Introduction

The cloud trend in technology is growing quickly around the world and companies are increasingly boosting their use of cloud systems, so today we can do almost anything (if not everything) from the cloud. This manual focuses on SAP HANA and SAP Data Intelligence in the cloud and provides an approach to learn easily and fast how to deploy data models and machine learning scenarios using both SAP cloud technologies. A description of the book's chapters follows:

- ***Chapter 1. SAP HANA cloud overview:*** An overview of what SAP HANA is:
  - *Database services*
  - *Application development*
  - *Data virtualization*
  - *On-premise vs cloud*
  - *SAP business technology Platform (BTP)*

- ***Chapter 2. SAP HANA cloud initial setup:*** The initial setup of the SAP HANA cloud system:
  - *Database provisioning*
  - *Business application studio (BAS)*
  - *Building a database project*
  - *Git configuration*

- ***Chapter 3. SAP HANA cloud modeling concepts:*** Learn how to create a data model and convert it to a database design:
  - *Data model overview*
  - *Entity relationship model (ERM)*
  - *Normalization*
  - *HANA modeling concepts*
  - *Conversion of the ERM to a database design*

- ***Chapter 4. SAP HANA cloud deployment:*** Deploy the data model on SAP HANA cloud:
  - *Deploying the data model*
  - *Loading the tables*
  - *HDI infrastructure overview*

- ***Chapter 5. SAP HANA cloud development:*** Learn the basics of SAP HANA cloud development:
  - *Cloud foundry environment overview*
  - *SQLScript basics*
  - *Information views*

- ***Chapter 6. SAP Data Intelligence:*** Deploy a machine learning scenario using SAP Data Intelligence integrated with SAP HANA cloud:
  - *Launchpad*
  - *Connection management*
  - *Monitoring*
  - *Metadata Explorer*
  - *System Management*
  - *ML Scenario Manager*
  - *Modeler*
  - *HANA Cloud Connection*
  - *Data science using SAP DI*

5.4. Information Views

SAP HANA modeling uses the virtual data model (VDM) approach as a key concept of the SAP HANA platform. A virtual model does not store data itself but calculates the result on the fly.

Information views are used to create the reports on top on the SAP HANA database tables based on the business requirements and can be consumed from other tools like SAP Analytics Cloud, SAP Lumira, SAP Business Object or SAP Data Intelligence.

The tool used by SAP HANA Cloud to create information views is SAP Web IDE for SAP HANA as we know from previous chapters. Some of the benefits of the information views are as follows:

- Every information view is a reusable object that you can transport to other environments and be called from other information views.
- All the calculations are performed on the fly.

With SAP HANA Cloud we can build different types of information views:

- *Graphical calculation views:* As we see in the section 3.5.2 there are three types of graphical calculation views:
  - **Dimension views:** data master like customer or products.
  - **Star join views:** facts like sales order details.
  - **Cube views:** we can use several star join views together as actual versus budget.

- *Script-based views:* We use SQLScript language to create complex reports where graphical calculation views can not fit the business requirements:
  - **Table functions:** The table function replaces the old, scripted calculation views. We use SQLScript to create complex functions that use loops, rankings and so on and they are read-only, so we cannot use statements like INSERT, UPDATE or DELETE.

During the next business questions, we will address some of the most important topics of the modeling with SAP HANA calculation views. The way of explaining it will be based on hands-on exercises.

5.4.2. Business question #1

The business question #1 explains the following calculation view modeling concepts:

- Type of nodes
- Add to output
- Data preview
- Semantics
- Start join
- Join definition
- Calculated columns
- Raw and analysis view

Business question 1*: The sales department wants to know a list of customers for each order, their products and the quantity ordered and the total cost by product.*

Initial Analysis: What do we need to address this question in terms of SAP HANA modeling? Please check the sales diagram (as seen in chapter 3 figure 3.1) to know what tables we must include in the final report and what type of calculation views we need to model:

- We need *customers* and *products* tables as a data master (calculation view of type dimension)
- We need *orders* and *order details* as fact table or data foundation (calculation view of type star)

Before modeling the views, it is recommendable to think the naming convention like: *<project name>-cv-<type of calculation view>-<table>-<business question number>* to identify it easily and reuse later. So, and for the example above we will create the following artifacts:

- *Salesapp-cv-dim-customers-BQ1*
- *Salesapp-cv-dim-products-BQ1*
- *Salesapp-cv-star-orders-BQ1*

Then we will use the above example to explain some of the key concepts of SAP HANA graphical calculation views. From SAP Web IDE we will start the wizard to create the HANA Artifacts so from the workspaces called: *DB-HANACLOUD-BOOK* we will navigate until the folder *src* and then press *F1,* search by *HANA* and click on *SAP HANA Create HANA Database Artifact* (as seen in figure 5.13).

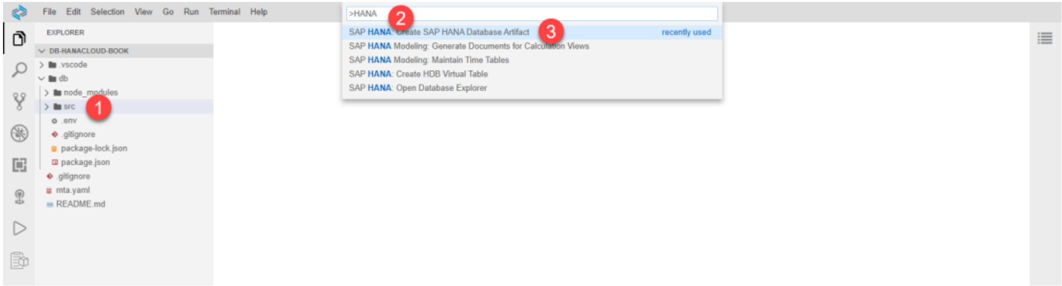
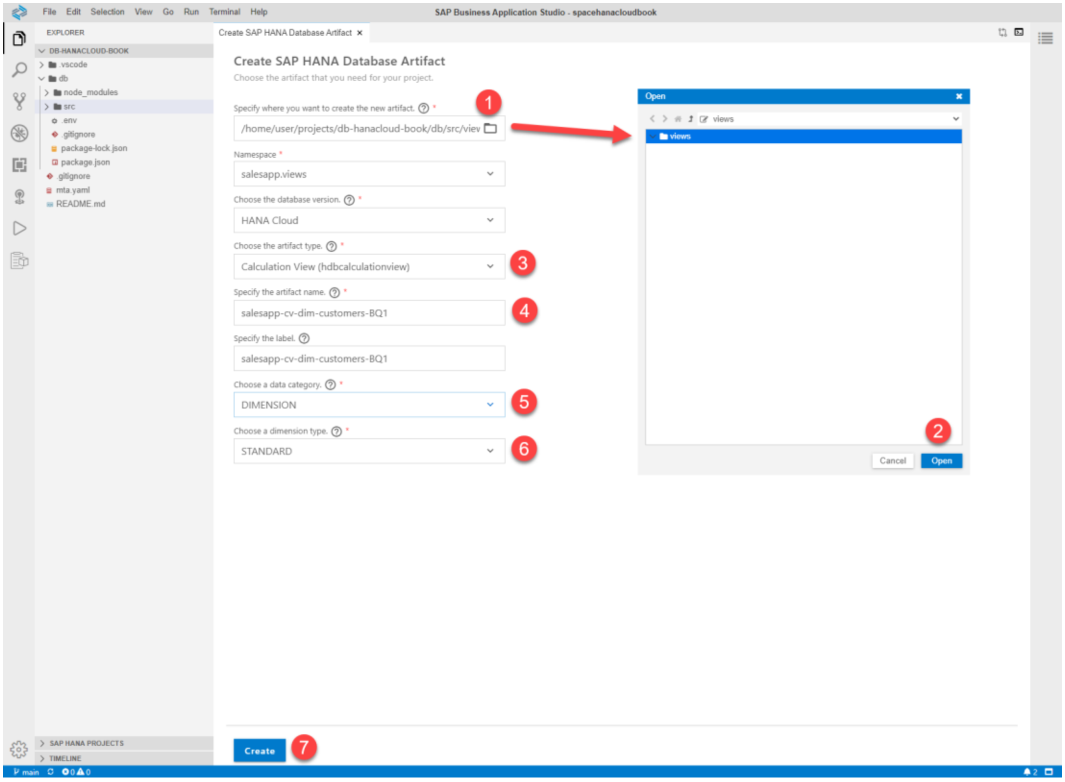

File
Edit
Selection
View
Go
Run
Terminal
Help
EXPLORER
DB-HANACLOUD-BOOK
.vscode
db
node_modules
src
1
.env
.gitignore
package-lock.json
package.json
.gitignore
mta.yaml
README.md
>HANA
2
SAP HANA: Create SAP HANA Database Artifact
3
recently used
SAP HANA Modeling: Generate Documents for Calculation Views
SAP HANA Modeling: Maintain Time Tables
SAP HANA: Create HDB Virtual Table
SAP HANA: Open Database Explorer

Figure 5.13 Open the wizard

(Copyright © 2022 SAP SE or an SAP affiliate company. All rights reserved)

5.4.2.1. Customers dimension

The next step will be creating the *Salesapp-cv-dim-customers-BQ1* calculation view and use the customers table (as seen in figure 5.14). In our example we only need one table but in some business scenarios where the data is not in the same table, we would need to JOIN several tables inside the same calculation view of type dimension.



Create SAP HANA Database Artifact

Choose the artifact that you need for your project.

Specify where you want to create the new artifact. *: /home/user/projects/db-hanacloud-book/db/src/viev

Namespace *: salesapp.views

Choose the database version. *: HANA Cloud

Choose the artifact type. *: Calculation View (hdbcalculationview)

Specify the artifact name. *: salesapp-cv-dim-customers-BQ1

Specify the label.: salesapp-cv-dim-customers-BQ1

Choose a data category. *: DIMENSION

Choose a dimension type. *: STANDARD

Figure 5.14 Creation of the calculation view for customers table

(Copyright © 2022 SAP SE or an SAP affiliate company. All rights reserved)

Before creating the calculation view, we need to setup the following options:

- Assign the folder where the calculation view will be stored (step 1 and step 2)
- Choose the type of artifact: calculation view – *hdbcalculationview* (step 3)
- Specify the artifact name based on the naming convention (step 4)
- Choose the data category *DIMENSION* as we will create dimension tables for data master (step 5)
- And then choose *STANDARD* as we are not going to create a TIME dimension (step 6)
- And press on *Create* button to finish (step 7)

Finally, and from the *views* folder we will see our HANA artifact already created (as seen in figure 5.15) and we will start with the definition itself.

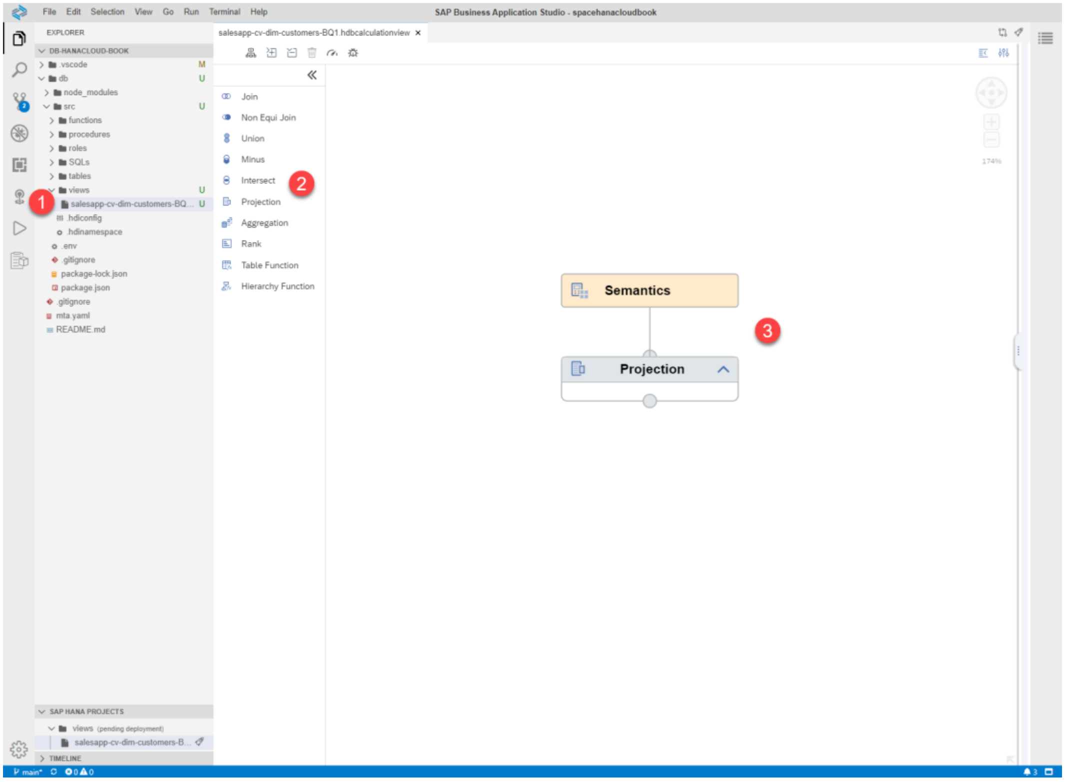


Figure 5.15 Calculation view for customers table

(Copyright © 2022 SAP SE or an SAP affiliate company. All rights reserved)

Before moving forward, we will explain the concept of nodes and the types of nodes that we can use (as seen in figure 5.15 step 2). A node is an element that we will use to build our calculation views, by default a calculation view of type dimension uses the *projection* and *semantics* nodes (as seen in figure 5.15 step 3) and the different types of nodes are as follows:

- **Join:** It is used to combine rows from more than one table using only the comparison operator equal (=) in the key field. The types of joins have been explained in the chapter 3 modeling concepts section 3.5.2.3.

- **Non Equi Join:** This type of join was introduced in SAP HANA 2.0 SPS 03 and allows us use comparison operators like non equal (!=) , equal (=), Greater than (>), less than (<), greater than equal to (>=) and less than equal to (<=) in the key field.

- **Union:** We can use the union operation to combine several result sets. The union expects that the result sets contain the same number of columns, in the same order and the matching columns should have similar data types. We have different types of unions:

- **Union all:** return all records
- **Union:** return only the unique values
- **Union with constant values:** we can use separate columns using constant values to format the output of the report.

- **Minus:** To find out which records occurs only in one of the two result sets.

- **Intersect:** To find out which records occurs in both result sets.

- **Projection:** This node can be used to remove some fields in the output fields, rename fields, filter data, and calculate columns.

- **Aggregation:** We can calculate single summary values using functions like SUM, COUNT, MAX, MIN, AVERAGE and so on and it works like GROUP BY clause in SQL language.

- **Rank:** We can use this node to create top N or bottom N lists of values to sort results.

- **Table Function:** It is code written in SQLScript or SQL which returns a result set. This works like a table. We will see an example later in the next business question.

- **Hierarchy Function:** it is a new function from SAP HANA 2.0 SPS 03 that enables you to build information models with hierarchical data from tables.

The next step will be adding the data source of our dimension table so click on *projection* node and push on the **+** symbol (as seen in figure 5.16), then select the table customers by searching by "*cust*", select the table and click on finish.

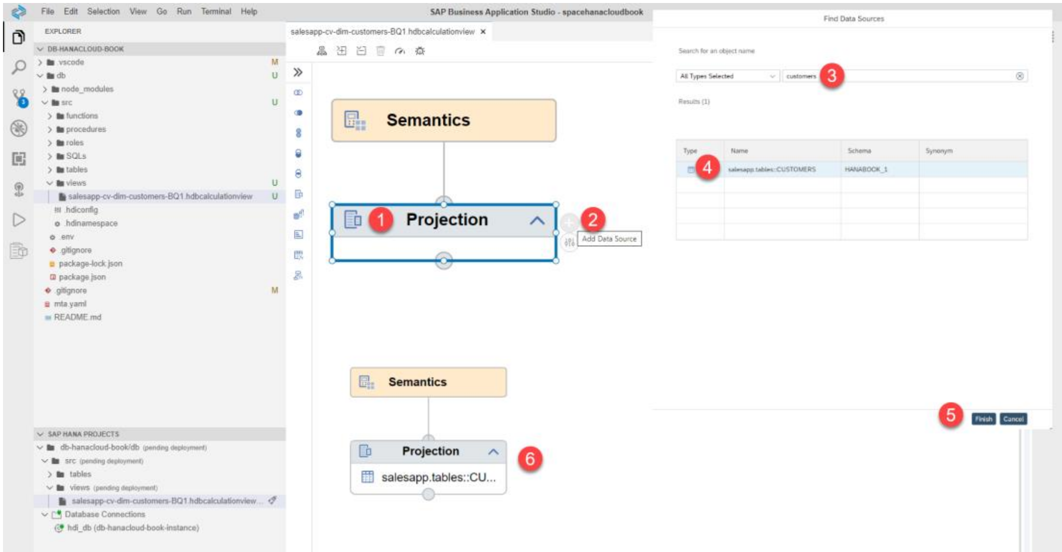


Figure 5.16 Adding customers data source

(Copyright © 2022 SAP SE or an SAP affiliate company. All rights reserved)

In case we are facing issues trying to find the table we need to unbind and bind again the container as a workaround, so you can use the option *bind/unbind* to do it (as seen in figure 5.17)

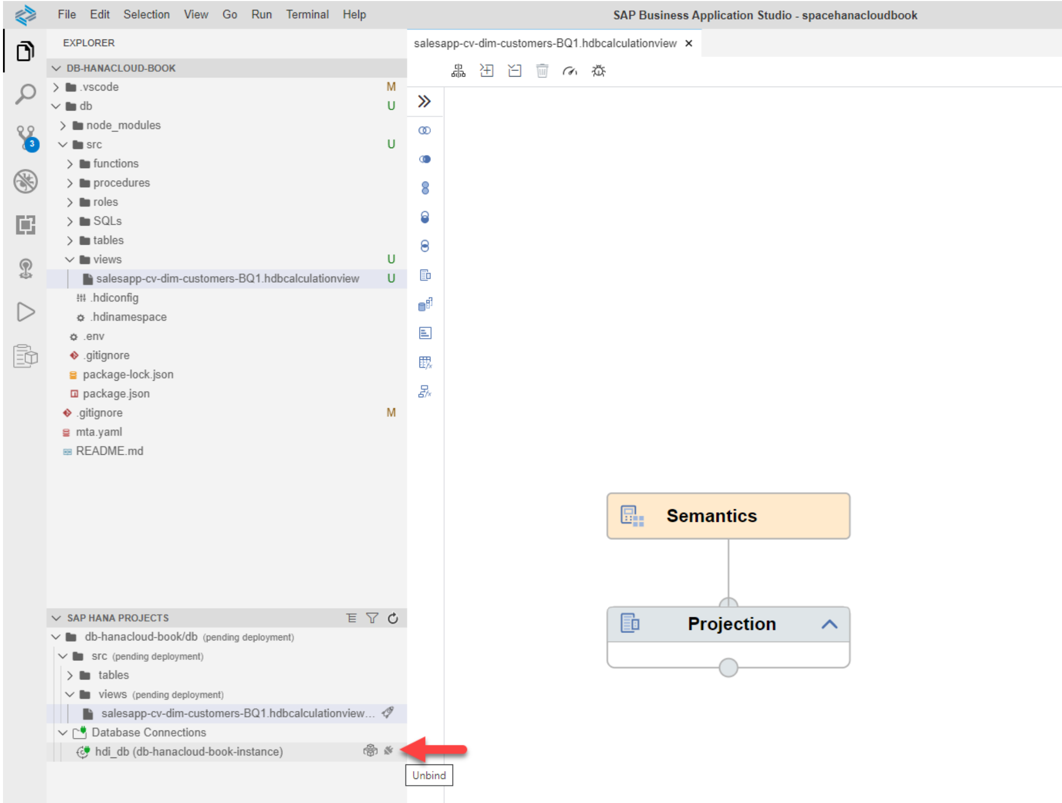

File Edit Selection View Go Run Terminal Help
SAP Business Application Studio - spacehanacloudbook
EXPLORER
salesapp-cv-dim-customers-BQ1.hdbcalculationview
DB-HANACLOUD-BOOK
.vscode
db
node_modules
src
functions
procedures
roles
SQLs
tables
views
salesapp-cv-dim-customers-BQ1.hdbcalculationview
.hdiconfig
.hdinamespace
.env
.gitignore
package-lock.json
package.json
.gitignore
mta.yaml
README.md
SAP HANA PROJECTS
db-hanacloud-book/db (pending deployment)
src (pending deployment)
tables
views (pending deployment)
salesapp-cv-dim-customers-BQ1.hdbcalculationview...
Database Connections
hdi_db (db-hanacloud-book-instance)
Unbind
Semantics
Projection

Figure 5.17 bind/unbind DB the HDI container

(Copyright © 2022 SAP SE or an SAP affiliate company. All rights reserved)

From mapping tab, we will add to the output all the fields of our customers table (as seen in figure 5.18) and then we will deploy the calculation view (step 3)

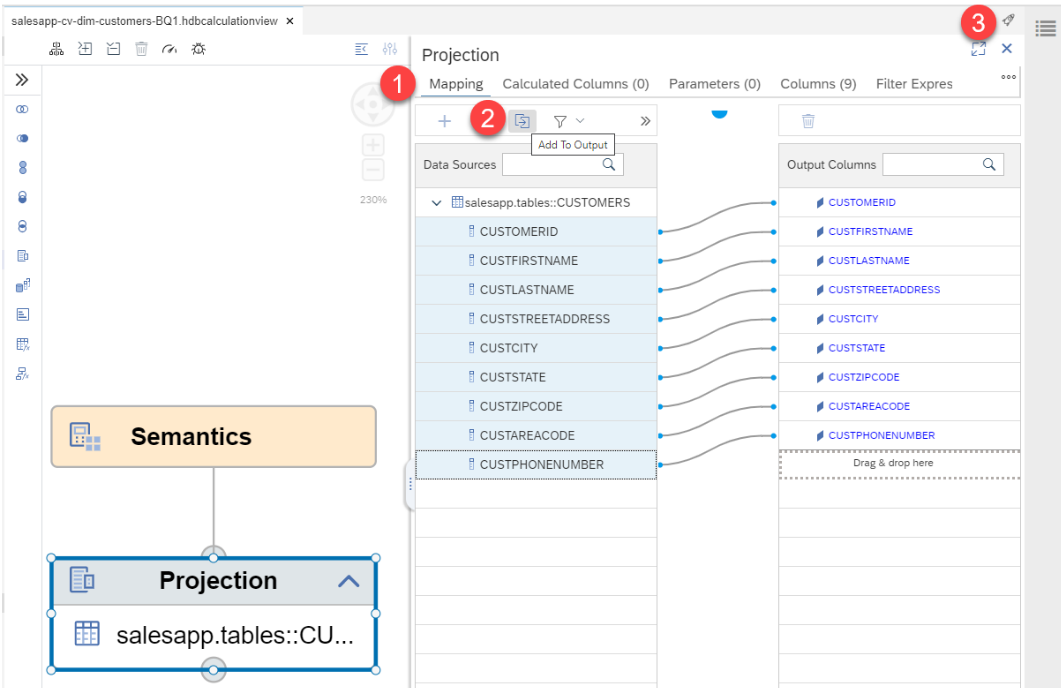


Figure 5.18 Add to Output

(Copyright © 2022 SAP SE or an SAP affiliate company. All rights reserved)

Finally, we can check the output of the calculation view of type dimension for the customers table with the *data preview* option and from the raw data (as seen in figure 5.19).

6.12.1. Data Understanding

The data understanding phase will guide us in collecting, analyzing, exploring, and checking the quality of all the datasets of the project:

- ***Collect Initial data:*** As we have defined previously in SAP HANA chapters the tables that we will use (from the chapter 3, section 3.2, and figure 3.1), we only need to check it in the next sub-phases. The following are the tables to be used:

  - *ORDERS*
  - *ORDERDETAILS*
  - *PRODUCTS*

- ***Describe data:*** From the browse connections in metadata explorer, we will find the *HANABOOK_1* schema where all the tables are stored (as seen in figure 6.46) and from there we can check the columns available (step 5), the number of rows, column types and so on (step 6) for all the three tables.

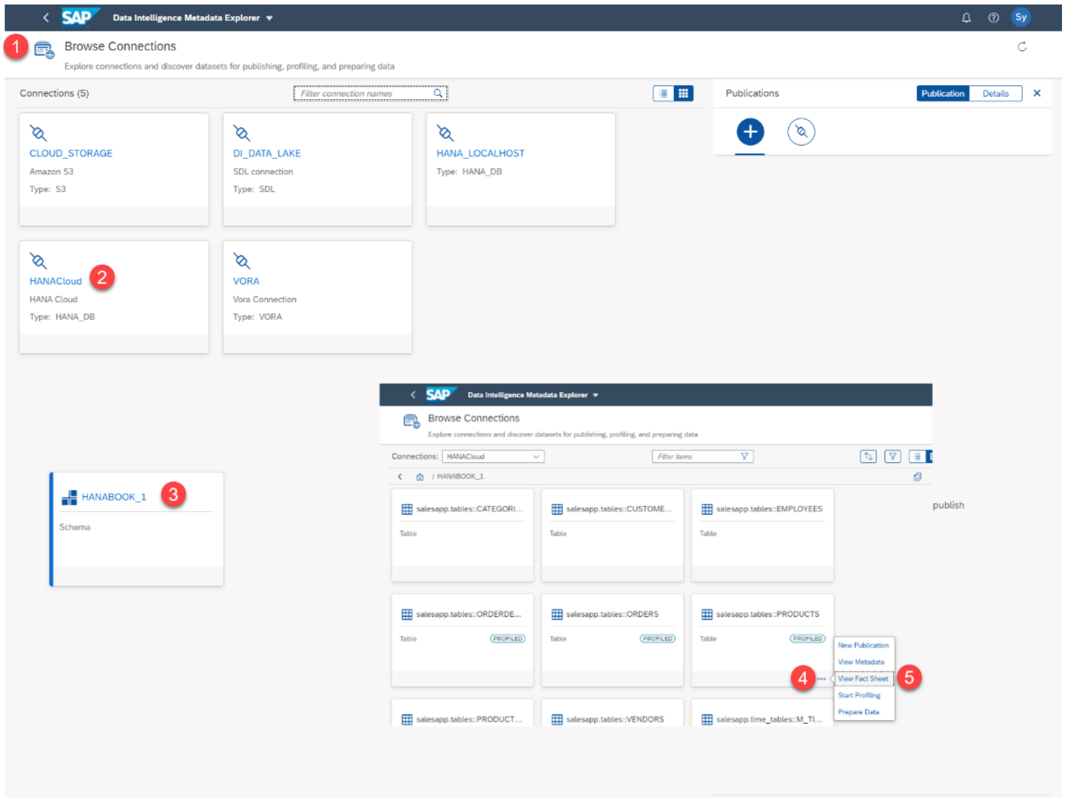
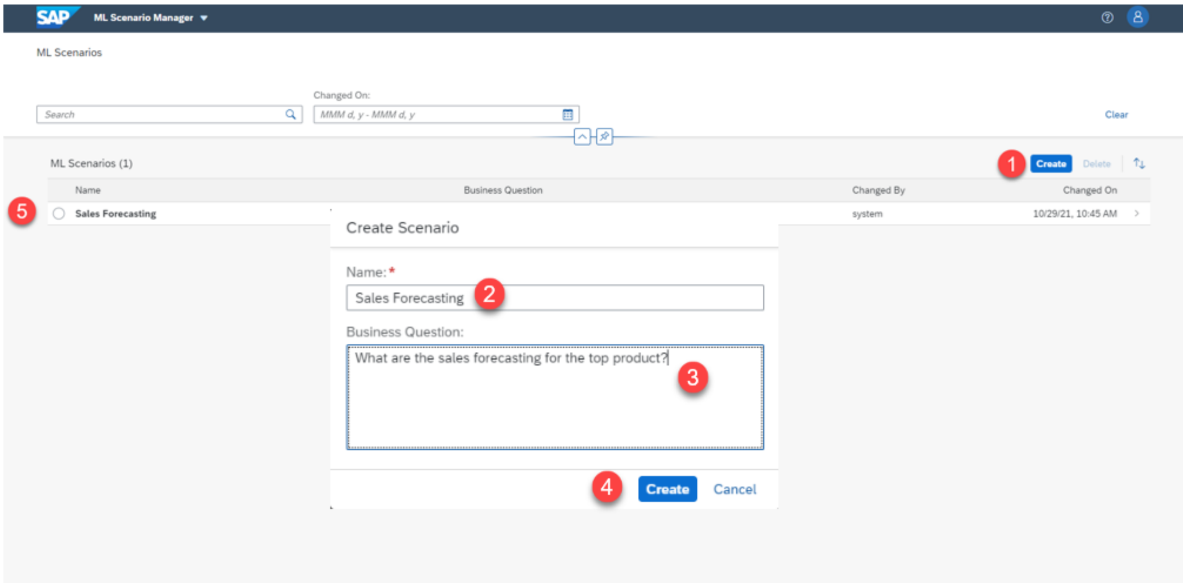


Figure 6.46 Metadata and fact sheet

(Copyright © 2022 SAP SE or an SAP affiliate company. All rights reserved)

- ***Explore data:*** The data exploration sub-phase will allow us to get a better understanding of datasets like missing values, data distribution, hypothesis generation, etc. and from SAP Data Intelligence we can use jupyter notebooks to do the exploration, so the first step is creating the ML scenario project with a specific title and business questions (as seen in figure 6.47).



Name	Business Question	Changed By	Changed On
○ Sales Forecasting		system	10/29/21, 10:45 AM

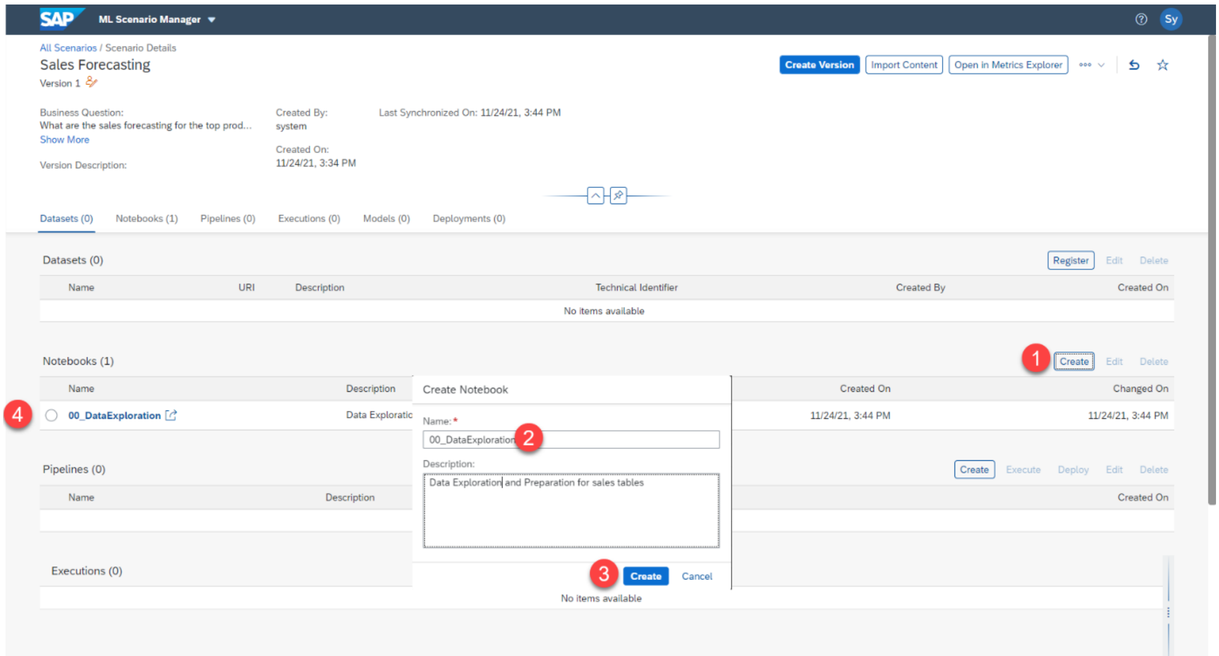
Name:*
Sales Forecasting

Business Question:
What are the sales forecasting for the top product?

Figure 6.47 ML Scenario for sales forecasting

(Copyright © 2022 SAP SE or an SAP affiliate company. All rights reserved)

Then we have to create the jupyter notebooks from the *Notebooks* section, and as we are going to create several ones, we can use a number in the name (like 00, 10, 20, …) to see them sorted out later (as seen in figure 6.48) and to have a clear naming convention.



Sales Forecasting

Version 1

Business Question:
What are the sales forecasting for the top prod...
Show More

Version Description:

Created By:
system

Created On:
11/24/21, 3:34 PM

Last Synchronized On: 11/24/21, 3:44 PM

Datasets (0)

Name	URI	Description	Technical Identifier	Created By	Created On
No items available					

Notebooks (1)

Name	Description	Created On	Changed On
00_DataExploration	Data Exploratic	11/24/21, 3:44 PM	11/24/21, 3:44 PM

Create Notebook

Name:* 00_DataExploration

Description: Data Exploration and Preparation for sales tables

Pipelines (0)

Name	Description	Created On

Executions (0)

No items available

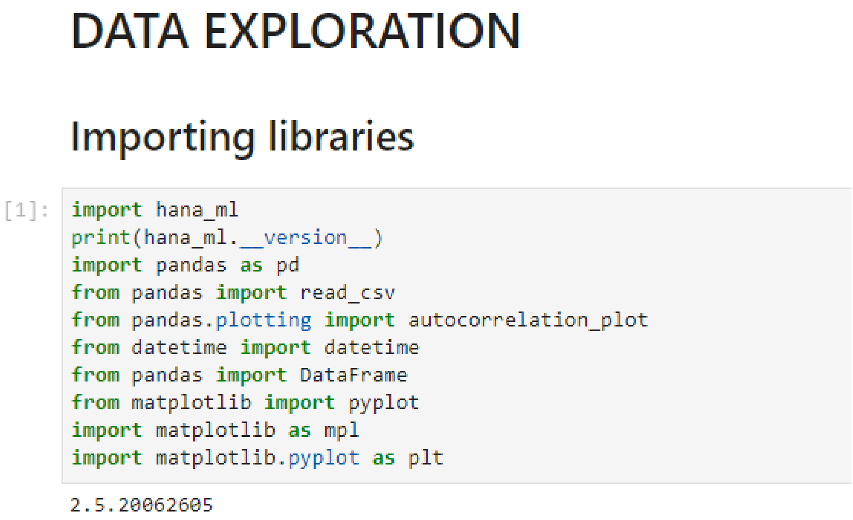
Figure 6.48 Create notebook

(Copyright © 2022 SAP SE or an SAP affiliate company. All rights reserved)

The next step will be starting the data exploration sub-phase, so we have to load the libraries and the datasets from the SAP HANA cloud database.

Importing libraries:

We will import all the libraries needed for the data exploration (as seen in figure 6.49)



DATA EXPLORATION

Importing libraries

```
[1]: import hana_ml
     print(hana_ml.__version__)
     import pandas as pd
     from pandas import read_csv
     from pandas.plotting import autocorrelation_plot
     from datetime import datetime
     from pandas import DataFrame
     from matplotlib import pyplot
     import matplotlib as mpl
     import matplotlib.pyplot as plt
```

```
2.5.20062605
```

Figure 6.49 Importing libraries