MuleSoft Certified Developer – Integration Professional (Mule 3) Certification Exam

Summary

A *MuleSoft Certified Developer – Integration Professional* should have extensive, real-world MuleSoft development experience and be able to develop and provide technical leadership on complex Anypoint Platform integration projects. The *MCD – Integration Professional (Mule 3)* exam validates that a developer has the required knowledge and skills to:

- Create and configure flows, subflows, and flow processing strategies for reusability and performance.
- Select and use appropriate types of transformers and modules to persist data and write MEL expressions to access and modify messages.
- Write DataWeave expressions to transform data.
- Control message content, flow, and processing by selecting and implementing appropriate connectors, routers, scopes, and filters.
- Design and implement comprehensive error handling strategies for applications.
- Implement and consume REST and SOAP web services.
- Create and use custom Java components.
- Use JUnit and MUnit to test Mule applications.
- Configure and deploy Mule applications to CloudHub and/or single or clusters of customer-hosted Mule runtimes.

Format

- Format: Multiple-choice, closed book, in a testing center or proctored online using a qualifying, external webcam
- Length: 100 questions
- Duration: 120 minutes (2 hours)
- Pass score: 80%
- Language: English

The exam can be taken a maximum of 5 times, with a 24 hour wait between each attempt.

Cost

The exam can be purchased with one of the following. Each includes a coupon for one free retake.

- $250
- 1 Flexible Training Credit (FTC)

Additional retakes (attempts 3 to 5) are $125 or 0.5 FTC off and do not come with a free retake.

Validity

The certification expires two years from the date of passing.

Preparation

The best preparation for the exam is to take the instructor-led *Anypoint Platform Development: Fundamentals (Mule 3)* and *Anypoint Platform Development: Advanced (Mule 3)* courses, to review and get experience with the exam topics, and to get 6 months to several years of real-world development experience on various MuleSoft projects.

Note: This exam does not require a previous MCD - Integration and API Associate (Mule 3.8) certification but it is highly recommended.

Topics

The exam validates that the candidate can perform the following tasks.

General
• Explain basic MuleSoft implementation and design concepts • Track data movement through an application
Basics
• Identify when to use and use flow variables and session variables • Write Mule expressions • Define Mule properties and create properties files
HTTP Connector
• Create and configure inbound and outbound HTTP endpoints • Use HTTP and HTTPS • Define HTTP content-type and explain its effect on browser types
Flows
• Use flows, sub-flows, and flow references • Explain the differences between inbound and outbound endpoints • Configure flow processing strategies • Code and test exchange patterns (including request-response and one-way) • Test Mule applications using JUnit and MUnit cases • Send a Mule message from a test class to a Mule application

Flow Control
• Use splitters, aggregators, and multicast routers • Use the For-each scope • Use filters
Error Handling
• Debug flows and expression handlers • List the different exception strategies that are available • Use exception strategies and explain how they affect flows and sub-flows • Change and return a message from an exception strategy • Configure global application exception handling • Use routers (including First Successful and Until Successful) to handle potential error conditions
Transformations with DataWeave
• Write DataWeave scripts to convert JSON, XML, and Java data structures to different data structures and data types • Use DataWeave operators • Define and use custom data types • Apply correct DataWeave syntax to coerce data types • Apply correct DataWeave syntax to format strings, numbers, and dates • Call Mule flows from a DataWeave script • Call global MEL functions from a DataWeave script
Web Services
• Implement REST services with GET, POST, PUT, and DELETE methods • Use annotations on REST methods to create unique signatures • Create REST clients and working with dynamic endpoints • Publish and consume SOAP messages • Use CXF interfaces to create service definitions • Extend interfaces to create CXF implementations
Scopes
• Configure and use batch processing • Use the Cache Scope to store and reuse frequently called data • Create and manage caching strategies • Use Enrichers to enhance a Mule message

Deployment
• Explain the general concepts and benefits for building Mule clusters • Manage runtime clusters • Use queues to distribute application flows for processing in clusters • Describe how clustering supports various Mule transport mechanisms • Deploy applications to customer-hosted Mule runtimes • Deploy applications to CloudHub • Organize Spring properties and Spring property file configuration
Java Components
• Create and test Java custom components and integrating them into flows • Use advanced Java concepts to invoke service calls for passing Mule messages • Create custom filters with Java • Configure Java components to be prototypes or singletons • Use the default entry point resolver with Java components
Connectors and Transports
• Configure and use Database connectors • Explain how Database inbound and outbound endpoints differ and their limitations • Configure JMS connectors for two-way communications, temporary queues, and object serialization over transports • Use back channels and creating two-way communication through JMS connections • Describe how JMS uses correlation IDs • Use VM Transport to control how messages are sent and received by components in a system • Use VM Transport for communication between Mule flows • Explain queue usage with VM Transport and configuration structure • Configure and use File and FTP connectors
Transactions
• Explain transaction management • Identify which endpoints support transactions • Manage and configure resource transactions for inbound and outbound messages • List the various transaction types and usage techniques

Delivery methods

The exam is administered via the Kryterion Webassessor testing platform. The exam can be taken in-person at a testing center or online using a web camera.

In-person at a Kryterion Testing Center:

- Over 1000 locations worldwide
- Onsite instructions
- Test-taker guide

Online using the Kryterion Webassessor testing platform:

- Requires a webcam - a laptop webcam can be used, an external camera is not required
- Requires internet connectivity with 1 Mbps upload, 1 Mbps download, jitter <50ms, ping <200ms
- Check internet speed and reliability
    - Note: Some candidates are expelled from the exam for an unstable connection even after checking reliability with the tool. If you think your connection could potentially be unreliable, we **strongly** recommend scheduling your exam at a test center.
- Online instructions
- Test-taker guide

Registration

To register for the exam:

- Go to https://training.mulesoft.com/webassessor.
- Create a user profile.
- Log in.
- Select Register for an Exam.
- Select the ***MuleSoft Certified Developer – Integration Professional (Mule 3)*** exam.
- Select either the Online Proctoring Option or the Kryterion Test Center option.
- On the payment screen, select to pay by credit card or enter a voucher/coupon code.

    Note: A fee applies if an exam is cancelled or rescheduled within 72 hours of its scheduled time, even if the exam was purchased with a voucher.

More information

For more information, visit http://help.learn.mulesoft.com.

Sample questions

Answers are provided at the end.

1. **Can a flow have more than one message source?**
    A. Yes, using the composite source
    B. Yes, if declared sequentially at the beginning of the flow
    C. No, Mule validates on startup that the next element after a message source is a message processor
    D. It depends on if the two message sources have compatible transport types

2. **Fill in the Blank: The Mule runtime and CloudHub can have multiple _____, each of which can be transmitting concurrent _____.**
    A. Applications, Messages
    B. Nodes, Loads
    C. Flows, Endpoints
    D. Endpoints, Workers

3. **Where can't I define an exception strategy?**
    A. Sub-flow
    B. Flow
    C. Private flow
    D. Global configuration

4. **Expressions can be used in a MuleSoft application:**
    A. To extract information from the current message
    B. With routers and filters for defining routing logic
    C. For filtering out unwanted messages
    D. All of the above

5. **One can use Mule Expression Language in which of the following contexts?**
    A. Application
    B. Message
    C. Server
    D. All

6. **When receiving HTTP responses, the payload of the Mule message will always be:**
    A. A HashMap of all query parameters
    B. An InputStream
    C. Dependent on the URL and how it comes to Mule (webserver, proxy, etc.)
    D. Either JSON, a string or a HashMap depending on the request

7. **If the component implements Callable lifecycle interface, what method would you use to receive the message?**
    A. onCall()
    B. initialize()
    C. Start()
    D. None of the above

8. **What are the databases that MuleSoft integrations can connect to?**
    A. Any JDBC compliant database
    B. Oracle, MySQL, Postgres, Derby, DB2, MSSQL
    C. Any database
    D. Oracle and Postgres

9. **What is the purpose of setting the maxRedelivery attribute on the connector URL?**
    A. To deliver each message in the queue up to the value set in maxRedelivery, without differentiating between messages that generated errors and others
    B. That attribute is not possible to set on a JMS connector
    C. To avoid an infinite loop trying to process a message that will always generate an error
    D. To define the maximum number of receivers for the message

10. **What's the correct way to define a dynamic inbound endpoint?**
    A. Dynamic endpoints are only supported for outbounds
    B. <http:inbound-endpoint host="0.0.0.0" path="/#[expresion.for.path]" port="port" />
    C. <http:inbound-endpoint address="http://host:port/#[expresion.for.path]" />
    D. <dynamic-source expression=""[expresion.for.path]">
        <http:inbound-endpoint rootAddress="http://host:port/" />
        </dynamic-source>

11. **What is a Scope?**
    A. It is a construct that contains message processors and limits the scope of flow variables
    B. It is another name for flows and sub-flows
    C. It is a construct that contains message processors, and changes the way they execute
    D. It is a construct that executes message processors asynchronously to the container flow

12. **What is the appropriate way to log SOAP messages in CXF?**
    A. Logging the payload as a String
    B. Using cxf:inInterceptors
    C. Enabling logging in the CXF configuration

Question	Answer
1	A
2	A
3	A
4	D
5	D
6	B
7	A
8	A
9	C
10	A
11	C
12	B