

EXERCISE 03 - BUILD A HELLOWORLD EXTENSION ON CLOUD FOUNDRY USING SAP CLOUD SDK ARCHETYPE

SAP Partner Workshop



30 min

Description

In this exercise, you'll learn how

- to create an SAP S/4HANA SDK project for Cloud Foundry from an archetype using the IntelliJ Idea IDE
- to build the project with IntelliJ
- to deploy the application to your Cloud Foundry environment

For further reading on SAP Cloud SDK, click link below.

<https://www.sap.com/germany/developer/topics/s4hana-cloud-sdk.html>

Target group

- Developers
- People interested in learning about S/4HANA extension and SAP Cloud SDK

Goal

The goal of this exercise is to create a Maven project for Cloud Foundry using an archetype, build the project and deploy to SAP Cloud Platform.

Prerequisites

Here below are prerequisites for this exercise.

- A trial account on the SAP Cloud Platform. You can get one by registering here <https://account.hanatrial.ondemand.com>
- Cloud Foundry CLI Tool
- Apache Maven
- Java JDK 8
- IntelliJ IDEA

Refer to Exercise 2 to get step-by-step guide on system setup and pre-requisites.

Step 1 - Generate Project from Archetype

To generate your first project from the Maven archetype, run the following command:

```
mvn archetype:generate "-DarchetypeGroupId=com.sap.cloud.s4hana.archetypes" "-DarchetypeArtifactId=scp-cf-tomee" "-DarchetypeVersion=2.18.1" "-DgroupId=com.sap.cloud.sdk.tutorial" "-DartifactId=address-manager" "-Dversion=1.0-SNAPSHOT"
```

During the generation process, Maven require following parameters to form your project:

groupId	An identifier representing your group, company or organization (e.g. <code>com.sap.cloud.sdk.tutorial</code>)
artifactId	An identifier for your application (e.g. <code>address-manager</code>)
version	The version of your application (e.g. <code>2.18.1</code>)
package	The name of the top-level package your source code will reside in (typically equal to your groupId , e.g. <code>com.sap.cloud.sdk.tutorial</code>). Please pay attention to package and directory names in any upcoming source code when using a different package name than suggested here.

Since we have already provided these values in our maven command, it may not prompt to provide the values again.

Once maven command is executed, it will generate your project from the archetype.

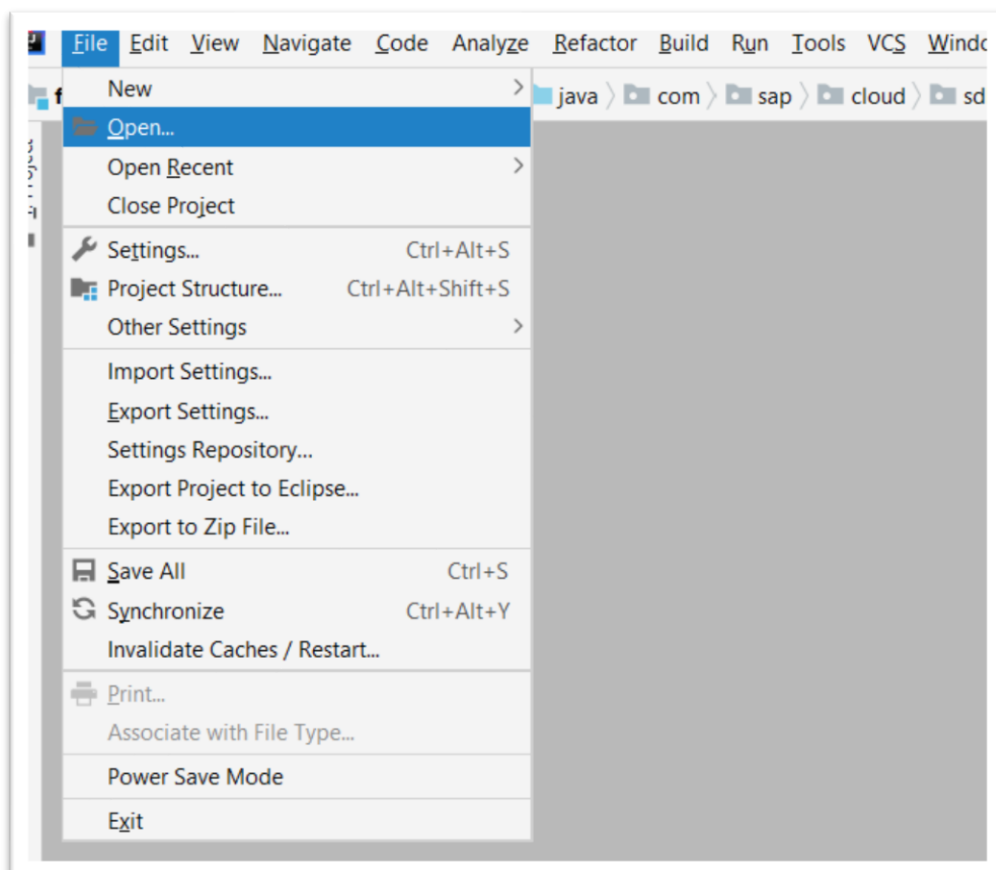
```

C:\Windows\system32\cmd.exe
groupId: com.sap.cloud.sdk.tutorial
artifactId: address-manager
version: 1.0-SNAPSHOT
package: com.sap.cloud.sdk.tutorial
gitignore: .gitignore
skipUsageAnalytics: false
Y: : y
[INFO] -----
[INFO] Using following parameters for creating project from Archetype: scp-cf-tomee:2.18.1
[INFO] -----
[INFO] Parameter: groupId, Value: com.sap.cloud.sdk.tutorial
[INFO] Parameter: artifactId, Value: address-manager
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] Parameter: package, Value: com.sap.cloud.sdk.tutorial
[INFO] Parameter: packageInPathFormat, Value: com/sap/cloud/sdk/tutorial
[INFO] Parameter: package, Value: com.sap.cloud.sdk.tutorial
[INFO] Parameter: skipUsageAnalytics, Value: false
[INFO] Parameter: gitignore, Value: .gitignore
[INFO] Parameter: groupId, Value: com.sap.cloud.sdk.tutorial
[INFO] Parameter: artifactId, Value: address-manager
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] Project created from Archetype in dir: C:\s4hana_extension\address-manager
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 15.304 s
[INFO] Finished at: 2019-08-26T17:21:46+05:30
[INFO] -----

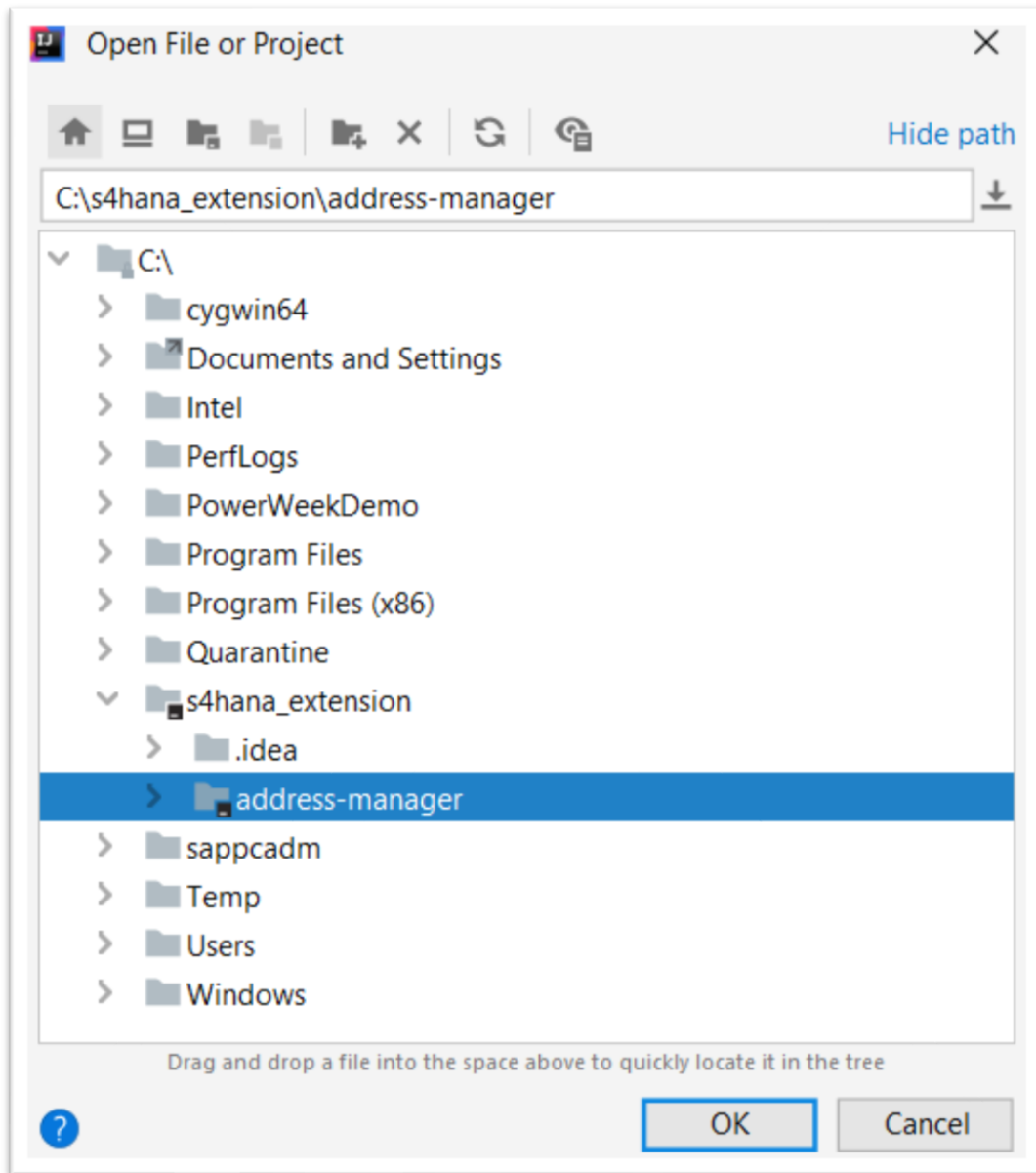
```

Step 2 – Build the project using IntelliJ IDEA

Open IntelliJ IDEA. Go to File → Open

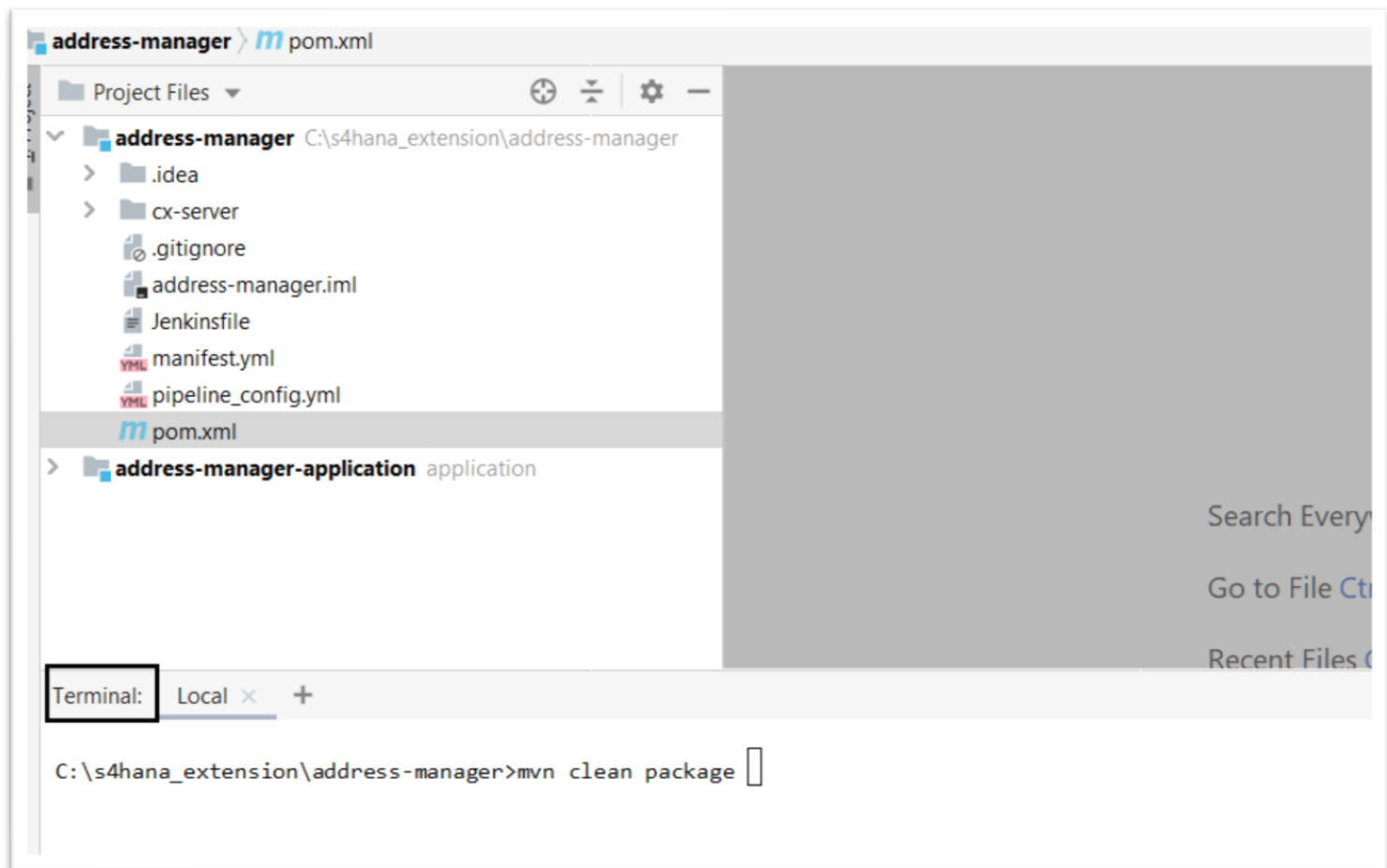


Select the folder which contains generated project and open it.



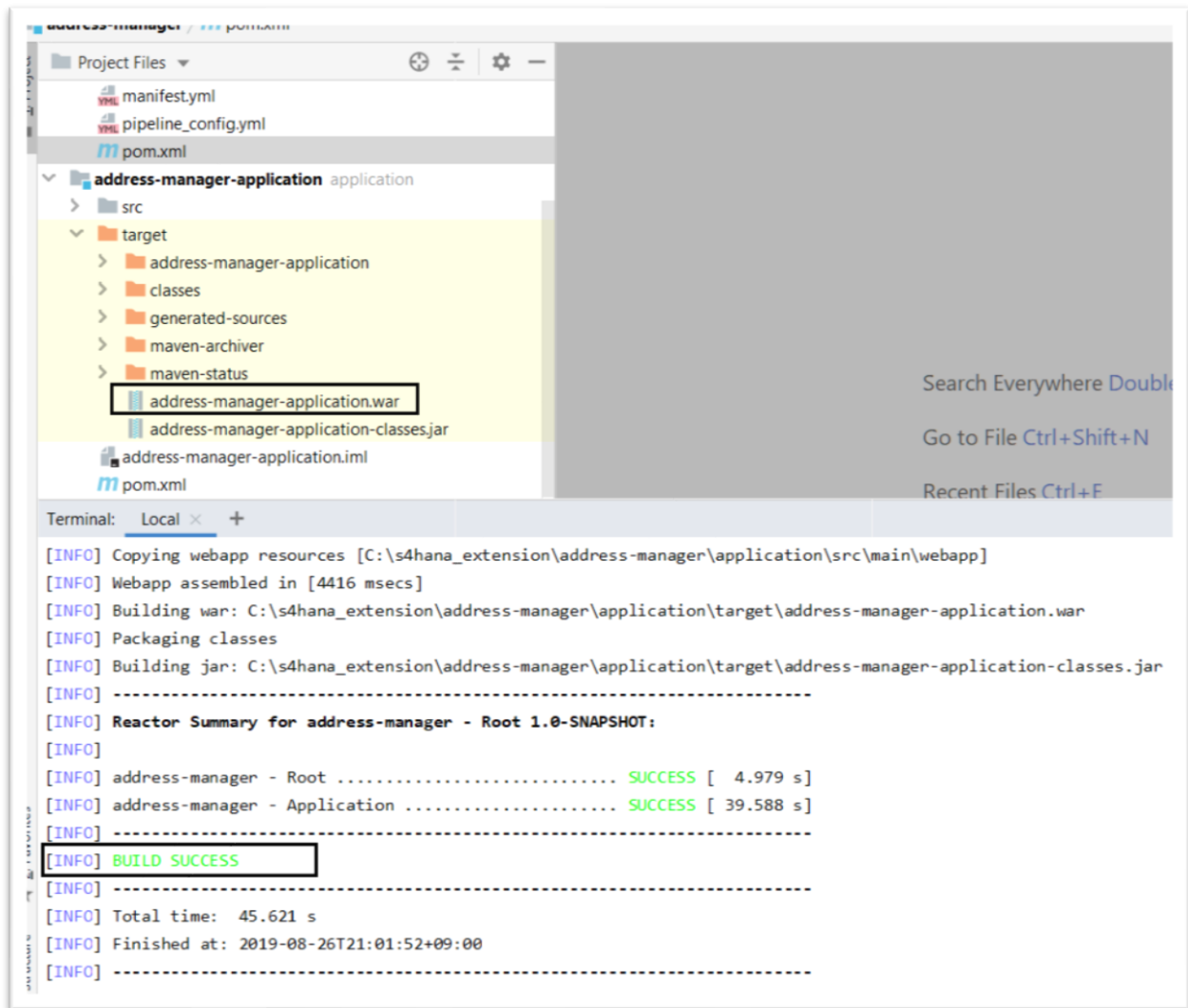
Open the terminal and run below command to build the project.

```
mvn clean package
```



This tells maven to **remove any files from previous assemblies (clean)** and to **assemble our project into a deployable artifact – a .war file (package)**.

After running the command there should be a directory target inside of the application directory, containing a file called address-manager-application.war. This is the file that we will deploy to Cloud Foundry (or locally).



Step 3: Deploy to Cloud Foundry

In order to deploy applications on SAP Cloud Foundry we need to provide cf command with an API endpoint. The API endpoint depends on the region you chose for your account.

To get the API end point, refer to the SAP Cloud Cockpit.

For example, in case of EU region, it is - <https://api.cf.eu10.hana.ondemand.com>

Enter the following command.

```
cf api https://api.cf.eu10.hana.ondemand.com
cf login
```

Enter your user id and password for SAP Cloud Platform account.

Now, enter below command to deploy your application to SAP Cloud Platform Cloud Foundry environment.

cf push

After the deployment is finished, output should look like this:

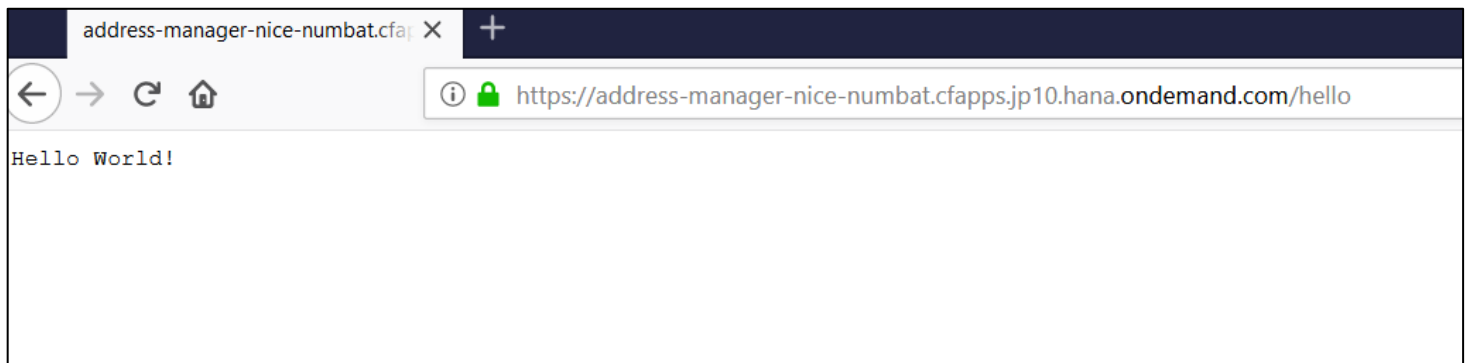
```
Terminal: Local x +

name:          address-manager
requested state: started
routes:        address-manager-nice-numbat.cfapps.jp10.hana.ondemand.com
last uploaded: Mon 26 Aug 21:33:50 KST 2019
stack:         cflinuxfs3
buildpacks:    sap_java_buildpack

type:          web
instances:     1/1
memory usage:  1024M
start command: JRE_HOME="META-INF/.sap_java_buildpack/sapjvm" JBP_CONFIG_SAPJVM_MEMORY_?
               JAVA_HOME="META-INF/.sap_java_buildpack/sapjvm" CATALINA_HOME="META-INF/
               -Djava.io.tmpdir=$TMPDIR -Dhttp.port=$PORT -Daccess.logging.enabled=false
               -DSAPJVM_EXTENSION_COMMAND_HANDLER=com.sap.xs2rt.dropletaddon.JvmExtensio
               -agentpath:/app/META-INF/.sap_java_buildpack/jvm_kill/jvmkill-1.12.0.RELI
               ./META-INF/.sap_java_buildpack/tomee7/bin/catalina.sh run

state  since          cpu  memory          disk          details
#0    running        2019-08-26T12:34:32Z  90.1%  367.5M of 1G  193.9M of 1G
```

Now we can visit the application under its corresponding URL as it is shown in the output above. Take the value from “routes: ...” and append the “/hello” path.



Congratulation! You have successfully created an SAP S/4HANA SDK project for Cloud Foundry from an archetype, built it and deploy to SAP Cloud Platform.