

# Anypoint Platform Development: Fundamentals (Mule 4)

## Summary

This instructor-led course is for developers and architects who want to get hands-on experience using Anypoint Platform to build APIs and integrations. In the first part, students use Anypoint Platform discover, consume, design, build, deploy, manage, and govern APIs. In the second and third parts, students focus on using Mule and Anypoint Studio to build applications for use as API implementations and integrations.

This course prepares students to take the *MuleSoft Certified Developer – Level 1 (Mule 4)* exam and includes a voucher for two exam attempts.

## Duration

5 days in-person or online

## Objectives

At the end of this course, students should be able to:

- Build an application network using API-led connectivity and Anypoint Platform.
- Use Anypoint Platform to discover, consume, design, build, deploy, manage, and govern APIs.
- Connect to databases, files, web services, SaaS applications, JMS queues, and more.
- Add application logic, handle errors, and transform data using DataWeave.
- Structure applications to facilitate development and deployment.
- Handle batch data processing.

## Prerequisites

- Experience with an object-oriented language
- A basic understanding of data formats such as XML, CSV, and JSON
- A basic understanding of typical integration technologies such as HTTP, JMS, JDBC, REST, and SOAP

## Setup requirements

- A computer with at least 8-16 GB (16 highly recommended) available RAM, 2GHz CPU, and 10GB available storage and a minimum screen resolution of 1024x768
- Internet access to ports 80 and 3306 (with > 5Mbps download and > 2Mbps upload)
- The latest version of Chrome
- An Anypoint Platform account  
<http://anypoint.mulesoft.com>

- JDK 1.8  
<http://www.oracle.com/technetwork/java/javase/downloads/index.html>
- Anypoint Studio 7 with embedded Mule 4.1 runtime  
<https://www.mulesoft.com/lp/dl/studio>
- Advanced REST Client (or some other REST client application)  
<http://install.advancedrestclient.com>
- A Salesforce Developer account (NOT a standard account) and API Access token  
<https://developer.salesforce.com/en/signup>

A detailed setup document can be downloaded from here: <https://training.mulesoft.com/downloads>.

## Outline

### **PART 1: Building Application Networks with Anypoint Platform**

#### **Module 1: Introducing Application Networks and API-Led Connectivity**

- Explain what an application network is and its benefits
- Describe how to build an application network using API-led connectivity
- Explain what web services and APIs are
- Make calls to secure and unsecured APIs

#### **Module 2: Introducing Anypoint Platform**

- Describe the benefits of Anypoint Platform and MuleSoft's approach to be successful with it
- Describe the role of each component in building application networks
- Navigate Anypoint Platform
- Locate APIs and other assets needed to build integrations and APIs in Anypoint Exchange
- Build basic integrations to connect systems using flow designer

#### **Module 3: Designing APIs**

- Define APIs with RAML, the Restful API Modeling Language
- Mock APIs to test their design before they are built
- Make APIs discoverable by adding them to the private Anypoint Exchange
- Create public API portals for external developers

#### **Module 4: Building APIs**

- Use Anypoint Studio to build, run, and test Mule applications
- Use a connector to connect to databases
- Use the graphical DataWeave editor to transform data
- Create RESTful interfaces for applications from RAML files
- Connect API interfaces to API implementations

## **Module 5: Deploying and Managing APIs**

- Describe the options for deploying Mule applications
- Deploy Mule applications to CloudHub
- Use API Manager to create and deploy API proxies
- Use API Manager to restrict access to API proxies

## **PART 2: Building Applications with Anypoint Studio**

### **Module 6: Accessing and Modifying Mule Events**

- Log event data
- Debug Mule applications
- Read and write event properties
- Write expressions with the DataWeave expression language
- Create variables

### **Module 7: Structuring Mule Applications**

- Create applications composed of multiple flows and subflows
- Pass messages between flows using asynchronous queues
- Encapsulate global elements in separate configuration files
- Specify application properties in a separate properties file and use them in the application
- Describe the purpose of each file and folder in a Mule project
- Define and manage application metadata

### **Module 8: Consuming Web Services**

- Consume web services that have a connector in Anypoint Exchange
- Consume RESTful web services
- Consume SOAP web services
- Pass parameters to SOAP web services using the Transform Message component
- Transform data from multiple services to a canonical format

### **Module 9: Controlling Event Flow**

- Multicast events
- Route events based on conditions
- Validate events

## Module 10: Handling Errors

- Handle messaging errors at the application, flow, and processor level
- Handle different types of errors, including custom errors
- Use different error scopes to either handle an error and continue execution of the parent flow or propagate an error to the parent flow
- Set the success and error response settings for an HTTP Listener
- Set reconnection strategies for system errors

## Module 11: Writing DataWeave Transformations

- Write DataWeave expressions for basic XML, JSON, and Java transformations
- Write DataWeave transformations for complex data structures with repeated elements
- Define and use global and local variables and functions
- Use DataWeave functions
- Coerce and format strings, numbers, and dates
- Define and use custom data types
- Call Mule flows from DataWeave expressions
- Store DataWeave scripts in external files

## PART 3: Building Applications to Synchronize Data

### Module 12: Triggering Flows

- Read and write files
- Trigger flows when files are added, created, or updated
- Trigger flows when new records are added to a database table
- Schedule flows to run at a certain time or frequency
- Persist and share data in flows using the Object Store
- Publish and consume JMS messages

### Module 13: Processing Records

- Process items in a collection using the For Each scope
- Process records using the Batch Job scope
- Use filtering and aggregation in a batch step