Reference: https://docs.mulesoft.com/runtime-manager/developing-applications-for-cloudhub

Question: 36

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

Answer: B

Explanation:

As dependency is not present in Mulesoft Maven repository, we need to install the dependency on computer's local Maven repository.
https://docs.mulesoft.com/mule-runtime/4.3/maven-reference

Question: 37

A web client submits a request to http://localhost:8081?flrstName=john. What is the correct DataWeave expression to access the firstName parameter?

A. #[attributes.queryParams.firstName]
B. #[message.queryParams.hrstName]
C. #[message.inboundProperties.'http.query.params'.firstName]
D. #[attributes.'http.query.params'.firstName]

Answer: A

Explanation:

Question: 38

By default, what happens to a file after it is read using an FTP connector Read operation?

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

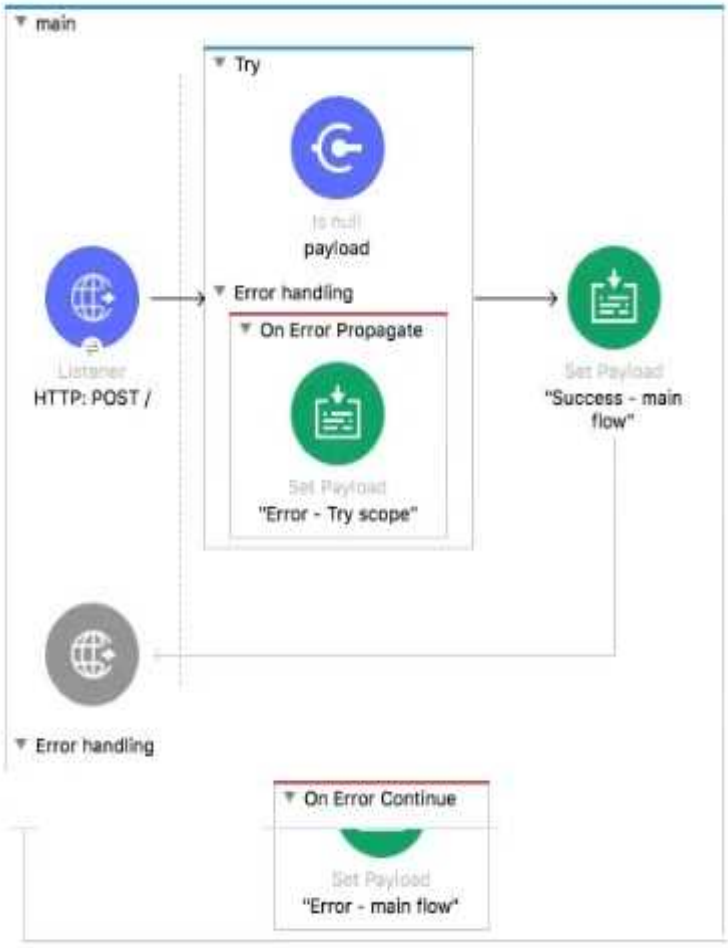
Answer: C

Explanation:

File is not updated when FTP read operations is performed.
MuleSoft Doc Ref : https://docs.mulesoft.com/file-connector/1.3/file-read

Question: 39

Refer to the exhibits.



```
<flow name="main">
    <http:listener doc:name="HTTP: POST /" config-ref="HTTP_Listener_config" path="/" />
    <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="Error - Try scope" doc:name='"Error - Try scope"'/>
        </on-error-propagate>
        </error-handler>
    </try>
    <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>
```

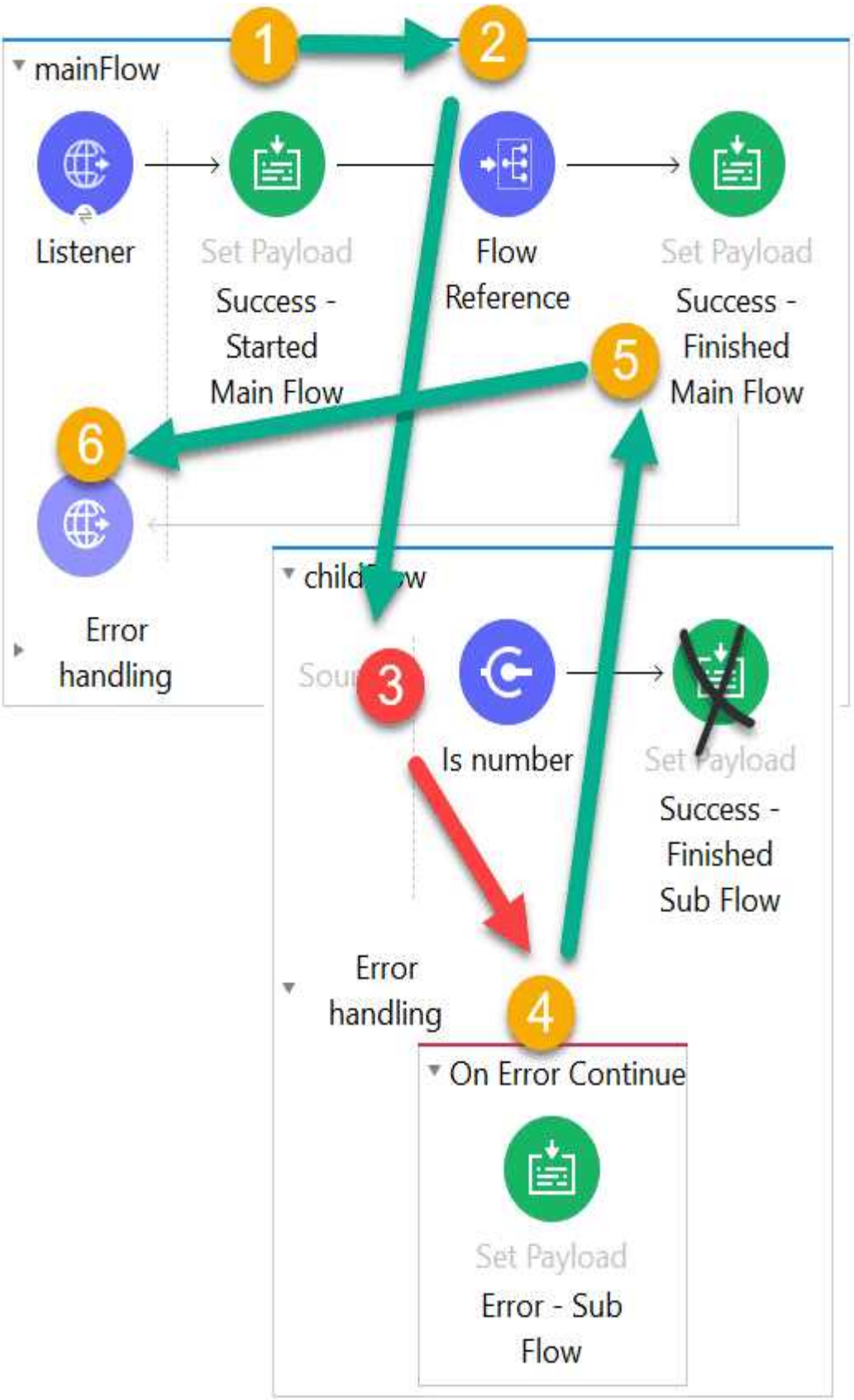
The Validation component in the Try scope throws an error.
What response message is returned to a client request to the main flow's HTTP Listener?
The Validation component in the Try scope throws an error. What response message is returned to a client request to the main flow's HTTP Listener?

A. Success - main flow
B. Error - main flow
C. Error - Try scope
D. Validation Error

Answer: A

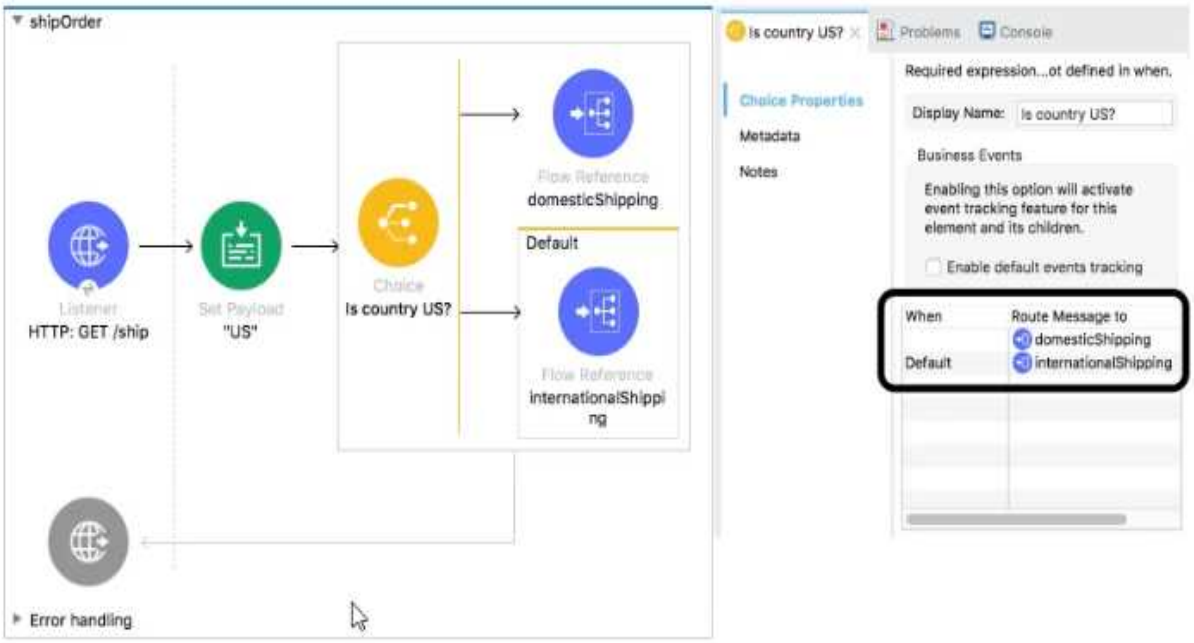
Explanation:

Note that private flow has error scope defined as On Error Continue . So when error occurs in private flow , it is handled by this On Error Continue scope which sends success response back to main flow and does not throw back an error. So main continues normally and payload is set to Success - main flow.
Hence correct answer is Success - main flow
1) HTTP listener received request
2) The Flow Reference calls the child flow
3) The Is Number validator creates an Error Object because the payload isn't an integer. Child Flow execution stops
#[error.description] = "payload is not a valid INTEGER value"
#[error.errorType] = VALIDATION:INVALID_NUMBER
4) The On Error Continue handles the errorThe payload is set to "Error – Sub Flow"
5) "Error – Sub Flow" is returned to the main flow as if the child flow was a success. The Set Payload is executed. The payload is reset to "Success – Finished Main Flow"
6) "Success – Main Flow" is returned to the requestor in the body of the HTTP request. HTTP Status Code: 200
As you can see, in the above example, because the error was caught by an On Error Continue scope in the child flow (RED in, GREEN out) when the Mule Message returns to the parent flow, the parent flow knows none-the-different that there was a failure because the on error continue returns a 200 success message. Note that because, to the mainFlow, the childFlow appeared to succeed, the processing of mainFlow resumed after the flow reference.



Question: 40

Refer to the exhibit.



What is a valid expression for the Choice router's when expression to route events to the documenticShipping flow?

A. 0#[payload = 'US']
B. #[payload == 'US' J
C. #[if(payload = 'US') J
D. #[if(payload == "US")]

Answer: B

Explanation:

Choice Router
The Choice router dynamically routes messages through a flow according to a set of DataWeave expressions that evaluate message content. Each expression is associated with a different routing option. The effect is to add conditional processing to a flow, similar to an if/then/else code block in most programming languages.
Only one of the routes in the Choice router executes, meaning that the first expression that evaluates to true triggers that route's execution and the others are not checked. If none of the expressions are true, then the default route executes.
Properties of <when>
PropertyDescription
Expression (expression)
Expression in DataWeave language to evaluate input.
If the expression evaluates to true, this routing option is used:
<when expression="#[vars.language == 'Spanish']" >

Mulesoft Doc Ref : https://docs.mulesoft.com/mule-runtime/4.3/choice-router-concept
With respect to above information ,
Option 1 is the correct syntax as others are incorrect because of below reasons
* Single = is not the correct syntax to validate the condition. It should be ==
* If keyword is not required in when condition.

Question: 41

A web client submits a request to http://localhost:8081?accountType=personal. The query parameter is captured using a Set Variable transformer to a variable named accountType.
What is the correct DataWeave expression to log accountType?

A. Account Type: #[flowVars.accountType]
B. Account Type: #[message.inboundProperties.accountType]
C. Account Type: # [attributes.accountType]
D. Account Type: #[vars.accountType]

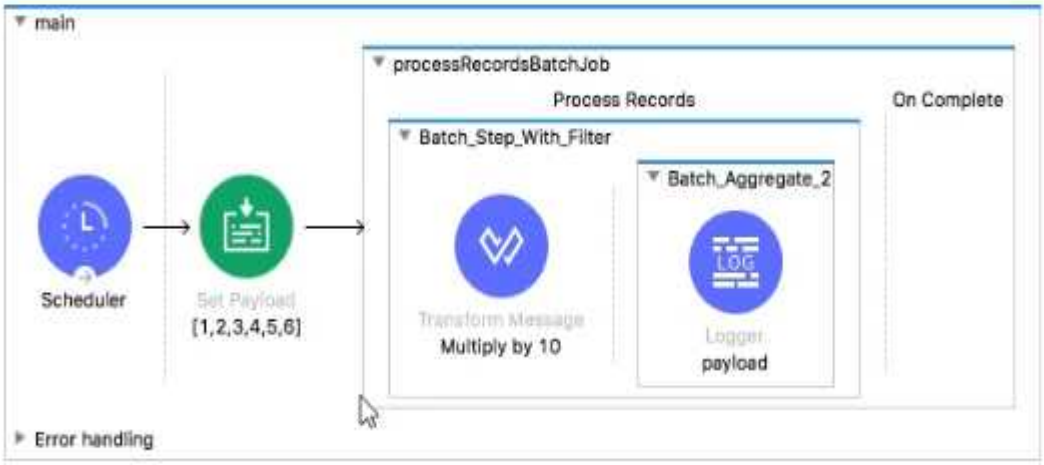
Answer: D

Explanation:

vars: Keyword for accessing a variable, for example, through a DataWeave expression in a Mule component, such as the Logger, or from an Input or Output parameter of an operation. If the name of your variable is myVar, you can access it like this: vars.myVar
Hence correct answer is Account Type: #[vars.accountType]

Question: 42

Refer to the exhibit. The Batch Job processes, filters and aggregates records, What is the expected output from the Logger component?



```
<flow name="main" >
    <scheduler doc:name="Scheduler" > <scheduling-strategy >
    <fixed-frequency frequency="10000"/></scheduling-strategy> </scheduler>
    <set-payload value="#[[1,2,3,4,5,6]]" doc:name="[1,2,3,4,5,6]" />
    <batch:job jobName="processRecordsBatchJob" >
        <batch:process-records >
            <batch:step name="Batch_Step_With_Filter" acceptExpression="#[(payload mod 2) == 0]">
                <ee:transform doc:name="Multiply by 10"><ee:message >
                        <ee:set-payload ><![CDATA[%dw 2.0
                            output application/java

                            payload * 10]]></ee:set-payload>
                    </ee:message></ee:transform>
                <batch:aggregator doc:name="Batch_Aggregate_2" size="2">
                    <logger level="INFO" doc:name="payload" message="#[payload]"/>
                </batch:aggregator>
            </batch:step>
        </batch:process-records>
    </batch:job>
</flow>
```

A. [10. 20, 30. 40, 50, 60]
B. [10. 20] [30, 40] [50, 60]
C. [20, 40, 60]
D. [20. 40] [60]

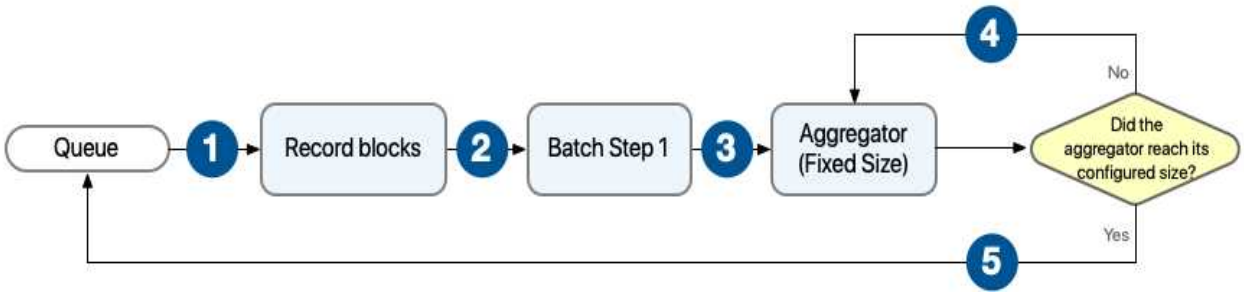
Answer: D

Explanation:

* Batch scope has filter criteria which says paylod mod 2 = 0 whch means only 2, 4 and 6 will be in batch scope.
* So payload for each of these will be incremented by 10.
* Aggregator has batch size defined as 2. So it will process in batch of two records.
* Hence option 3 is correct answer.

[20,40]
[60]

Behavior with aggregator configured with fixed size
In this scenario, the batch step sends the processed records to an aggregator, which starts processing the records and buffering them until the configured aggregator's size is reached. After that, the aggregator sends the aggregated records to the stepping queue.

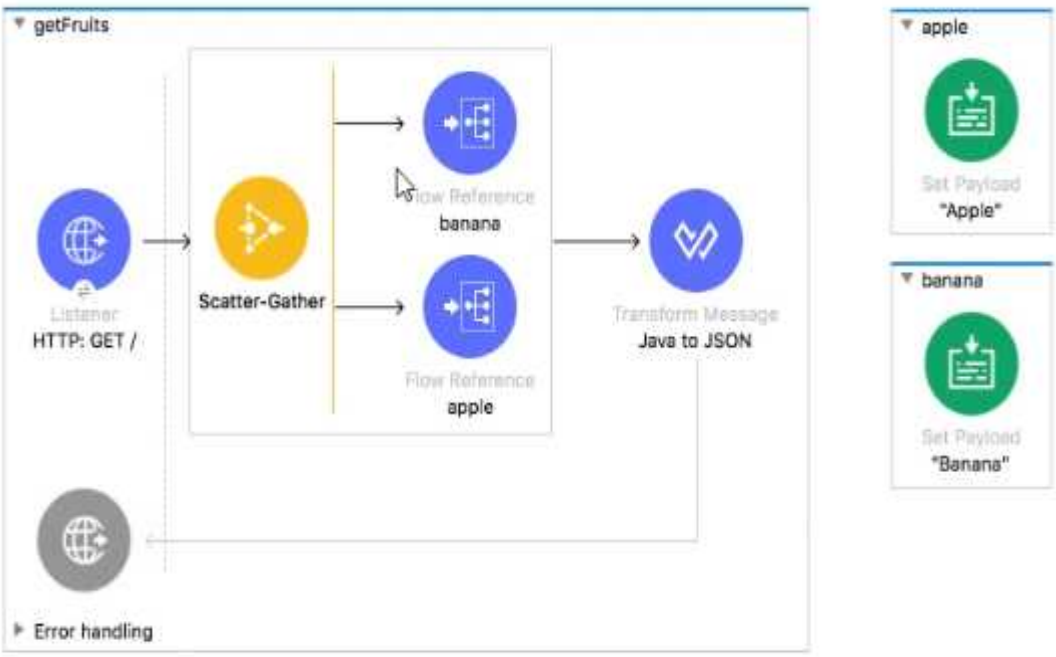


The batch job builds record blocks of the configured block size and sends them to their corresponding batch step for processing. Each batch step receives one or more record blocks and starts processing them in parallel. After the batch step processes a record, the batch step sends the record to the aggregator for further processing. The aggregator continues processing records until the number of

aggregated records reaches the configured aggregator's size.
https://docs.mulesoft.com/mule-runtime/4.3/batch-processing-concept

Question: 43

Refer to the exhibits.




```
<flow name="getFruits">
    <http:listener doc:name="HTTP: GET /" config-ref="HTTP_Listener_config" path="/" />
    <scatter-gather doc:name="Scatter-Gather">
        <route> <flow-ref doc:name="banana" name="banana"/>  </route>
        <route> <flow-ref doc:name="apple" name="apple"/>    </route>
    </scatter-gather>
```

```
<sub-flow name="apple" ><set-payload value="Apple" doc:name='"Apple"' /></sub-flow>
<sub-flow name="banana"><set-payload value="Banana" doc:name='"Banana"' /></sub-flow>
```

A web client submits a request to http://localhQst:8081. What is the structure of the payload at the end of the flow?

A)

```
['Banana', 'Apple']
```

B)

```
{
    "0": "Banana",
    "1": "Apple"
}
```

C)

```
{
    "attributes": ...,
    "payload": ['Banana','Apple']
}
```

D)

```
{
    "0": {
        "attributes": ...,
        "payload": "Banana"
    }
    "1": {
        "attributes": ...,
        "payload: "Apple"
    }
}
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: D

Explanation:

Scatter-Gather Router
The Scatter-Gather component is a routing event processor that processes a Mule event through different parallel processing routes that contain different event processors. Each route receives a reference to the Mule event and executes a sequence of one or more event processors. Each of these routes uses a separate thread to execute the event processors, and the resulting Mule event can be either the same Mule event without modifications or a new Mule event with its own payload, attributes, and variables. The Scatter-Gather component then combines the Mule events returned by each processing route into a new Mule event that is passed to the next event processor only after every route completes successfully.
The Scatter-Gather component executes each route in parallel, not sequentially. Parallel execution of routes can greatly increase the efficiency of your Mule application and may provide more information than sequential processing.
Mule Ref Doc : https://docs.mulesoft.com/mule-runtime/4.3/scatter-gather-concept

Question: 44

Refer to the exhibits. A web client submits a request to http://localhost:8081/fliqhts?destination=SFO and the 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               : 10.02.06-soaperrorFlow/processors/0 @ 10.02.06-soaperror:10.02.06-soaperror.xml:21 (Consume)
Element XML           : <wsc:consume doc:name="Consume" doc:id="49a84a74-2848-4a65-b93f-dcbf59fd4412" config-
ref="Web_Service_Consumer_Config" operation="findFlight"></wsc:consume>

  (set debug level logging or '-Dmule.verbose.exceptions=true' for everything)
********************************************************************************
```

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

Answer: B

Explanation:

As can be seen in error message , SOAP service findFlights expects the SOAP paylaod. This can be set using transform message processor which forms SOAP payload before the Consume operation that contains the destination query parameter

Question: 45

What valid RAML retrieves details on a specific by its orderId as a URL parameter?
A)

```
/orders:
    /{orderId}:
        get:
```

B)

```
/orders:
    /orderId:
        get:
```

C)

```
/orders:
    get:
        /{orderId}:
```

D)

```
/orders:
    get:
        /orderId:
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: A

Explanation:

Question: 46

An HTTP Request operation sends an HTTP request with a non-empty JSON object payload to an external HTTP endpoint. The response from the external HTTP endpoint returns an XML body. The result is stored in a target named the Result.
What is the payload at the event processor after the HTTP Request?

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

Answer: C

Explanation:

Question: 47

According to MuleSoft. what is the first step to create a Modern API?

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

Answer: B

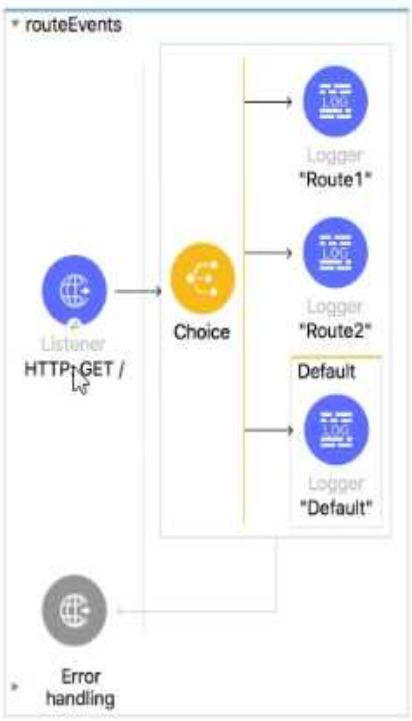
Explanation:

First step in creating Modern API is to create an API specification and get feedback from stakeholders so that any future issues can be identified at early stage thereby reducing overall delivery time

Reference: https://developer.mulesoft.com/tutorials-and-howtos/quick-start/designing-your-first-api

Question: 48

Refer to the exhibit.




```
<flow name="routeEvents" >
    <http:listener doc:name="HTTP: GET /" config-ref="HTTP_Listener_config"
    path="/" />
    <choice doc:name="Choice" >
        <when expression="true" >
            <logger level="INFO" doc:name='"Route1"' message="Route1"/>
        </when>
        <when expression="true" >
            <logger level="INFO" doc:name='"Route2"' message="Route2"/>
        </when>
        <otherwise >
            <logger level="INFO" doc:name='"Default"'  message="Default"/>
        </otherwise>
    </choice>
</flow>
```

All three of the condition for the Choice router are true. What log messages are written?

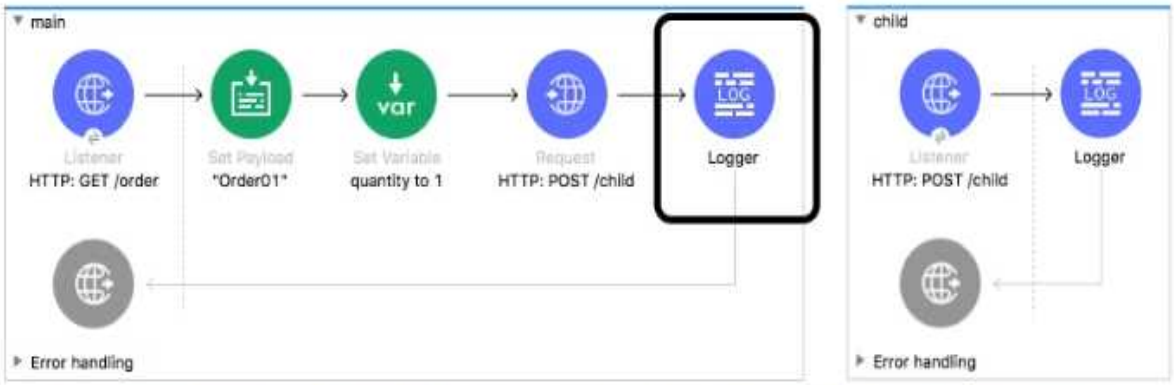
A. Route 1
B. Route2
C. Route1, Route2
D. Route1, Route2, Default

Answer: A

Explanation:

Question: 49

Refer to the exhibit. The main flow contains an HTTP Request in the middle of the flow. The HTTP Listeners and HTTP request use default configurations.
What values are accessible to the Logger at the end of the flow after a web client submit request to http://local:801/order?color=red?




```
    <http:request method="POST" doc:name="HTTP: POST /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 quantity var
C. payload color query param
D. payload quantity var color query param

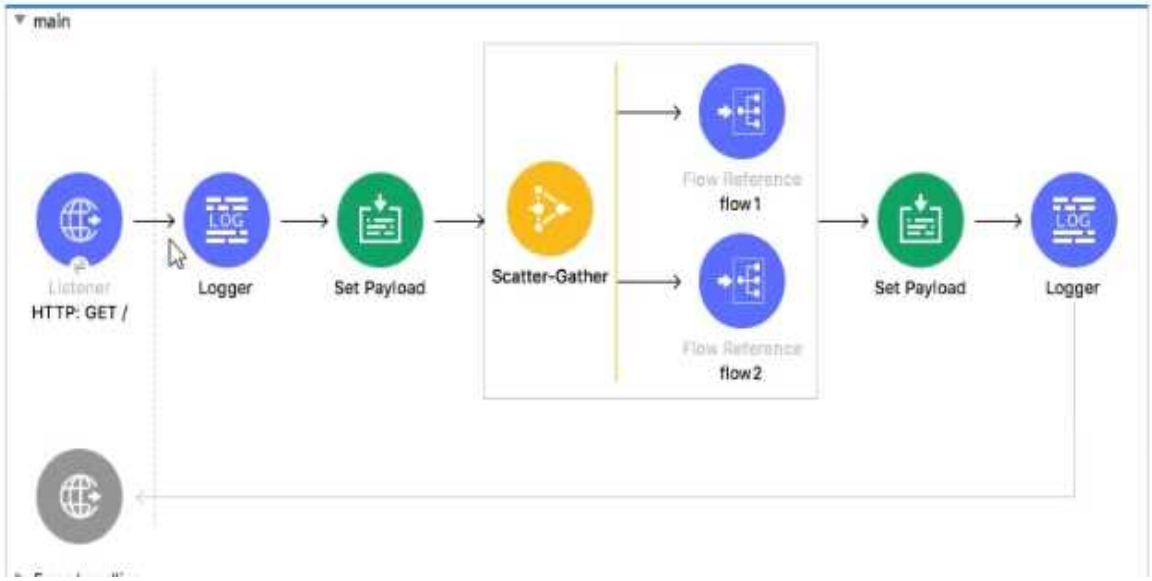
Answer: B

Explanation:

In this case as outbound call is made using HTTP: POST /child , all attributes will be replaced by this invocation. Hence query parameter will not be accessible at logger.

Question: 50

Refer to the exhibit.



In the execution of the Scatter_Gather, the flow1 route completes after 10 seconds and the flow2 route completes after 20 seconds.
How many seconds does it take for the Scatter_Gather to complete?

A. 0
B. 10
C. 20
D. 30

Answer: C

Explanation:

Question: 51

To avoid hard-coding values, a flow uses some property placeholders and the corresponding values are stored in a configuration file.
Where does the configuration file's location need to be specified in the Mule application?

A. The pom.xml file
B. A global element
C. The mule-art if act .json file
D. a flow attribute

Answer: B

Explanation:

Correct answer is A global element
When we create a configuration file , that file needs to added as Global Configuration file in Global element. A global element is a reusable object containing parameters that any number of elements in a flow can share. You reference a global element from Anypoint Connectors or components in your Mule application

Question: 52

Refer to the exhibit.

What is the correct syntax to add an employee ID as a URI parameter in an HTTP Listener path?

A. (employeeID)
B. ${emp!oyeeID}
C. {employeeID}
D. # [employeeID]

Answer: C

Explanation:

Paths
The path of an HTTP listener can be static, which requires exact matches, or feature placeholders. Placeholders can be wildcards (*), which match against anything they are compared to, or parameters ({param}), which not only match against anything but also capture those values on a URI parameters map.
Take the following example paths for three listeners using a configuration that establishes api/v1 as the base path:
account/mulesoft/main-contact: only match the exact path request http://awesome-company.com/api/v1/account/mulesoft/main-contact
account/{accountId}/main-contact: matches all path requests structured similarly, such as http://awesome-company.com/api/v1/account/salesforce/main-contact, and save salesforce as the value of accountId.
account/{accountId}/*: matches all path requests different from main-contact, such as http://awesome-company.com/api/v1/account/mulesoft/users, and save mulesoft as the value of accountId.
Mule Ref Doc : https://docs.mulesoft.com/http-connector/1.6/http-listener-ref#paths

Question: 53

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

Answer: D

Explanation:

Correct answer is Add required headers to RAML specification and redeploy new API proxy
MuleSoft Doc Ref : https://docs.mulesoft.com/api-manager/2.x/tutorial-manage-an-api
Steps are as below :
Add the Required RAML Snippet
SLA-based rate limiting requires adding a RAML or OAS snippet to your API. This procedure demonstrates adding a RAML snippet.
Specify the client ID and secret as query parameters.
Add a section called traits: at the RAML root level to define query parameters:
traits:
- client-id-required:
queryParameters:
client_id:
type: string
client_secret:
type: string
Add the client-id-required trait to every method that requires these query parameters:

/users:
get:
is: [client-id-required]
description: Gets a list of JSONPlaceholder users.

Step 2 : Add the SLA Tier in API Manager
Step 3 : Apply the policy and redeploy
Reference: https://docs.mulesoft.com/api-manager/2.x/tutorial-manage-an-api

Question: 54

What payload is returned by a Database SELECT operation that does not match any rows in the database?

A. false
B. null
C. Exception
D. Empty Array

Answer: D

Explanation:

Empty array is returned when no rows are matched.
MuleSoft Doc Ref : https://docs.mulesoft.com/db-connector/1.9/database-connector-select

Question: 55

What is the correct syntax to define and call a function in Database?
A)

```
fun addKV( object: Object, key: String, value: Any ) =
  object ++ { (key):value }
---
addKV ( {hello: "world"}, "hola","mundo" )
```

B)

```
%function addKV( object: Object, key: String, value: Any ) =
  object ++ { (key):value }
---
addKV ( {hello: "world"}, "hola","mundo" )
```

C)

```
%function addKV( object: Object, key: String, value: Any ) =
  object ++ { (key):value }
---
{ hello: "world" } addKV  ( "hola","mundo" )
```

D)

```
fun addKV( object: Object, key: String, value: Any ) =
  object ++ { (key):value }
---
{ hello: "world" } addKV  ( "hola","mundo" )
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: A

Explanation:

Keyword to ad function in Dataweave transformation is fun. Hence option 2 and 4 are invalid. Also parameters needs to be passed exactly in same order as defined in function definition. Hence correct answer is'
fun addKV(object: Object, key: String, value: Any) =
object ++ {(key):(value)}
---
addKV ({"hello': "world"}, "hola", "mundo")
MuleSoft Documentation Reference : https://docs.mulesoft.com/mule-runtime/4.3/dataweave-functions

DataWeave Function Definition Syntax
To define a function in DataWeave use the following syntax:
fun myFunction(param1, param2, ...) = <code to execute>
The fun keyword starts the definition of a function.
myFunction is the name you define for the function.
Function names must be valid identifiers.
(param1, param2, … , paramn) represents the parameters that your function accepts.
You can specify from zero to any number of parameters, separated by commas (,) and enclosed in parentheses.
The = sign marks the beginning of the code block to execute when the function is called.
<code to execute> represents the actual code that you define for your function.

Question: 56

A Utility.dwl file is located in a Mule project at src/main/resources/modules. The Utility.dwl hie defines a function named pascalize that reformats strings to pascal case.
What is the correct DataWeave to call the pascalize function in a Transform Message component?
A)

```
%dw 2.0
output application/json
import modules.Utility
---
pascalize( "max mule" )
```

B)

```
%dw 2.0
output application/json
import modules::Utility
---
pascalize( "max mule" )
```

C)

```
%dw 2.0
output application/json
import modules::Utility
---
Utility::pascalize( "max mule" )
```

D)

```
%dw 2.0
output application/json
import modules.Utility
---
Utility.pascalize( "max mule" )
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: B

Explanation:

Question: 57

A RAML specification is defined to manage customers with a unique identifier for each customer record. What URI does MuleSoft recommend to uniquely access the customer identified with the unique ID 1234?

A. /customers?custid=true&custid=1234
B. /customers/1234
C. /customers/custid=1234
D. /customers?operation=get&custid=1234

Answer: B

Explanation:

URI parameter (Path Param) is basically used to identify a specific resource or resources . For eg : the URL to get employee details on the basis of employeeID will be GET /employees/{employeeID} where employees is resource and {employeeID} is URI parameter. Hence option 1is the correct

answer

Question: 58

Refer to the exhibits.

orders.csv

```
orderId,account
100, partnerA
101, acme.com
102, mybank.com
103, onlineSales
```

The orders.csv file is read, then processed to look up the orders in a database. The Mule application is debugged in Any point Studio and stops at the breakpoint.
What is the payload shown in the debugger at this breakpoint?

A. "none"
B. The entire CSV file
C. The database response
D. 100

Answer: D

Explanation:

Question: 59

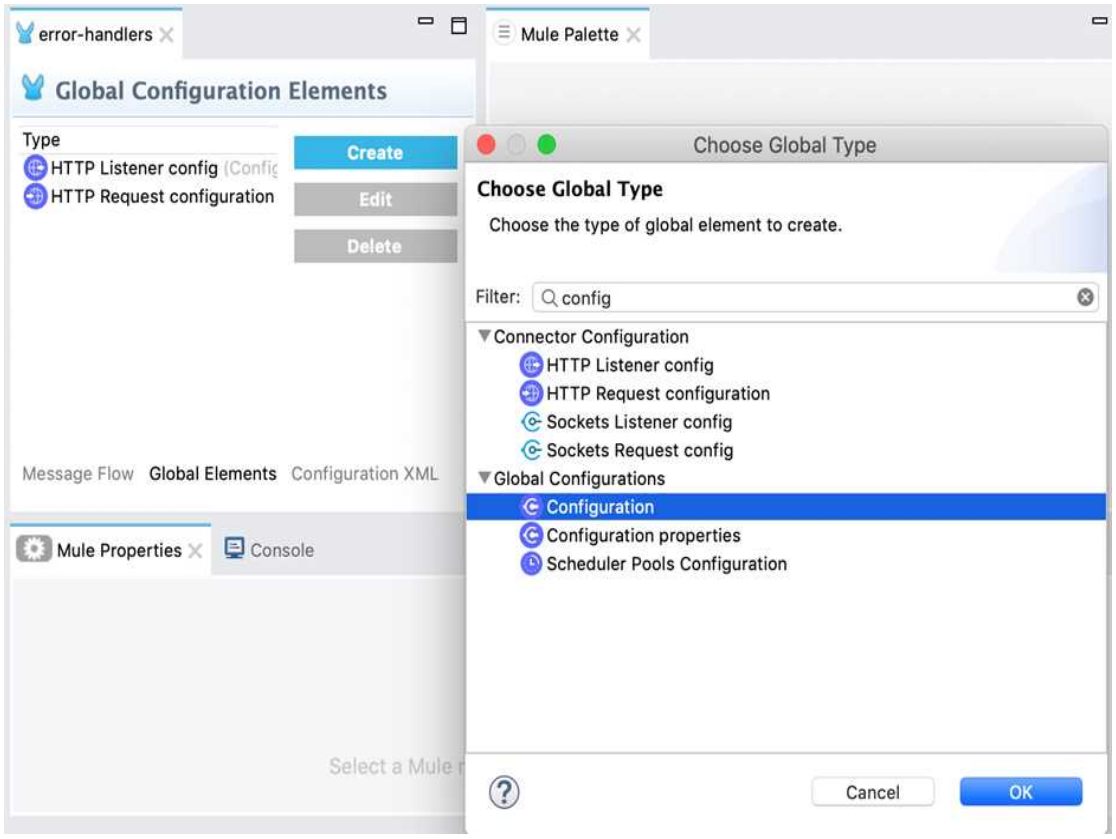
A Mule application contains a global error handler configured to catch any errors.
Where must the global error handler be specified so that the global error handler catches all errors from flows without their own error handlers?

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

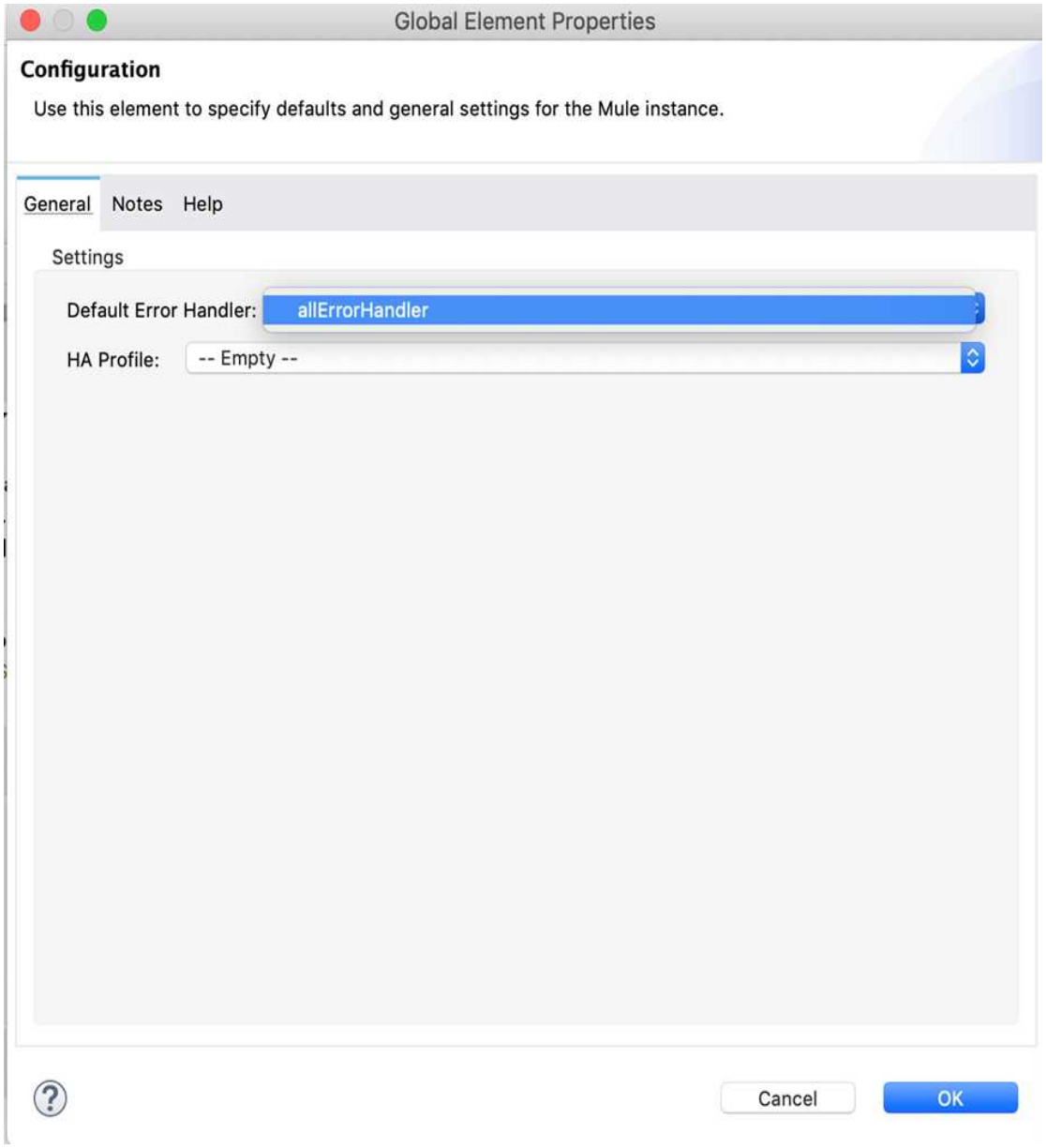
Answer: C

Explanation:

Correct answer is A global element
Global error handlers are to be created in global element .
Quick note to remember here is Global error handlers come in to picture only when there are no error handlers specified as flow level.
Steps to create Global error handler
1) Click Global Elements to open Global Configuration Elements. Global Elements is located below the Studio canvas
2) In Global Configuration Elements, click Create to open the Choose Global Type dialog



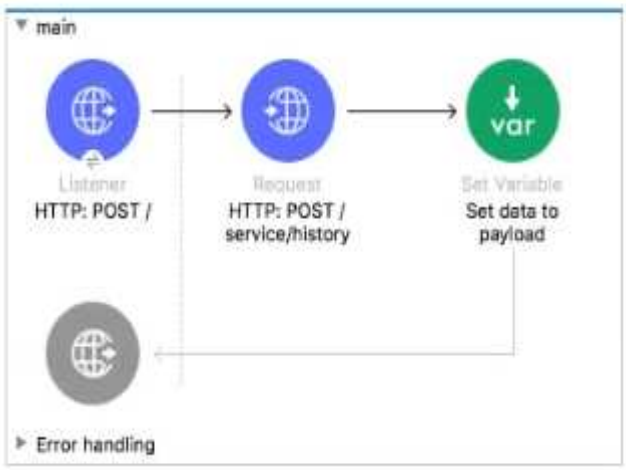
3) From the dialog, select Global Configuration -→ Configuration, and then click OK to open the Configuration dialog.



4) From the select Configuration dialog, select allErrorHandler for the Default Error Handler field, and click OK.

Question: 60

Refer to the exhibit.



What can be added to the flow to persist data across different flow executions?

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

Answer: A

Explanation:

An object store is a facility for storing objects in or across Mule applications. Mule runtime engine (Mule) uses object stores to persist data for eventual retrieval. Internally, Mule uses object stores in various filters, routers, and other message processors that need to store states between messages. Object stores are available in all deployment targets. If you deploy your application to CloudHub, you can also use Object Store V2.
Correct answer is Key/value pair in Object store
MuleSoft Documentation reference : https://docs.mulesoft.com/mule-runtime/4.3/mule-object-stores#use-cases

Question: 61

Refer to the exhibit.