

MuleSoft.U Mule 4 for Mule 3 Users Setup Instructions

Note: If you need help with the setup instructions, use the MuleSoft Training forum at <http://training.mulesoft.com/forums>.

Make sure your computer meets the minimum requirements

To complete the exercises in this course, you need:

1. A computer with the following:

- At least 8-16 GB RAM (16 highly recommended), 2GHz CPU, and 10GB available storage
- A minimum screen resolution of 1024x768

These are the requirements to run Anypoint Studio and Anypoint Platform.

<https://docs.mulesoft.com/release-notes/anypoint-studio-7.1-with-4.1-runtime-release-notes>

<https://docs.mulesoft.com/mule-user-guide/v/4.1/hardware-and-software-requirements>

Note: If your computer does not meet these requirements, you need to get access to or borrow one that does.

2. Internet access to the following (with > 5Mbps download and > 2Mbps upload)

Port	Host
80	mulesoft.com (and subdomains)
80	cloudhub.io (and subdomains)
3306	learn.mulesoft.com

Note: If you have restricted internet access and cannot access port 3306, you can instead use a MuleSoft training services Java application that is included in the student files for the course.

Create required accounts

You will use the following accounts during this course:

1. An Anypoint Platform account

<http://anypoint.mulesoft.com>

Note: You can sign up for a free, 30-day trial account or you can use your company account (if you already have one).

Install and test required software

To complete the exercises in this course, install and test the required software:

- 1. Check and make sure you have the latest version of Chrome.**

<https://www.google.com/chrome/browser/>

This is required to use Anypoint Platform.

<https://docs.mulesoft.com/release-notes/anypoint-studio-7.1-with-4.1-runtime-release-notes>

- 2. Install Advanced REST Client (or some other REST client application).**

<http://install.advancedrestclient.com>

- 3. Check your JDK version.**

https://www.java.com/en/download/help/version_manual.xml

For example, in a command-line interface, type:

```
java -version
```

Make sure you have JDK 1.8 and NOT JDK 1.7 or 1.10. If you do not have JDK 1.8, complete the next step to install it.

- 4. Install JDK 1.8 (NOT JDK 1.7 or JDK 1.10)**

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

JDK 1.8 is required for Anypoint Studio. You cannot use JDK 1.7 or JDK 1.10.

Note: Be sure to install the 64-bit version if you are running a 64-bit operating system or the 32-bit version if you are running a 32-bit operating system.

After installation, confirm with `java -version` in a command-line interface.

Install and test Anypoint Studio 7

Download, install, test, and troubleshoot the installation BEFORE class.

- 1. Download Anypoint Studio 7 with embedded Mule 4.1 runtime.**

<https://www.mulesoft.com/lp/dl/studio>

Note: If you have any issues, make sure the JDK and Anypoint Studio are BOTH 64-bit or BOTH 32-bit and that you have enough memory (at least 8-16 GB) to run Anypoint Studio.

- 2. Start Anypoint Studio and make sure you do not get any errors.**

Some of the more common errors and their solutions are included below.

Java was started but returned exit code=13

This signifies that the Java version Anypoint Studio is referencing is not compatible with the operating system (32-bit vs 64-bit). Make sure to install the same bit version that matches your operating system.

JDK path not found (MAC OS and Linux)

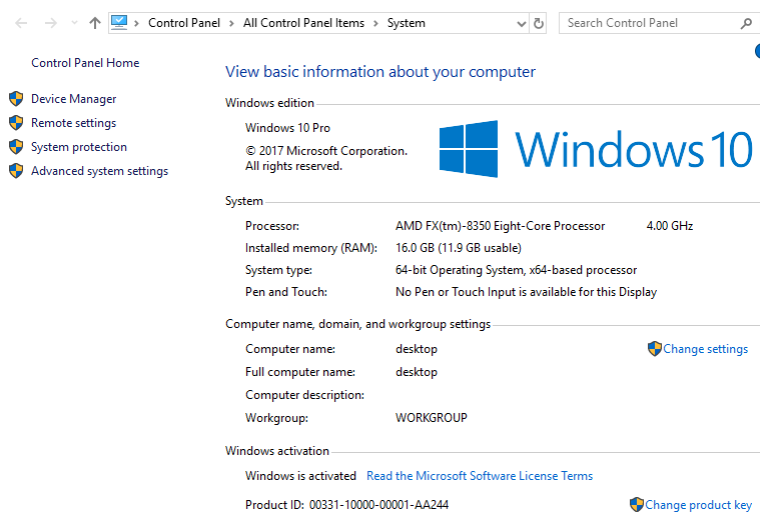
This error occurs because Anypoint Studio 7 cannot find the JDK 1.8 folder in your system environment. To configure your path, follow the instructions here:

<https://www.java.com/en/download/help/path.xml>

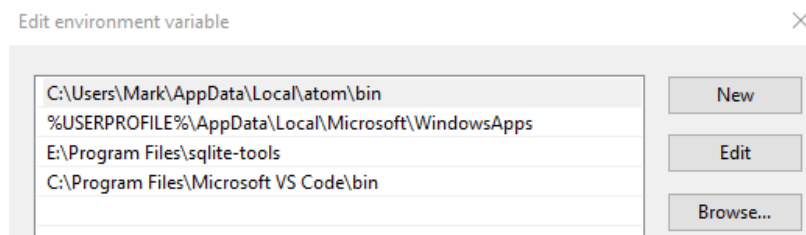
JDK path not found (Windows)

This error occurs because Anypoint Studio 7 cannot find the JDK 1.8 folder in your Windows environment. To fix, complete the following steps.

1. Clear old paths and remove any paths set by Oracle JDK in the User Environment Variables and System Environment Variables.
 - i. Search for Control Panel.
 - ii. Click the Advanced systems settings link.

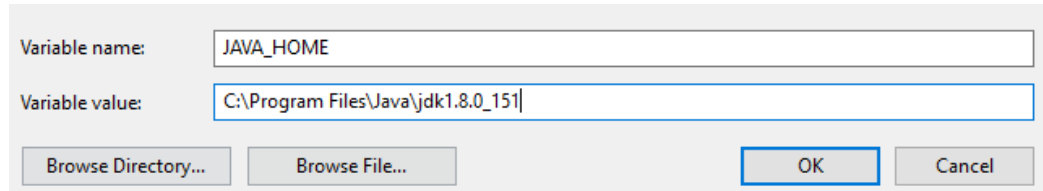


- iii. Click Environment Variables at the bottom of the dialog box.
2. Clear any Java related User Environment Variables.
 - i. In the User variables section, double-click the Path variable.
 - ii. Remove any instances of any Java path set by Oracle or any other tool you may have.

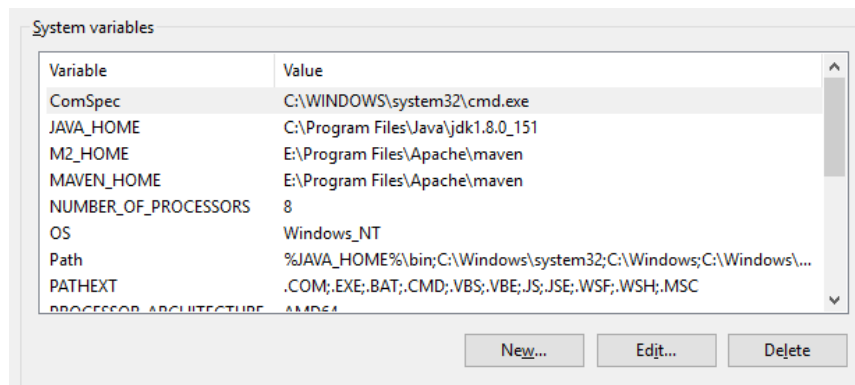


- iii. Close the dialog box.
3. Clear any Java related System Environment Variables.
 - i. In the System variables section, double-click the Path variable.
 - ii. Remove any instances of any Java path set by Oracle or any other tool you may have.
 - iii. Close the dialog box.

4. Add or edit the JAVA_HOME System Environment Variable to point to the JDK 1.8 directory.
 - i. If the JAVA_HOME System Environment Variable is not set, click New at the bottom of the dialog box.
 - ii. Set the Variable name to: JAVA_HOME.
 - iii. Set the Variable value to the location of your JDK 1.8 installation.
 - iv. If the JAVA_HOME System Environment Variable exists already, verify that the Variable value refers to your JDK 1.8 installation directory.

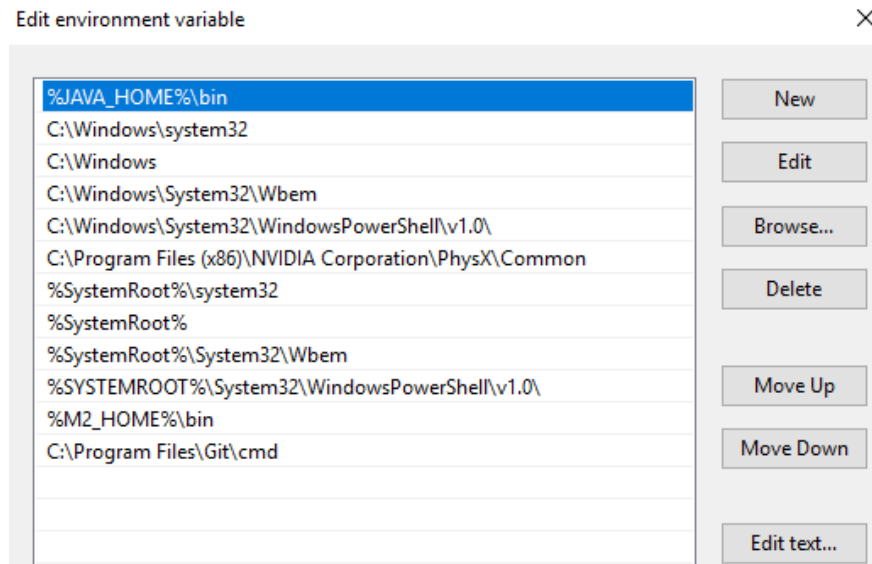


5. Add the JAVA_HOME in front of the Path for System Environment Variables.
 - i. In the System Environment Variables, double-click the Path variable.



Variable	Value
ComSpec	C:\WINDOWS\system32\cmd.exe
JAVA_HOME	C:\Program Files\Java\jdk1.8.0_151
M2_HOME	E:\Program Files\Apache\maven
MAVEN_HOME	E:\Program Files\Apache\maven
NUMBER_OF_PROCESSORS	8
OS	Windows_NT
Path	%JAVA_HOME%\bin;C:\Windows\system32;C:\Windows;C:\Windows\...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE	AMD64

- ii. Click New.
- iii. Enter: %JAVA_HOME%\bin
- iv. Move this path to the very top of the Path list.



Edit environment variable ✕

%JAVA_HOME%\bin	New
C:\Windows\system32	Edit
C:\Windows	Browse...
C:\Windows\System32\Wbem	Delete
C:\Windows\System32\WindowsPowerShell\v1.0\	Move Up
C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common	Move Down
%SystemRoot%\system32	Edit text...
%SystemRoot%	
%SystemRoot%\System32\Wbem	
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\	
%M2_HOME%\bin	
C:\Program Files\Git\cmd	

- v. If there are any other Java paths referred in the Path list, remove them if you do not need them for your development.
- vi. Verify your System variables have the JAVA_HOME and the Path variables set using the JAVA_HOME variable.

3. Ensure your Anypoint Studio workspace path has no spaces.

The Anypoint Studio tooling service will not work if there is a space in the path to the workspace. By default, the workspace that is created and used by Anypoint Studio is:

`/Users/Username/AnypointStudio/workspace`

If your username is Max Mule, the default workspace will be the following:

`/Users/Max Mule/AnypointStudio/workspace`

You need to make sure your workspace path has no spaces, by either making sure your username has none, or by setting a different workspace that does not include the username.

4. Download a test project and uncompress the ZIP file.

https://training.mulesoft.com/static/public_downloadables/setup/testApp.jar.zip

This is a JAR file compressed in a ZIP file. Mule 4 projects are now exported as JAR files in Anypoint Studio 7.

5. Import the project into Anypoint Studio.

6. Wait for Anypoint Studio 7 to download all Maven dependencies.

This can take a few minutes based on your internet connection.

Note: You will get errors if your system does not have the proper access to Maven servers over the internet.

7. Run the project.

8. In a web client, make a request to <http://localhost:8081/jms>.

Verify that the message: "Accounts sent for processing to JMS queue." is returned.

9. In a web client, make a request to <http://localhost:8081/pop>.

Verify the endpoint returns accounts from the remote MySQL database setup in the project.