SALESFORCE-MULESOFT-DEVELOPER-II$^{Q\&As}$

Salesforce Certified MuleSoft Developer 2 (SP24)

Pass Salesforce SALESFORCE-MULESOFT-DEVELOPER-II Exam with 100% Guarantee

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

QUESTION 1

A heathcare customer wants to use hospital system data, which includes code that was developed using legacy tools and methods. The customer has created reusable Java libraries in order to read the data from the system. What is the most effective way to develop an API retrieve the data from the hospital system?

A. Refer to JAR files in the code

B. Include the libraries writes deploying the code into the runtime

C. Create the Java code in your project and invoice the data from the code

D. Install libraries in a local repository and refer to it in the pm.xml file

Correct Answer: D

To develop an API that retrieves data from a hospital system using reusable Java libraries, the developer should install libraries in a local repository and refer to it in the pom.xml file. This way, the developer can use Maven to manage dependencies and invoke Java code from Mule applications using Java Module operations.
https://docs.mulesoft.com/mule-runtime.3/java- module-reference#add-the-java-module-to-your-project
https://docs.mulesoft.com/muleruntime/4.3/ java-module-reference#invoke-java-code

QUESTION 2

Which statement is true about using mutual TLS to secure an application?

A. Mutual TLS requires a hardware security module to be used

B. Mutual TLS authenticates the identity of the server before the identity of the client

C. Mutual TLS ensures only authorized end users are allowed to access an endpoint

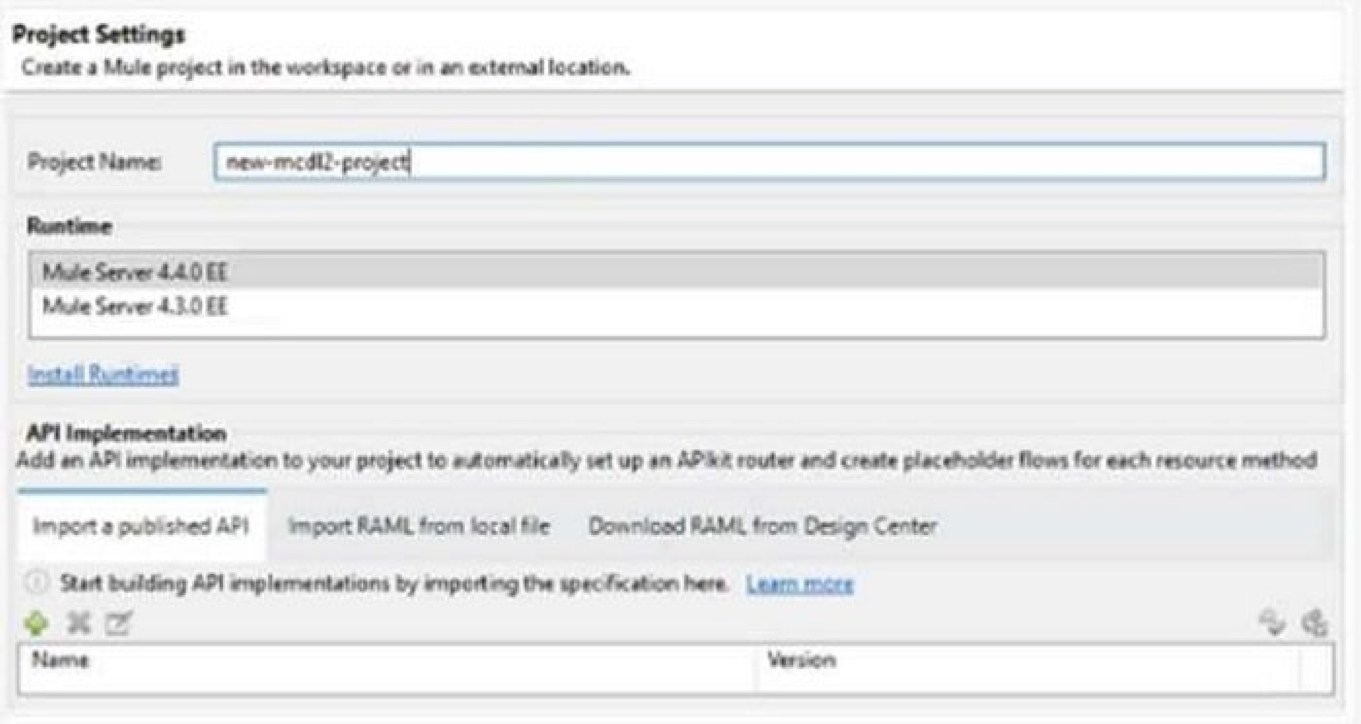
D. Mutual TLS increases the encryption strength versus server-side TLS alone

Correct Answer: B

Mutual TLS (mTLS) is an extension of TLS that requires both parties (client and server) to present their certificates to each other during the handshake process. This way, both parties can verify each other\\'s identity and establish a secure connection. The authentication of the server happens before the authentication of the client, as the server sends its certificate first and then requests the client\\'s certificate. https:// docs.mulesoft.com/mule-runtime.3/tls-configuration#mutualauthentication

QUESTION 3

Refer to the exhibit.



Project Settings
Create a Mule project in the workspace or in an external location.

Project Name: new-mcdl2-project

Runtime
Mule Server 4.4.0 EE
Mule Server 4.3.0 EE

Install Runtimes

API Implementation
Add an API implementation to your project to automatically set up an APIkit router and create placeholder flows for each resource method

Start building API implementations by importing the specification here. Learn more

When creating a new project, which API implementation allows for selecting the correct API version and scaffolding the flows from the API specification?

A. Import a published API

B. Generate a local RAML from anypoint Studio

C. Download RAML from Design Center

D. Import RAML from local file

Correct Answer: C

To create a new project that selects the correct API version and scaffolds the flows from the API specification, the developer should import a published API. This option allows importing an API specification that has been published to Anypoint Exchange or Design Center, and selecting a specific version of that API specification. The developer can also choose to scaffold flows based on that API specification. https://docs.mulesoft.com/ apikit.x/apikit-4-new-project-task

QUESTION 4

A company deploys 10 public APIs to CloudHub. Each API has its individual health endpoint defined. The platform operation team wants to configure API Functional Monitoring to monitor the health of the APIs periodically while minimizing

operational overhead and cost.

How should API Functional Monitoring be configured?

A. From one public location with each API in its own schedule

B. From one private location with all 10 APIs in a single schedule

C. From one public location with all 10 APIs in a single schedule

D. From 10 public locations with each API in its own schedule

Correct Answer: C

To configure API Functional Monitoring to monitor the health of 10 public APIs periodically while minimizing operational overhead and cost, the developer should use one public location with all 10 APIs in a single schedule. A public location is a worker that runs in a CloudHub shared environment, which is cheaper and easier to maintain than a private location. A single schedule allows running all 10 APIs tests at the same time and frequency, which reduces complexity and resource consumption. https://docs.mulesoft.com/functional-monitoring/fm-create-monitor#create-a-monitor

QUESTION 5

An order processing system is composed of multiple Mule application responsible for warehouse, sales and shipping. Each application communication using Anypoint MQ. Each message must be correlated against the original order ID for observability and tracing. How should a developer propagate the order ID as the correlation ID across each message?

A. Use the underlying HTTP request of Anypoint MQ to set the "X-CORRELATION_ID\\' header to the order ID

B. Set a custom Anypoint MQ user property to propagate the order ID and set the correlation ID in the receiving applications.

C. Use the default correlation ID, Anypoint MQ will sutomatically propagate it.

D. Wrap all Anypoint MQ Publish operations within a With CorrelationID scope from the Tracing module, setting the correlation ID to the order ID

Correct Answer: D

To propagate the order ID as the correlation ID across each message using Anypoint MQ, the developer should wrap all Anypoint MQ Publish operations within a With CorrelationID scope from the Tracing module, setting the correlation ID to the order ID. The With CorrelationID scope allows setting a custom correlation ID for any event that occurs within it. The Tracing module also enables distributed tracing across different Mule applications and services using Anypoint Monitoring. https://docs.mulesoft.com/tracing-module.0/tracing- module-reference#withcorrelation-id-scope https://docs.mulesoft.com/tracing-module.0/tracing-module-concepts

QUESTION 6

A Mule implementation uses a HTTP Request within an Unit Successful scope to connect to an API. How should a permanent error response like HTTP:UNAUTHORIZED be handle inside Until Successful to reduce latency?

A. Configure retrying until a MULERETRY_EXHAUSTED error is raised or the API responds back with a successful response.

B. In Until Successful configuration, set the retry count to 1 for error type HTTP: UNAUTHORIZED.

C. Put the HTTP Request inside a try scope in Unit Successful. In the error handler, use On Error Continue to catch permanent errors like HTTP UNAUTHORIZED.

D. Put the HTTP Request inside a try scope in Unit Successful. In the error handler, use On Error Propagate to catch permanent errors like HTTP UNAUTHORIZED.

Correct Answer: C

To handle a permanent error response like HTTP:UNAUTHORIZED inside Until Successful, the developer should put the HTTP Request inside a try scope in Unit Successful, and use On Error Continue to catch permanent errors like HTTP UNAUTHORIZED in the error handler. This way, the developer can avoid retrying requests that will always fail due to a permanent error, and reduce latency. On Error Continue allows the flow to continue processing after handling the error. Reference: https://docs.mulesoft.com/mule-runtime.3/until-successful-scope https://docs.mulesoft.com/mule-runtime.3/on-error-continue-concept

QUESTION 7

The Center for Enablement team published a common application as a reusable module to the central Nexus repository. How can the common application be included in all API implementations?

A. Download the common application from Naxus and copy it to the src/main/resources folder in the API

B. Copy the common application\\'s source XML file and out it in a new flow file in the src/main/mule folder

C. Add a Maven dependency in the PCM file with multiple-plugin as

D. Add a Maven dependency in the POM file with jar as

Correct Answer: D

To include a common application as a reusable module in all API implementations, the developer should add a Maven dependency in the POM file with jar as . This way, the developer can reuse Mule code from another application by packaging it as a JAR file and adding it as a dependency in the POM file of the API implementation. The classifier element specifies that it is a JAR file. https://docs.mulesoft.com/mule-runtime.3/mmp-concept#add-a-maven-dependencytothe-pom-file

QUESTION 8

A Mule application uses API autodiscovery to access and enforce policies for a RESTful implementation.

A. Northing because flowRef is an optional attribute which can be passed runtime

B. The name of the flow that has APIkit Console to receive all incoming RESTful operation requests.

C. Any of the APIkit generate implement flows

D. The name of the flow that has HTTP listener to receive all incoming RESTful operation requests

Correct Answer: D

To use API autodiscovery to access and enforce policies for a RESTful implementation, flowRef must be set to the name of the flow that has HTTP listener to receive all incoming RESTful operation requests. This way, API autodiscovery can identify the API implementation and associate it with the corresponding API specification and policies in API Manager. The flow that has HTTP listener is usually the main flow that contains the APIKit Router. https://docs.mulesoft.com/apimanager/ 2.x/api-auto-discovery-new-concept#flowref

QUESTION 9

A Mule application defines as SSL/TLS keystore properly "tis,keystore.keyPassword\\'\\' as secure. How can this property be referenced to access its value within the application?

A. #{secure::tiskeystore,keyPassowrd}

B. ${secure::tiskeystore,keyPassowrd}

C. ${secure::tiskeystore,keyPassowrd}

D. p{secure::tiskeystore,keyPassowrd}

Correct Answer: B

secure::tiskeystore,keyPassowrd ShortExplanationofCorrectAnswerOnly:Toreferenceasecureproper tyvaluewithintheapplication,thedeveloperneedstousethesyntax{secure::}. In this case, the property name is tiskeystore,keyPassword, so the correct syntax is ${secure::tiskeystore,keyPassowrd}. https:// docs.mulesoft.com/mule-runtime.3/secure-configurationproperties#referencing-secure-properties

QUESTION 10

A Mule application need to invoice an API hosted by an external system to initiate a process. The external API takes anywhere between one minute and 24 hours to compute its process. Which implementation should be used to get response data from the external API after it completes processing?

A. Use an HTTP Connector to invoke the API and wait for a response

B. Use a Scheduler to check for a response every minute

C. Use an HTTP Connector inside Async scope to invoice the API and wait for a response

D. Expose an HTTP callback API in Mule and register it with the external system

Correct Answer: D

To get response data from the external API after it completes processing, the developer should expose an HTTP callback API in Mule and register it with the external system. This way, the external API can invoke the callback API with the

response data when it is ready, instead of making the Mule application wait for a long time or poll for a response repeatedly.

Reference:

https://docs.mulesoft.com/mule-runtime.3/http-listener-ref#callback