

PDFVCE



Choose the version that fits your needs	PDF Version	Desktop Test Engine	Online Test Engine
Latest and Up-to-Date exam dumps with real exam questions answers.	✓	✓	✓
Get 12-Months free updates without any extra charges.	✓	✓	✓
Experience same exam environment before appearing in the certification exam.	✗	✓	✓
100% exam passing guarantee in the first attempt.	✓	✓	✓
20% discount on more than one license and 30% discount on 5+ license purchases.	✗	✓	✓
100% secure purchase on SSL.	✓	✓	✓
Completely private purchase without sharing your personal info with anyone.	✓	✓	✓

<http://www.pdfvce.com>

Highly Efficiently Exam Tool and Effective Exam Practice Materials

Exam : **Mule-Dev-202**

Title : Salesforce Certified MuleSoft
Hyperautomation Developer

Vendor : Salesforce

Version : DEMO

NO.1 The MuleSoft development team at Northern Trail Outfitters creates a Mule application that interacts with several APIs and RPA processes. The team needs to share this application with other teams to help them create similar applications.

How should the Mule application be published in Anypoint Exchange to meet this requirement?

- A.** Template asset
- B.** Custom asset
- C.** Connector asset
- D.** API asset

Answer: A

Explanation:

To share a Mule application that interacts with several APIs and RPA processes with other teams for creating similar applications, publishing it as a Template asset is the best approach:

Template Asset:

A Template in Anypoint Exchange is designed to provide a reusable solution that can be easily adapted for different use cases. It includes predefined integration logic and configurations that can be used as a starting point for new projects.

By publishing the Mule application as a Template, other teams can leverage this pre-built solution, customize it to their specific needs, and ensure consistency in integration practices across the organization.

References:

Anypoint Exchange Templates Documentation

NO.2 Which component of Anypoint Platform is responsible for enforcing API policies?

- A.** API Analytics
- B.** API Runtime
- C.** API Gateway
- D.** API Manager

Answer: C

Explanation:

The component of Anypoint Platform responsible for enforcing API policies is the API Gateway:

API Gateway:

The API Gateway is a runtime component that enforces policies applied to APIs. It acts as an intermediary that manages API traffic, security, and performance by applying the policies configured in API Manager.

Policies such as rate limiting, security, transformation, and monitoring are enforced at the API Gateway to ensure that APIs are accessed securely and efficiently.

References:

Anypoint Platform API Gateway Documentation

NO.3 AnyAirlines has MuleSoft Composer installed on their production Salesforce environment.

To test flows with data in multiple non-production environments, what does the hyperautomation specialist need to do?

- A.** Create a connection to each of the non-production environments within the Composer UI.
- B.** Install MuleSoft Composer in each of the non-production Salesforce environments.
- C.** Install MuleSoft Composer in only one non-production Salesforce environment and create a proxy

to all other non-production environments.

D. Use mocked data because non-production data is not available to MuleSoft Composer.

Answer: A

Explanation:

Create Connections: To test flows with data in multiple non-production environments, creating connections to each environment within the MuleSoft Composer UI is necessary. This allows the Composer to access and manipulate data across different environments, ensuring comprehensive testing.

Reference: MuleSoft Composer Connections

Installing Composer in Non-Production Environments: While installing Composer in each environment is technically possible, creating individual connections is more efficient and aligns with best practices.

Reference: Composer Setup and Configuration

Using Mocked Data: Mocked data can be useful for initial testing, but connecting to actual non-production environments provides more realistic test scenarios.

Reference: Testing with Mock Data

Proxy Setup: Creating a proxy to other environments is complex and unnecessary when Composer supports direct connections.

Reference: Proxy Configuration

NO.4 AnyAirlines wants to create a new marketing campaign that sends customers special offers every month based on their accrued loyalty points. There is an existing integration for customer data using MuleSoft's API-led three-tier strategy. Loyalty information exists in an external system that can be accessed via an HTTP endpoint provided by the system, but has no current integration. The external ID used will be email address.

The desired output is a CSV file containing customers that includes only the top 10 percent of loyalty point holders.

What is the most efficient way to meet this requirement?

A. 1. Have the MuleSoft team develop a new integration that includes a System API to the Loyalty system and uses the existing Customer System API. 2. Create a Process API to output the final results. 3. Create an Experience API for the business consumers to initiate the integration.

B. 1. Create a MuleSoft Composer flow that utilizes the current Customer integration to select all customers. 2. Create an additional MuleSoft Composer flow that retrieves all the Loyalty information. 3.

Create a MuleSoft Composer flow that combines the two previous results and outputs the top 10 percent to a CSV file.

C. 1. Have the MuleSoft team develop a new integration that includes a new System API to both the Customer and Loyalty systems. 2. Create a Process API to output the final results. 3. Create an Experience API for the business consumers to initiate the integration.

D. 1. Create a Salesforce Flow that retrieves the Contact data. 2. Create a Salesforce Flow that retrieves the Loyalty data. 3. Create a Flow Orchestration that uses the two flows and outputs the result to a CSV file.

Answer: A

Explanation:

Develop System API for Loyalty System: The first step is to develop a new System API that integrates with the Loyalty system. This API will handle communication with the external system via the

provided HTTP endpoint.

Reference: MuleSoft API-led Connectivity

Utilize Existing Customer System API: Use the existing System API for customer data to retrieve necessary customer information. Combining these APIs ensures a modular approach and reuse of existing assets.

Reference: API-led Connectivity

Create Process API: Develop a Process API that combines data from both the Customer and Loyalty System APIs. This API will process the data, apply business logic to filter the top 10 percent of loyalty point holders, and format the results.

Reference: Designing Process APIs

Create Experience API: Develop an Experience API to serve the business consumers. This API will provide a user-friendly interface for initiating the integration and retrieving the results as a CSV file.

Reference: API Experience Layer

NO.5 Northern Trail Outfitters is building a hyperautomation solution using Salesforce and MuleSoft. They need to use Salesforce Flow to automate a multi-departmental process in an external system and capture the outcome in Salesforce.

How should the Salesforce Flow solution be structured to meet this requirement?

- A.** An autolaunched flow invoked by REST API to update Salesforce after the process is completed
- B.** A Flow Orchestration to automate the multi-departmental process and update Salesforce records
- C.** Parent and subflows invoked by REST API to capture user inputs and update Salesforce records
- D.** An evaluation flow which evaluates when the process is completed and updates Salesforce records

Answer: B

Explanation:

Salesforce Flow Orchestration is designed to manage complex, multi-step business processes that span multiple departments and systems. Here's how it can be structured to meet the requirement:

Automate Multi-Departmental Process:

Use Flow Orchestration to define and manage the steps involved in the multi-departmental process. It allows you to break down the process into stages and define the sequence of actions and approvals required.

Capture Outcome in Salesforce:

After completing the external process, Flow Orchestration can be configured to update Salesforce records with the outcome. This ensures that the results of the automated process are reflected within Salesforce.

Orchestration Capabilities:

Salesforce Flow Orchestration provides features such as task assignments, decision elements, and complex branching logic, which are ideal for managing multi-departmental workflows.

References:

Salesforce Flow Orchestration Documentation

NO.6 AnyAirlines is developing an RPA process and is implementing testing best practices. They want to take the RPA process through rigorous testing.

During these tests, where do RPA process test plans execute?

- A.** On a configured RPA Bot

- B. In RPA Manager
- C. In RPA Builder
- D. In an RPA process runtime

Answer: A

Explanation:

During testing of an RPA process, test plans are executed on a configured RPA Bot. This allows you to simulate real-world scenarios and ensure the RPA process works correctly under various conditions:

On a Configured RPA Bot:

RPA Bots are configured to execute the automated tasks defined in the RPA process. By running test plans on these bots, you can verify the functionality and performance of the RPA process.

This approach ensures that the RPA process is thoroughly tested in an environment that closely mirrors production conditions.

References:

MuleSoft RPA Documentation