EXERCISE 05 - BUILD A BASIC EXTENSION APP ON CLOUD FOUNDRY

SAP Partner Workshop

30 min

Description

In this exercise, you'll learn how

- to build a Basic Extension App on Cloud Foundry
- to connect to OData Service using mock server

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 build a basic extension app on Cloud Foundry using Business Partner Example.

Prerequisites

Here below are prerequisites for this exercise.

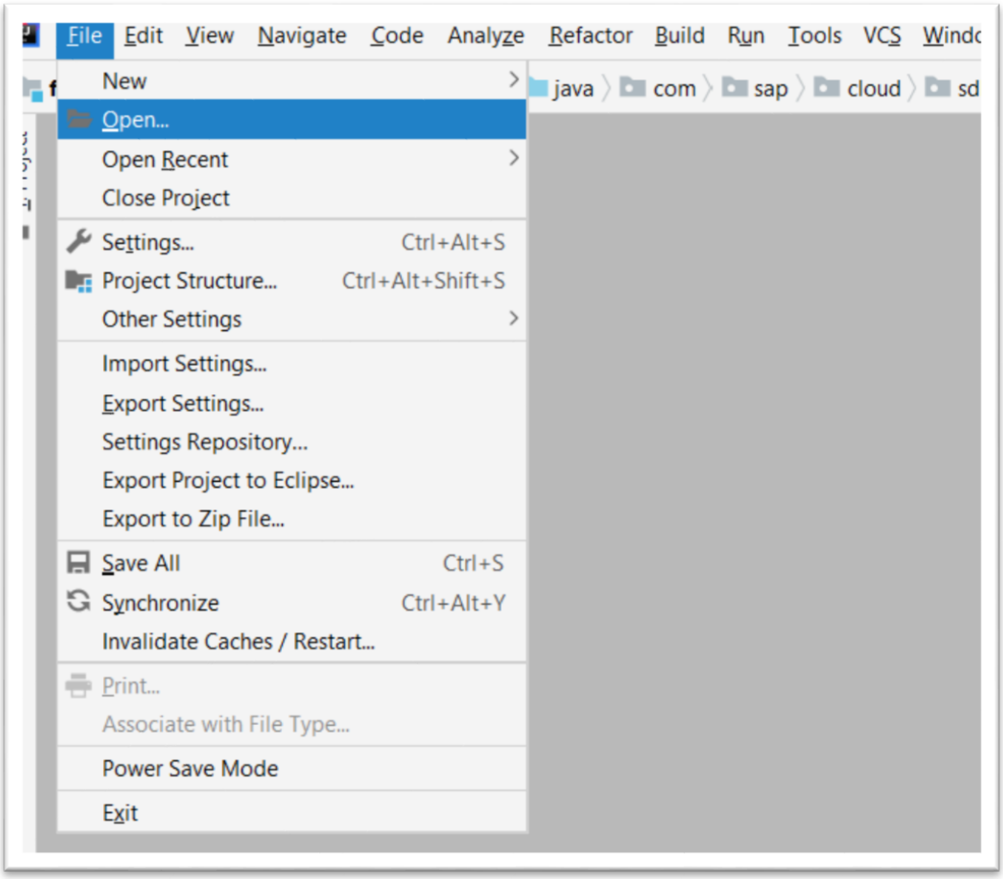
- A trial account on the SAP Cloud Platform. You can get one by registering here https://account.hanatrial.ondemand.com
- Cloud Foundry CLI Tool
- Apache Maven
- Java JDK 8
- IntelliJ IDEA

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

Step 1 – Download and build the project

Download the Exercise_5_Starting.zip file from here and extract it locally.

Similar to the previous exercise, open the project in IntelliJ Web IDEA.



Step 2 – Change GetBusinessPartner Commands

Open the Servlet file BusinessPartnerServlet and check doGet() method.

Open the below command files and check the execute method.

- GetAllBusinessPartnersCommand
- `GetSingleBusinessPartnerByIdCommand`

In both the command files, in execute method, comment below line:

```
return null;
```

And uncomment remaining sections. Finally execute method in GetAllBusinessPartnersCommand file should be as below.

```
public List<BusinessPartner> execute() throws Exception {
        //TODO:
       //return null;

       final List<BusinessPartner> businessPartners = service
                .getAllBusinessPartner()
                .select(BusinessPartner.BUSINESS_PARTNER,
                        BusinessPartner.LAST_NAME,
                        BusinessPartner.FIRST_NAME)
.filter(BusinessPartner.BUSINESS_PARTNER_CATEGORY.eq(CATEGORY_PERSON))
                .orderBy(BusinessPartner.LAST_NAME, Order.ASC)
                .execute();

        return businessPartners;
    }
```

Similarly make the changes in GetSingleBusinessPartnerByIdCommand file as below.

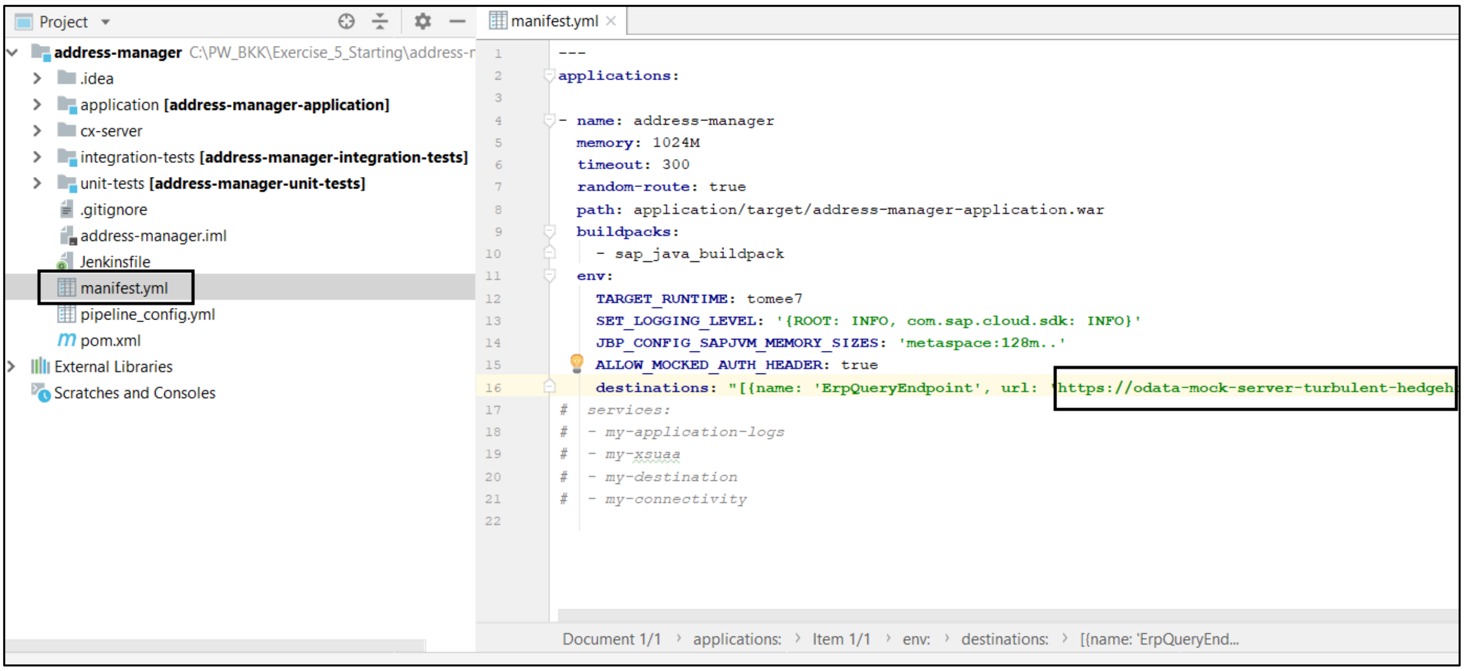
```
    public BusinessPartner execute() throws Exception {
        //TODO
        //return null;

        final BusinessPartner businessPartner = service
                .getBusinessPartnerByKey(id)
                .select(BusinessPartner.BUSINESS_PARTNER,
                        BusinessPartner.LAST_NAME,
                        BusinessPartner.FIRST_NAME,
                        BusinessPartner.IS_MALE,
                        BusinessPartner.IS_FEMALE,
                        BusinessPartner.CREATION_DATE,
                        BusinessPartner.TO_BUSINESS_PARTNER_ADDRESS.select(
                                BusinessPartnerAddress.BUSINESS_PARTNER,
                                BusinessPartnerAddress.ADDRESS_ID,
                                BusinessPartnerAddress.COUNTRY,
                                BusinessPartnerAddress.POSTAL_CODE,
                                        BusinessPartnerAddress.CITY_NAME,
                                        BusinessPartnerAddress.STREET_NAME,
                                        BusinessPartnerAddress.HOUSE_NUMBER))
                    .execute();
            return businessPartner;
        }
```

Build the project using below command

```
mvn clean package
```

Step 3 – Update Mock Server URL in manifest.yml file

Open the file manifest.yml and update destinations URL. Specify the your mock server URL created as part of exercise 4.



Project
address-manager C:\PW_BKK\Exercise_5_Starting\address-r
.idea
application [address-manager-application]
cx-server
integration-tests [address-manager-integration-tests]
unit-tests [address-manager-unit-tests]
.gitignore
address-manager.iml
Jenkinsfile
manifest.yml
pipeline_config.yml
pom.xml
External Libraries
Scratches and Consoles
manifest.yml
Document 1/1 › applications: › Item 1/1 › env: › destinations: › [{name: 'ErpQueryEnd...

```
---
applications:

- name: address-manager
  memory: 1024M
  timeout: 300
  random-route: true
  path: application/target/address-manager-application.war
  buildpacks:
    - sap_java_buildpack
  env:
    TARGET_RUNTIME: tomee7
    SET_LOGGING_LEVEL: '{ROOT: INFO, com.sap.cloud.sdk: INFO}'
    JBP_CONFIG_SAPJVM_MEMORY_SIZES: 'metaspace:128m..'
    ALLOW_MOCKED_AUTH_HEADER: true
    destinations: "[{name: 'ErpQueryEndpoint', url: 'https://odata-mock-server-turbulent-hedgeh
#  services:
#  - my-application-logs
#  - my-xsuaa
#  - my-destination
#  - my-connectivity
```

Step 4 - 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.

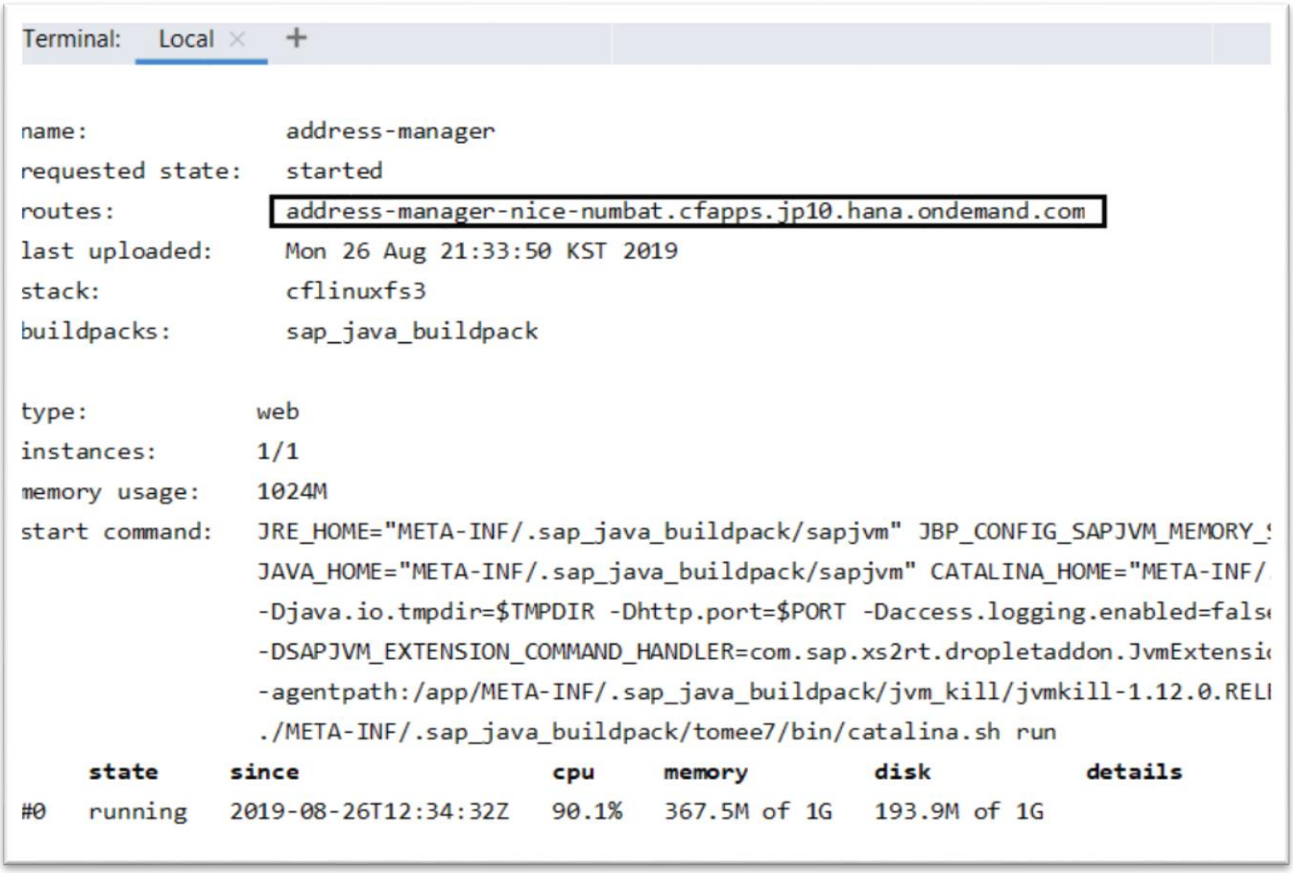
```
cf api https://api.cf.eu10.hana.ondemand.com
cf login
```

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.

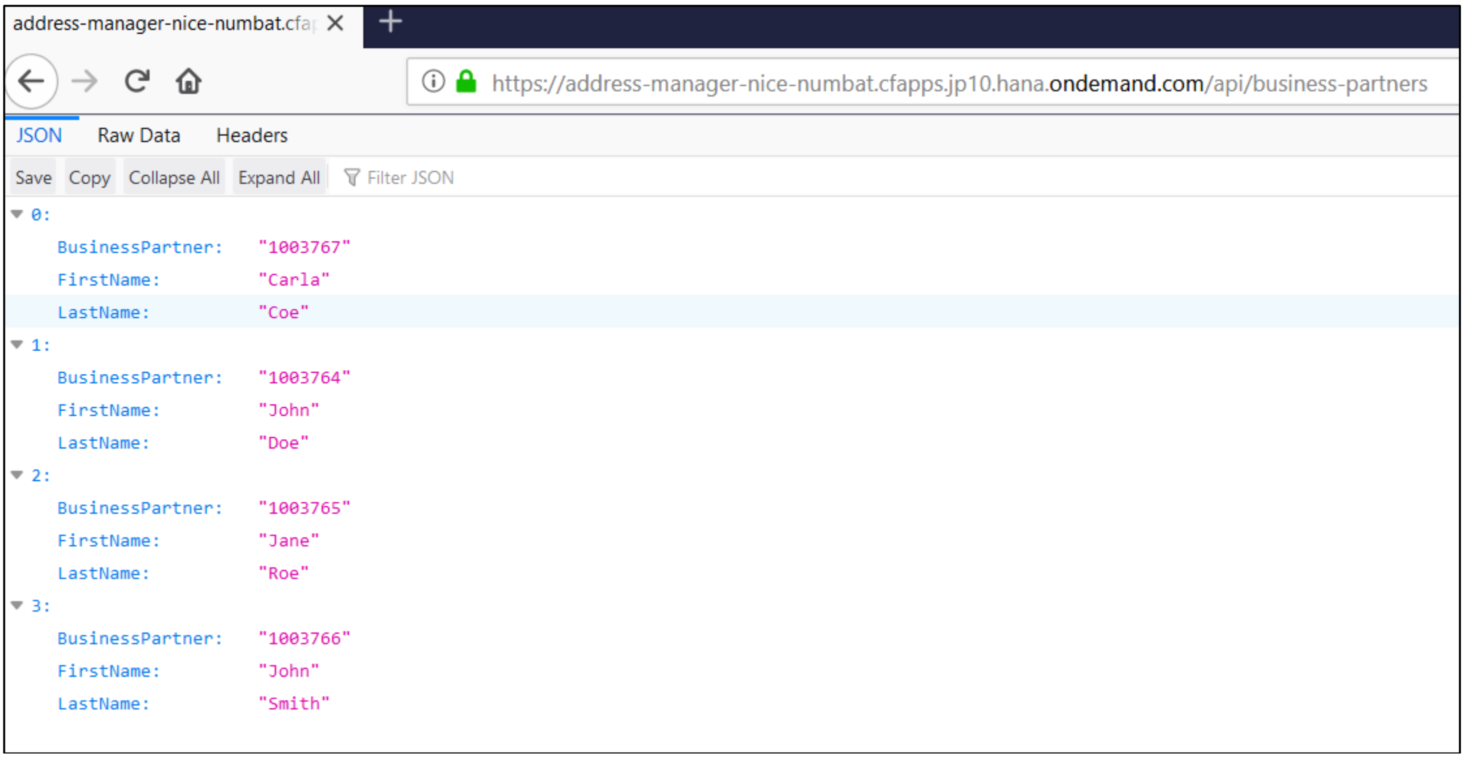
```
cf push
```

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



```
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=fals
                 -DSAPJVM_EXTENSION_COMMAND_HANDLER=com.sap.xs2rt.dropletaddon.JvmExtensi
                 -agentpath:/app/META-INF/.sap_java_buildpack/jvm_kill/jvmkill-1.12.0.REL
                 ./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
```

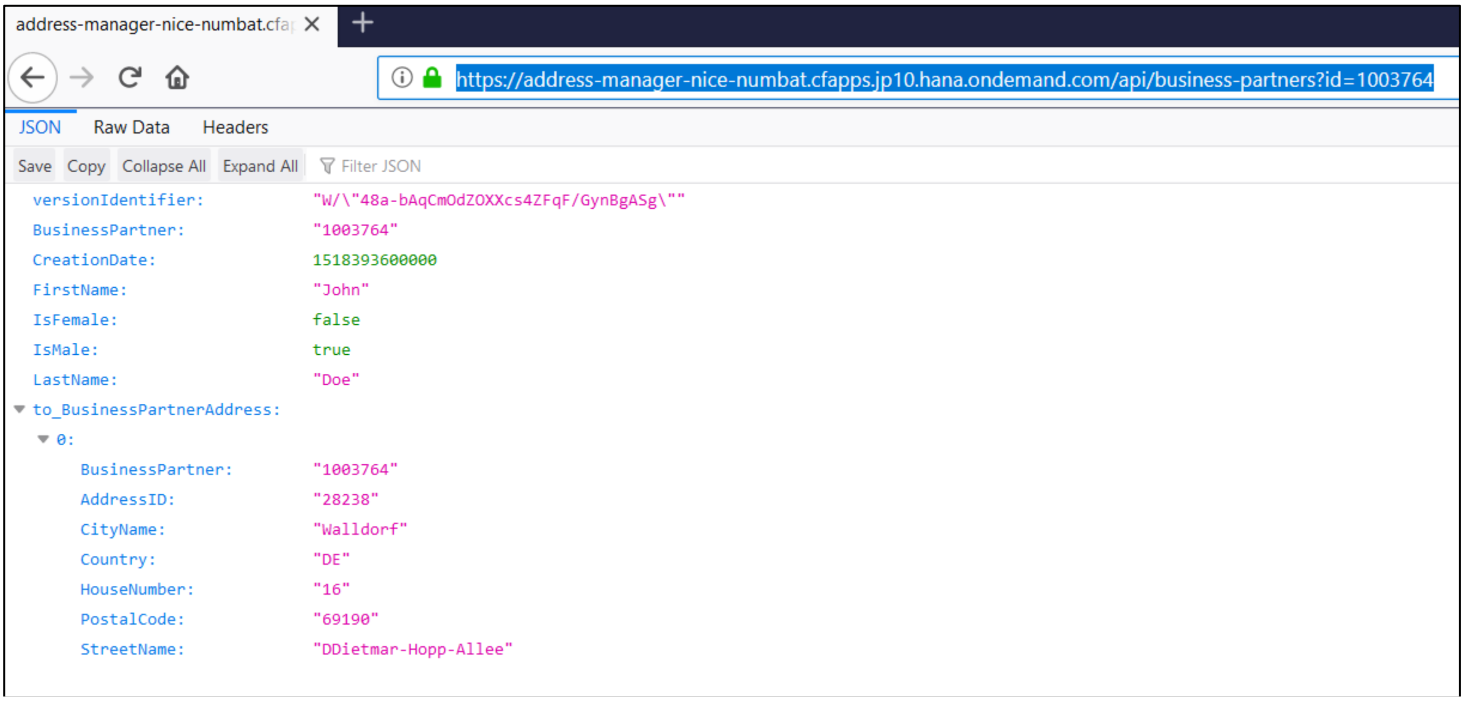
Now we can visit the application under its corresponding URL as it is shown in the output above. Take the value from *“routes: ...”* and append the ***“/api/business-partners”*** path.


address-manager-nice-numbat.cfa
https://address-manager-nice-numbat.cfapps.jp10.hana.ondemand.com/api/business-partners
JSON
Raw Data
Headers
Save
Copy
Collapse All
Expand All
Filter JSON

```
▼ 0:
    BusinessPartner:    "1003767"
    FirstName:          "Carla"
    LastName:           "Coe"
▼ 1:
    BusinessPartner:    "1003764"
    FirstName:          "John"
    LastName:           "Doe"
▼ 2:
    BusinessPartner:    "1003765"
    FirstName:          "Jane"
    LastName:           "Roe"
▼ 3:
    BusinessPartner:    "1003766"
    FirstName:          "John"
    LastName:           "Smith"
```

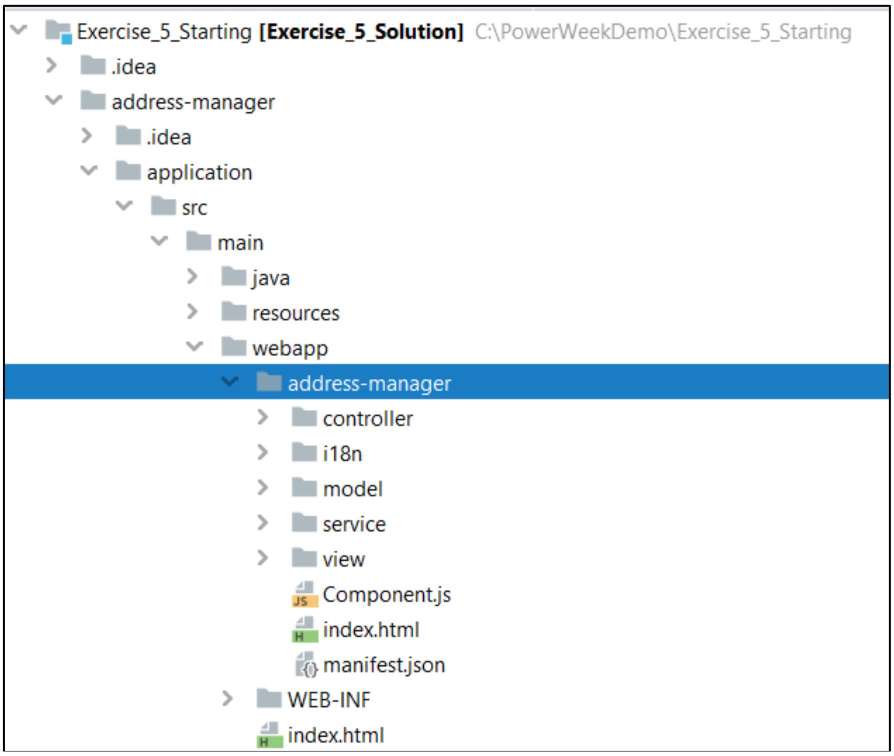
Copy any BusinessPartner id and provide it as query parameter. For example https://address-manager-nice-numbat.cfapps.jp10.hana.ondemand.com/api/business-partners?id=1003764

The output should be like below.


address-manager-nice-numbat.cfa
https://address-manager-nice-numbat.cfapps.jp10.hana.ondemand.com/api/business-partners?id=1003764
JSON
Raw Data
Headers
Save
Copy
Collapse All
Expand All
Filter JSON

```
  versionIdentifier:          "W/\"48a-bAqCmOdZOXXcs4ZFqF/GynBgASg\""
  BusinessPartner:            "1003764"
  CreationDate:               1518393600000
  FirstName:                  "John"
  IsFemale:                   false
  IsMale:                     true
  LastName:                   "Doe"
▼ to_BusinessPartnerAddress:
  ▼ 0:
      BusinessPartner:        "1003764"
      AddressID:              "28238"
      CityName:               "Walldorf"
      Country:                "DE"
      HouseNumber:            "16"
      PostalCode:             "69190"
      StreetName:             "DDietmar-Hopp-Allee"
```

We have also provided a SAPUI5 module in the sample project to showcase these data.

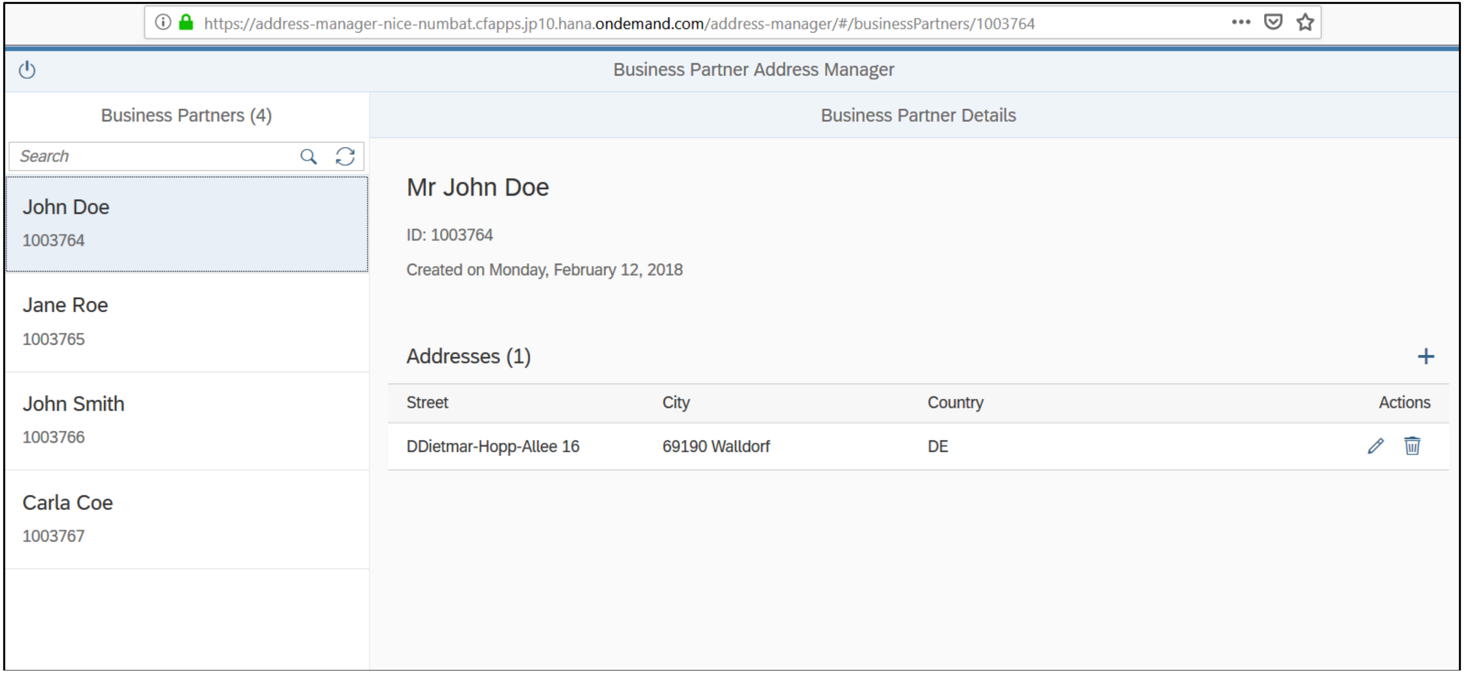


```
Exercise_5_Starting [Exercise_5_Solution] C:\PowerWeekDemo\Exercise_5_Starting
  .idea
  address-manager
    .idea
    application
      src
        main
          java
          resources
          webapp
            address-manager
              controller
              i18n
              model
              service
              view
              Component.js
              index.html
              manifest.json
            WEB-INF
            index.html
```

Append the **“/address-manager”** to base URL to open UI. For example.

https://address-manager-nice-numbat.cfapps.jp10.hana.ondemand.com/address-manager/

Explore the UI.



https://address-manager-nice-numbat.cfapps.jp10.hana.ondemand.com/address-manager/#/businessPartners/1003764

Business Partner Address Manager

Business Partners (4)

Search

John Doe
1003764

Jane Roe
1003765

John Smith
1003766

Carla Coe
1003767

Business Partner Details

Mr John Doe

ID: 1003764

Created on Monday, February 12, 2018

Addresses (1)

Street	City	Country	Actions
DDietmar-Hopp-Allee 16	69190 Walldorf	DE	

Step 5 – Explore other command files

Open the command files for Create, Update and Delete. Uncomment the code in execute method and observe the behavior in UI.

Congratulation! You have successfully build a Basic Extension App on Cloud Foundry and deploy to SAP Cloud Platform.