

Salesforce-MuleSoft-Developer-I Answers Free | Valid Salesforce-MuleSoft-Developer-I Test Practice



BTW, DOWNLOAD part of TestPassed Salesforce-MuleSoft-Developer-I dumps from Cloud Storage:
https://drive.google.com/open?id=1Futc5X61zen6lf_y2AAQFb1usFyNDJ4J

The Salesforce-MuleSoft-Developer-I exam is the right way to learn new in-demand skills and upgrade knowledge. After passing the Salesforce Certified MuleSoft Developer (Mule-Dev-201) (Salesforce-MuleSoft-Developer-I) exam the successful candidates can gain multiple personal and professional benefits with the real Salesforce Salesforce-MuleSoft-Developer-I Exam Questions. Validation of skills, more career opportunities, increases in salary, and increases in the chances of promotion are some prominent benefits of the Salesforce Salesforce-MuleSoft-Developer-I certification exam.

Salesforce Salesforce-MuleSoft-Developer-I Exam Syllabus Topics:

Topic	Details
Topic 1	<ul style="list-style-type: none"> Creating Application Networks: The topic of creating Application Networks encompasses understanding MuleSoft's proposal for closing the IT delivery gap and describing the role and characteristics of the modern API. It also includes the purpose and roles of a Center for Enablement (C4E), and the benefits of API-led.
Topic 2	<ul style="list-style-type: none"> Building API Implementation Interfaces: This topic involves manually creating a RESTful interface for a Mule application and generating a REST Connector from a RAML specification. It also includes describing the features and benefits of APIkit.
Topic 3	<ul style="list-style-type: none"> Routing Events: It focuses on using the Choice router for conditional logic and the Scatter-Gather router to multicast events. This topic also involves validating data by using the Validation module.
Topic 4	<ul style="list-style-type: none"> Handling Errors: Handling errors includes describing default error handling in Mule applications and defining custom global default error handlers. It involves comparing On Error Continue and On Error Propagate scopes, creating error handlers for a flow, using the Try scope, and mapping errors to custom application errors.
Topic 5	<ul style="list-style-type: none"> Debugging and Troubleshooting Mule Applications: Using breakpoints to inspect a Mule event during runtime, installing missing Maven dependencies, and reading and deciphering Mule log error messages are sub-topics of this topic.
Topic 6	<ul style="list-style-type: none"> Transforming Data with DataWeave: It involves writing DataWeave scripts and using DataWeave functions. This topic also includes defining and using DataWeave variables, functions, and modules, and applying correct syntax.

Topic 7	<ul style="list-style-type: none"> • Designing APIs: Designing APIs involves describing the lifecycle of the modern API and using RAML to define various aspects of an API. It includes identifying when to use query parameters vs URI parameters, and defining API parameters.
Topic 8	<ul style="list-style-type: none"> • Using Connectors: It focuses on retrieving data from REST services using HTTP Request or REST Connector. Moreover, the topic covers using a Web Service Consumer connector for SOAP web services and the Transform Message component.
Topic 9	<ul style="list-style-type: none"> • Deploying and Managing APIs and Integrations: It includes packaging Mule applications for deployment and deploying them to CloudHub. This topic also involves using CloudHub properties, creating and deploying API proxies, connecting an API implementation to API Manager, and applying policies to secure an API.

>> Salesforce-MuleSoft-Developer-I Answers Free <<

100% Pass Salesforce - Salesforce-MuleSoft-Developer-I - Useful Salesforce Certified MuleSoft Developer (Mule-Dev-201) Answers Free

Salesforce Salesforce-MuleSoft-Developer-I preparation materials will be the good helper for your qualification certification. We are concentrating on providing high-quality authorized Salesforce-MuleSoft-Developer-I study guide all over the world so that you can clear exam one time. As we all know, the preparation process for an exam is very laborious and time-consuming. We had to spare time to do other things to prepare for Salesforce Salesforce-MuleSoft-Developer-I Exam, which delayed a lot of important things.

Salesforce Certified MuleSoft Developer (Mule-Dev-201) Sample Questions (Q20-Q25):

NEW QUESTION # 20

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?

```

%dw 2.0
output application/json
import * from modules.MyModule
---
formatString( "annie point" )

```

- A.

```

%dw 2.0
output application/json
import * from modules::MyModule
---
MyModule::formatString( "annie point" )

```

- B.

```

) %dw 2.0
output application/json
import * from modules::MyModule
---
formatString( "annie point" )

```

- C.

```

%dw 2.0
output application/json
import * from modules.MyModule
---
MyModule.formatString( "annie point" )

```

- D.

Answer: C

NEW QUESTION # 21

Refer to the below exhibit.

A Mule application configures a property placeholder file named config.yaml to set some property placeholders for an HTTP connector.

What is the valid properties placeholder file to set these values?

Global Element Properties

HTTP Listener config

Configuration element for a HttpListener.

General Notes Help

Name: HTTP_Listener_config

Connection

General TLS Advanced

Connection

Protocol: HTTP (Default)

Host: \${http.host}

Port: \${http.port}

General

Base path:

Test Connection... OK Cancel

- A. 1. http:
2. host = "localhost"
3. port = "8081"
- B. 1. http:
2. basepath: "api"
3. host : "localhost"
4. port : "8081"
- C. 1. {
2. http:
3. basePath: "api",
4. port: "8081",
- D. 1. http.host = localhost
2. http.port = 8081

Answer: B

Explanation:

5. host: "localhost"

Explanation:

Correct answer is as below as it follows the correct syntax.

http:

basepath: "api"

host : "localhost"

port : "8081"

Mule Doc Reference : <https://docs.mulesoft.com/mule-runtime/4.3/mule-app-properties-to-configure>

NEW QUESTION # 22

Refer to the exhibit.



What

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

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

Answer: A

Explanation:

The screenshot shows the MuleSoft Studio interface. On the left, the Package Explorer shows a project named 'question2' with a message flow 'question2Flow'. The flow diagram shows an HTTP Listener (HTTP: GET), Set Payload 'Before', Is null payload, Set Payload 'After', and a Logger. The Message Flow configuration XML is visible below the diagram. On the right, the Advanced REST client is open, showing a GET request to <http://localhost:8081/api/question2>. 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 Properties console at the bottom shows the following log entries:

```
INFO 2020-02-04 21:28:12,899 [(MuleRuntime).cpuLight.19] [question2]
INFO 2020-02-04 21:28:12,912 [(MuleRuntime).cpuLight.19] [question2]
ERROR 2020-02-04 21:28:12,899 [(MuleRuntime).cpuLight.18] [question2]
Message : Validation Error.
Error type : VALIDATION:NOT_NULL
Element : <validation:is-null doc:name="" doc:id="" value="#[payload]" message="Validation Error" />
Element XML : <validation:is-null doc:name="" doc:id="" value="#[payload]" message="Validation Error" />
```

NEW QUESTION # 23

Refer to the exhibits.

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

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-45YQ</transaction_id>
  <customer_id>anniepoint@acme.com</customer_id>
  <customer>Annie Point</customer>
  <item>Car Seat</item>
  <qty>1</qty>
  <price>59</price>
</sale>
```



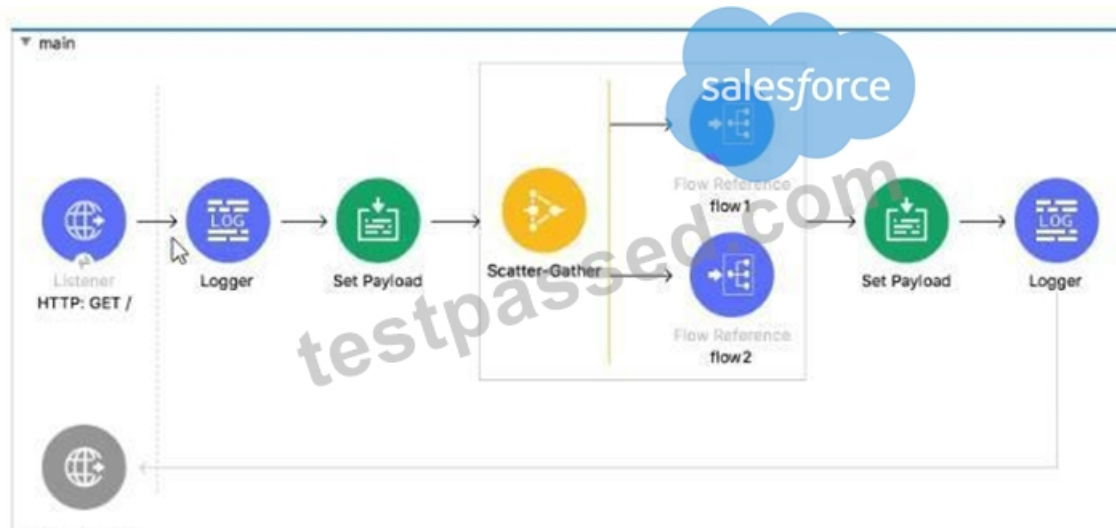
```
<flow name="writeRecords" >
  <http:listener doc:name="HTTP: POST /sales" config-ref="HTTP_Listener_config"
    path="/sales" allowedMethods="POST"/>
  <ee:transform doc:name="Add vendor name" >
    <ee:message >
      <ee:set-payload ><![CDATA[%dw 2.0
        output application/json
        ---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

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

Answer: C

NEW QUESTION # 24

Refer to the exhibit.



jeangcbs080262.snack-blog.com, thesocialcircles.com, emilyudou661491.bloggerswise.com, sltskills.com,
elaineixnn001451.goabroadblog.com, zubairkavl815835.signalwiki.com, heidilcia157662.blog5star.com, myportal.utt.edu.tt,
myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt,
myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, Disposable vapes

P.S. Free & New Salesforce-MuleSoft-Developer-I dumps are available on Google Drive shared by TestPassed:
https://drive.google.com/open?id=1Futc5X61zen6lf_y2AAQFb1usFyNDJ4J