**LAB- Understanding classloading and reusing flow in a separate project**

**STEP1**

IN this step, you will be working on **01-classloading-start** project

Open sharedflowsapp.xml inside **01-sharedflows-app** application and observe that there is a simple private flow with name “**mysharedflow**”

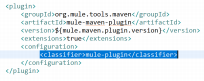
We want this flow to be reused in other applications.

So, we need to add this app as a maven dependency in another app. So, we have to do a mvn install.

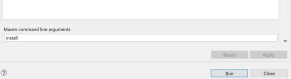
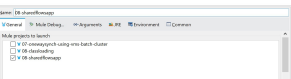
Before doing mvn install, do the below step:

Open pom.xml of **01-sharedflows-app** and add the below tag under configuration tag inside mule-maven-plugin

<classifier>mule-plugin</classifier>



Right click on sharedflowsapp . Select Run as “ Mule Application(Configure)”

Under Maven command line arguments, give install as shown below: 

Once application starts, go to .m2 folder under you user directory (c:\\Users\\{your username} in windows).

See that

.m2/repository/com.way2learnonline/sharedflowsapp/1.0.0.SNAPSHOT/sharedflo wsapp-1.0.0-SNAPSHOT-mule-plugin.jar is created after mvn install.

Now open pom.xml of **01-classloading-start** application and add a dependency as shown below:

<dependency>

<groupId>com.way2learnonline</groupId>

<artifactId>sharedflowsapp</artifactId>

<version>1.0.0-SNAPSHOT</version>

<classifier>mule-plugin</classifier>

</dependency>

Open Classloading.xml and observe the flow.

Click on Global elements at the bottom, click on Create and select import under Global Configuration as shown below





Give the file name as sharedflowsapp.xml in the import window.

Drag a flow reference at the end of the flow. You should be able to select “mysharedflow”

Run the Classloading application, give request to the flow and observe that log from sharedflow is seen in logs

**Congratulations!! You understood how to invoke shared flows in another mule application**

**STEP2**

Open Classloading.xml and observe that we are using a java module to invoke a static method in a class.

In our app, we have a java class com.way2learnonloine.HelloWorld inside classpath.

Our app is an artifact and the java module is an artifact to mule runtime. How are classes defined in my app visible to the java module?

As we understood, every artifact has to export the packages which it wants to be visible to other artifacts.

In 08-classloading project open target/META-INF/mule-artifact/mule-artifact.json and observe the exportedPackages and exportedResources

So, as my app is exporting the packages, they are visible to the java module.

**Congratulations!! You understood how classes and resources of one module are visible in other modules**