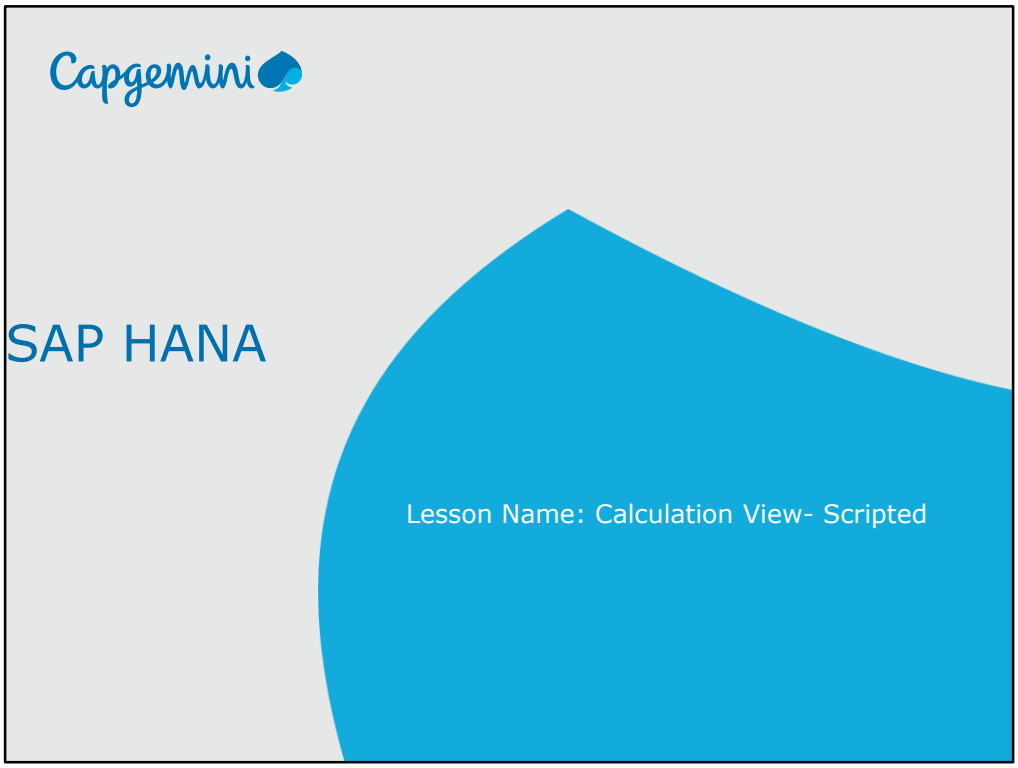

Capgemini
SAP HANA
Lesson Name: Calculation View- Scripted

Add instructor notes here.

Lesson Objectives

After completing this lesson, participants will be able to -

- Create calculation view - scripted using HANA Modeler
- Invoke calculation view using ADBC

Calculation View (Scripted)

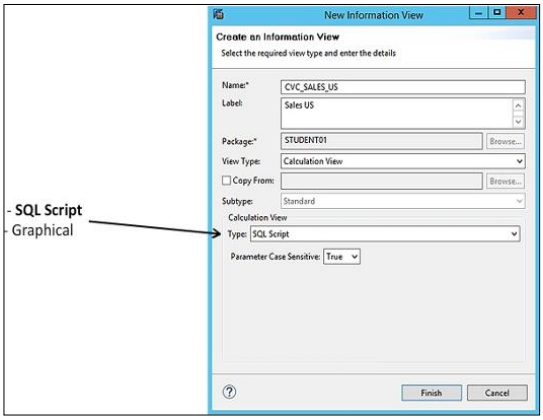
- SQLScript is used to create scripted calculation view.
- SQLScript is an SAP HANA query language developed by SAP.
- It is based on standard SQL but includes many extra functions to allow the developer to include conditional flow control logic such as If, Then, Else and While.
- Calculation Views can be created by the following 2 techniques
  - Scripted Calculation View
  - Graphical Calculation view.
- Scripted View is useful when you need to apply a complex logic that is not supported by graphical information views.
- For example, when you want to use syntax such as conditions(IF…THEN…ELSE), loops (FOR or WHILE), and so on.

There is a way to build calculation views using SQLScript code. We call these types of views **scripted calculation views**. These types of calculation views allow the developer more freedom to use standard SQL and SQLScript functions and have more control over the data flow logic.

The outcome is largely the same as if you were creating the calculation view using the graphical approach — you produce a data set based on original database tables by joining, filtering, aggregating measures over any number of attributes. The view can then be consumed directly by a reporting tool or via SQL. You can certainly have measure and attributes. Note :SQLScript is an SAP HANA query language developed by SAP. It is based on standard SQL but includes many extra functions to allow the developer to include conditional flow control logic such as If, Then, Else and While. This provides a similar level of control that is found in application programming code such as ABAP. An example : we can read each record and depending on the value in columns we can jump to different parts of the flow. This is not possible in standard SQL. To get started simply choose a calculation view but remember to select the subtype as **SQL Script** and not Graphical.

Creating Calculation View (Scripted)

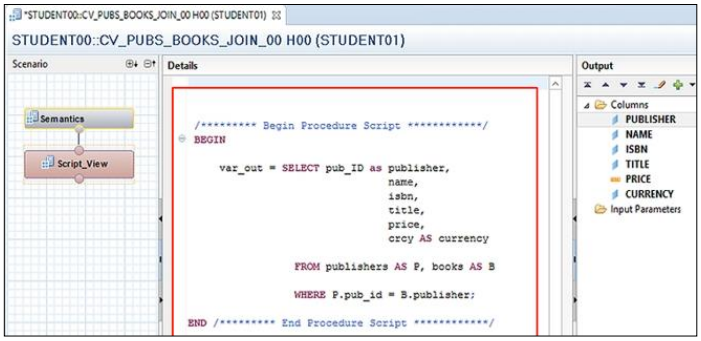
- To get started, simply choose a calculation view but remember to select the **subtype as SQL Script** and not Graphical.



Script-Based Calculation Views can be used when the graphical view types (DIMENSION and CUBE) do not fulfill the reporting requirement.

Writing the Script for Scripted Calculation View

- **Define the output columns:** After selecting SQL Script as shown in the last slide, you would then define the output columns.
- You can also have the system define the columns automatically by referring to an existing table or view.



You would then define the output columns. This can be done manually where you type a column name and its data type and length if applicable. Since SPS11 you can also have the system define the columns automatically by referring to an existing table or view.

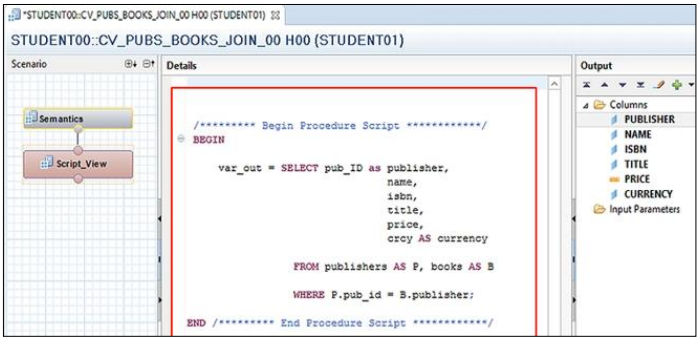
To write the script you must select the **Script View** node in the scenario pane. The script editor now appears in the centre of the screen. Here you describe what you would like to happen using the language **SQLScript**. Note: SQLScript is based on standard SQL but includes additional keywords and functions which have been developed by SAP.

Make sure you align the output columns with the column in the script using the same names.

Finally, rename any columns in the semantic node and add any other semantic information, then activate and preview. Note : It should be noted that from SPS9 onwards, SAP recommends that instead of creating scripted calculation views you should consider creating an SQL table functions. The reason for this is that an SQL table function offer far more flexibility in how it can be used as an input sources in other views compared to a scripted calculation view.

Writing the Script for Scripted Calculation View

- **Write the script :**To write the script you must select the Script View node in the scenario pane.
- The script editor now appears in the centre of the screen.
- Here you describe what you would like to happen using the language SQLScript.



You would then define the output columns. This can be done manually where you type a column name and its data type and length if applicable. Since SPS11 you can also have the system define the columns automatically by referring to an existing table or view.

To write the script you must select the **Script View** node in the scenario pane. The script editor now appears in the centre of the screen. Here you describe what you would like to happen using the language **SQLScript**. Note: SQLScript is based on standard SQL but includes additional keywords and functions which have been developed by SAP.

Make sure you align the output columns with the column in the script using the same names.

Finally, rename any columns in the semantic node and add any other semantic information, then activate and preview. Note : It should be noted that from SPS9 onwards, SAP recommends that instead of creating scripted calculation views you should consider creating an SQL table functions. The reason for this is that an SQL table function offer far more flexibility in how it can be used as an input sources in other views compared to a scripted calculation view.

Demo

Creation of Scripted Calculation View.

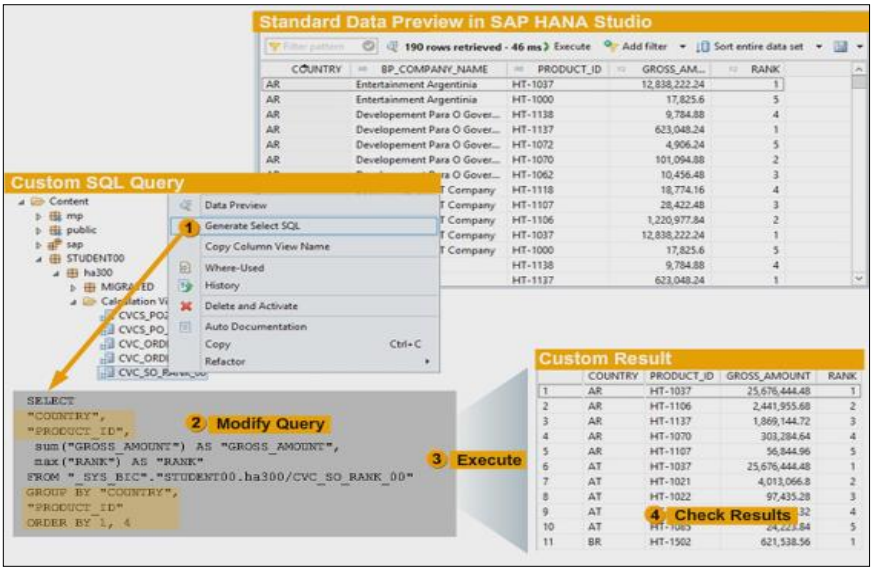
Querying Calculation Views in SAP HANA Studio

When building an Information View, SAP HANA Studio provides two ways to query the data of the view. It is important to understand between these approaches.

- i. Standard Data Preview
  - With the Standard Data Preview, **you select all the columns that** are included in the Semantics of the information view (provided that they are not hidden).
  - You can move the columns (drag and drop), and also order the result set by one (and only one) column.
- ii. Custom SQL Query
  - An alternative to the Standard Data Preview is to execute a Custom SQL Query. As shown in the figure in the next slide, after generating the SQL Statement equivalent to the data preview, **you can modify it**, and for example change the selected columns or the group by clause, order the result set by several columns.

When building an Information View, SAP HANA Studio provides two ways to query the data of the view. It is important to understand between these approaches.
Standard Data Preview
With the Standard Data Preview, you select all the columns that are included in the Semantics of the information view (provided that they are not hidden).
You can move the columns (drag and drop), and also order the result set by one (and only one) column.
Custom SQL Query
An alternative to the Standard Data Preview is to execute a Custom SQL Query. As shown in the figure, after generating the SQL Statement equivalent to the data preview, you can modify it, and for example change the selected columns or the group by clause, order the result set by several columns.
This is particularly useful to perform a thorough test of Calculation Views with a complex scenario (stacked calculation views, counter measures, join between several aggregation nodes with different GROUP BY columns, and so on).
Note: Indeed, Calculation Views behave differently depending on which columns are selected, on whether you explicitly define a group by or not. You must ensure that a View does not give wrong results if it is not correctly queried upon, or document the view so that it is correctly consumed by end-users with reporting applications.

Querying Calculation Views in SAP HANA Studio



When building an Information View, SAP HANA Studio provides two ways to query the data of the view. It is important to understand between these approaches.

Standard Data Preview

With the Standard Data Preview, you select all the columns that are included in the Semantics of the information view (provided that they are not hidden).

You can move the columns (drag and drop), and also order the result set by one (and only one) column.

Custom SQL Query

An alternative to the Standard Data Preview is to execute a Custom SQL Query. As shown in the figure, after generating the SQL Statement equivalent to the data preview, you can modify it, and for example change the selected columns or the group by clause, order the result set by several columns.

This is particularly useful to perform a thorough test of Calculation Views with a complex scenario (stacked calculation views, counter measures, join between several aggregation nodes with different GROUP BY columns, and so on).

Note: Indeed, Calculation Views behave differently depending on which columns are selected, on whether you explicitly define a group by or not. You must ensure that a View does not give wrong results if it is not correctly queried upon, or document the view so that it is correctly consumed by end-users with reporting applications.

Demo

Creation of Calculation view in SAP HANA Studio and query it.

Also modify the query and execute to illustrate other options too.

Invoke the calculation script using ADBC

Add instructor notes here.

Summary

In this lesson, you have learnt:

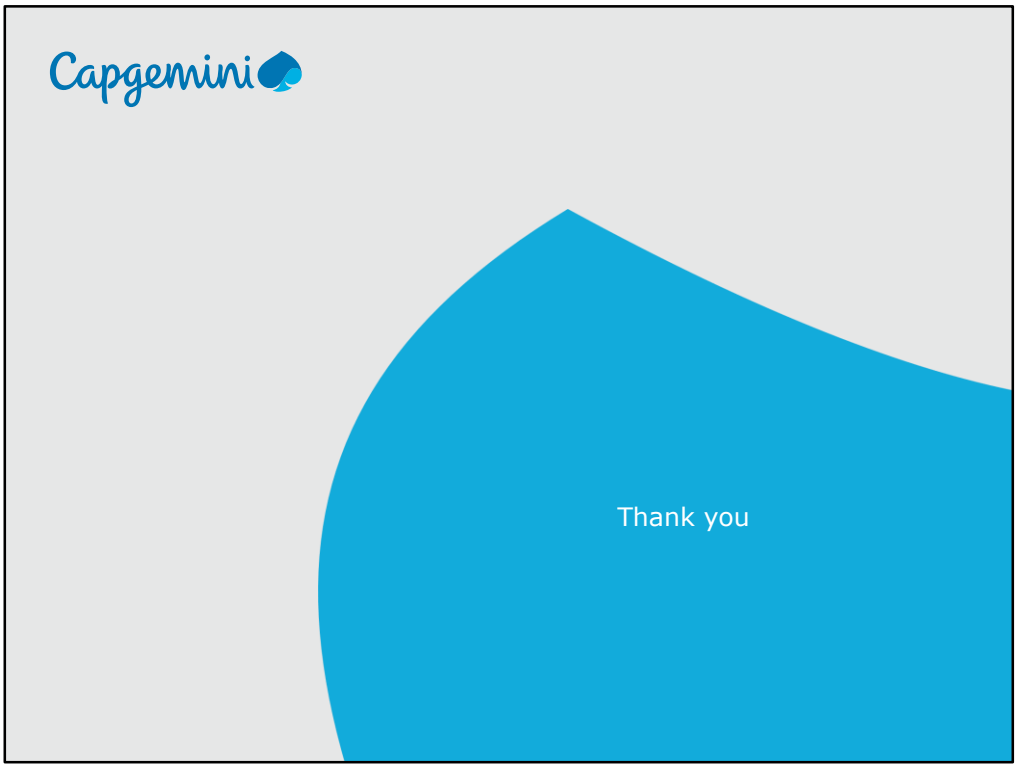
- How to create a scripted calculation view scripts using HANA Modeler
- How to use ADBC to invoke the scripted calculation view

Add the notes here.

Add instructor notes here.

Review Question

Add the notes here.


Capgemini
Thank you