# SAP Workshop 4.x - HANA Database

Author: [dejim.juang@mulesoft.com](mailto:dejim.juang@mulesoft.com)

Date: 9/3/2018

Version 1.0.0

## Overview

This guide shows how to connect to a SAP HANA Database using the MuleSoft Anypoint Studio7.x and the Database Connector. This assumes you have access to a SAP HANA database for this guide. There is an instance that is referenced in this guide but connection information may have changed. Please refer to your instructor for more details.

You will create a flow that uses the Database Connector and the JDBC Driver. Additionally you will learn how to pass in URL parameters to the SQL Query to select specific records from a table.

## Requirements

* [Anypoint Studio 7.x](https://www.mulesoft.com/lp/dl/studio)
* [SAP HANA JDBC Driver](https://dejim.s3.amazonaws.com/software/ngdbc.jar)
* SAP HANA Database

## Table of Contents

[Overview](#_ekijsl4qsgjb)

[Requirements](#_6f8yv0na6rzp)

[1. Connect and select all from a table](#_w7awktw89d80)

[2. Pass a parameter to the SQL Query](#_d2qtigesvmaq)

[Additional Resources](#_4khm4nc9g4tf)

## 

## 

## 1. Connect and select all from a table

|  |  |  |
| --- | --- | --- |
| 1.1 | Open Studio and create a new **Mule Project** |  |
| 1.2 | In the **New Mule Project** window, give the project a name (e.g. workshop-database), select a **Runtime**, and then click on **Finish** |  |
| 1.3 | Once the new project is created, you’ll be presented with a blank canvas.  In the **Mule Palette** on the right, click on **HTTP** and then drag and drop the **Listener** into the canvas |  |
| 1.4 | If the **Mule Properties** tab doesn’t open, click the **Listener** icon and click on the green plus sign to create a new **Connector configuration.** |  |
| 1.5 | Under the **General** tab, and in the **Connection** section, fill in the host and the port with the following:  Host: 0.0.0.0  Port: 8082  And then click on **OK** |  |
| 1.6 | Back in the **Listener** Mule properties tab, fill in the **Path** field with the following:  /records |  |
| 1,7 | Back in the **Mule Palette**, we need to add the **Database** **Connector**.  Click on **Add Modules** |  |
| 1.8 | Click and drag the **Database** component to the area that says **Drag and drop here to add to project** |  |
| 1.9 | Once the **Database** component is added to the palette, you can select and add any operation to the canvas.  Click on **Select** and drag that into the canvas and place it after the **Listener** component. |  |
| 1.10 | If the **Mule Properties** tab doesn’t open, click the **Select** icon and click on the green plus sign to create a new **Connector configuration.** |  |
| 1.11 | In the **Database Config** window, under the **General** tab, click on the **Connection** drop-down and select **Generic Connection** |  |
| 1.12 | Next, click on **Configure** for the **JDBC Driver** field to add the driver for SAP HANA. |  |
| 1.13 | For this guide, we’ll add a local file for the JDBC driver but Studio also allows you to add a Maven dependency if needed.  Click on **Use local file** |  |
| 1.14 | In the **Choose local file** window, browse to and select the \*.jar file.  If you don’t have the JDBC driver, you can download a copy from [here](https://dejim.s3.amazonaws.com/software/ngdbc.jar).  In the **Advanced settings** section, fill in the following fields:  Group ID: com.sap.db.jdbc  Version: 2.3.48 |  |
| 1.15 | Back in the **Database Config** window, use the following for the **Connection** configuration:  **URL**: jdbc:sap://<host>:39015/?currentschema=<schema>&user=<user>&password=<password>  **Driver Class Name**: com.sap.db.jdbc.Driver  Here are configuration properties for a sandbox instance of SAP HANA if needed:  Host: 34.201.35.10  Schema: SAP\_REST\_API  User: SYSTEM  Password: Mule1379  Click on **Test Connection** to make sure the credentials are correct. If it returns back a successful test, click on **OK** to continue. |  |
| 1.16 | Back in the **Select** properties tab, (1) enter in a query for a table in your SAP HANA database. You can use the following if you’re using the sandbox instance.  (2) Click on **Refresh Metadata** to use DataSenseon the table.  (3) Once the metadata is refreshed, click on the **Output** tab and expand the Mule Message and Payload to see the metadata for the table. |  |
| 1.17 | In the **Mule Palette**, find and select the **Tranform Message** component and drag and drop it into the canvas after the **Select** component. |  |
| 1.18 | In the **Tranform Message** properties tab, modify the DataWeave script to look like the following:  %dw 2.0  output application/json  ---  payload |  |
| 1.19 | On the canvas, right mouse click the canvas and select **Run project workshop-database**  You can also click on the green circle in the navigation bar and select **Run As > Mule Application** |  |
| 1.20 | Switch to Google Chrome and open the following URL:  http://localhost:8081/records |  |

## 2. Pass a parameter to the SQL Query

|  |  |  |
| --- | --- | --- |
| 2.1 | Select the **Select** component to open the Mule Properties window. Modify the **SQL Query Text** field and add a ‘WHERE’ clause  where ID = :record\_id  In the **Input Parameters** field, enter in the following:  #[{'record\_id' : attributes.queryParams.id}] |  |
| 2.2 | Save and re-run the project. If the project was already running, once you save the project, it will automatically redploy the changes in Studio. |  |
| 2.3 | Switch back to Google Chrome and open the following URL:  http://localhost:8082/records?id=2 |  |

## 

## Additional Resources

* MuleSoft Database Connector
  + <https://docs.mulesoft.com/connectors/db-connector-index>
* SAP HANA Tools
  + <https://tools.hana.ondemand.com/#hanatools>
* How to create an SAP HANA Developer Edition in the Cloud
  + <https://www.sap.com/developer/tutorials/hana-setup-cloud.html>
* Getting started with SAP HANA Express
  + <https://caldocs.hana.ondemand.com/caldocs/help/Getting_started_HANA_Express.pdf>
* Connect to SAP HANA using JDBC
  + <https://www.sap.com/developer/tutorials/hxe-connect-hxe-using-jdbc.html>