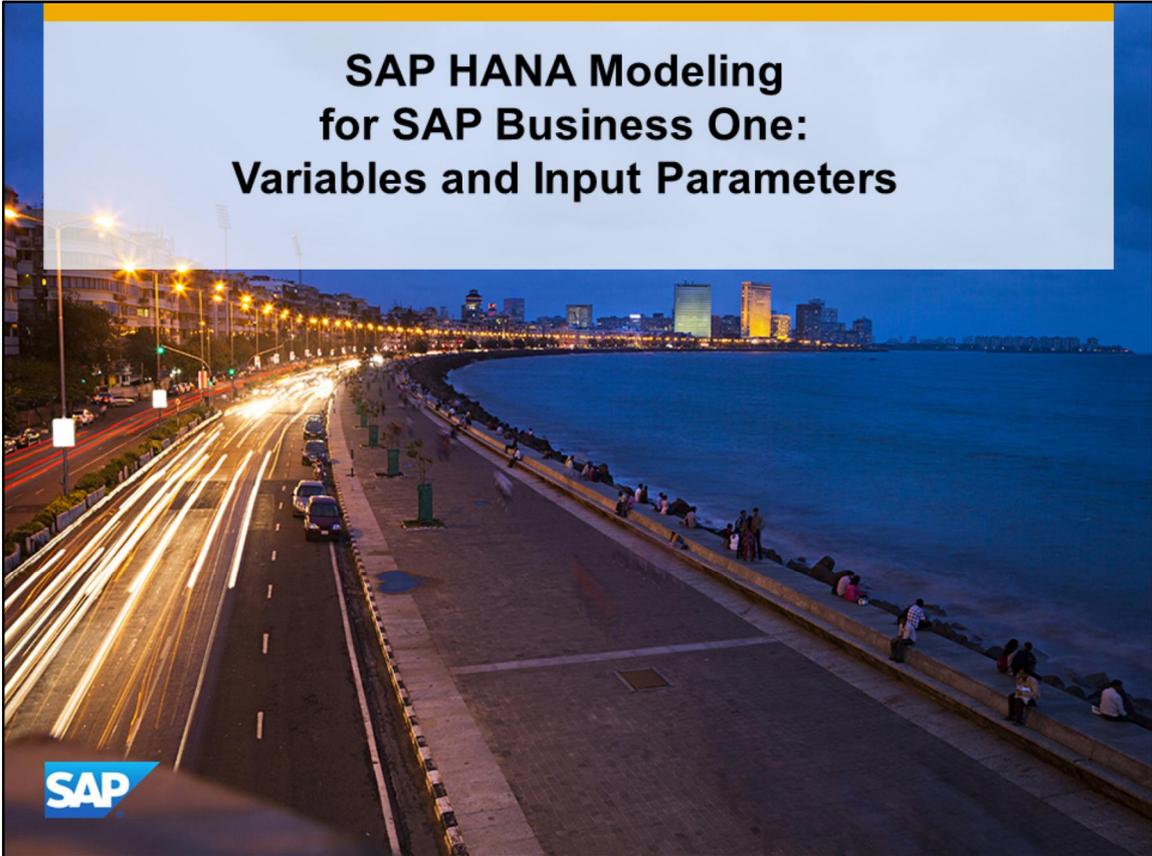


SAP HANA Modeling for SAP Business One: Variables and Input Parameters



Welcome to the topic on variables and input parameters.

Objectives

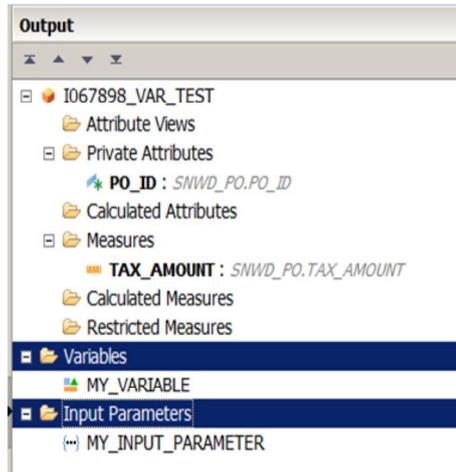


At the end of this course, you will be able to:

- Describe how to create Variables and Input Parameters

At the end of this course, you will be able to describe how to create variables and input parameters.

Object Types for Getting Data from Report Users



Variable

- Bound to an attribute
- Used for filtering values
- Can only contain the values available in the related attribute

Input Parameter

- Can contain any value the reporting user want to enter.
- Data type for the Input Parameter must be specified.

For query views, you can use variables and input parameters to get data from report users.

Variables are bound to attributes and are used for filtering. As such, they can only contain the values available in the attribute they relate to.

Input parameters can contain any value the reporting user wants to enter. Therefore, a data type for the input parameter must be specified.

Concepts for Variables

- You use variables to filter data at runtime.
- You assign values to these variables by entering the value manually, or by selecting it from the drop-down list.
- You do not need to decide on the restriction on the value of attributes at the design time.



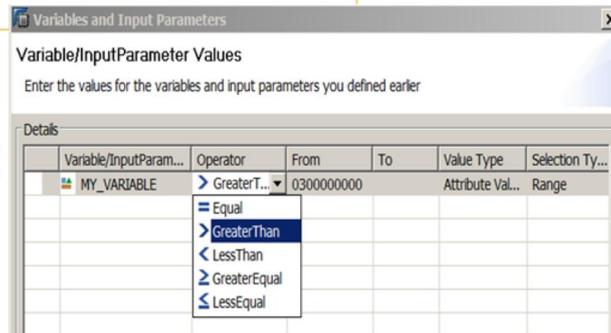
- You use variables to filter data at runtime. You assign values to these variables by entering the value manually, or by selecting it from the drop-down list.
- Using variables means that you do not need to decide on the restriction on the value of attributes at the design time.
- You can apply variables in the analytic and calculation views.
- If a calculation view is created using an analytic view with variables, those variables are also available in the calculation view but cannot be edited.

Variable types

The following types of variables are supported:

Type	Description
Single Value	Use this to apply a filter to a Single Value
Interval	Use this where you want the user to specify a set start and end to a selected interval.
Range	Use this when you want the end user to be able to use operators such as "Greater Than" or "Less Than".

Range Variable Example:



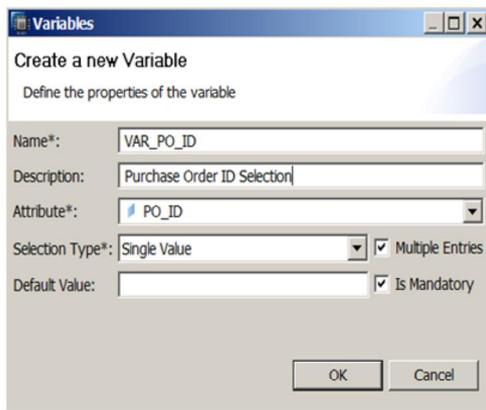
The following types of variables are supported:

Single Value: use this to apply a filter to a single value.

Interval: use this where you want the user to specify a set start and end to a selected interval.

Range: Use this when you want the end user to be able to use operators such as "Greater Than" or "Lesser Than"

Creating Variables



Select the attribute to filter and define:

- **Selection Type:** Whether selections should be based on intervals, ranges or single values.
- **Multiple Entries:** Whether multiple values of the selection types should be allowed.
- **Default Value:** You have the option to define a default value for the variable.
- **Mandatory:** The variable can be set as mandatory.

A variable restricts the results in the view for the selected attribute. To create a variable, select the attribute in the view that you want to filter on, then define the following values:

Selection Type: You can choose whether selections should be based on intervals, ranges or single values.

Multiple Entries: You control whether multiple values of the selection types should be allowed.

Default Value: You have the option to define a default value for the variable.

You also have the option to set the variable as **Mandatory**.

Input parameter concepts



You might want to take input from the user and process it, returning dynamic data based on the user selection.

Using **Input Parameters** makes this possible.

Sometimes using a variable is not enough. You might want to do more than restrict the value of an attribute in a view.

You might want to take input from the user and process it, returning dynamic data based on the user selection.

Using Input Parameters makes this possible.

Input parameter types

The following types of Input variables are supported:

Type	Description
Direct	Use this to specify an input parameter as currency and date during currency conversion, and target unit during unit of measure conversion.
Column	Use this when the value of a parameter comes from an attribute or table column.
Static List	Use this when the value of a parameter comes from a user-defined list of values.
Derived from Table	Use this when the value of a parameter comes from a table column based on some filter conditions and you do not need to provide any input at runtime.

The following types of Input variables are supported:

Direct: Use this to specify an input parameter as currency and date during currency conversion, and target unit during unit of measure conversion.

Column: Use this when the value of a parameter comes from an attribute or table column.

Static List: Use this when the value of a parameter comes from a user-defined list of values.

Derived from Table: Use this when the value of a parameter comes from a table column based on some filter conditions and you do not need to provide any input at runtime.

Example of an Input Parameter

Imagine that you have an Aging Query.

You would like to use the same query to run aging for accounts receivable or vendor liabilities.

You can use an input parameter for *Business Partner Type* to allow the report user to make this decision when running the report.



Imagine that you have an Aging Query.

You would like to use the same query to run aging for accounts receivable or vendor liabilities.

You can use an input parameter for Business Partner Type to allow the report user to make this decision when running the report.

When running the report the user could choose Customer or Vendor.

Input Parameters definition

Edit Input Parameter Definition

Input parameters are used to parameterize the view execution such as, to parameterize currency conversion, calculated columns or

Name	Description
C	Customer
S	Vendor

Here is the input parameter definition for our example

- Choose any suitable parameter type, for example a StaticList type.
- Define whether the parameter is mandatory.
- Set any default values. In this case "Customer"
- Define the data type for the parameter.
- This shows a list of values for our static list.

When creating an input parameter definition, first add a name and give it a descriptive label.

Choose a suitable parameter type, for example a StaticList type. We used this type for the input parameter for our Aging Query because the value of this parameter will come from a user-defined list of values.

You have the option to define whether a parameter is mandatory. You also can set a default value. In this case, we have chosen to set a constant value of C.

Input parameters must have a data type defined because they are not directly associated with an attribute unlike variables.

In this example we chose to have a static list with values that the user can choose, therefore we entered the values "Customer" or "Vendor" in a list of values.

Case Study Variable

Create a Variable to filter the NetSales by year

- **Create a Variable in our Query View:**
 - Link the Variable to the YEAR attribute

- **Run the Data Preview, you will be prompted to enter a YEAR before the view is executed.**

Create a Variable to filter the NetSales by year

- Create a Variable in our Query View:
 - Link the Variable to the YEAR attribute

- Run the Data Preview, you will be prompted to enter a YEAR before the view is executed.

Case Study: Input Parameter

We are going to calculate the commission charges based on an input parameter: % commission.

Add an Input Parameter to your Query view where the user will enter a % commission on sales:

□ **Create the Input Parameter:**

- Type – Decimal (21,6)

□ **Create a calculated column**

- Call the calculated column “Commission Total”
- Multiply the NetSales with the % Discount Input Parameter and divide by 100.

We are going to calculate the commission charges based on an input parameter: % commission.

Add an Input Parameter to your Query view where the user will enter a % commission on sales:

□ **Create the Input Parameter:**

- Type – Decimal (21,6)

□ **Create a calculated column**

- Call the calculated column “Commission Total”
- Multiply the NetSales with the % Discount Input Parameter and divide by 100.

Summary



Here are some key points:

- For query views, you can use variables and input parameters to get data from report users.
- Variables are bound to attributes and are used for filtering.
- Variables can only contain the values available in the attribute they relate to. You do not need to decide on the restriction on the value of attributes at the design time.
- Input parameters are more flexible. They can contain any value the reporting user wants to enter.
- A data type for the input parameter must be specified.

- Variables are bound to attributes and are used for filtering.
- As such, they can only contain the values available in the attribute they relate to. You do not need to decide on the restriction on the value of attributes at the design time.
- Input parameters are more flexible. They can contain any value the reporting user wants to enter.
- Therefore, a data type for the input parameter must be specified.

Thank you

This concludes the topic on variables and
input parameters.

Thank you for your time.

This concludes the topic on variables and input parameters in the SAP Business One, version for SAP HANA. Thank you for your time.

© 2015 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Excel, Outlook, PowerPoint, Silverlight, and VisualStudio are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, System i, System i5, System p, System p5, System x, System z, System z10, z10, z/VM, z/OS, OS/390, zEnterprise, PowerVM, Power Architecture, PowerSystems, POWER7, POWER6+, POWER6, POWER, PowerHA, pureScale, PowerPC, BladeCenter, System Storage, Storwize, XIV, GPFS, HACMP, RETAIN, DB2 Connect, RACF, Redbooks, OS/2, AIX, Intelligent Miner, WebSphere, Tivoli, Informix, and Smarter Planet are trademarks or registered trademarks of IBM Corporation.

Linux is the registered trademark of Linus Torvalds in the United States and other countries.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are trademarks or registered trademarks of Adobe Systems Incorporated in the United States and other countries.

Oracle and Java are registered trademarks of Oracle and its affiliates.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems Inc.

HTML, XML, XHTML, and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Apple, App Store, iBooks, iPad, iPhone, iPhoto, iPod, iTunes, Multi-Touch, Objective-C, Retina, Safari, Siri, and Xcode are trademarks or registered trademarks of Apple Inc.

IOS is a registered trademark of Cisco Systems Inc.

RIM, BlackBerry, BBM, BlackBerry Curve, BlackBerry Bold, BlackBerry Pearl, BlackBerry Torch, BlackBerry Storm, BlackBerry Storm2, BlackBerry PlayBook, and BlackBerry App World are trademarks or registered trademarks of Research In Motion Limited.

Google App Engine, Google Apps, Google Checkout, Google Data API, Google Maps, Google Mobile Ads, Google Mobile Updater, Google Mobile, Google Store, Google Sync, Google Updater, Google Voice, Google Mail, Gmail, YouTube, Dalvik and Android are trademarks or registered trademarks of Google Inc.

INTERMEC is a registered trademark of Intermec Technologies Corporation.

Wi-Fi is a registered trademark of Wi-Fi Alliance.

Bluetooth is a registered trademark of Bluetooth SIG Inc.

Motorola is a registered trademark of Motorola Trademark Holdings LLC.

Computop is a registered trademark of Computop Wirtschaftsinformatik GmbH.

SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP BusinessObjects Explorer, StreamWork, SAP HANA, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries.

Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects Software Ltd. Business Objects is an SAP company.

Sybase and Adaptive Server, iAnywhere, Sybase 365, SQL Anywhere, and other Sybase products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Sybase Inc. Sybase is an SAP company.

Crossgate, m@gic EDDY, B2B 360°, and B2B 360° Services are registered trademarks of Crossgate AG in Germany and other countries. Crossgate is an SAP company.

All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

The information in this document is proprietary to SAP. No part of this document may be reproduced, copied, or transmitted in any form or for any purpose without the express prior written permission of SAP AG.