact.json file
- C. A flow attribute
- D. The pom.xml file

Mark this item for later review.



## MCD - Level 1 (Mule 4)

Time Remaining: 1:58:16

- 3 of 60.** Refer to the exhibit. The error occurs when a project is run in Anypoint Studio. The project, which has a dependency that is not in the MuleSoft Maven repository, was created and successfully run on a different computer.

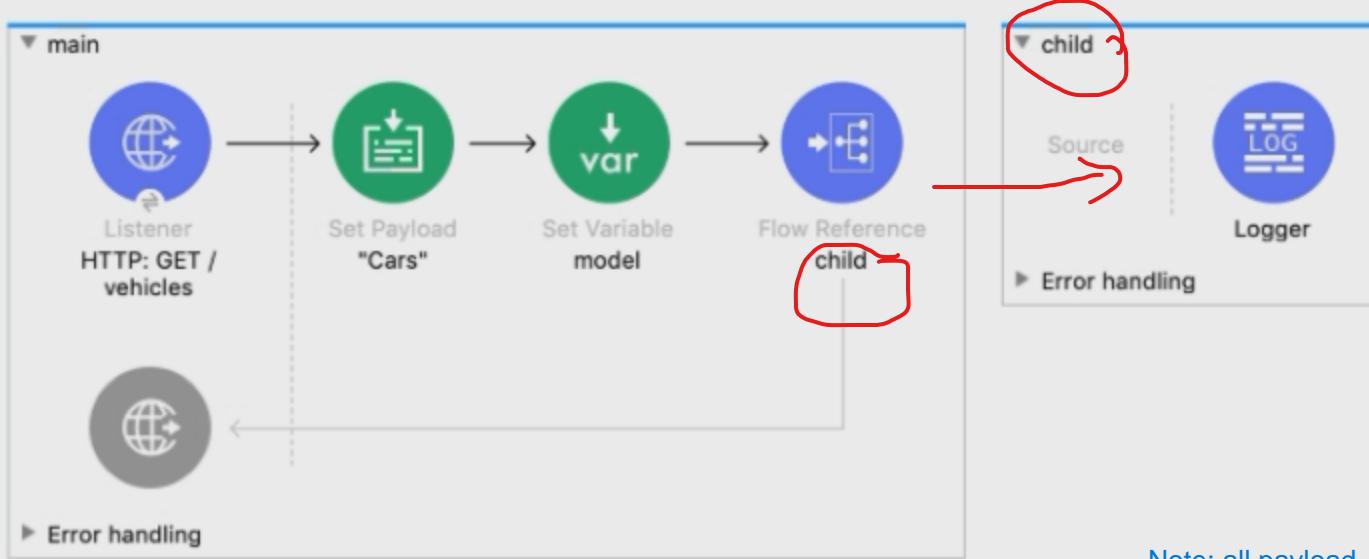
What is the next step to fix the error to get the project to run successfully?

```
[WARNING] The POM for com.mulesoft.training:mock-servers:jar:1.1.2 is missing, no dependency information available
[INFO] -----
[INFO] BUILD FAILURE
[INFO] -----
[INFO] Total time: 0.253 s
[INFO] Finished at: 2018-06-14T10:52:39-07:00
[INFO] Final Memory: 190M/867M
[INFO] -----
[ERROR] Failed to execute goal on project app-server: Could not resolve dependencies for project com.mycompany:app-server:mule-application:1.0.0-SNAPSHOT: Failure to find com.mulesoft.training:mock-servers:jar:1.1.2 in https://repository.mulesoft.org/nexus-ee/content/repositories/releases-ee/ was cached in the local repository, resolution will not be reattempted until the update interval of MuleRepository has elapsed or updates are forced -> [Help 1]
```

- A.  Add the dependency to the MULE\_HOME/bin folder
- B.  Edit the dependency in the Mule project's pom.xml file
- C.  Install the dependency to the computer's local Maven repository
- D.  Deploy the dependency to MuleSoft's Maven repository

- 4 Refer to the exhibits. The main flow contains a Flow Reference to the child flow.
- of
60. A web client sends a GET request to the main flow's HTTP Listener that includes a make query parameter.

What values are accessible in the child flow?



- A.  payload
- B.  payload  
make query param
- C.  payload  
model var
- D.  payload  
make query param  
model 

Note: all payload ,  
q.param and vars are  
accessible by the logger  
(child flow)

- 5 of 60.** A Mule application contains an ActiveMQ JMS dependency. The Mule application was developed in Anypoint Studio and runs successfully in Anypoint Studio.

The Mule application must now be exported from Anypoint Studio and shared with another developer.

What export options create the smallest JAR file that can be imported into the other developer's Anypoint Studio and run successfully?

- A.   Attach project sources  
 Include project modules and dependencies
- B.   Attach project sources  
 Include project modules and dependencies
- C.   Attach project sources  
 Include project modules and dependencies
- D.   Attach project sources  
 Include project modules and dependencies

Note: project modules and dependencies (like codes) doesn't need to include in the JAR file. no more need to executing app

- Repeated 5 of** A Mule application contains an ActiveMQ JMS dependency. The Mule application was developed in Anypoint Studio and runs successfully in Anypoint Studio.

The Mule application must now be exported from Anypoint Studio and shared with another developer.

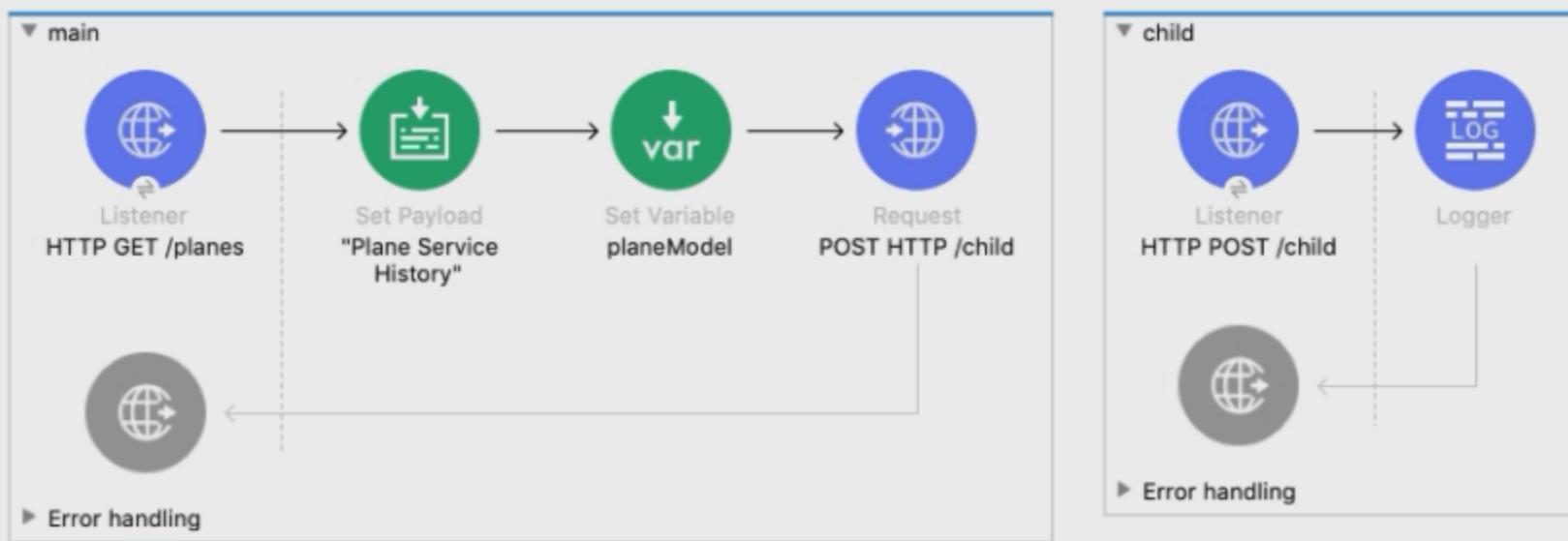
What export options create the smallest JAR file that can be imported into the other developer's Anypoint Studio and run successfully?

- A.   Attach project sources  
 Include project modules and dependencies
- B.   Attach project sources  
 Include project modules and dependencies
- C.   Attach project sources  
 Include project modules and dependencies
- D.   Attach project sources  
 Include project modules and dependencies

**6 of** Refer to the exhibits. The main flow contains an HTTP Request. The HTTP Listeners and HTTP Request use default configurations.  
60.

A web client sends a GET request to the main flow's HTTP Listener that includes a modelName query parameter

What values are accessible in the child flow?

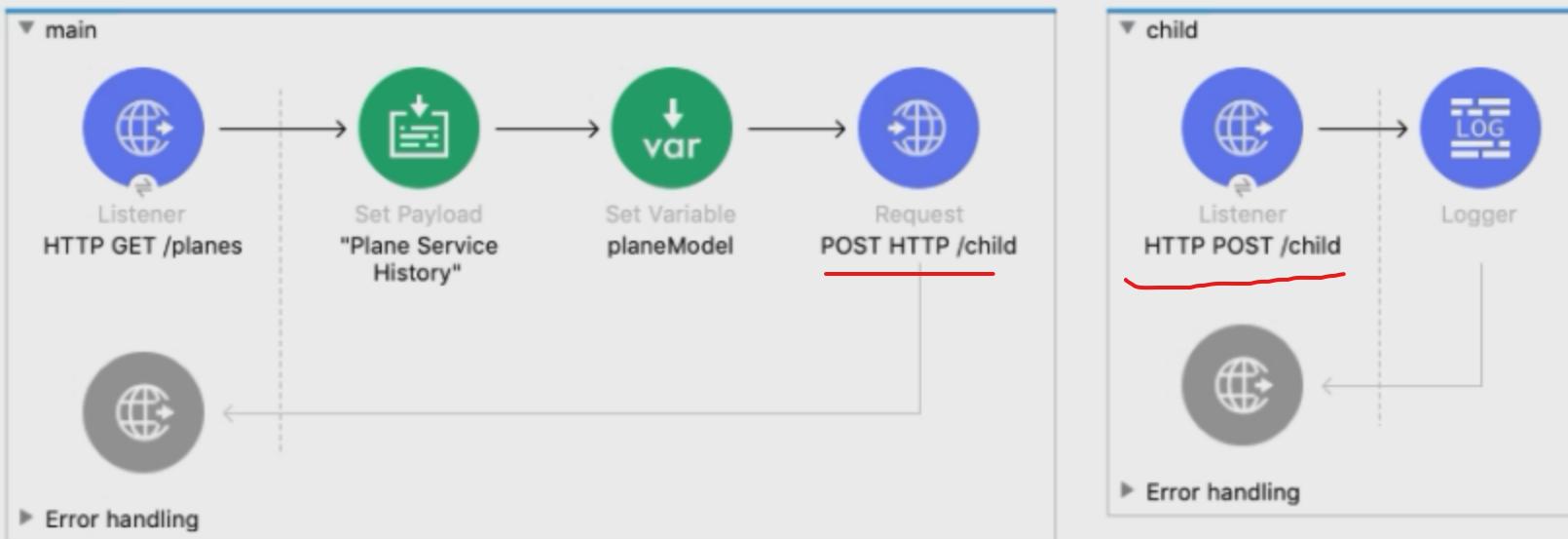


```
<http:request method="POST" doc:name="POST HTTP /child" url="http://localhost:8081/child">
</http:request>
```

- A.  payload
- B.  payload  
modelName query param
- C.  payload  
planeModel var
- D.  payload



What values are accessible in the child flow?



```
<http:request method="POST" doc:name="POST HTTP /child" url="http://localhost:8081/child">  
</http:request>
```

- A.  payload
- B.  payload  
modelName query param
- C.  payload  
planeModel var
- D.  payload  
modelName query param  
planeModel var

this is confusing if the main flow can be referenced or nor? Payload is passed but not sure if attribute can be? I prefer to mark A anyways



**7 of 60.** An API specification is designed using RAML. What is the next step to create a REST Connector from this API specification?

- A.  Add the specification to a Mule project's src/main/resources/api folder
- B.  Publish the API specification to Anypoint Exchange
- C.  Download the API specification and build the interface using APIkit
- D.  Implement the API specification using flow designer in Design Center

Mark this item for later review.

after you have tested your API specification, publish it to your private exchange so others in your organization can reuse it

- 8 of 60 A Mule project contains a DataWeave module called MyModule.dwl that defines a function named formatString. The module is located in the project's src/main/resources/modules folder.

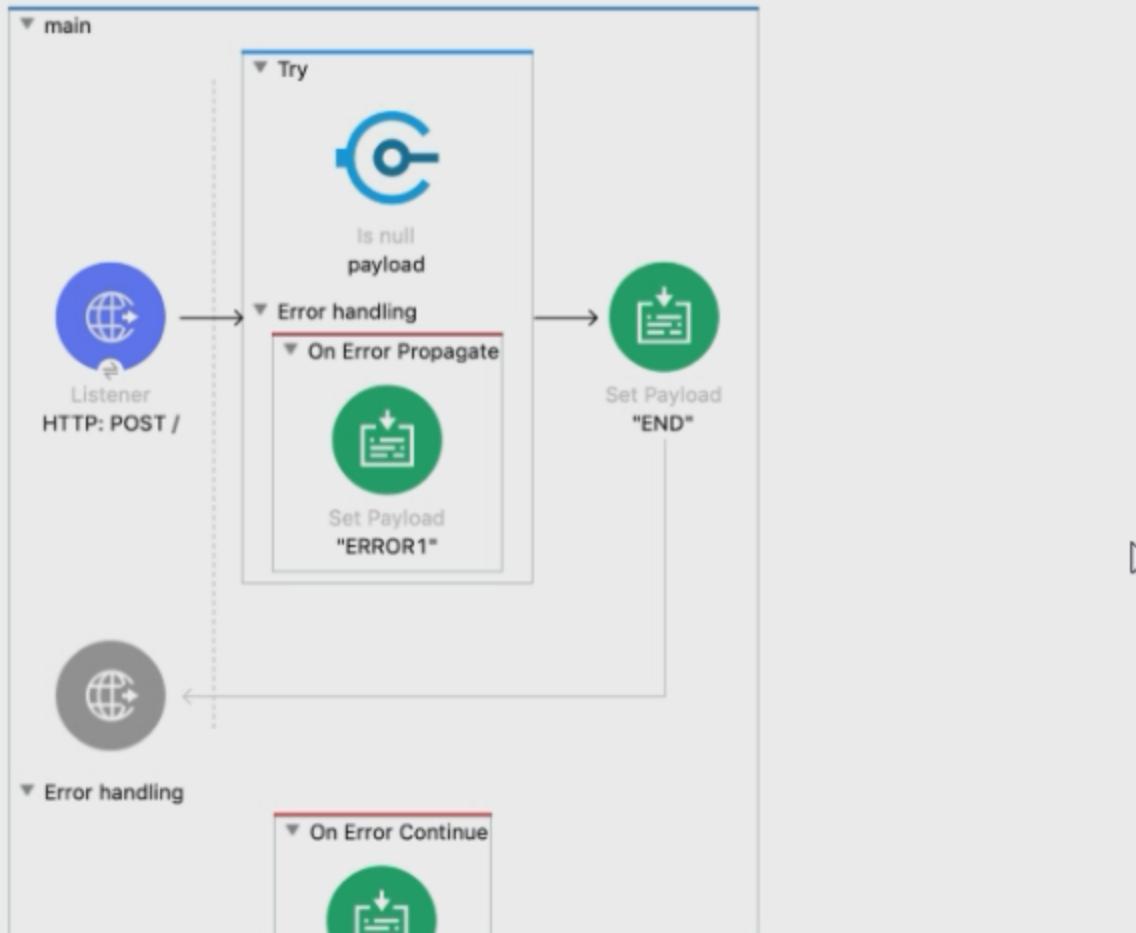
What is the correct way in DataWeave code to import MyModule using a wildcard and then call the module's formatString function?

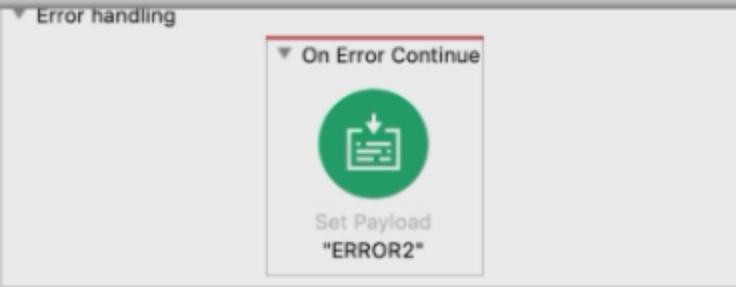
- A.  %dw 2.0  
output application/json  
import \* from modules::MyModule  
---  
formatString( "annie point" )
- B.  %dw 2.0  
output application/json  
import \* from modules.MyModule  
---  
MyModule.formatString( "annie point" )
- C.  %dw 2.0  
output application/json  
import \* from modules.MyModule  
---  
formatString( "annie point" )
- D.  %dw 2.0  
output application/json  
import \* from modules::MyModule  
---  
MyModule::formatString( "annie point" )



- 9 of 60 Refer to the exhibits. A web client sends a POST request to the HTTP Listener and the Validation component in the Try scope throws an error.

What response message is returned to the web client?





```

<flow name="main">
    <http:listener doc:name="HTTP: POST /" config-ref="HTTP_Listener_config" path="/" >
    </http:listener>
    <try doc:name="Try" >
        <validation:is-null doc:name="payload" value="#[payload]" message="Validation Error"/>
        <error-handler >
            <on-error-propagate enableNotifications="true" logException="true"
            doc:name="On Error Propagate">
                <set-payload value='ERROR1' doc:name='ERROR1' />
            </on-error-propagate>
        </error-handler>
    </try>
    <set-payload value='END' doc:name='END' />
    <error-handler >
        <on-error-continue enableNotifications="true" logException="true"  →
        doc:name="On Error Continue" >
            <set-payload value='ERROR2' doc:name='ERROR2' />
        </on-error-continue>
    </error-handler>
</flow>

```

Refer to the exhibits. A web client sends a POST request to the HTTP Listener and the Validation component in the Try scope throws an error.

What response message is returned to the web client?

```
<set-payload value='''ERROR1''' doc:name='''ERROR1'''>
</on-error-propagate>
</error-handler>
</try>
<set-payload value='''END''' doc:name='''END''' />
<error-handler >
    <on-error-continue enableNotifications="true" logException="true"
    doc:name="On Error Continue" >
        <set-payload value='''ERROR2''' doc:name='''ERROR2''' />
    </on-error-continue>
</error-handler>
</flow>
```

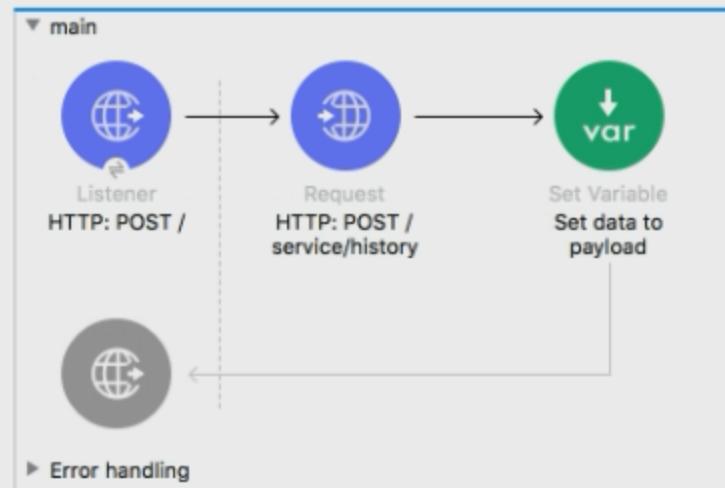
Refer to the exhibits. A web client sends a POST request to the HTTP Listener and the Validation component in the Try scope throws an error.

What response message is returned to the web client?

- A.  "END"
- B.  Validation Error
- C.  "ERROR1"
- D.  "ERROR2"

on error propagate has thrown the error in the main flow, so the control goes to the default error handler. Here the actual error should be the error configure in the HTTP listener ...ERROR 1?????????

10 of 60 Refer to the exhibit. What can be added to the flow to persist data across different flow executions?



- A.  Session variables
- B.  Properties of the Mule runtime flow object
- C.  Properties of the Mule runtime app object
- D.  Key/value pairs in the ObjectStore

Mark this item for later review.



- 11 of 60. A web client submits a request to <http://localhost:8081/books/0471767840>. The value "0471767840" is captured by a Set Variable transformer to a variable named bookISBN.

What is the DataWeave expression to access bookISBN later in the flow?

- A.  attributes.bookISBN
- B.  bookISBN
- C.  flowVars.bookISBN
- D.  vars.bookISBN

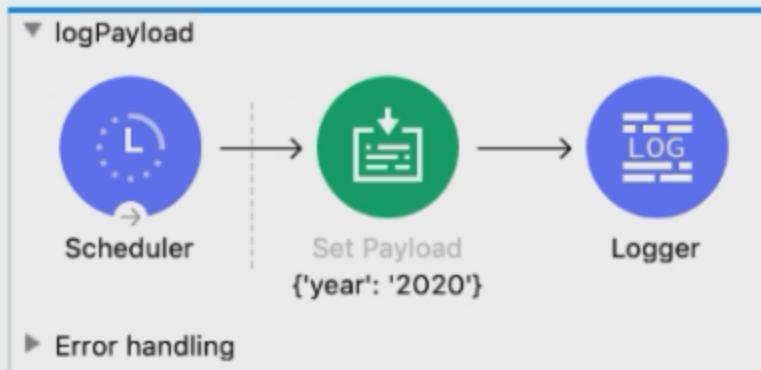
**VARS is must to access a value of any flow variable**

Mark this item for later review.

12 of Refer to the exhibits. The Set Payload transformer's value is set to {'year': '2020'}.

60.

What message value should be added to the Logger component to output the message 'The year is 2020', without hardcoding 2020?



```
<flow name="logPayload" >
    <scheduler doc:name="Scheduler" >
        <scheduling-strategy >
            <fixed-frequency />
        </scheduling-strategy>
    </scheduler>
    <set-payload value="#[{'year': '2020'}]" doc:name="{'year': '2020'}" />
    <logger level="INFO" doc:name="Logger" message="????? " />
</flow>
```

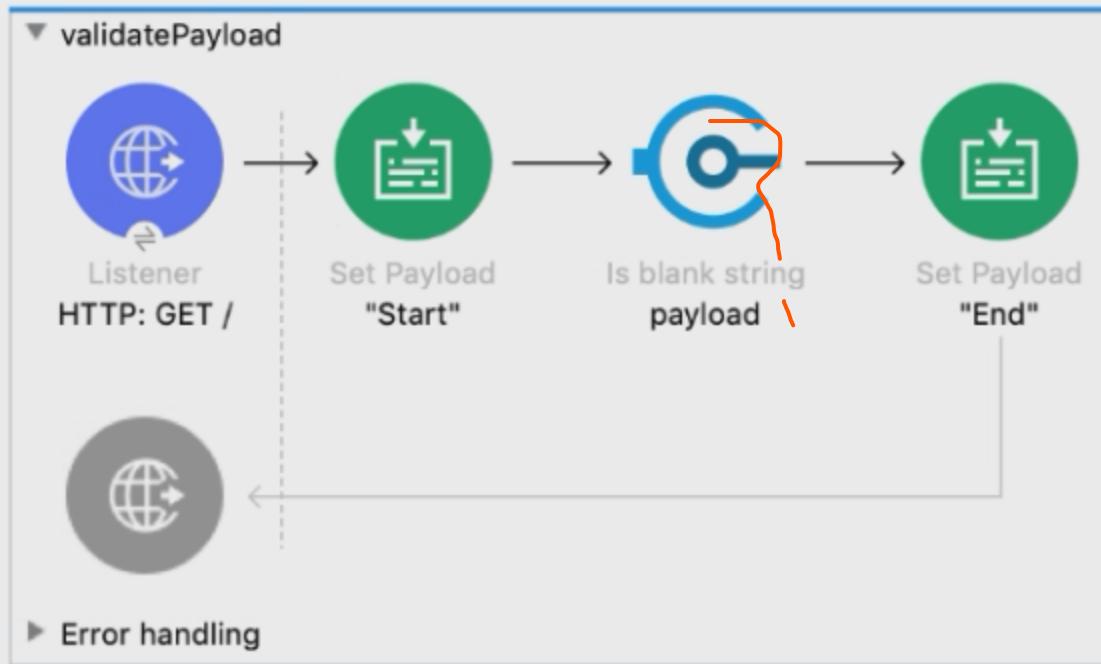
- A.  '#["The year is " + payload.year]'
- B.  '#[The year is \$(payload.year)]'
- C.  '#["The year is ++ payload.year"]'
- D.  The year is #[payload.year]'



13 of Refer to the exhibits. A web client sends a GET request to the HTTP Listener.

60.

What response message is returned to the web client?



```
<flow name="validatePayload" >
    <http:listener doc:name="HTTP: GET /" config-ref="HTTP_Listener_config" path="/" />
    <set-payload value="Start" doc:name=' "Start"' />
    <validation:is-blank-string doc:name="payload" value="#{payload}" />
    <set-payload value="End" doc:name=' "End"' />
</flow>
```



```

<flow name="validatePayload" >
    <http:listener doc:name="HTTP: GET /" config-ref="HTTP_Listener_config" path="/" />
    <set-payload value="Start" doc:name='"Start"' />
    <validation:is-blank-string doc:name="payload" value="#[payload]" />
    <set-payload value="End" doc:name='"End"' />
</flow>
  
```

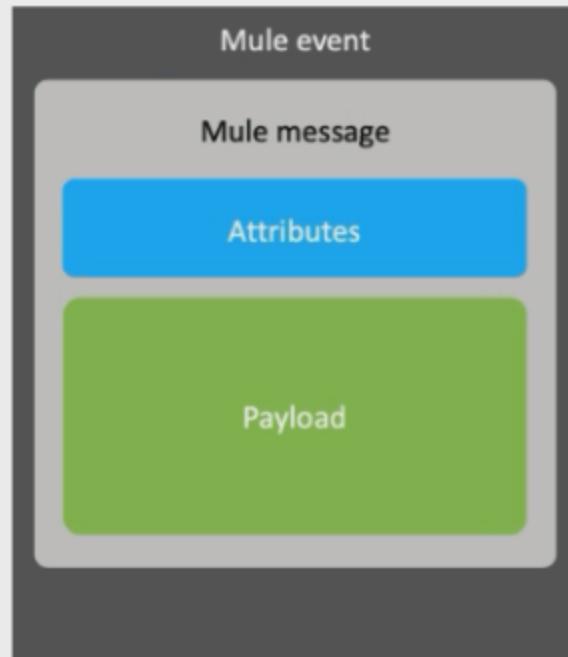
- A.  "End"
- B. "String is not blank"
- C. ""
- D. "Start"

Mark this item for later review.



**14 of 60.** Refer to the exhibit. A Mule event is composed of a hierarchy of objects.

Where in the hierarchy are variables stored?



- A.  Mule message attributes
- B.  Mule message
- C.  Mule event
- D.  Mule message payload

C.  Mule event

D.  Mule message payload

## MCD - Level 1 (Mule 4)

Time Remaining: 1:53:36

- 15 of 60.** A flow needs to combine and return data from two different data sources. It contains a Database SELECT operation followed by an HTTP Request operation.

What is the method to capture both payloads so the payload from the second request does not overwrite that from the first?

- A.  Nothing, previous payloads are combined into the next payload
- B.  Put the Database SELECT operation inside a Message Enricher scope
- C.  Put the Database SELECT operation inside a Cache scope
- D.  Save the payload from the Database SELECT operation to a variable

Mark this item for later review.

Note: Both the SELECT operation and the HTTP request are going to store response in the payload, which will lead to overwriting of data. To resolve this limitation (overwriting) before making HTTP request we need to store the content of the payload into variable. We will end up with two different data stored in payload and variable. We can merge this data (bring them to one) using transform message. (source internet)

## MCD - Level 1 (Mule 4)

Time Remaining: 1:53:25

- 16 of** A Database On Table Row listener retrieves data from a CUSTOMER table that contains a primary key user\_id column and an  
60. increasing login\_date\_time column. Neither column allows duplicate values.

How should the listener be configured so it retrieves each row at most one time?

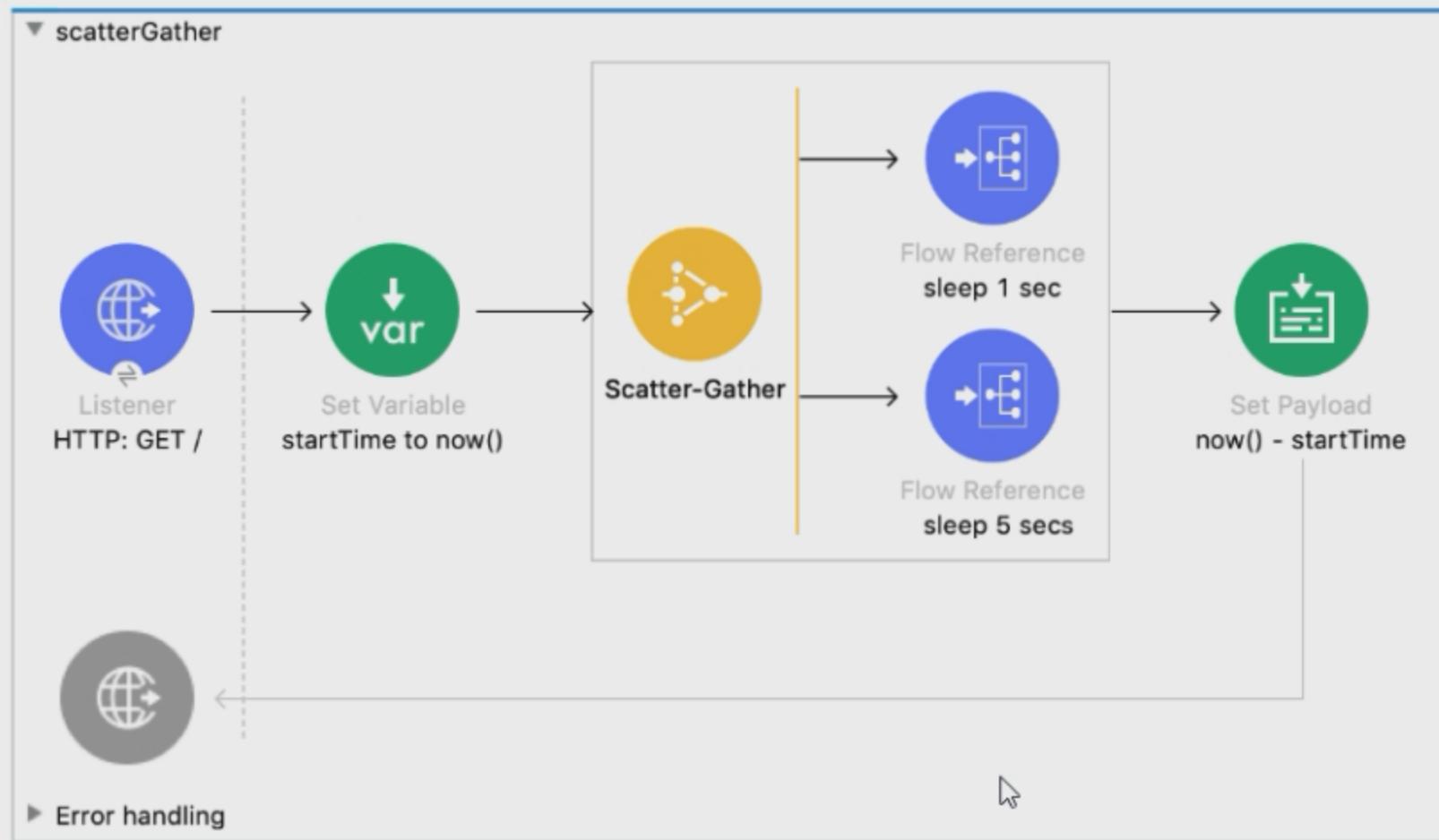
- A.  Set the watermark column to the login\_date\_time column
- B.  Set the target value to the last retrieved login\_date\_time value
- C.  Set the watermark column to the user\_id column
- D.  Set the target value to the last retrieved user\_id value

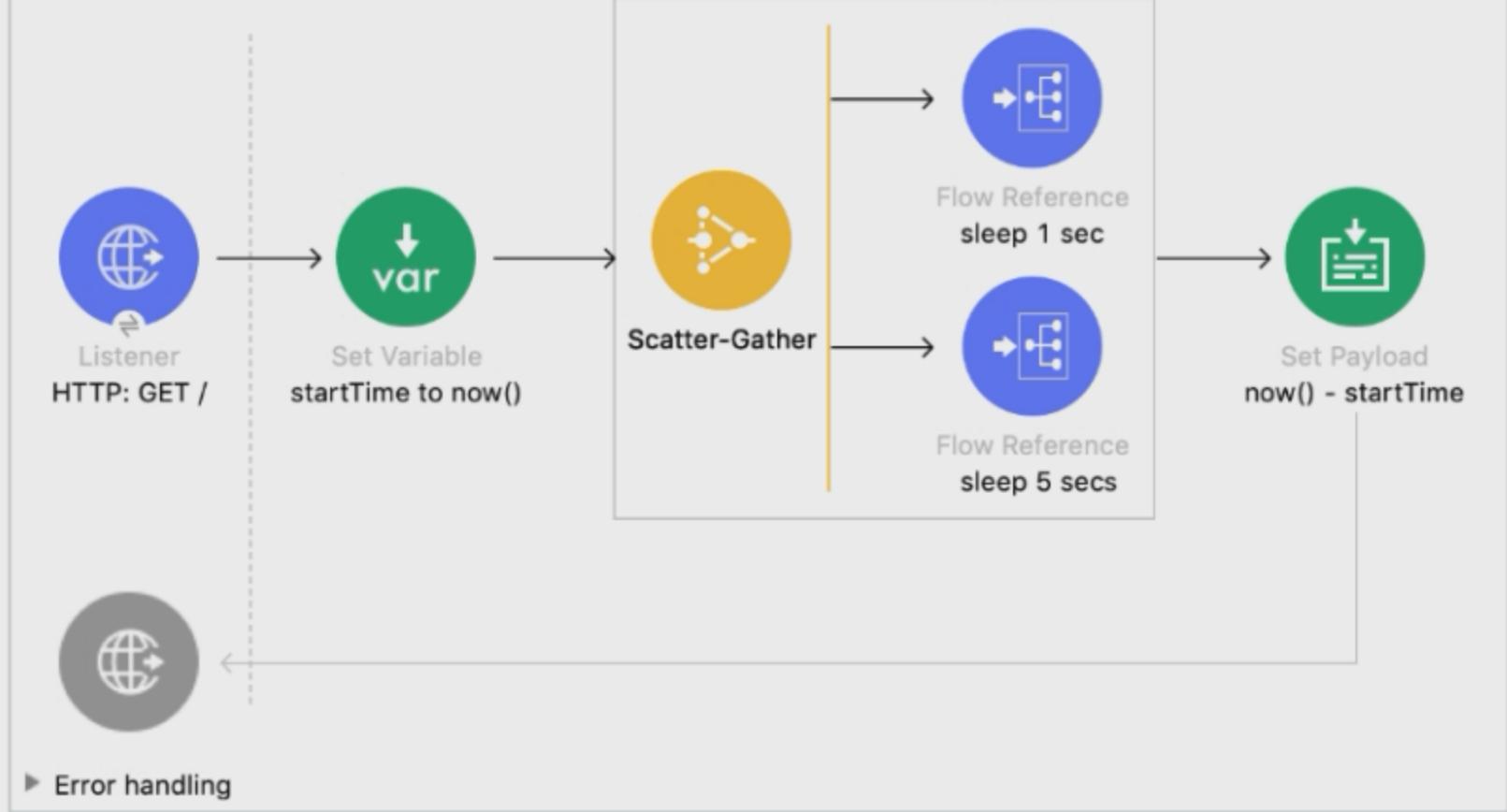
Mark this item for later review.



- 17 of** Refer to the exhibit. In the execution of the Scatter-Gather, the "sleep 1 sec" Flow Reference takes about 1 second to complete, and  
**60.** the "sleep 5 secs" Flow Reference takes about 5 seconds to complete.

About how many seconds does it take from the time the Scatter-Gather is called until the Set Payload transformer is called?





- A.  0
- B.  1
- C.  5
- D.  6

the duration of scatter-gather executed in parallel will be the time which is for the longest process to be finished, while for sequential is the same of both



## MCD - Level 1 (Mule 4)

Time Remaining: 1:52:37

**18 of 60.** What execution model is used by For Each and Batch Job scopes?

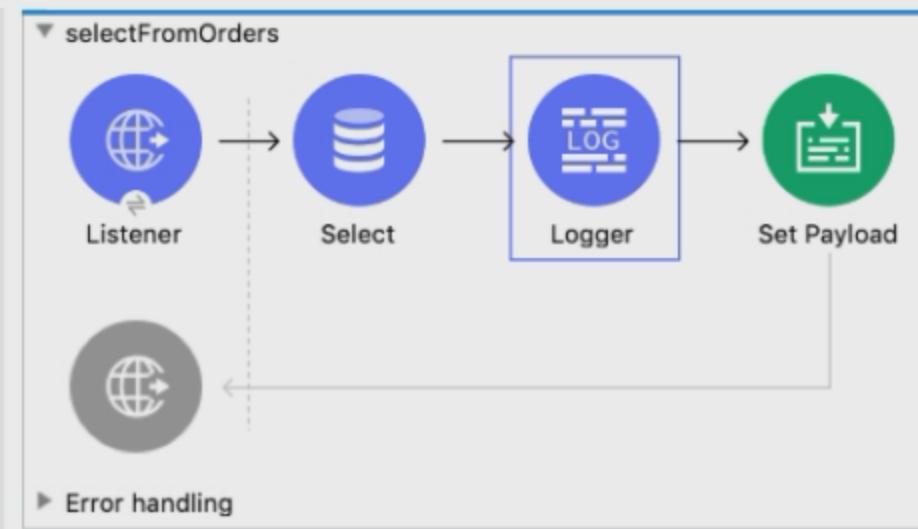
- A.  Batch Job is single-threaded and For Each is multi-threaded
- B.  Both are single-threaded
- C.  Both are multi-threaded
- D.  For Each is single-threaded and Batch Job is multi-threaded

Mark this item for later review.



19 of 60. Refer to the exhibit. The Database Select operation returns five rows from a database.

What is logged by the Logger component?



Message Flow Global Elements Configuration XML

The screenshot shows the configuration interface for the "Logger" component. The top navigation bar includes tabs for "Logger" (selected), "Problems", and "Console". The left sidebar has sections for "General", "Metadata", "Notes", and "Help". The main panel displays the following configuration:

- General**:
  - Display Name:
  - Message:
- Metadata**:
  - Generic

A status message at the top right says "There are no errors." with a green checkmark icon.



▶ Error handling

Message Flow Global Elements Configuration XML

Logger X Problems Console

General

Metadata Notes Help

Generic

Display Name: Logger

Message: `#[ typeOf(payload) ]`

Level: INFO (Default)

Category:

?

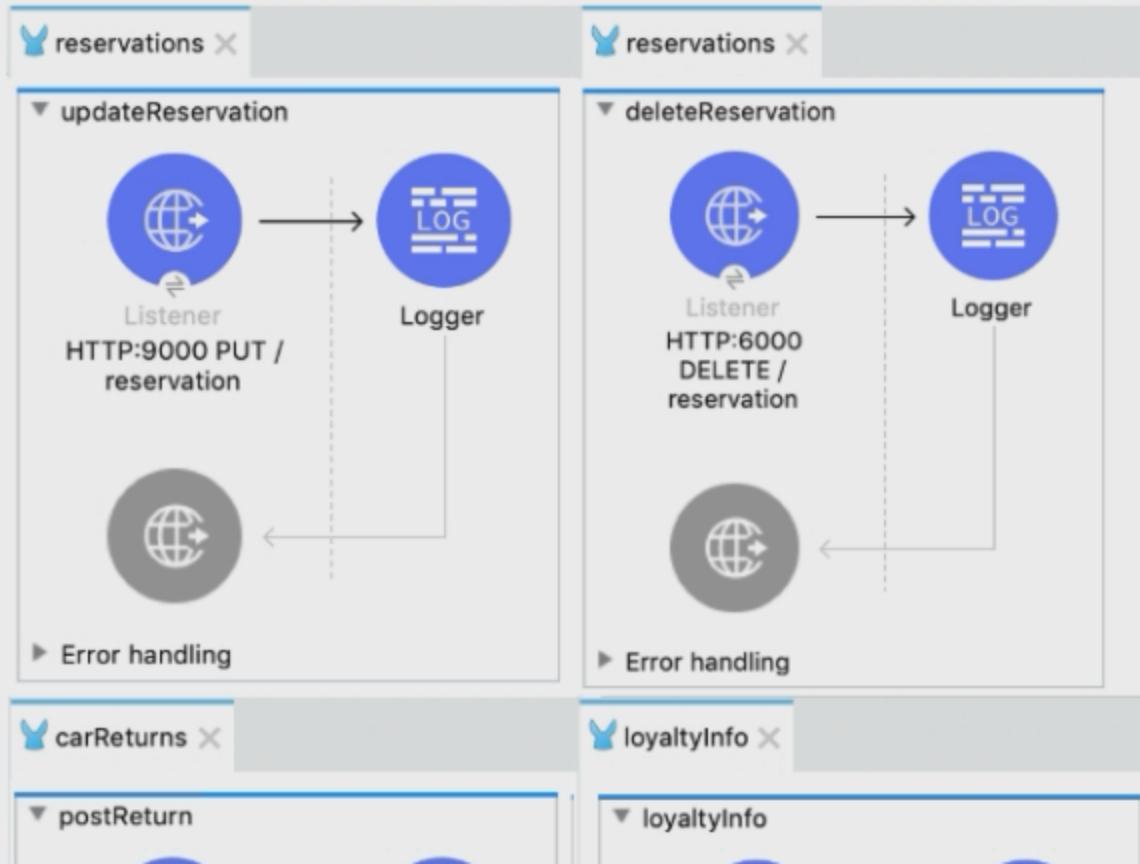
- A.  "Object"
- B.  "LinkedHashMap"
- C.  "Array"
- D.  "CaseInsensitiveHashMap"

Mark this item for later review.



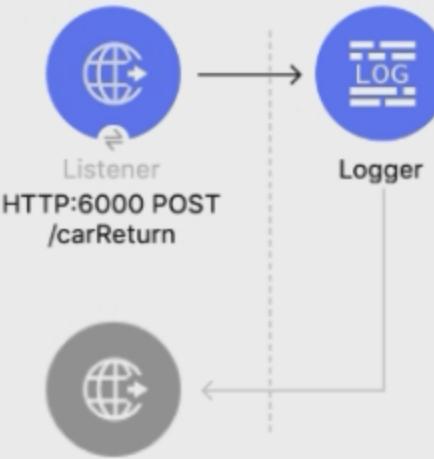
- 20 Refer to the exhibits. The Mule application has multiple HTTP Listeners contained in various configuration XML files. Each HTTP Listener is configured with the same host and with the port number, path, and operation shown in its display name.
- 60.

What is the minimum number of global elements that must be defined to support all these HTTP Listeners?



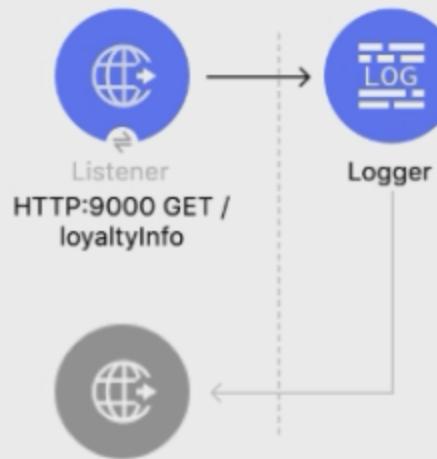
### carReturns

#### postReturn



### loyaltyInfo

#### loyaltyInfo



Refer to the exhibits. The Mule application has multiple HTTP Listeners contained in various configuration XML files. Each HTTP Listener is configured with the same host and with the port number, path, and operation shown in its display name.

What is the minimum number of global elements that must be defined to support all these HTTP Listeners?

- A.  1
- B.  2
- C.  3
- D.  4



How many private flows does APIkit generate from the RAML specification?

```
#%RAML 1.0
title: ACME Car Rentals API
version: 1.0

/rentals:
  get:
    responses:
      200:
      404:

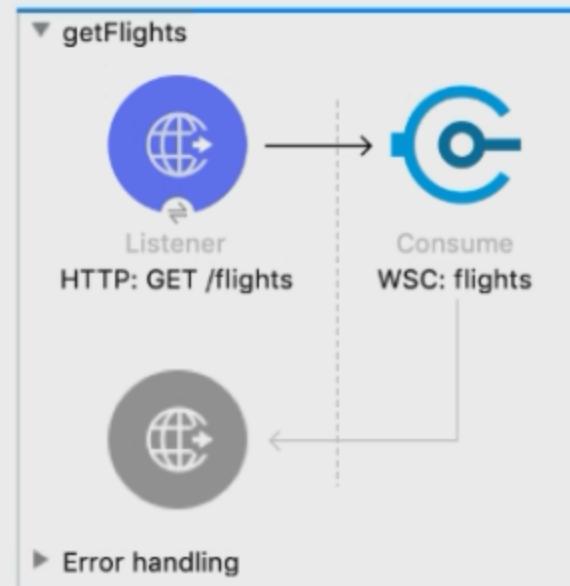
/customers:
  get:
    responses:
      200:
      400:
  post:
    responses:
      201:
```



- A.  1
- B.  2
- C.  3
- D.  5

- 22 of Refer to the exhibits. A web client sends a GET request that includes a destination query parameter to the flow's HTTP Listener. The  
60. Web Service Consumer throws a WSC:BAD\_REQUEST error.

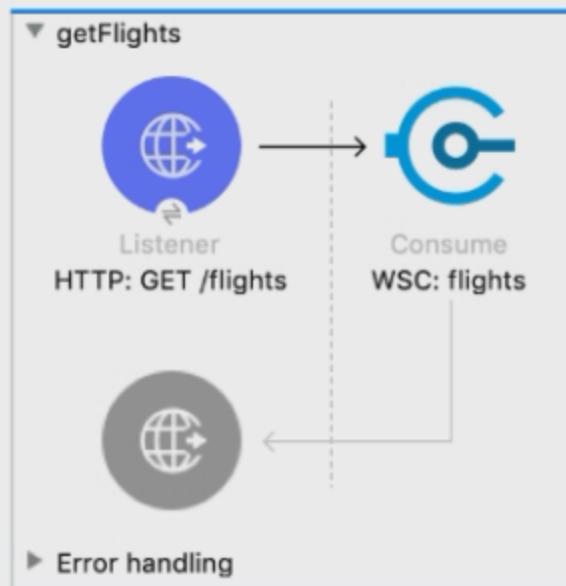
What is the next step to fix this error?



```
*****
Message           : Cannot build default body request for operation [findFlight], the operation requires input parameters.
Error type       : WSC:BAD_REQUEST
Element          : getFlights/processors/0 @ 10.02.06_v1.1-alt3:10.02.06_v1.1-alt3.xml:36 (WSC: flights)
Element XML      : <wsc:consume doc:name="WSC: flights" config-ref="Web_Service_Consumer_Config"
operation="findFlight"></wsc:consume>
(set debug level logging or '-Dmule.verbose.exceptions=true' for everything)
*****
```

- A.  Set a SOAP payload before the Consume operation that contains the destination query parameter
- B.  Set a JSON payload before the Consume operation that contains the destination query parameter

What is the next step to fix this error?



```
*****
Message           : Cannot build default body request for operation [findFlight], the operation requires input parameters.
Error type       : WSC:BAD_REQUEST
Element          : getFlights/processors/0 @ 10.02.06_v1.1-alt3:10.02.06_v1.1-alt3.xml:36 (WSC: flights)
Element XML      : <wsc:consume doc:name="WSC: flights" config-ref="Web_Service_Consumer_Config"
operation="findFlight"></wsc:consume>
(set debug level logging or '-Dmule.verbose.exceptions=true' for everything)
*****
```

- A.  Set a SOAP payload before the Consume operation that contains the destination query parameter
- B.  Set a JSON payload before the Consume operation that contains the destination query parameter
- C.  Set a property in the Consume operation equal to the destination query parameter
- D.  Set a header in the Consume operation equal to the destination query parameter

## MCD - Level 1 (Mule 4)

Time Remaining: 1:50:24

**23 of 60.** What is the output type of the DataWeave map operator?

- A.  Array
- B.  Map
- C.  String
- D.  Object

Mark this item for later review.

24 of 60. Refer to the exhibit.

What DataWeave expression transforms the conductorIds array to the XML output?

```
1 %dw 2.0
2   output application/xml
3   var conductorIds = [592, 921]
4   ---
5
6
7
8
9
10
```

```
<?xml version='1.0' encoding='UTF-8'?>
<trains>
  <train>
    <engineerId>592</engineerId>
  </train>
  <train>
    <engineerId>921</engineerId>
  </train>
</trains>
```

A.  {  
 trains: conductorIds map ( (engId, index) ->  
 train: {  
 engineerId: engId  
 }  
 )  
}

B.  trains:  
 conductorIds map ( (engId, index) ->  
 train: {  
 engineerId: engId  
 }  
 )



A.  {  
 trains: conductorIds map ( (engId, index) ->  
 train: {  
 engineerId: engId  
 }  
 )  
}

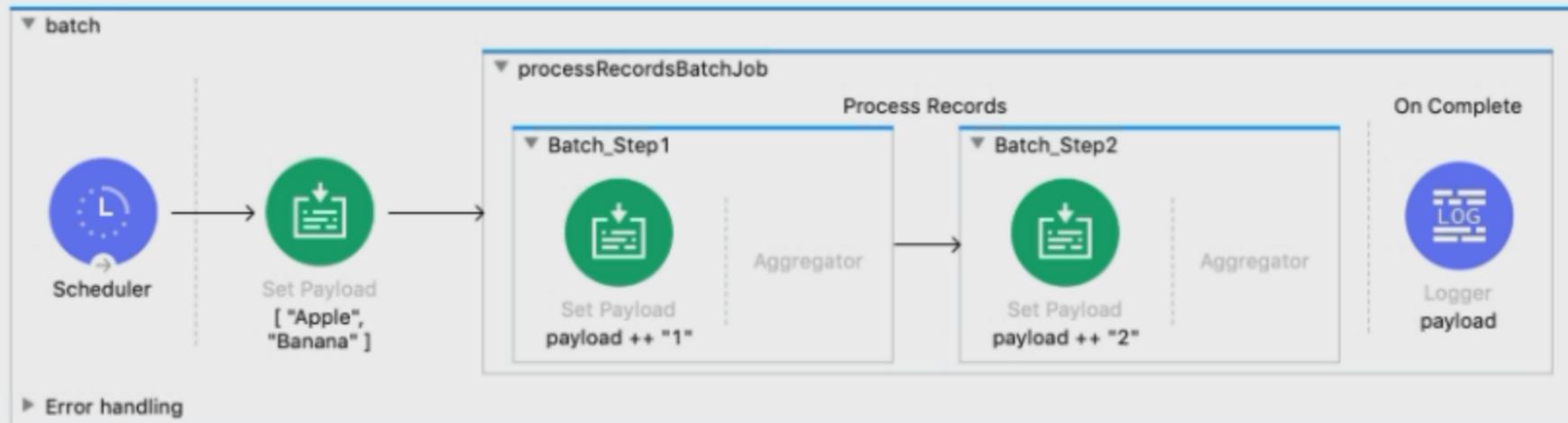
B.  trains:  
 conductorIds map ( (engId, index) ->  
 train: {  
 engineerId: engId  
 }  
 )

C.  {{  
 trains: conductorIds map ( (engId, index) ->  
 train: {  
 engineerId: engId  
 }  
 )  
}}

D.  trains:  
  {{  
 conductorIds map ( (engId, index) ->  
 train: {  
 engineerId: engId  
 }  
 )  
}}

60.

What payload is logged by the Logger component?



- A. [ "Apple12", "Banana12" ]
- B. [ "Apple", "Banana" ]
- C. [ "Apple1", "Banana1", 2 ]
- D. Summary report of processed records

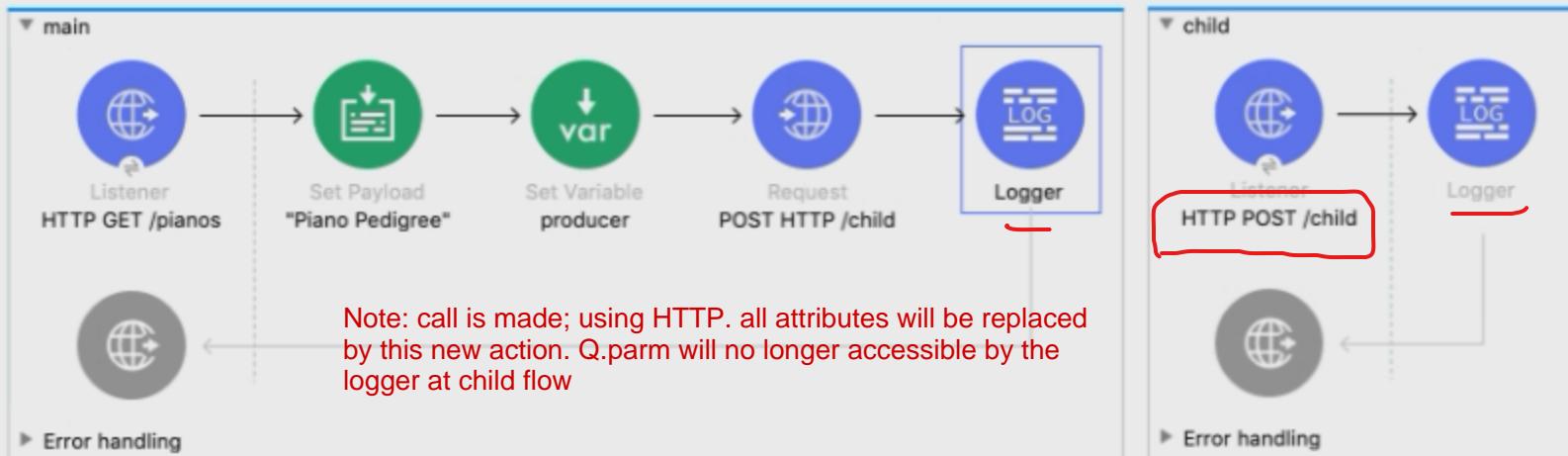
Mark this item for later review.

26 Refer to the exhibits. The main flow contains an HTTP Request in the middle of the flow. The HTTP Listeners and HTTP Request use of default configurations.

60.

A web client sends a GET request to the main flow's HTTP Listener. The GET request includes query parameters for the pedigree of a piano.

What values are accessible to the Logger component at the end of the main flow?



```

<http:request method="POST" doc:name="POST HTTP /child" url="http://localhost:8081/child">
</http:request>
<logger level="INFO" doc:name="Logger"/>
</flow>

<flow name="child" >
    <http:listener doc:name="HTTP POST /child" config-ref="HTTP_Listener_config" path="/child"/>

```



```

<http:request method="POST" doc:name="POST HTTP /child" url="http://localhost:8081/child">
</http:request>
<logger level="INFO" doc:name="Logger"/>
</flow>

<flow name="child" >
    <http:listener doc:name="HTTP POST /child" config-ref="HTTP_Listener_config" path="/child"/>

```

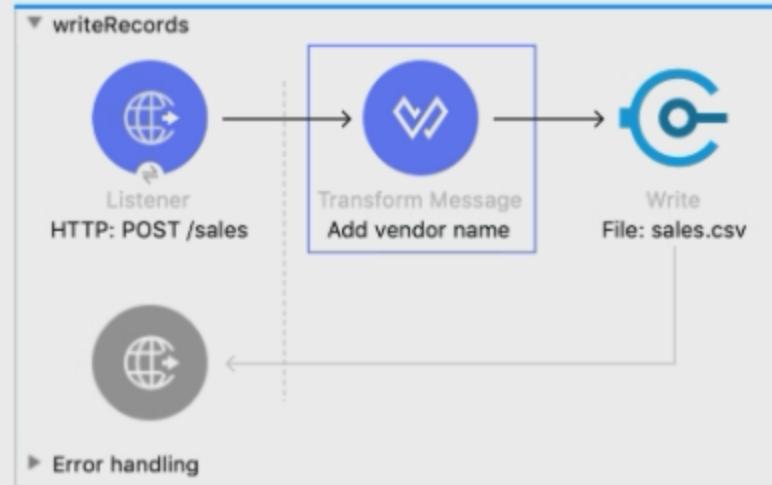
- A.  payload
- B.  payload  
pedigree query params
- C.  payload  
producer var
- D.  payload  
pedigree query params  
producer var

- 27 Refer to the exhibits. A web client sends sale data in a POST request to the Mule application. The Transform Message component then enriches the payload by prepending a vendor name to the sale data.

60.

What is written to the sales.csv file when the flow executes?

```
<?xml version="1.0" encoding="UTF-8"?>
<sale>
    <transaction_id>SS-4848-44KK-4SYQ</transaction_id>
    <customer_id>anniepoint@acme.com</customer_id>
    <customer>Annie Point</customer>
    <item>Car Seat</item>
    <qty>1</qty>
    <price>59</price>
</sale>
```



```
<flow name="writeRecords" >
    <http:listener doc:name="HTTP: POST /sales" config-ref="HTTP_Listener_config"
    path="/sales" allowedMethods="POST"/>
    <ee:transform doc:name="Add vendor name" >
        <ee:message >
            <ee:set-payload ><![CDATA[%dw 2.0
                output application/json
                ---
                {vendor: "Acme"} ++ payload.sale]]>
            </ee:set-payload>
```

```
<flow name="writeRecords" >
    <http:listener doc:name="HTTP: POST /sales" config-ref="HTTP_Listener_config"
        path="/sales" allowedMethods="POST"/>
    <ee:transform doc:name="Add vendor name" >
        <ee:message >
            <ee:set-payload ><![CDATA[%dw 2.0
                output application/json
                ---
                {vendor: "Acme"} ++ payload.sale]]>
            </ee:set-payload>
        </ee:message>
    </ee:transform>
    <file:write doc:name="File: sales.csv" path="file-store/sales.csv">
        </file:write>
    </flow>
```

- A.  The enriched payload in JSON format
- B.  An error message
- C.  The enriched payload in XML format
- D.  The enriched payload in CSV format

- 28 of Refer to the exhibits. This Mule application has an HTTP Request that is configured with hardcoded values. To change this, the Mule  
60. application is configured to use a properties file named config.yaml.

What valid expression can the HTTP Request host value be set to so that it is no longer hardcoded?

The screenshot shows the Mule Studio interface with two main panes. On the left is the component configuration pane, and on the right is the properties file pane.

**Component Configuration (Left Pane):**

- General Tab:** Selected tab.
- Settings Tab:** Unselected tab.
- Notes Tab:** Unselected tab.
- Help Tab:** Unselected tab.
- Basic Settings:**
  - Name:** HTTP\_Request\_config
- URL Configuration:**
  - Base path:** /
- Connection:**
  - Configuration:**
    - Protocol:** HTTP (Default)
    - Host:** mu.learn.mulesoft.com
    - Port:** 80

### config.yaml

```
1 @ training:  
2   host: "mu.learn.mulesoft.com"  
3   port: "80"  
4   basepath: "/"  
5   protocol: "HTTP"
```

- A.  #[training.host]
- B.  \${training:host}
- C.  #[training:host]
- D.  \${training.host}

## MCD - Level 1 (Mule 4)

Time Remaining: 1:45:56

29 of 60. Why would a Mule application use the \${http.port} property placeholder for its HTTP Listener port when it is deployed to CloudHub?

- A.  Allows MuleSoft Support to troubleshoot the application by connecting directly to the HTTP Listener
- B.  Allows clients to VPN directly to the application at the Mule application's configured HTTP port
- C.  Allows CloudHub to automatically change the HTTP port to allow external clients to connect to the HTTP Listener
- D.  Allows CloudHub to automatically register the application with API Manager

Mark this item for later review.



## MCD - Level 1 (Mule 4)

Time Remaining: 1:45:36

**30 of 60.** By default, what happens to a file after it is read using an FTP connector Read operation?

- A.  The file is renamed in the same folder
- B.  The file stays in the same folder unchanged
- C.  The file is moved to a different folder
- D.  The file is deleted from the folder

Mark this item for later review.

## MCD - Level 1 (Mule 4)

Time Remaining: 1:45:18

**31 of 60.** According to MuleSoft, what is the first step to create a Modern API?

- A.  Performance tune and optimize the backend systems and network
- B.  Create a prototype of the API implementation
- C.  Create an API specification and get feedback from stakeholders
- D.  Gather a list of requirements to secure the API

Mark this item for later review.

## MCD - Level 1 (Mule 4)

Time Remaining: 1:45:07

**32 of** A Mule application contains a global error handler configured to catch any errors.

**60.**

Where must the global error handler be specified so that the global error handler catches all errors from flows without their own error handlers?

- A.  A configuration properties file
- B.  The pom.xml file
- C.  Nowhere, the global error handler is automatically used
- D.  A global element

---

Mark this item for later review.

## MCD - Level 1 (Mule 4)

Time Remaining: 1:45:02

**33 of 60.** A web client sends a request to <http://localhost:8081?dept=sales>.

What is the correct DataWeave expression to access the value of dept?

- A.  attributes.dept
- B.  message.queryParams.dept
- C.  attributes.queryParams.dept
- D.  vars.dept

Mark this item for later review.

## MCD - Level 1 (Mule 4)

Time Remaining: 1:44:55

**34 of 60.** Refer to the exhibit. What data is expected by the POST /accounts endpoint?

```
#%RAML 1.0
title: Accounts API
version: 1.0

/accounts:
  get:
    description: Get all accounts
    responses:
      200:
        body:
          application/json:
            example:
              id: "48292"
              name: Geordi La Forge
              address: 1 Forge Way, Midgard, CA 95928
              customer_since: "2014-01-04"
              balance: 4829.29
  post:
    description: Create an account
    body:
      application/json:
        example:
          name: Geordi La Forge
```



```
responses:
  200:
    body:
      application/json:
        example:
          id: "48292"
          name: Geordi La Forge
          address: 1 Forge Way, Midgard, CA 95928
          customer_since: "2014-01-04"
          balance: 4829.29
  post:
    description: Create an account
    body:
      application/json:
        example:
          name: Geordi La Forge
          address: 1 Forge Way, Midgard, CA 95928
          customer_since: "2014-01-04"
          balance: 4829.29
          bank_agent_id: "48-SJT-282924-KL"
```

Refer to the exhibit. What data is expected by the POST /accounts endpoint?

- A.  <item>
- ```
<name>Geordi La Forge</name>
<address>1 Forge Way, Midgard, CA 95928</address>
<customer_since>2014-01-04</customer_since>
<balance>4829.29</balance>
<bank_agent_id>48-SJT-282924-KL</bank_agent_id>
</item>
```

- B.  {

A.  <item>  
    <name>Geordi La Forge</name>  
    <address>1 Forge Way, Midgard, CA 95928</address>  
    <customer\_since>2014-01-04</customer\_since>  
    <balance>4829.29</balance>  
    <bank\_agent\_id>48-SJT-282924-KL</bank\_agent\_id>  
  </item>

B.  {  
    "name": "Geordi La Forge",  
    "address": "1 Forge Way, Midgard, CA 95928",  
    "customer\_since": "2014-01-04",  
    "balance": 4829.29,  
    "bank\_agent\_id": "48-SJT-282924-KL"  
}

C.  {  
    "id": "48292",  
    "name": "Geordi La Forge",  
    "address": "1 Forge Way, Midgard, CA 95928",  
    "customer\_since": "2014-01-04",  
    "balance": 4829.29  
}

D.  <item>    
    <id>48292</id>  
    <name>Geordi La Forge</name>  
    <address>1 Forge Way, Midgard, CA 95928</address>  
    <customer\_since>2014-01-04</customer\_since>  
    <balance>4829.29</balance>  
  </item>

## MCD - Level 1 (Mule 4)

Time Remaining: 1:44:27

- 35 of A company has an API to manage departments, with each department identified by a unique deptId. The API was built with RAML  
60. according to MuleSoft best practices.

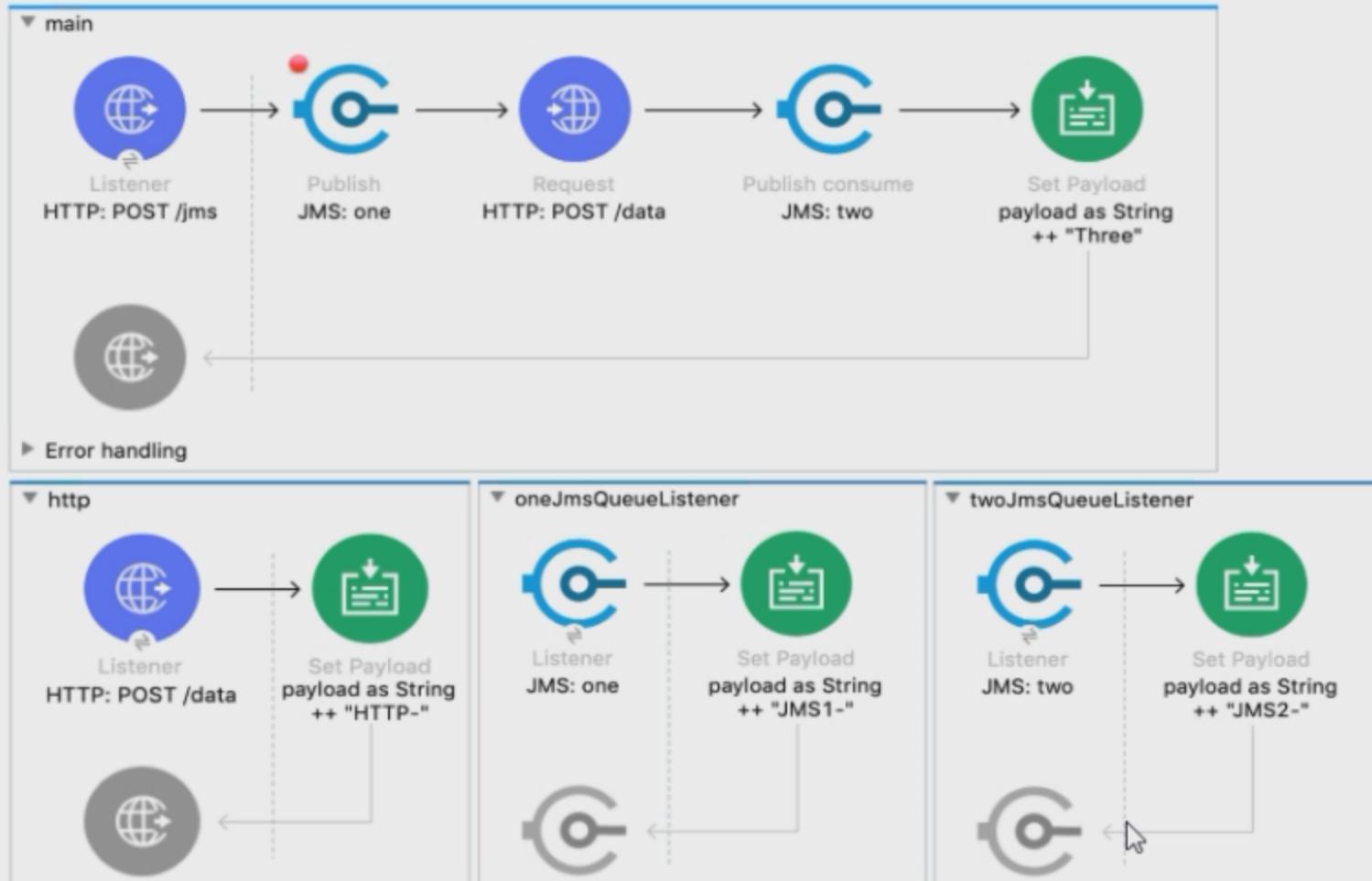
What is valid RAML to specify a method to update the details for a specific department?

- A.  /departments:  
    patch:  
    /deptId:
- B.  /departments:  
    /deptId:  
    patch:
- C.  /departments:  
    patch:  
    queryParameters:  
    deptId:
- D.  /departments:  
    /{deptId}:  
    patch:



Mark this item for later review.

- 36 Refer to the exhibits. A web client sends a POST request to the HTTP Listener with the payload "Hello-".  
of  
60. What response is returned to the web client?





▶ Error handling

▼ http



Listener  
HTTP: POST /data



Set Payload  
payload as String  
++ "HTTP-"



▶ Error handling

▼ oneJmsQueueListener



Listener  
JMS: one



Set Payload  
payload as String  
++ "JMS1-"



▶ Error handling

▼ twoJmsQueueListener



Listener  
JMS: two



Set Payload  
payload as String  
++ "JMS2-"



▶ Error handling

```
<flow name="main">
    <http:listener doc:name="HTTP: POST /jms" config-ref="HTTP_Listener_config" path="/jms" />
    <jms:publish doc:name="JMS: one" config-ref="JMS_Config" destination="one" >
        <jms:message outboundContentType="text/plain" />
    </jms:publish>
    <http:request method="POST" doc:name="HTTP: POST /data" url="http://localhost:8081/data"/>
    <jms:publish-consume doc:name="JMS: two" config-ref="JMS_Config" destination="two" >
        <jms:message outboundContentType="text/plain" />
    </jms:publish-consume>
    <set-payload value="#[payload ++ "Three"]" doc:name='payload as String ++ "Three"' />
```

Refer to the exhibits. A web client sends a POST request to the HTTP Listener with the payload "Hello".



▶ Error handling

▶ Error handling

▶ Error handling

```
<flow name="main">
    <http:listener doc:name="HTTP: POST /jms" config-ref="HTTP_Listener_config" path="/jms" />
    <jms:publish doc:name="JMS: one" config-ref="JMS_Config" destination="one" >
        <jms:message outboundContentType="text/plain" />
    </jms:publish>
    <http:request method="POST" doc:name="HTTP: POST /data" url="http://localhost:8081/data"/>
    <jms:publish-consume doc:name="JMS: two" config-ref="JMS_Config" destination="two">
        <jms:message outboundContentType="text/plain" />
    </jms:publish-consume>
    <set-payload value="#[payload ++ "Three"]" doc:name='payload as String ++ "Three"' />
```

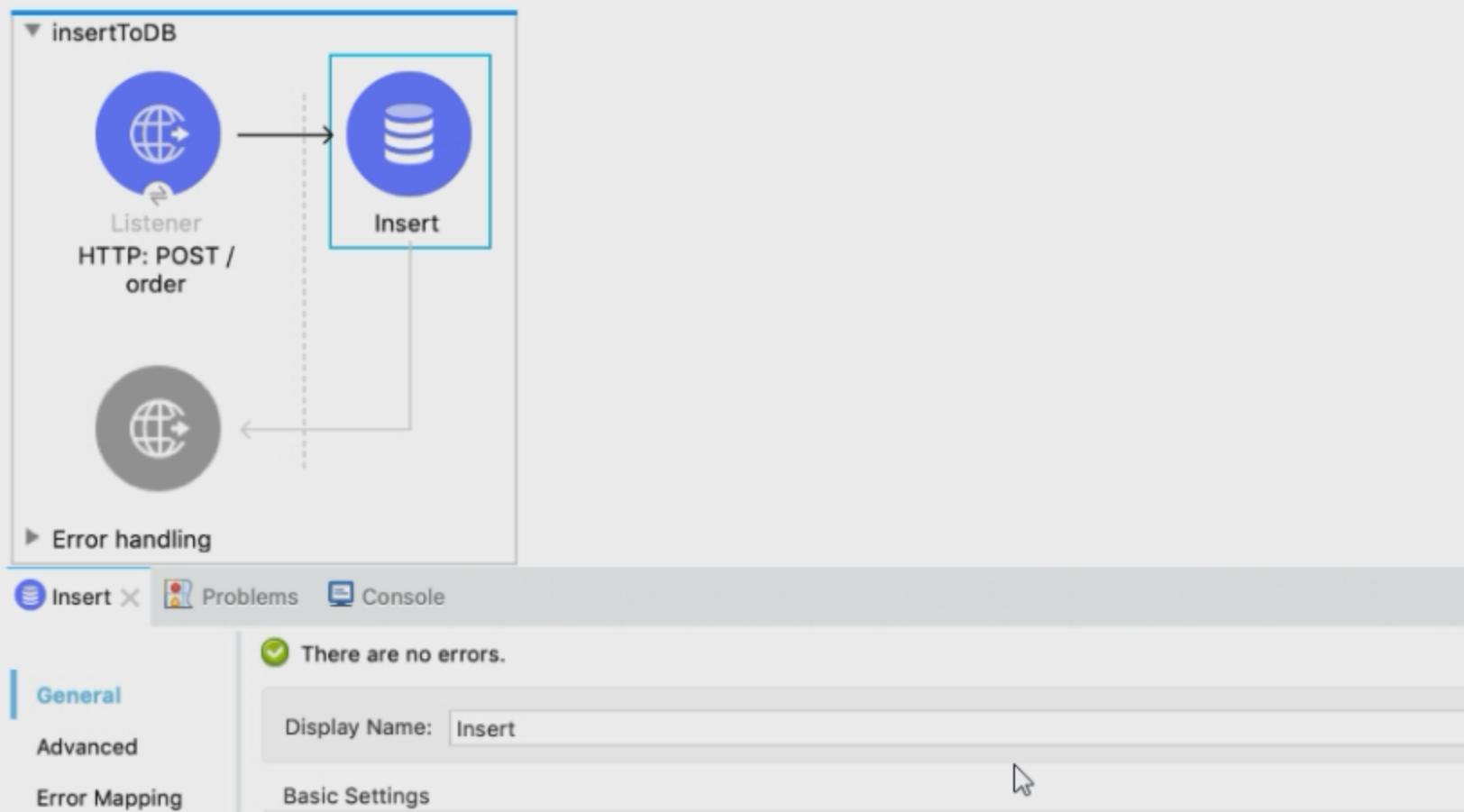
Refer to the exhibits. A web client sends a POST request to the HTTP Listener with the payload "Hello-".

What response is returned to the web client?

- A.  Hello-JMS1-HTTP-JMS2-Three
- B.  HTTP-JMS2-Three
- C.  Hello-HTTP-JMS2-Three
- D.  Hello-HTTP-Three

- 37 of** Refer to the exhibits. A web client sends a new order record { "oid": "100", "custId": "annie@acme.com", "status": "NEW ORDER" } in  
60. the payload of a POST request to the Mule application.

What value must be used in the Input Parameters field of the Database Insert operation to properly pass the order record values to the SQL statement?



The screenshot shows the Mule Studio interface with the following details:

- Toolbar:** Includes icons for Insert (selected), Problems, and Console.
- Left Sidebar:** Contains tabs for General (selected), Advanced, Error Mapping, Metadata, Notes, and Help.
- Central Area:**
  - Message Status:** A green checkmark icon with the text "There are no errors."
  - Basic Settings:** Display Name: Insert
  - Connector configuration:** Database\_Config
  - Query:** SQL Query Text:  
INSERT INTO orders.ORDER (orderId, CustomerName, status, startDate) VALUES (:oid, :customerId, :status, now() )
  - Input Parameters:** fx 1

Refer to the exhibits. A web client sends a new order record { "oid": "100", "customerId": "annie@acme.com", "status": "NEW ORDER" } in the payload of a POST request to the Mule application.

What value must be used in the Input Parameters field of the Database Insert operation to properly pass the order record values to the SQL statement?

What value must be used in the Input Parameters field of the Database Insert operation to properly pass the order record values to the SQL statement?

- A.  "#["  
    {  
        oid: payload.oid,  
        custId: payload.custId,  
        status: payload.status  
    }  
]"
- B.  "#["  
    inputParams: [  
        payload.oid,  
        payload.custId,  
        payload.status  
    ]  
]"
- C.  "#["  
    [  
        payload.oid,  
        payload.custId,  
        payload.status  
    ]  
]"
- D.  "#["  
    {  
        orderId: payload.oid,  
        customerName: payload.custId,  
        status: payload.status  
    }  
]"

- 38 Refer to the exhibits. A web client sends a GET request to the HTTP Listener and the HTTP Request throws an error.  
of  
60. What payload and status code are returned to the web client?

HTTP: GET / X

General

MIME Type

Redelivery

**Responses**

Advanced

Metadata

Notes

Help

There are no errors.

Response

Body:

Headers:

Status code:

Reason phrase:

Error Response

Body:

Headers:

main

Listener HTTP: GET / → Set Payload "START" → Request HTTP: GET /data → Set Payload "END"

Error handling

On Error Continue

Set Payload "ERROR1"

## Error Response

Body:



```
1@ output text/plain  
2 --- error.description
```

Headers:



## Headers



Name

Value

Status code:



Reason phrase:



```
<flow name="main">  
    <http:listener doc:name="HTTP: GET /" config-ref="HTTP_Listener_config" path="/" >  
    </http:listener>  
    <set-payload value='START' doc:name='START' />  
    <http:request method="GET" doc:name="HTTP: GET /data" url="http://mu.learn.mulesoft.com/data"/>  
    <set-payload value='END' doc:name='END' />  
    <error-handler>  
        <on-error-continue enableNotifications="true" logException="true" doc:name="On Error Continue">  
            <set-payload value="ERROR1" doc:name="ERROR1" />  
        </on-error-continue>  
    </error-handler>  
</flow>
```

Set Payload  
"ERROR1"

Refer to the exhibits. A web client sends a GET request to the HTTP Listener and the HTTP Request throws an error.

What payload and status code are returned to the web client?

- A.  Response body: "START"

```
</http:listener>
<set-payload value='"START"' doc:name='"START"' />
<http:request method="GET" doc:name="HTTP: GET /data" url="http://mu.learn.mulesoft.com/data"/>
<set-payload value='"END"' doc:name='"END"' />
<error-handler>
    <on-error-continue enableNotifications="true" logException="true" doc:name="On Error Continue">
        <set-payload value="ERROR1" doc:name='"ERROR1"' />
    </on-error-continue>
</error-handler>
</flow>
```

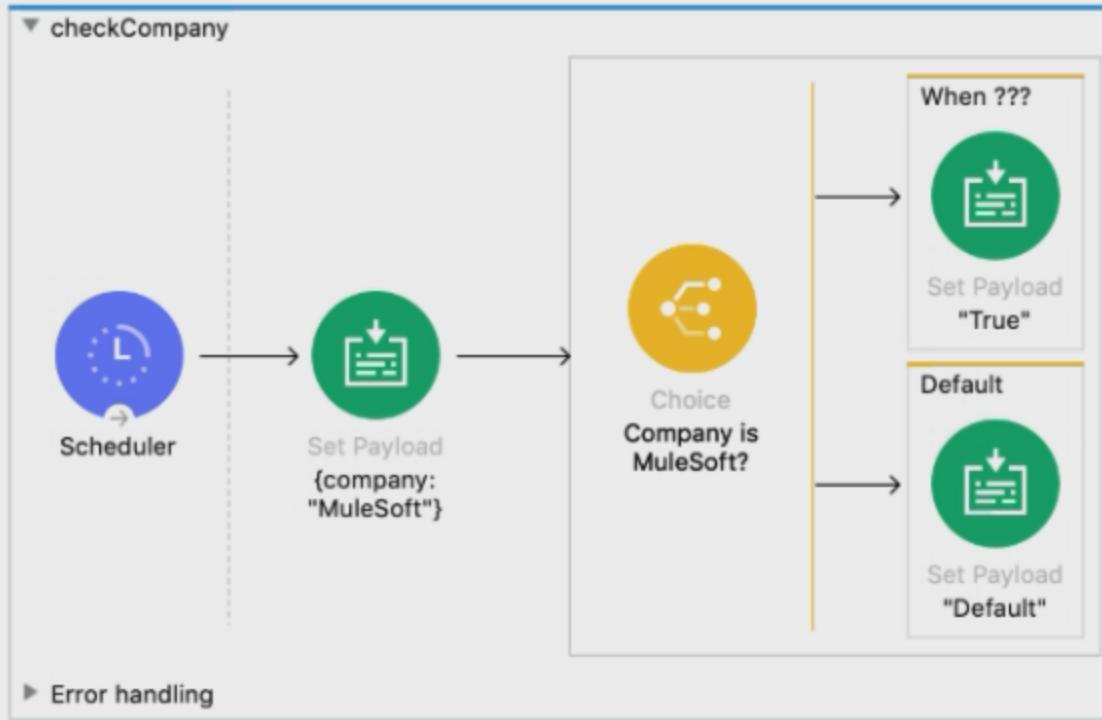
Refer to the exhibits. A web client sends a GET request to the HTTP Listener and the HTTP Request throws an error.

What payload and status code are returned to the web client?

- A.  Response body: "START"  
Default response status code: 200
- B.  Error response body: error.description  
Default error response status code: 500
- C.  Response body: "ERROR1"  
~~Default response status code: 200~~
- D.  Response body: "END"  
Default response status code: 200

39 of 60. Refer to the exhibits. The <when> expression for the Choice router needs to be written.

What is a valid <when> expression to route Mule events to the non-default flow?



```
<flow name="checkCompany">
  <scheduler doc:name="Scheduler">
    <scheduling-strategy>
      <fixed-frequency frequency="5000" />
    </scheduling-strategy>
  </scheduler>
  <choice>
    <when>Company is MuleSoft?</when>
      <set-payload value="True" />
    <otherwise>
      <set-payload value="Default" />
    </otherwise>
  </choice>
</flow>
```

## ▶ Error handling

```
<flow name="checkCompany">
    <scheduler doc:name="Scheduler">
        <scheduling-strategy>
            <fixed-frequency frequency="5000" />
        </scheduling-strategy>
    </scheduler>
    <set-payload value="#[{company: "MuleSoft"}]" doc:name="{company: "MuleSoft"}" />
    <choice doc:name="Company is MuleSoft?">
        <when expression="When ???">
            <set-payload value="#["True"]" doc:name="True" />
        </when>
        <otherwise>
            <set-payload value="#["Default"]" doc:name="Default" />
        </otherwise>
    </choice>
</flow>
```

Refer to the exhibits. The <when> expression for the Choice router needs to be written.

What is a valid <when> expression to route Mule events to the non-default flow?

- A.  #[ company = "MuleSoft" ]
- B.  #[ if( 'MuleSoft' == payload.company ) ]
- C.  #[ if( company = "MuleSoft" ) ]
- D.  #[ 'MuleSoft' == payload.'company' ]

- 40 of** Refer to the exhibits. A web client sends a POST request with the payload { "oid": "1000", "itemid": "AC200", "qty": "4" } to the Mule application. The File Write operation throws a FILE:CONNECTIVITY error.
- 60.** What response message is returned to the web client?



type: FILE:CONNECTIVITY

The screenshot shows the configuration for a 'Write' component. The 'Error Mapping' tab is selected. It maps a 'FILE:CONNECTIVITY' error to an 'ORDER:NOT\_CREATED' target. A note indicates there are no errors, and a 'Set Payload' step is configured to return 'OTHER ERROR'.

General Advanced Error Mapping Metadata Notes

There are no errors.

Error types to be mapped: FILE:CONNECTIVITY Map to: ORDER:NOT\_CREATED

Set Payload "OTHER ERROR"

```
<flow name="acceptOrder">
    <http:listener doc:name="HTTP: POST /order" config-ref="HTTP_Listener_config"
        path="/order" allowedMethods="POST">
        <http:error-response>
            <http:body><!CDATA[#[output text/plain --- payload]]></http:body>
        </http:error-response>
    </http:listener>
    <file:write doc:name="Write" config-ref="File_Config" path="newOrder.json">
        <error-mapping sourceType="FILE:CONNECTIVITY" targetType="ORDER:NOT_CREATED" />
        <file:content><!CDATA[#[output application/json --- payload]]></file:content>
    </file:write>
    <set-payload value="#["File written"]" doc:name="File written" />
```

Refer to the exhibits. A web client sends a POST request with the payload { "oid": "1000", "itemid": "AC200", "qty": "4" } to the Mule application. The File Write operation throws a FILE:CONNECTIVITY error.

What response message is returned to the web client?

- A.  "FILE:CONNECTIVITY"

## Error Mapping

Metadata

Notes

Error types to be mapped:

FILE:CONNECTIVITY

Map to:

ORDER:NOT\_CREATED

Set Payload  
"OTHER ERROR"

```
<flow name="acceptOrder">
    <http:listener doc:name="HTTP: POST /order" config-ref="HTTP_Listener_config"
        path="/order" allowedMethods="POST">
        <http:error-response>
            <http:body><![CDATA[#[output text/plain --- payload]]]></http:body>
        </http:error-response>
    </http:listener>
    <file:write doc:name="Write" config-ref="File_Config" path="newOrder.json">
        <error-mapping sourceType="FILE:CONNECTIVITY" targetType="ORDER:NOT_CREATED" />
        <file:content><![CDATA[#[output application/json --- payload]]]></file:content>
    </file:write>
    <set-payload value="#["File written"]" doc:name='File written'" />
```

Refer to the exhibits. A web client sends a POST request with the payload { "oid": "1000", "itemid": "AC200", "qty": "4" } to the Mule application. The File Write operation throws a FILE:CONNECTIVITY error.

What response message is returned to the web client?

- A.  "FILE:CONNECTIVITY"
- B.  "OTHER ERROR"
- C.  "File written"
- D.  "ORDER:NOT\_CREATED"



## MCD - Level 1 (Mule 4)

Time Remaining: 1:42:27

- 41 of** A company has an API to manage purchase orders, with each record identified by a unique purchase order ID. The API was built  
60. with RAML according to MuleSoft best practices.

What URI should a web client use to request order PO5555?

- A.  /orders/{PO5555}
- B.  /orders/order=PO5555
- C.  /orders/PO5555
- D.  /orders?order=PO5555

Mark this item for later review.

Time Remaining: 1:42:18

- 42 An app team is developing a mobile banking app. It took them two months to create their own APIs to access transaction information  
of from a central database. The app team later found out that another team had already built an API that accesses the transaction  
60. information they need.

According to MuleSoft, what organization structure could have saved the app team two months of development time?

- A.  Center for Enablement
- B.  Central API Review Board
- C.  Center of Excellence
- D.  MuleSoft Support Center

Mark this item for later review.

43 of 60. Refer to the exhibits. A company has defined this Book data type and Book example to be used in APIs.

What is valid RAML for an API that uses this Book data type and Book example?

```
#%RAML 1.0 DataType          #%RAML 1.0 NamedExample
# bookDataType.raml           # bookExample.raml

type: object                  bookExample:
properties:                   ID: 101
    ID?: integer              title: Shakespeare
    title: string              author: Encyclopaedia Britannica
    author: string             publisher: John Wiley & Sons
    publisher?: string         year: 2007
    year: integer              ISBN: "0471767840"
    ISBN:
        type: string
        required: true
```

- A.  #%RAML 1.0  
title: Books  
  
Book: BookDataType.raml  
  
/books:  
 post:  
 body:  
 application/json:  
 type: Book  
 examples:



```
    type: string  
    required: true
```

A.  #**%RAML 1.0**  
title: Books

Book: BookDataType.raml

```
/books:  
  post:  
    body:  
      application/json:  
        type: Book  
        examples:  
          input: BookExample.raml  
    responses:  
      201:  
        body:  
          application/json:  
            example:  
              message: Book added
```

B.  #**%RAML 1.0**  
title: Books

Book: !include BookDataType.raml

```
/books:  
  post:  
    body:  
      application/json:  
        type: Book  
        examples:  
          input: !include BookExample.raml  
    responses:  
      201:  
        body:  
          application/json:  
            example:  
              message: Book added
```

message: Book added

C.  #**RAML 1.0**  
title: Books

types:  
    Book: ABC/DataTypes/BookDataType.raml

/books:  
    post:  
        body:  
            application/json:  
                type: Book  
                examples:  
                    input: ABC/Examples/BookExample.raml  
                responses:  
                    201:  
                        body:  
                            application/json:  
                            example:  
                                message: Book added

D.  #**RAML 1.0**  
title: Books

types:  
    Book: !include BookDataType.raml

/books:  
    post:  
        body:  
            application/json:  
                type: Book  
                examples:  
                    input: !include BookExample.raml  
                responses: 201:  
                    body:  
                            application/json:  
                            example:  
                                message: Book added

44 of 60. Refer to the exhibit. This RAML specification includes a resource and method to retrieve accounts by account\_type and industry.

What is the correct URI to get all retail finance accounts?

```
#%RAML 1.0
title: Accounts API

/accounts:
  get:
    queryParameters:
      account_type:
        required: true
        enum:
          - "retail"
          - "commercial"
      industry:
        required: true
        enum:
          - "finance"
          - "construction"
          - "government"
```



- A.  /accounts/account\_type=retail/industry=finance
- B.  /accounts?account\_type:retail&industry:finance
- C.  /accounts/retail/finance
- D.  /accounts?account\_type=retail&industry=finance

## MCD - Level 1 (Mule 4)

Time Remaining: 1:41:38

**45 of** An SLA based policy has been enabled in API Manager. What is the next step to configure the API proxy to enforce the new SLA policy?

- A.  Add new property placeholders and redeploy the API proxy
- B.  Add required headers to the RAML specification and redeploy the new API proxy
- C.  Add new environment variables and restart the API proxy
- D.  Restart the API proxy to clear the API policy cache

Mark this item for later review.



## MCD - Level 1 (Mule 4)

Time Remaining: 1:41:34

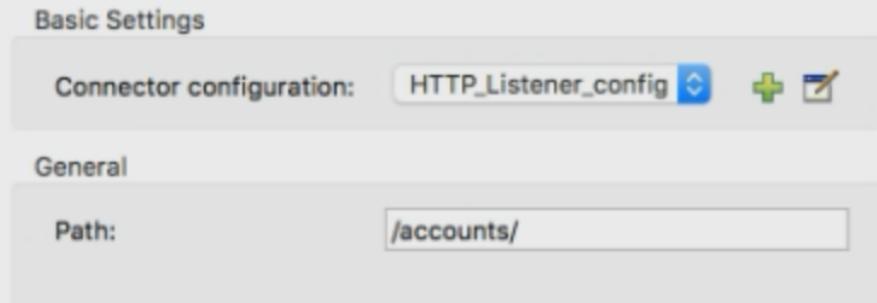
- 46 of** Refer to the exhibit. What is the correct syntax to add a customer ID as a URI parameter in an HTTP Listener path?  
60.

Basic Settings

Connector configuration: **HTTP\_Listener\_config**   

General

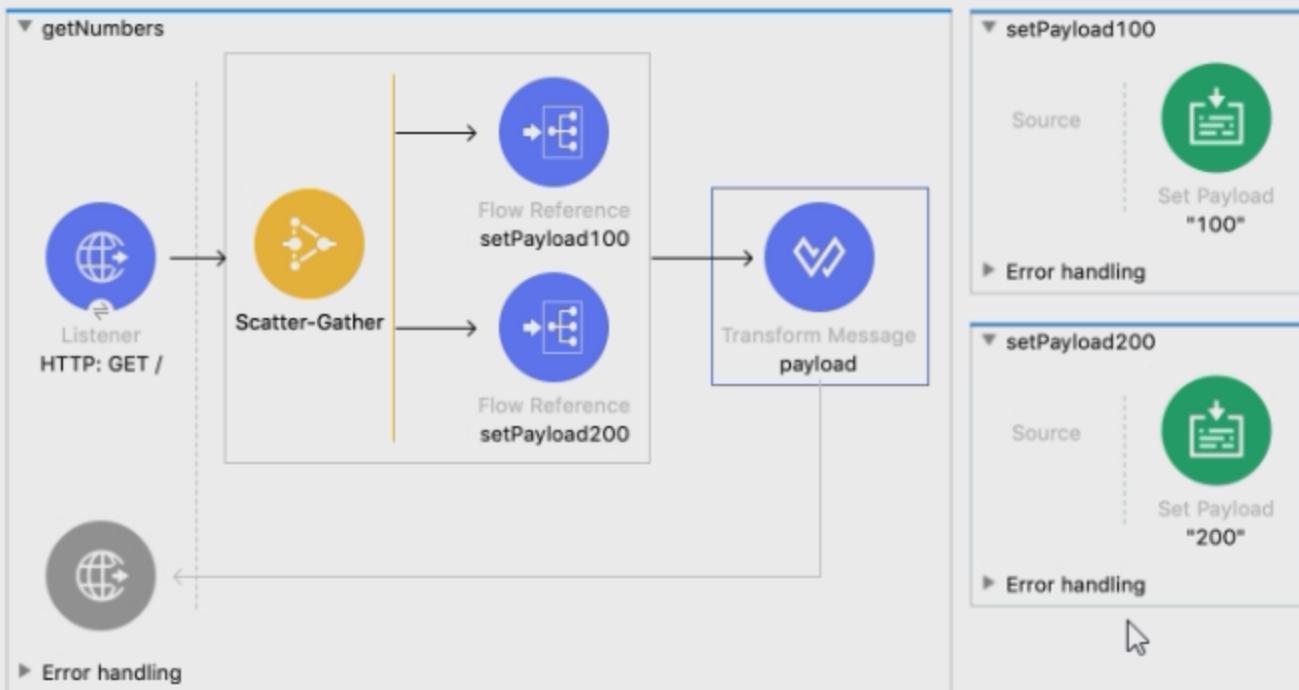
Path: **/accounts/**



- A.  #[customerID]
- B.  \${customerID}
- C.  {customerID}
- D.  (customerID)

- 47 of 60 Refer to the exhibits. Each route in the Scatter-Gather sets the payload to the number shown in the label.

What response is returned to a web client request to the HTTP Listener?



## ▶ Error handling

```
<flow name="getNumbers" >
    <http:listener doc:name="HTTP: GET /" config-ref="HTTP_Listener_config" path="/" />
    <scatter-gather doc:name="Scatter-Gather" >
        <route >
            <flow-ref doc:name='setPayload100' name='setPayload100' />
        </route>
        <route >
            <flow-ref doc:name="setPayload200" name="setPayload200" />
        </route>
    </scatter-gather>
    <ee:transform doc:name="payload">
        <ee:message >
            <ee:set-payload ><! [CDATA[%dw 2.0
                output application/json
                ---
                payload]]></ee:set-payload>
        </ee:message>
    </ee:transform>
</flow>
<flow name="setPayload100" ><set-payload value="#["100"]" doc:name='"100"' /></flow>
<flow name="setPayload200" ><set-payload value="#["200"]" doc:name='"200"' /></flow>
```

Refer to the exhibits. Each route in the Scatter-Gather sets the payload to the number shown in the label.

What response is returned to a web client request to the HTTP Listener?

- A.  {  
 "0": {  
 "attributes":



Refer to the exhibits. Each route in the Scatter-Gather sets the payload to the number shown in the label.

What response is returned to a web client request to the HTTP Listener?

A.  {

```
    "0": {
        "attributes": ...,
        "payload": "100"
    },
    "1": {
        "attributes": ...,
        "payload": "200"
    }
}
```

B.  [

```
{
    "0": {
        "attributes": ...,
        "payload": "100"
    },
    "1": {
        "attributes": ...,
        "payload": "200"
    }
}
```

C.  {

```
    "0": "100",
    "1": "200"
}
```

D.  ["100", "200"]

## MCD - Level 1 (Mule 4)

Time Remaining: 1:41:14

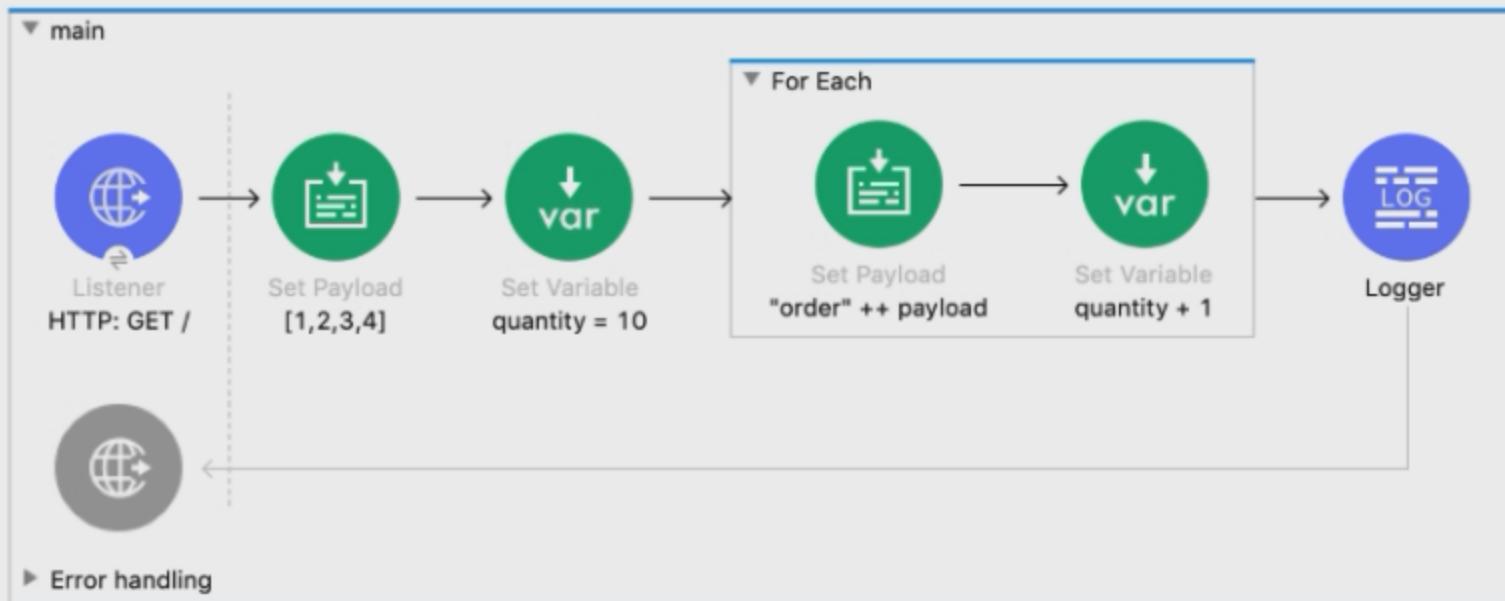
- 48 of A function named newProdCode needs to be defined that accepts two input parameters, an integer value for itemID and a string  
60. value for productCategory, and returns a new product code.

What is the correct DataWeave code to define the newProdCode function?

- A.  `fun newProdCode(itemID: Number, productCategory: String) ->  
 "PC-" ++ productCategory ++ (itemID as String)`
- B.  `var newProdCode(itemID: Number, productCategory: String) ->  
 "PC-" ++ productCategory ++ (itemID as String)`
- C.  `function newProdCode(itemID: Number, productCategory: String) =  
 "PC-" ++ productCategory ++ (itemID as String)`
- D.  `fun newProdCode(itemID: Number, productCategory: String) =  
 "PC-" ++ productCategory ++ (itemID as String)`

49 of 60. Refer to the exhibits.

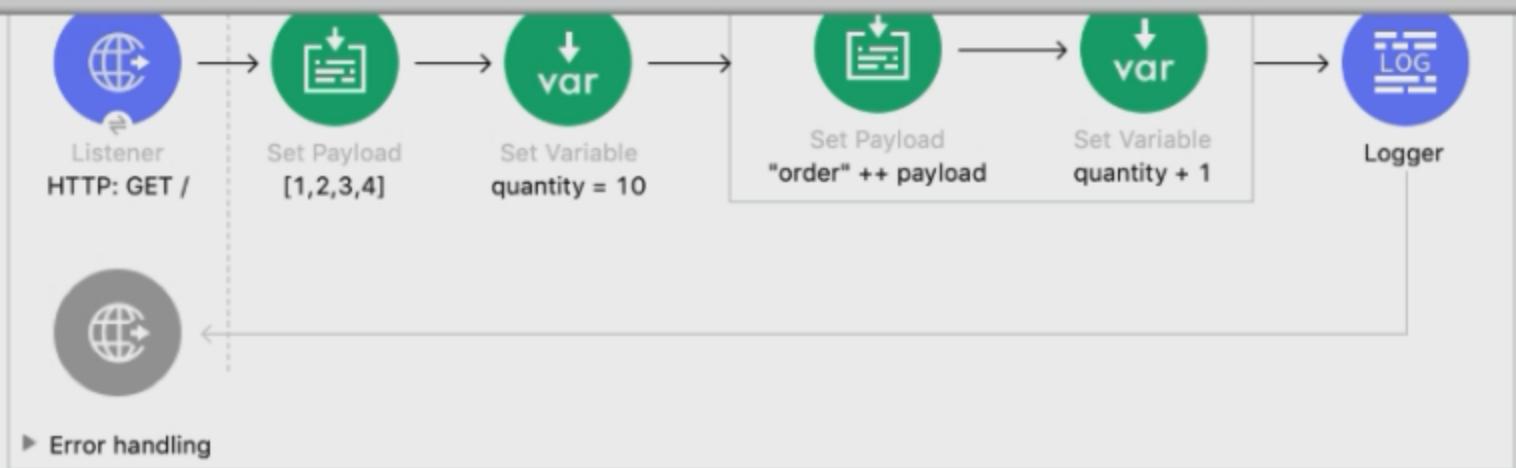
What payload and quantity are logged at the end of the main flow?



```

<?xml version="1.0" encoding="UTF-8"?>
<flow name="main">
    <http:listener doc:name="HTTP: GET /" config-ref="HTTP_Listener_config" path="/" allowedMethods="GET"/>
    <set-payload value="#[[1,2,3,4]]" doc:name="[[1,2,3,4]]" />
    <set-variable value='10' doc:name="quantity = 10" variableName="quantity" />
    <foreach doc:name="For Each">
        <set-payload value="#["order" ++ payload]" doc:name="order" ++ payload" />
        <set-variable value="#[vars.quantity + 1]" doc:name="quantity + 1" variableName="quantity" />
    </foreach>
    <logger level="INFO" doc:name="Logger" message="#[[ payload, vars.quantity ]]" />
</flow>

```



```

<flow name="main" >
    <http:listener doc:name="HTTP: GET /" config-ref="HTTP_Listener_config" path="/" allowedMethods="GET"/>
    <set-payload value="#[[1,2,3,4]]" doc:name="[1,2,3,4]" />
    <set-variable value='10' doc:name="quantity = 10" variableName="quantity" />
    <foreach doc:name="For Each" >
        <set-payload value='#[ "order" ++ payload ]' doc:name="order" ++ payload' />
        <set-variable value="#[vars.quantity + 1]" doc:name="quantity + 1" variableName="quantity" />
    </foreach>
    <logger level="INFO" doc:name="Logger" message='#[[ payload, vars.quantity ]]'/>
</flow>

```

- A.  [[order1, order2, order3, order4], 14]
- B.  [[1,2,3,4], 10]
- C.  [[1,2,3,4], 14]
- D.  [order1order2order3order4, 14]



## MCD - Level 1 (Mule 4)

Time Remaining: 1:40:49

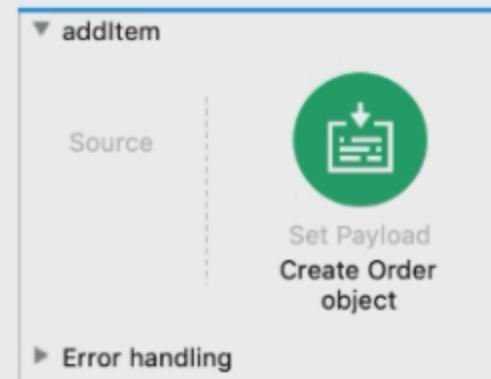
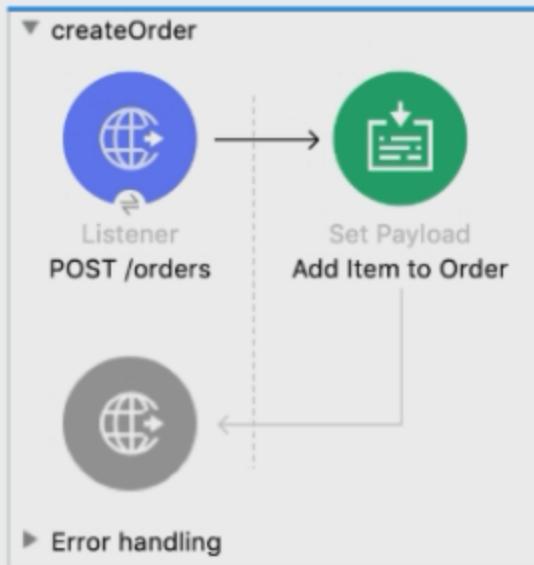
- 50 of** An API implementation has been deployed to CloudHub and now needs to be governed. IT will not allocate additional vCores for a new Mule application to act as an API proxy.

What is the next step to preserve the current vCore usage, but still allow the Mule application to be managed by API Manager?

- A.  Modify the API implementation to use auto-discovery to register with API Manager
- B.  Deploy the same API implementation behind a VPC and configure the VPC to connect to API Manager
- C.  Upload the Mule application's JAR file to the API instance in API Manager
- D.  Register the same API implementation in Runtime Manager to connect to API Manager

Mark this item for later review.

- 51 Refer to the exhibits. The Set Payload transformer in the addItem child flow uses DataWeave to create an order object.
- of
60. What is the correct DataWeave code for the Set Payload transformer in the createOrder flow to use the addItem child flow to add a router with the price of 100 to the order?



```

1 %dw 2.0
2 output application/json
3 ---
4 order: {
5   item: {
6     itemName: payload.item,
7     itemType: payload.itemType
8     price: payload.price
9   }
10 }
  
```

- A. `addItem( { price: "100", item: "router", itemType: "cable" } )`
- B. `lookup( "addItem", { payload: { price: "100", item: "router", itemType: "cable" } } )`
- C. `addItem( { payload: { price: "100", item: "router", itemType: "cable" } } )`
- D. `lookup( "addItem", { price: "100", item: "router", itemType: "cable" } )`

- 52 of Refer to the exhibits. The web client sends a POST request to the ACME Order API with an XML payload. An error is returned.  
60.

What should be changed in the request so that a success response code is returned to the web client?

```
%RAML 1.0
title: ACME Order API
version: 1.0

/order:
  post:
    body:
      application/xml:
        example: |
          <order oid="1001">
            <customerName>Annie Point
            </customerName>
            <itemName>Electric Standing Desk
            </itemName>
            <cost>300.00</cost>
          </order>
```

The screenshot shows a browser interface for making API requests. The URL is `http://localhost:8081/api/...`. The method is set to `POST`. The request URL is `http://localhost:8081/api/order`. The Headers tab shows `Content-Type: multipart/mixed`. The Body tab shows `Raw input`. The response status is `415 Unsupported Media Type` with a duration of `4097.00 ms`. The response body is `{"message": "Unsupported media type"}`.

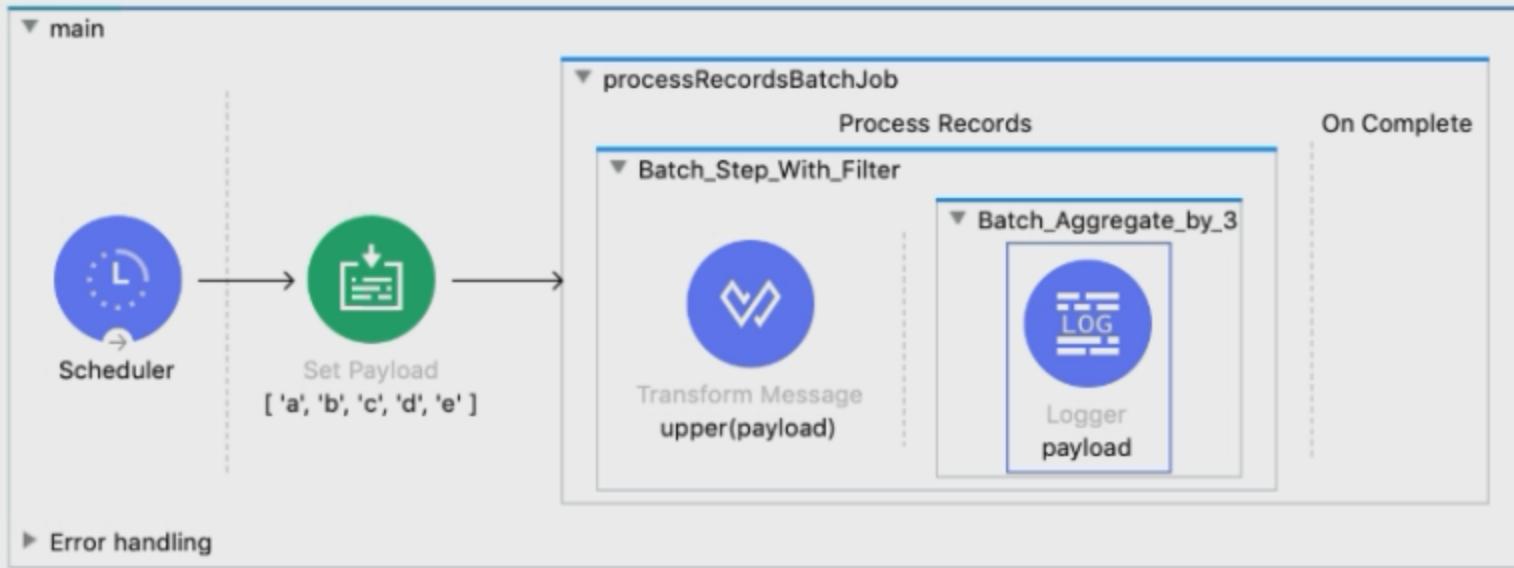
```
%%RAML 1.0
title: ACME Order API
version: 1.0

/order:
  post:
    body:
      application/xml:
        example: |
          <order oid="1001">
            <customerName>Annie Point
            </customerName>
            <itemName>Electric Standing Desk
            </itemName>
            <cost>300.00</cost>
          </order>
```

The screenshot shows a RAML 1.0 API definition on the left and a browser-like interface on the right. The API definition includes a POST method for the /order endpoint with an XML example. The browser interface shows a POST request to `http://localhost:8081/api/order`. The 'Body' tab is selected in the parameters panel, showing 'Body content type' set to 'multipart/mixed'. The response is an error: `415 Unsupported Media Type` with a duration of `4097.00 ms`. The response body is a JSON object with a single key 'message': "Unsupported media type".

- A.  Set a request header with the name Content-Type to a value of application/xml
- B.  Set a response header with the name Content-Type to a value of application/xml
- C.  Set a response header with the name Content-Type to a value of application/octet-stream
- D.  Set a request header with the name Content-Type to a value of application/octet-stream

- 53 Refer to the exhibits. The input array of strings is processed by the batch job that processes, filters, and aggregates the values.  
of  
60. What is the last message logged by the Logger component after the batch job completes processing?



```
<flow name="main" >
    <scheduler doc:name="Scheduler" > <scheduling-strategy >
        <fixed-frequency frequency="10000"/> </scheduling-strategy> </scheduler>
        <set-payload value="#[[ 'a', 'b', 'c', 'd', 'e' ]]" doc:name="["a", "b", "c", "d", "e"]" />
        <batch:job jobName="processRecordsBatchJob" >
            <batch:process-records >
                <batch:step name="Batch_Step_With_Filter"
                    acceptExpression="#[not(payload contains "b") ]">
                    <ee:transform doc:name="upper(payload)"><ee:message >
                        <ee:set-payload ><![CDATA[%dw 2.0
                            output application/json
                            ...]
```

## ▶ Error handling

```
<flow name="main" >
    <scheduler doc:name="Scheduler" > <scheduling-strategy >
        <fixed-frequency frequency="10000"/> </scheduling-strategy> </scheduler>
        <set-payload value="#[[ 'a', 'b', 'c', 'd', 'e' ]]" doc:name="['a', 'b', 'c', 'd', 'e']" />
        <batch:job jobName="processRecordsBatchJob" >
            <batch:process-records >
                <batch:step name="Batch_Step_With_Filter"
                    acceptExpression="#[not (payload contains "b") ]">
                    <ee:transform doc:name="upper(payload)"><ee:message >
                        <ee:set-payload ><!CDATA[%dw 2.0
                            output application/json
                            ---
                            upper(payload)]]></ee:set-payload>
                    </ee:message></ee:transform>
                    <batch:aggregator doc:name="Batch_Aggregate_by_3" size="3">
                        <logger level="INFO" doc:name="payload"
                            message="#[output application/json --- payload]"/>
                    </batch:aggregator>
                </batch:step>
            </batch:process-records>
        </batch:job>
    </flow>
```



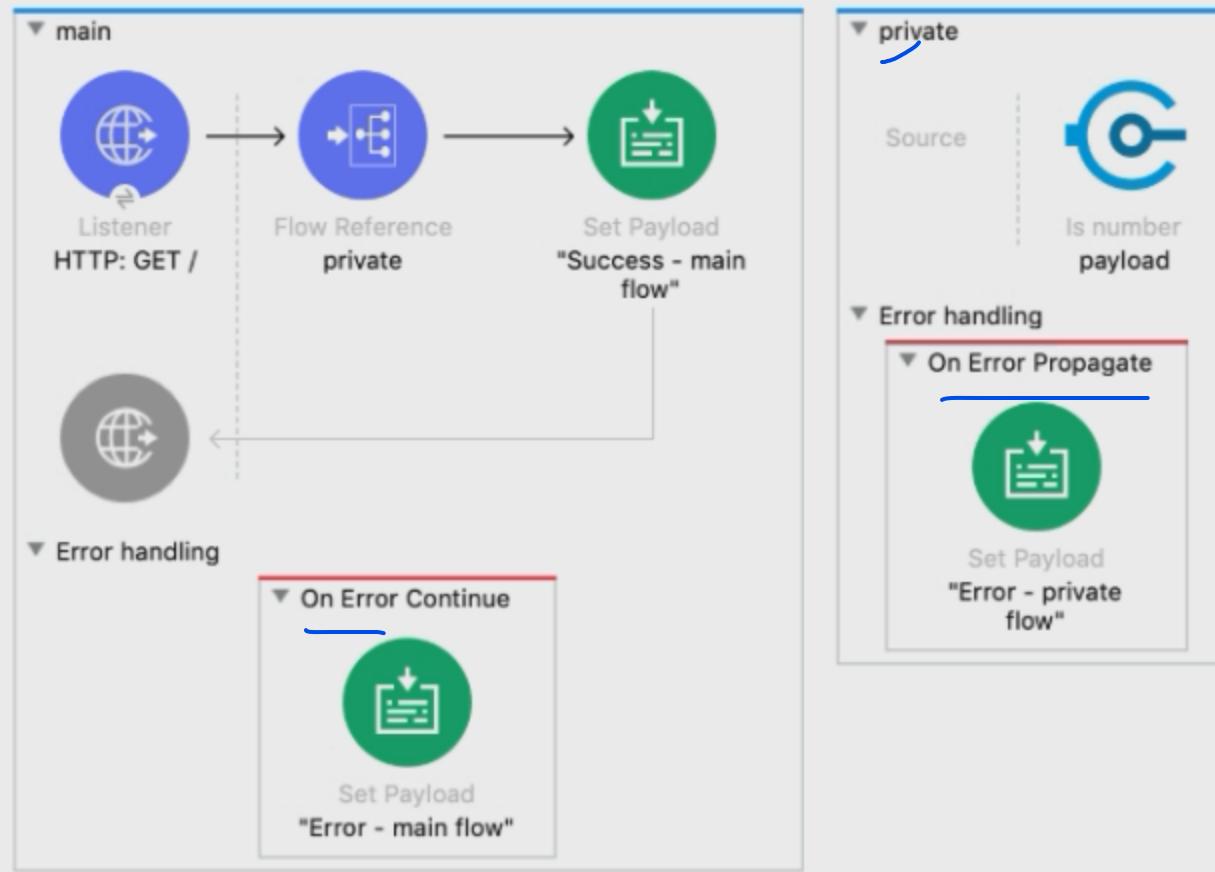
Refer to the exhibits. The input array of strings is processed by the batch job that processes, filters, and aggregates the values.

What is the last message logged by the Logger component after the batch job completes processing?

- A.  [[ "A", "C", "D" ], [ "E" ]]

54 of 60. Refer to the exhibits. The Validation component in the private flow throws an error.

What response message is returned to a web client request to the main flow's HTTP Listener?



```
<flow name="main" >
```

Set Payload  
"Error - main flow"

```
<flow name="main" >
    <http:listener doc:name="HTTP: GET /" config-ref="HTTP_Listener_config" path="/" />
    <flow-ref doc:name="private" name="private"/>
    <set-payload value="Success - main flow" doc:name="Success - main flow" />
    <error-handler>
        <on-error-continue enableNotifications="true" logException="true" doc:name="On Error Continue" >
            <set-payload value="Error - main flow" doc:name="Error - main flow" />
        </on-error-continue>
    </error-handler>
</flow>

<flow name="private" >
    <validation:is-number numberType="INTEGER" doc:name="payload" value="#[payload]" message="Validation Error" />
    <error-handler >
        <on-error-propagate enableNotifications="true" logException="true" doc:name="On Error Propagate" >
            <set-payload value="Error - private flow" doc:name="Error - private flow" />
        </on-error-propagate>
    </error-handler>
</flow>
```

Refer to the exhibits. The Validation component in the private flow throws an error.

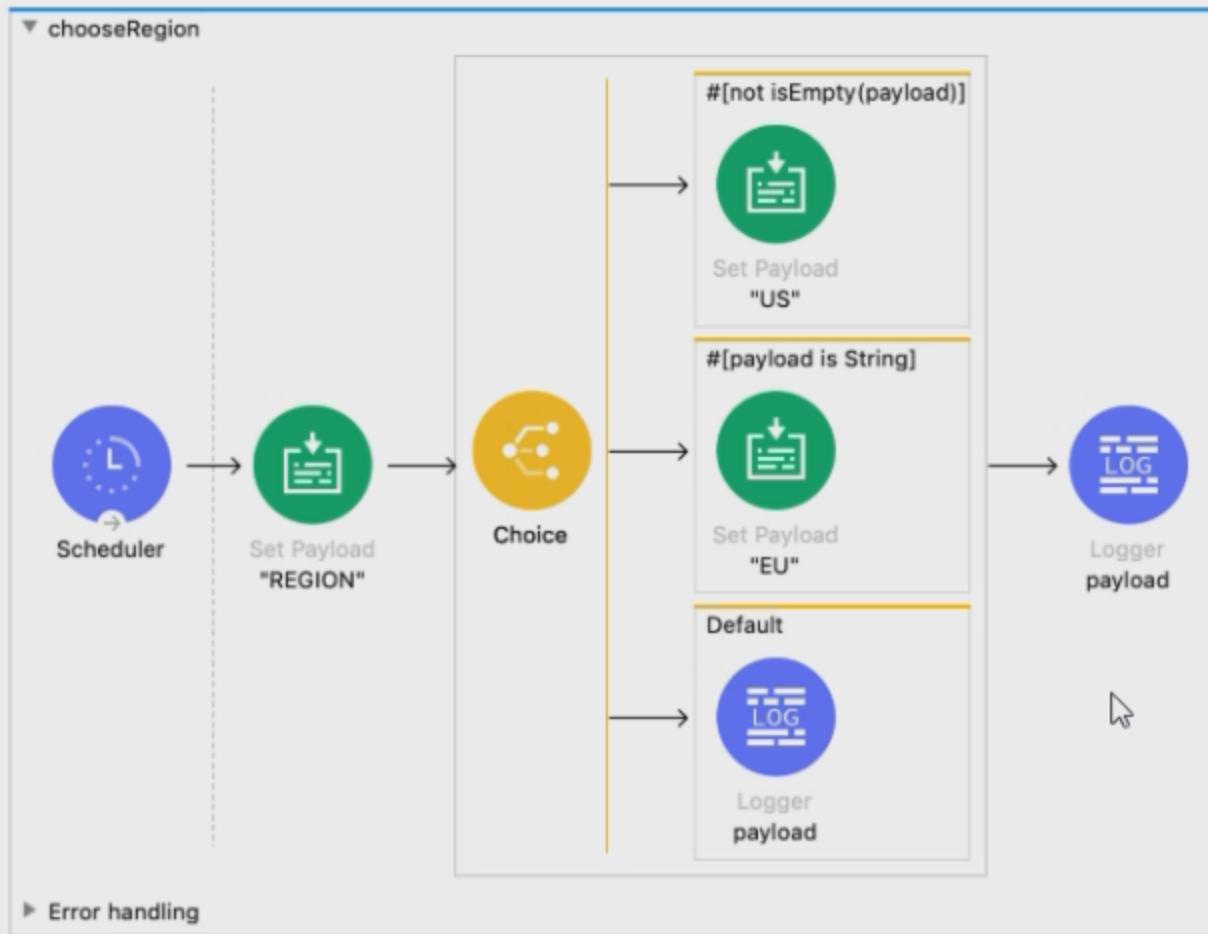
What response message is returned to a web client request to the main flow's HTTP Listener?

- A.  Error - main flow
- B.  Success - main flow
- C.  Validation Error
- D.  Error - private flow



55 of 60. Refer to the exhibits. The Mule application contains a Choice router.

What is logged when the flow completes?



```
<flow name="chooseRegion" >
    <scheduler doc:name="Scheduler" >
        <scheduling-strategy >
            <fixed-frequency frequency="5000"/>
        </scheduling-strategy>
    </scheduler>
    <set-payload value="#["REGION"]" doc:name='"REGION"' />
    <choice doc:name="Choice" >
        <when expression="#[not isEmpty(payload)]">
            <set-payload value="#["US"]" doc:name='"US"' />
        </when>
        <when expression="#[payload is String]">
            <set-payload value="#["EU"]" doc:name='"EU"' />
        </when>
        <otherwise>
            <logger level="INFO" doc:name="payload" message="#[payload]"/>
        </otherwise>
    </choice>
    <logger level="INFO" doc:name="payload" message="#[payload]"/>
</flow>
```

Refer to the exhibits. A Mule application contains a Choice router.

What is logged when the flow completes?

- A.  "REGION"
- B.  EU
- C.  ["US", "EU"]
- D.  US



56 of Refer to the exhibit.

60.

What DataWeave expression transforms the example XML input to the CSV output?

```
<?xml version="1.0" encoding="UTF-8"?>
<sale>
    <item itemId="592" saleId="1000">
        <shipping>international</shipping>
        <desc>T-shirt Navy</desc>
        <size>L</size>
        <quantity>1</quantity>
        <price>20</price>
    </item>
    <item itemId="972" saleId="1000">
        <shipping>domestic</shipping>
        <desc>Cargo Shorts</desc>
        <size>XL</size>
        <quantity>2</quantity>
        <price>30</price>
    </item>
</sale>
```

Output Payload

```
1@%dw 2.0
2   output application/csv
3   ---
4
5
6
7
8
9
10
11
12
13
14
15
16
17
+>
```

index	sale	itemName	itemPrice	item
0	1000	T-shirt Navy	20	592
1	1000	Cargo Shorts	60	972

- A.  payload.sale.\*item map ( (value,index) -> {
 index: index,
 sale: value.saleId,
 itemName: value.desc,
 itemPrice: (value.price) \* (value.quantity),
 item: value.itemId
 } )
- B.  payload.sale.item map ( (value,index) -> {

```
<quantity>2</quantity>          14  
<price>30</price>            15  
</item>                         16  
</sale>                        17  
+ n
```

- A.  payload.sale.\*item map ( (value,index) -> {  
 index: index,  
 sale: value.saleId,  
 itemName: value.desc,  
 itemPrice: (value.price) \* (value.quantity),  
 item: value.itemId  
} )
- B.  payload.sale.item map ( (value,index) -> {  
 index: index,  
 sale: value.@saleId,  
 itemName: value.desc,  
 itemPrice: (value.price) \* (value.quantity),  
 item: value.@itemId  
} )
- C.  payload.sale.item map ( (value,index) -> {  
 index: index,  
 sale: value.saleId,  
 itemName: value.desc,  
 itemPrice: (value.price) \* (value.quantity),  
 item: value.itemId  
} )
- D.  payload.sale.\*item map ( (value,index) -> {  
 index: index,  
 sale: value.@saleId,  
 itemName: value.desc,  
 itemPrice: (value.price) \* (value.quantity),  
 item: value.@itemId  
} )

## MCD - Level 1 (Mule 4)

Time Remaining: 1:39:26

- 57 of 60 A Mule application contains two HTTP Listeners, each configured for different API endpoints: <http://acme.com/apis/orders> and <http://acme.com/apis/customers>.

What base path value should be set in an HTTP Listener config element so that it can be used to configure both HTTP Listeners?

- A.  /apis/\*
- B.  /apis/
- C.  /apis/orders|customers
- D.  /apis/?

Mark this item for later review.

58 of 60. Refer to the exhibits. A JSON payload is set in the Set Payload transformer.

What is logged by the Logger?



```
<flow name="logPayload" >
    <scheduler doc:name="Scheduler" >
        <scheduling-strategy >
            <fixed-frequency />
        </scheduling-strategy>
    </scheduler>
    <set-payload doc:name="Set to JSON" value="#[{
        "accounts": {
            "account": {
                "accountName": "ABC Widgets",

```



Scheduler	Set Payload Set to JSON	Logger typeOf(payload)
-----------	----------------------------	---------------------------

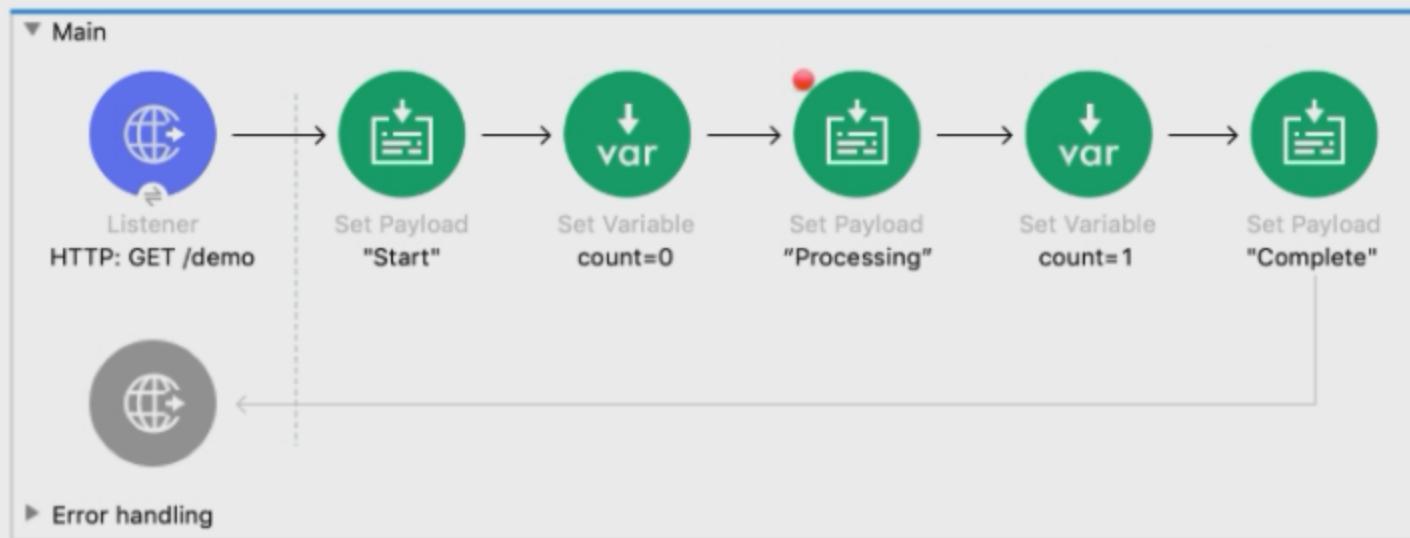
► Error handling

```
<flow name="logPayload" >
    <scheduler doc:name="Scheduler" >
        <scheduling-strategy >
            <fixed-frequency />
        </scheduling-strategy>
    </scheduler>
    <set-payload doc:name="Set to JSON" value="#[{
        "accounts": {
            "account": {
                "accountName": "ABC Widgets",
                "type": "New Customer",
                "stage": "Qualification"
            }
        }
    }]" />
    <logger level="INFO" doc:name="typeOf(payload)" message="#[typeOf(payload)]"/>
</flow>
```

- A.  "String"
- B.  "JSON"
- C.  "Array"
- D.  "Object" ?

59 of 60. Refer to the exhibit. The Mule application is debugged in Anypoint Studio and stops at the breakpoint.

What is the value of the payload displayed in the debugger at this breakpoint?

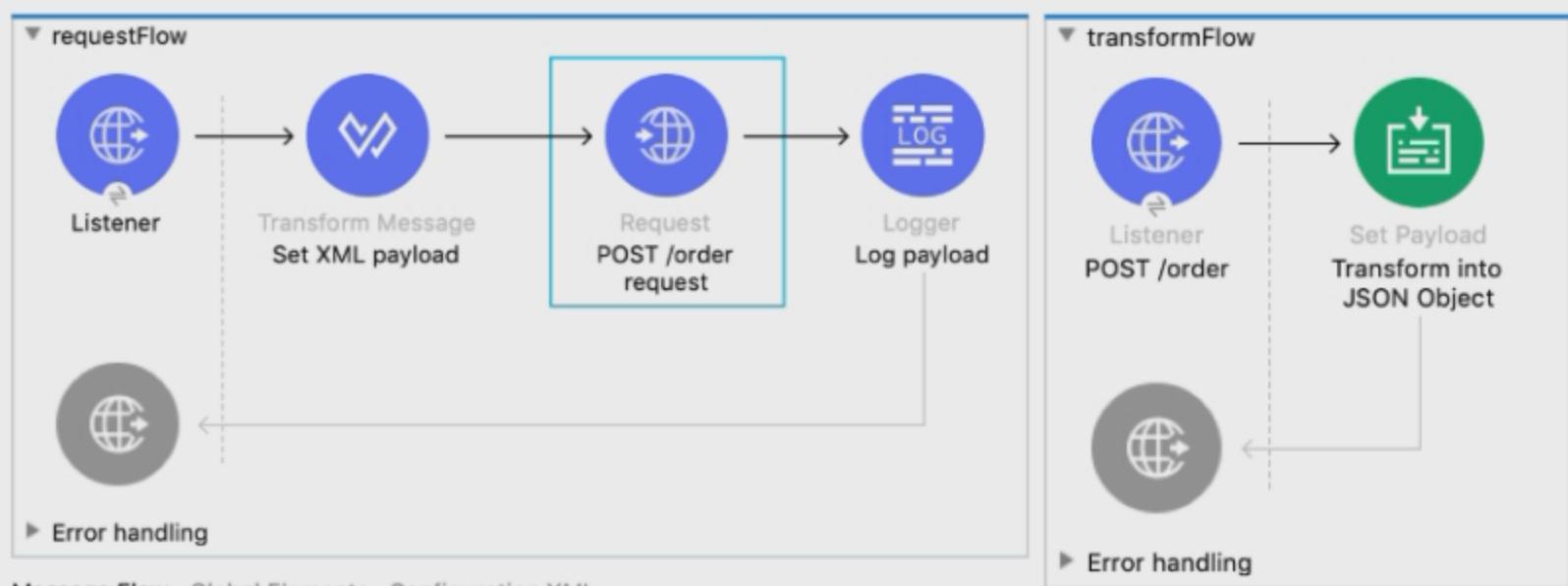


- A.  "Processing"
- B.  "Start"
- C.  0
- D.  "Complete"

- 60 of** Refer to the exhibits. In the requestFlow an HTTP Request operation is configured to send an HTTP request with an XML payload.
60. The request is sent to the HTTP Listener in the transformFlow.

That flow transforms the incoming payload into JSON format and returns the response to the HTTP request. The response of the request is stored in a target variable named theResult.

What is the payload at the Logger component after the HTTP Request?



Message Flow Global Elements Configuration XML

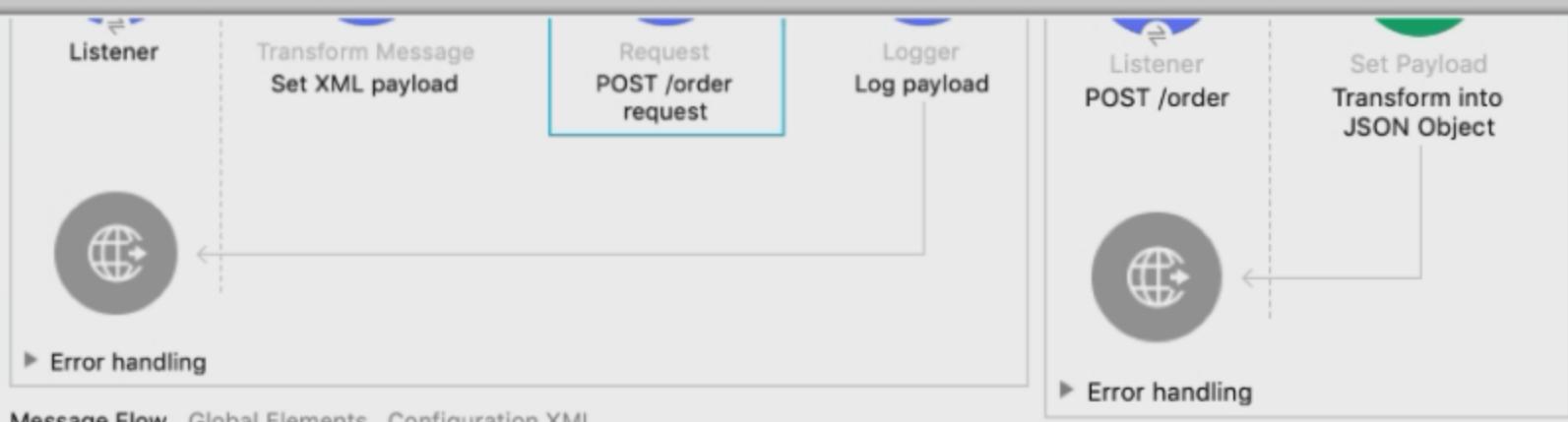
POST /order request X

Problems

There are no errors.

General

General



Message Flow Global Elements Configuration XML

**POST /order request** Problems

There are no errors.

General	Streaming strategy	Repeatable file store stream (Default)
MIME Type	In memory size:	1024
Request	Buffer unit:	KB (Default)
Response		
<b>Advanced</b>		
Error Mapping	Output	
Metadata	Target Variable:	theResult
Notes	Target Value:	<code>fx #[ payload</code>

Refer to the exhibits. In the requestFlow an HTTP Request operation is configured to send an HTTP request with an XML payload. The request is sent to the HTTP Listener in the transformFlow.

Streaming Strategy

Request	In memory size: <input type="text" value="1024"/>
Response	Buffer unit: <input type="text" value="KB (Default)"/>
<b>Advanced</b>	
Error Mapping	Output
Metadata	Target Variable: <input type="text" value="theResult"/>
Notes	Target Value: <input type="text" value="#[ payload ]"/> 

Refer to the exhibits. In the requestFlow an HTTP Request operation is configured to send an HTTP request with an XML payload. The request is sent to the HTTP Listener in the transformFlow.

That flow transforms the incoming payload into JSON format and returns the response to the HTTP request. The response of the request is stored in a target variable named theResult.

What is the payload at the Logger component after the HTTP Request?

- A.  The original XML payload
- B.  The returned JSON response
- C.  null
- D.  A non-empty Java object