Module 3: Exchange

1. What are connectors?

Connectors are software used for connecting mule flows and external resource.

We can use the connector to send or receive a message from Mule flow to one or more external services over protocol or API.

1. What are the different types of connectors?

Select

Premium

Mulesoft Certified

Community Connector

1. What is a REST connector?

Connectors which follow REST architectural pattern

Follows HTTP and HTTPS protocol

APIs available in Exchange can be used as REST Connectors

1. What is a SELECT Connector?

* Open source – can be used by everyone
* Maintained by Mulesoft
* Examples: HTTP, Database, Email, Salesforce, File Connector, ObjectStore
* HTTP Connector: Triggers mule flow, Consumes http service
* Salesforce: way to connect from Mule flow to Salesforce data
* File: Handle files and folders on a system, perform file operation from Mule flow
* ObjectStore: Used for storing synchronization information, user details and temporal information like token

1. What are premium connectors?

* Maintained by Mulesoft
* Must have premium subscription
* Examples: FTPS, AS2
* AS2: to send and receive business transactions over http and https
* FTPS: Connect to FTP server

1. What are mulesoft certified connectors?

* Developed by Mulesoft partners and developer community
* Reviewed and Certified by Mulesoft
* Examples: Google Analytics Connector, Atlassian JIRA REST Connector
* Google Analytics Connectors: to integrate google analytics data to other business apps
* Atlassian JIRA REST Connector: synchronize data between Jira and third party systems

1. What are community connectors?

* Created and maintained by Mulesoft or members of Mulesoft community
* Examples: Slack, Google calendar Connector
* Slack: Enables organizations to connect directly with Slack API
* Google calendar Connector: To use calendar events in real-time

1. What is a custom connector?

* Created by Mulesoft Developers
* Using the Mule SDK platform for Mule Runtime 4
* Mule 3 uses Devkit to create Custom connectors

1. How to create a custom connector?

* Go to the directory where you want to create connector (studio-workspace) and execute a command to create project structure with sample test cases:

$ mvn org.mule.extensions:mule-extensions-archetype-maven-plugin:generate

* Go to Anypoint studio, **File > Open Project from File System,** and select the project directory you have created in the last step.
* After you open this project in Anypoint studio, you will get the number of classes, annotated with mule SDK annotations as below:
* The connector is called as an **extension**in Mule 4, and it's class defined by the annotation **@Extension**. A **@Configurations** annotation is used to point the configuration class.
* We can use **@ConnectionProvider** and **@Operation** annotation with this class to point out the connection providers and the operations connector has implemented.
* This class is used to manage and provide the connection with the targeted system.
* The connection provider must implement the one of the connection provider available in Mule. Here, we are implementing the **PoolingConnectionProvider**, other options are **CachedConnectionProvider**and **ConnectionProvider**.
* Parameter required to establish connection must be defined into the ConnectionProvider class.
* Also, you must override connect, disconnect, and validate methods to provide the functionality specific to this connection provider.
* The Connection class is used by connection providers to handle the connection. By having the single connection class and multiple connection providers, we can create multiple connection configurations for our Connector.
* We can define any number of operation classes and all public methods from this class will be treated as a connector operation. We can inject configurations and connection to particular operation using **@Config** and **@Connection** annotation in method arguments.
* We can install this connector into local maven repository using the command:

$mvn clean install

* You can use this connector in your **Mule 4** application by adding the dependency in pom.xml