

**Connecting APIs Together using MuleSoft**

**Course Number:** MULE-108
**Duration:** 2.5 days

**Overview**

This hands-on, online Connecting APIs Together using MuleSoft training course teaches students how various processors impact the payload, attributes, and variables of a Mule event. Students learn how to prepare their APIs well by decentralizing flows, organizing application files, parameterizing application properties, and organizing project files. Students then consume REST and SOAP web services to orchestrate business logic according to use case requirements.

**Prerequisites**

Students must have experience with creating, deploying, and managing APIs in Anypoint Studio and Anypoint Platform. Some knowledge of SOAP web services is helpful but not required.

**Materials**

All MuleSoft training students receive comprehensive courseware.

**Software Needed on Each Student PC**

Students will not need to install any software on their computers for this class. The class will be conducted in a remote environment. Students need a local computer with a web browser (preferably Chrome), stable internet, two monitors, and a headset/microphone.

**Objectives**

* Analyze a Mule Event in the DataSense Explorer, in the console at runtime, and in the Mule Debugger
* Track the Mule Event through an application and between flows, paying close attention to the persistence of the payload, attributes, and variables
* Use DataWeave to set payloads, response headers, request query parameters, and a logger value
* Define global elements, property placeholders, configuration files, and import all metadata
* Consume a RESTful web service using a REST connector and an HTTP request
* Consume a SOAP web service using a web service connector
* Route events to event processors by applying conditional logic using the Choice Router
* Use the Scatter-Gather component to route events in parallel and collect the results upon successful completion of all processors
* Use validators to check if a query parameter is valid and create custom error messages

**Outline**

* Observing Mule Event Data
	+ In the DataSense Explorer
	+ In the Logger at Runtime
	+ Using the Mule Debugger
* Mule Event Data Persistence
	+ Flow, Subflows, and Private Flows
	+ HTTP Request (transport boundary)
	+ Flow Reference
	+ VM Connector
	+ Synchronous and Asynchronous
* Be an Orderly Mule
	+ Create a Global.xml Configuration File
	+ Create a Properties File
	+ Parameterize Application Properties
	+ Organize Files and Folders
* Metadata Mania
	+ Manage in Transform Message
	+ Manage in Metadata Tab
	+ Manage in Package Explorer
	+ Stored in application-types.xml
* Are You an Orderly Mule?
	+ Create Subflows Using Flow Reference
	+ Create Subflows Using Export To
	+ Create a Global.xml Configuration File
	+ Create a Properties File
	+ Parameterize Application Properties
	+ Organize Files and Folders
* Manage Your Metadata
	+ Manage in Project Explorer
* Consume Web Services – Our API
	+ Consume Using a REST Connector
	+ Consume Using an HTTP Request
	+ Consume Using a SOAP Web Service
* Consume Web Services – Your API
	+ Consume Using a REST Connector
	+ Consume Using an HTTP Request
	+ Consume Using a SOAP Web Service
* Choice, Scatter-Gather, and Validators – Our API
	+ Route Based on Conditions
	+ Run All Routes at the Same Time
	+ Use the Validation Module
* Choice, Scatter-Gather, and Validators – Your API
	+ Route Based on Conditions
	+ Run All Routes at the Same Time
	+ Use the Validation Module
* Conclusion