

MULESOFT-INTEGRATION- ASSOCIATE^{Q&As}

Salesforce Certified MuleSoft Integration Associate (SP24)

**Pass Salesforce MULESOFT-INTEGRATION-
ASSOCIATE Exam with 100% Guarantee**

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.leads4pass.com/mulesoft-integration-associate.html>

100% Passing Guarantee
100% Money Back Assurance

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

- ⚙️ **Instant Download** After Purchase
- ⚙️ **100% Money Back** Guarantee
- ⚙️ **365 Days** Free Update
- ⚙️ **800,000+** Satisfied Customers



QUESTION 1

An organization is not meeting its growth and innovation objectives because IT cannot deliver projects fast enough to keep up with the pace of change required by the business.

According to MuleSoft's IT delivery and operating model which step should the organization take to solve this problem?

- A. Adopt a new approach that decouples core IT projects from the innovation that happens within each line of business
- B. Switch from a design-first to a code-first approach for IT development
- C. Modify IT governance and security controls so that line of business developers can have direct access to the organization's systems of record
- D. Hire more IT developers, architects, and project managers to increase IT delivery

Correct Answer: A

MuleSoft's IT delivery and operating model suggests modernizing IT practices to better support business growth and innovation.

References:

MuleSoft Whitepaper: API-led Connectivity

IT Operating Model: Transforming IT Delivery

QUESTION 2

An IT integration team followed an API-led connectivity approach to implement an order- fulfillment business process. It created an order processing API that coordinates stateful interactions with a variety of microservices that validate, create and fulfill new product orders.

Which interaction composition pattern did the integration architect who designed this order processing API use?

- A. Multicasting
- B. Orchestration
- C. Streaming
- D. Aggregation

Correct Answer: B

In an API-led connectivity approach, different APIs are layered to provide modular and reusable services. For an order processing API that coordinates stateful interactions with various microservices, the integration architect used the orchestration interaction composition pattern. References: MuleSoft Documentation: Orchestration Pattern API-led Connectivity: MuleSoft API-led Connectivity

QUESTION 3

A developer is examining the responses from a RESTful web service that is compliant with the Hypertext Transfer Protocol (HTTP/1.1) as defined by the Internet Engineering Task Force (IETF).

In this HTTP/1.1-compliant web service, which class of HTTP response status codes should be specified to indicate when client requests are successfully received, understood and accepted by the web service?

- A. 2xx
- B. 3xx
- C. 5xx
- D. 4xx

Correct Answer: A

In HTTP/1.1, response status codes are categorized to indicate the result of a client's request.

References:

IETF RFC 7231: HTTP/1.1 Semantics and Content

HTTP Status Codes: HTTP Status Code Definitions

QUESTION 4

An organization's IT team follows an API-led connectivity approach and must use Anypoint Platform to implement a System API that securely accesses customer data. The organization uses Salesforce as the system of record for all customer data and its most important objective is to reduce the overall development time to release the System API.

The team's integration architect has identified four different approaches to access the customer data from within the implementation of the System API by using different Anypoint Connectors that all meet the technical requirements of the project.

Which approach should the team choose to meet the organization's objective to reduce the time to develop and release the System API?

- A. Use the Anypoint Connector for Salesforce to connect to the Salesforce APIs to directly access the customer data.
- B. Use the Anypoint Connector for HTTP to connect to the Salesforce APIs to directly access the customer data.
- C. Use the Anypoint Connector for Database to connect to a MySQL database to access a copy of the customer data.
- D. Use the Anypoint Connector for FTP to download a file containing a recent near-real time extract of the customer data.

Correct Answer: A

In an API-led connectivity approach, using the most efficient method to access system data can significantly reduce development time.

References:

MuleSoft Documentation: Salesforce Connector

API-led Connectivity: MuleSoft API-led Connectivity

QUESTION 5

A MuleSoft developer must implement an API as a Mule application, run the application locally, and execute unit tests against the running application

Which Anypoint Platform component can the developer use to fulfill all of these requirements?

- A. API Designer
- B. API Manager
- C. Anypoint Studio
- D. Anypoint CLI

Correct Answer: C

Anypoint Studio is the integrated development environment (IDE) provided by MuleSoft for designing, developing, and testing Mule applications.

References:

MuleSoft Documentation: Anypoint Studio

MUnit Testing: MUnit

QUESTION 6

What is a defining characteristic of an Integration-Platform-as-a-Service (iPaaS)?

- A. No-code
- B. Code-first
- C. On-premises
- D. Cloud-based

Correct Answer: D

An Integration-Platform-as-a-Service (iPaaS) is characterized by being a cloud-based solution that provides tools to develop, execute, and manage integration flows connecting multiple applications and data sources.

References:

iPaaS Overview: What is iPaaS?

Cloud-based Integration: iPaaS Characteristics

QUESTION 7

During a planning session with the executive leadership, the development team director presents plans for a new API to expose the data in the company's order database. An earlier effort to build an API on top of this data failed, so the director is recommending a design-first approach.

Which characteristics of a design-first approach will help make this API successful?

- A. Building MUnit tests so administrators can confirm code coverage percentage during deployment
- B. Publishing the fully implemented API to Exchange so all developers can reuse the API
- C. Developing a specification so consumers can test before the implementation is built
- D. Adding global policies to the API so all developers automatically secure the implementation before coding anything

Correct Answer: C

A design-first approach emphasizes creating the API specification before implementation, ensuring better alignment with consumer needs and reducing the risk of project failure.

References:

Design-First Approach: Design-First API Development API Mocking: API Designer Mocking Service

QUESTION 8

Which AnypointPlatform component helps integration developers discover and share reusable APIs, connectors and templates?

- A. Anypoint Exchange
- B. Design Center
- C. API Manager
- D. Anypoint Studio

Correct Answer: A

Anypoint Exchange is a component of the Anypoint Platform that helps integration developers discover and share reusable APIs, connectors, and templates.

References:

MuleSoft Documentation: Anypoint Exchange

Anypoint Platform Features: Discover and Share with Exchange

QUESTION 9

An application load balancer routes requests to a RESTful web API secured by Anypoint Flex Gateway

Which protocol is involved in the communication between the load balancer and the Gateway?

- A. LDAP
- B. HTTPS
- C. SFTP
- D. SMTP

Correct Answer: B

In scenarios where an application load balancer routes requests to a RESTful web API secured by Anypoint Flex Gateway, HTTPS is the protocol used.

References:

HTTPS Protocol:What is HTTPS?

API Gateway Security: Anypoint Flex Gateway

QUESTION 10

An integration architect is designing an API that must accept requests from API clients for both XML and JSON content over HTTP/1.1 by default.

Which API architectural style when used for its intended and typical purposes, should the architect choose to meet these requirements?

- A. SOAP
- B. GraphQL
- C. REST
- D. gRPC

Correct Answer: C

REST (Representational State Transfer) is an architectural style commonly used for designing networked applications, particularly APIs that need to handle multiple content types over HTTP.

References:

REST API Design:RESTful Web Services

Content Negotiation:HTTP Content Negotiation

[Latest MULESOFT-INTEGRATION-ASSOCIATE Dumps](#) | [MULESOFT-INTEGRATION-MULESOFT-INTEGRATION-ASSOCIATE PDF Dumps](#) | [ASSOCIATE Exam](#)

