

MuleSoft

*MCD-Level-1
MuleSoft Certified Developer - Level 1 (Mule 4)*

- Up to Date products, reliable and verified.
- Questions and Answers in PDF Format.

Full Version Features:

- 90 Days Free Updates
- 30 Days Money Back Guarantee
- Instant Download Once Purchased
- 24 Hours Live Chat Support

For More Information:

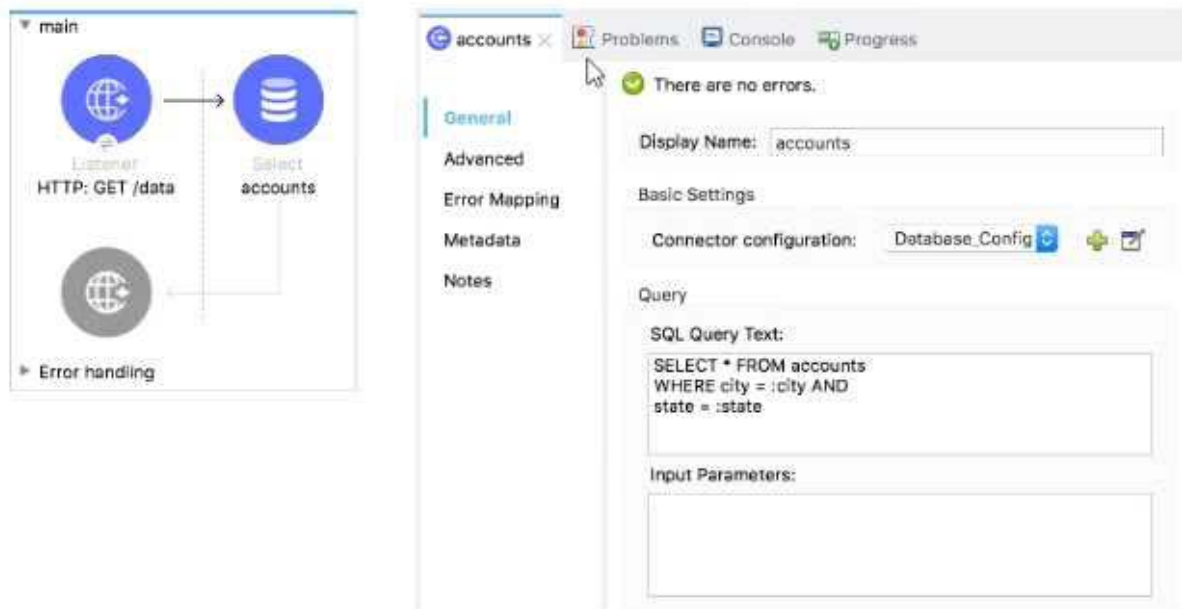
<https://www.testsexpert.com/>

- **Product Version**

Latest Version: 16.0

Question: 1

Refer to the exhibit.



What expression correctly specifies input parameters to pass the city and state values to the SQL query?

A)

```
#[
{
  city: "San Francisco",
  state: "CA"
}
]
```

B)

```
#[
[
  "San Francisco",
  "CA"
]
]
```

C)

```
#[
  inputParams: {
    city: "San Francisco",
    state: "CA"
  }
]
D)
#[
  inputParams: [
    "San Francisco",
    "CA"
  ]
]
```

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

Answer: A

Explanation:

MuleSoft Documentation Reference : <https://docs.mulesoft.com/db-connector/1.9/database-connectorselect>

Question: 2

A Mule flow has three Set Variable transformers. What global data structure can be used to access the variables?

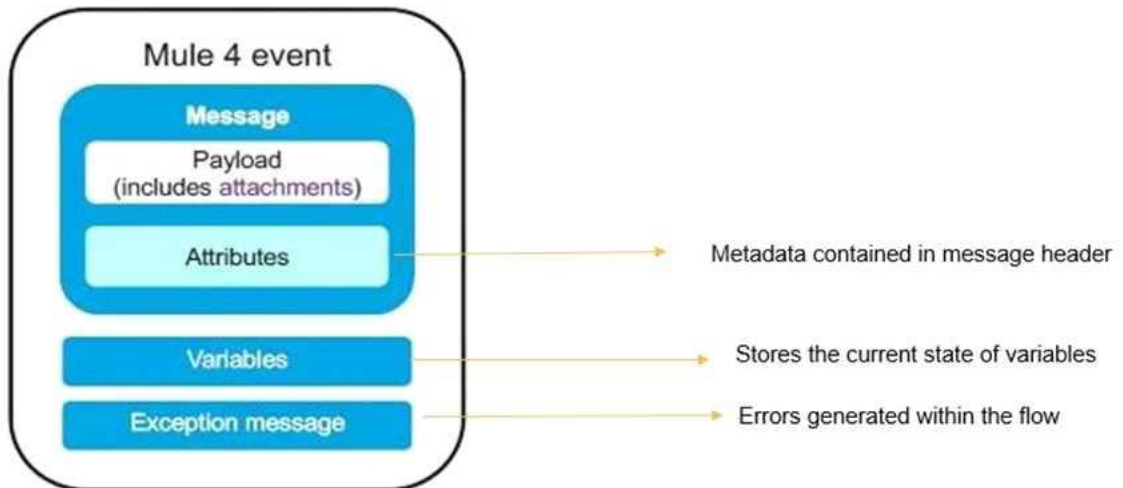
- A. Mule event attributes
- B. Mule event message
- C. Mule application properties
- D. Mule event

Answer: D

Explanation:

Mule event is correct answer. Mule event has two parts which are as follows

- 1) Message (which contains payload and attributes like headers and query/uri parameters)
- 2) Variables



Question: 3

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

- A. The applications associated with the other APIs must be restarted
- 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. Nothing needs to be changed in the other APIs or their associated applications

Answer: D

Explanation:

Correct answer is Nothing needs to be changed in the other APIs or their associated applications
This is the benefit of having separate interface layer. As there are no changes to interface , no changes are required on the API's which consumes this API in context

Question: 4

Refer to the exhibit.



```

<flow name="validatePayload" >
  <http:listener doc:name="HTTP: GET /" config-ref="HTTP_Listener_config" path="/" />
  <set-payload value="Before" doc:name="Before" />
  <validation:is-null doc:name="payload" value="#[payload]" message="Validation Error" />
  <set-payload value="After" doc:name="After" />
</flow>

```

What is the response to a web client request to <http://localhost:8081>?

- A. After
- B. before
- C. Validation Error
- D. null

Answer: C

Explanation:

The screenshot shows the Mule Studio interface with a flow named 'question12Flow' that includes an HTTP GET listener, a 'Before' payload setter, an 'is-null payload' validator, an 'After' payload setter, and a logger. The Advanced REST client is configured to send a GET request to <http://localhost:8086/q12>. The response is a '500 Server Error' with a status of 3941.33 ms. The error message is 'Validation Error', which is circled in blue. The Mule Studio console shows the following log entries:

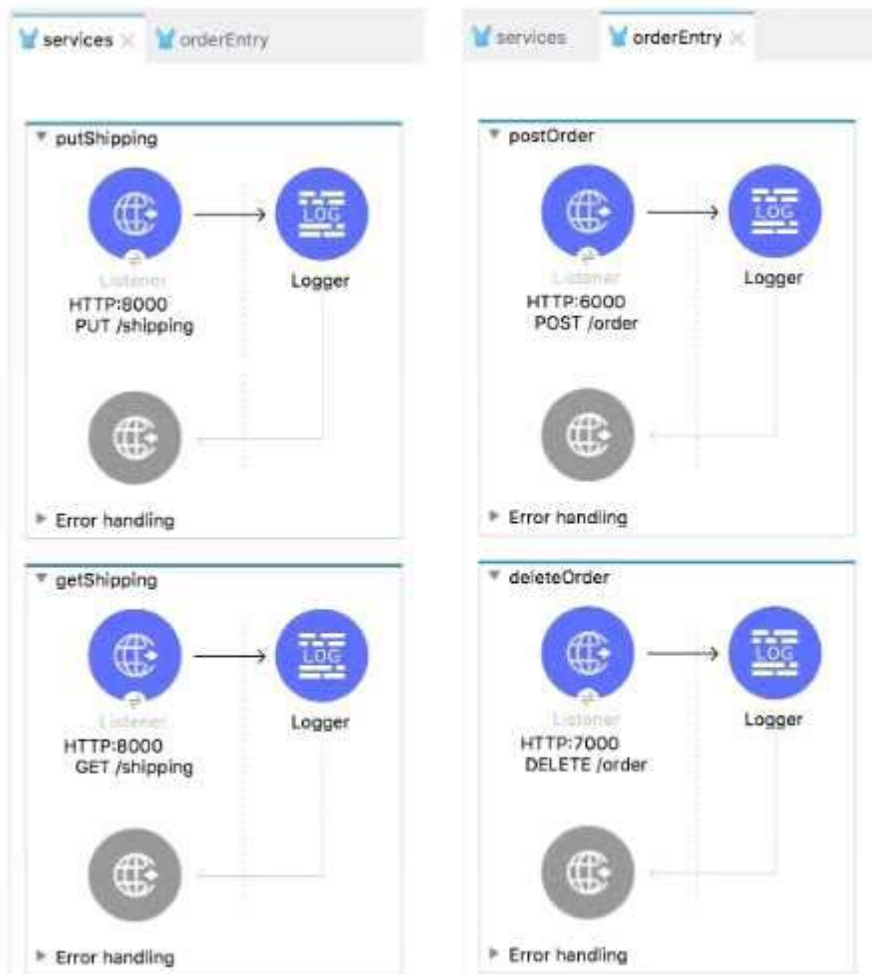
```

INFO 2020-02-06 21:20:20,499 [[MuleRuntime].cpuLight.18] [question12Flow]
INFO 2020-02-06 21:20:20,512 [[MuleRuntime].cpuLight.18] [question12Flow]
ERROR 2020-02-06 21:28:22,899 [[MuleRuntime].cpuLight.18] [question12Flow]
Message: Validation Error.
Error type: VALIDATION:NOT_NULL
Element: <validation:is-null doc:name="payload" doc:id="3d3272d6-2a27-46ea-bf46-bcc26b1e668b" value="#[payload]" message="Validation Error"></validation>

```

Question: 5

Refer to the exhibits.



The two Mule configuration files belong to the same Mule project. Each HTTP Listener is configured with the same host string and the port number, path, and operation values are shown in the display names.

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

Answer: C

Explanation:

In this case three configurations will be required each for port 8000, 6000 and 7000.

There would be three global elements defined for HTTP connections.
Each HTTP connection will have host and port. One example shown below with host as localhost and port 6000

The screenshot shows a 'Global Element Properties' dialog box with a title bar and a close button. The main title is 'HTTP Listener config' with a subtitle 'Configuration element for a HttpListener.' Below this are tabs for 'General', 'Notes', and 'Help', with 'General' selected. A 'Name' field contains 'HTTP_Listener_config'. Under the 'Connection' section, there is a sub-dialog box with tabs for 'General', 'TLS', and 'Advanced', with 'General' selected. This sub-dialog has fields for 'Protocol' (HTTP (Default)), 'Host' (All Interfaces [0.0.0.0] (default)), and 'Port' (6000). At the bottom of the main dialog are buttons for '?', 'Test Connection...', 'OK', and 'Cancel'.

To use an HTTP listener, you need to declare a configuration with a corresponding connection. This declaration establishes the HTTP server that will listen to requests.

Additionally, you can configure a base path that applies to all listeners using the configuration.

```
<http:listener-config name="HTTP_Listener_config" basePath="api/v1">
```

```
<http:listener-connection host="0.0.0.0" port="8081" />
```

```
</http:listener-config>
```

<https://docs.mulesoft.com/http-connector/1.6/http-listener-ref#http-listener-configuration>

For More Information – Visit link below:
<https://www.testsexpert.com/>

16\$ Discount Coupon: **9M2GK4NW**

Features:

■ Money Back Guarantee.....



■ 100% Course Coverage.....



■ 90 Days Free Updates.....



■ Instant Email Delivery after Order.....

