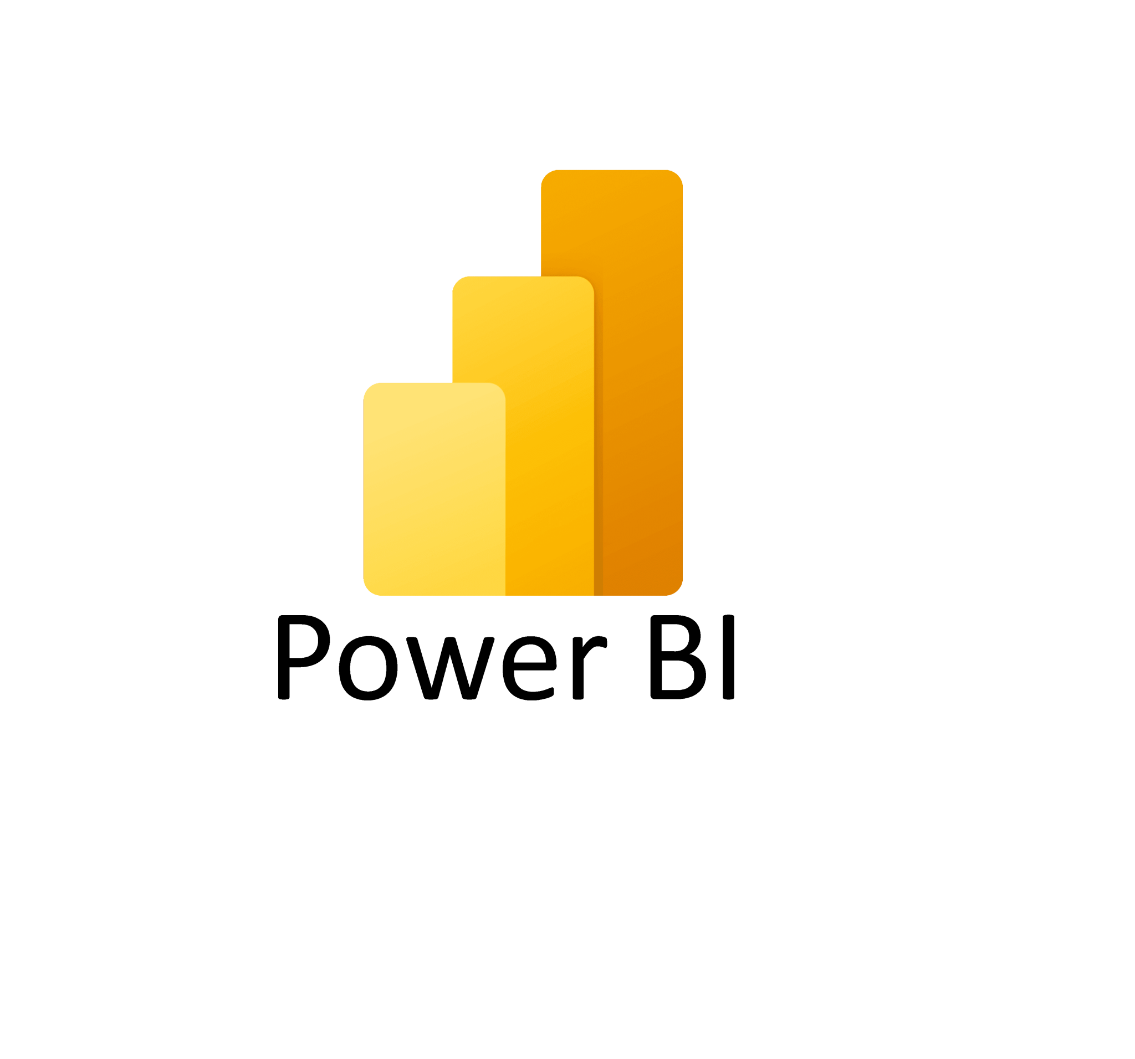
A logo for a video game

AI-generated content may be incorrect.

**Powered by Bokamoso Coders: Microsoft Power BI**

****

PARTICIPANT HANDBOOK

INSTRUCTOR-LED TRAINING

Course Version: 1

Course Duration: 4 hours

Material Number: 50160449

# Contents

Course Overview

|  |  |  |
| --- | --- | --- |
| 1 | Unit 1: | Course Introduction |
|  |  |  |
|  |  |  |
| 11 |  | Welcome & Objectives |
| 13 |  | Why Power BI? (business use case) |
| 53 | Unit 2: | Set-up and Installation |
|  |  |  |
| 55 |  | Creating a Microsoft PowerBI Account |
| 61 |  | Installing PowerBI |
| 66 |  | PowerBI walkthrough |
| 111 | Unit 3: | Connecting and shaping Data |
|  |  |  |
| 112 |  | Lesson: Importing different Files  Lesson: Connecting to different databases.  Lesson: Import data from folders |
| 115 |  |  |
| 131 | Unit 5: | Configuring Charts & Tables |
|  |  |  |
| 133 |  | Lesson: Configuring Tables |
| 138 |  | Lesson: Configuring Charts |
| 147 |  | Exercise 5: Configure Charts and Text Widgets |
| 167 |  | Lesson: Using Conditional Formatting |
| 169 |  | Exercise 6: Use Conditional Formatting (Optional) |
| 185 | Unit 6: | Sorting & Filtering |
|  |  |  |
| 187 |  | Lesson: Sorting Data in Stories |
| 188 |  | Lesson: Filtering Data in Stories |
| 197 |  | Exercise 7: Filtering |
| 217 |  | Exercise 8: Use Measure and Time Input Controls for a Time |
|  |  | Series and Scatterplot |

|  |  |  |
| --- | --- | --- |
| 235 | Unit 7: | Linking & Blending |
|  |  |  |
| 236 |  | Lesson: Linking and Blending Data in Stories |
| 239 |  | Exercise 9: Use Linked Analysis & Blending |
| 255 | Unit 8: | Using Calculations, Currency, Custom Time Hierarchies |
|  |  |  |
| 256 |  | Lesson: Calculating Data in Stories |
| 263 |  | Exercise 10: Create Story Calculations |
| 283 |  | Lesson: Converting Currencies |

# Course Overview

TARGET AUDIENCE

This course is intended for the following audiences:

* Application Consultant
* Technology Consultant
* Business User
* Business Analyst

**UNIT 1**

**Getting Started with PowerBi**

### Lesson 1

Getting Started with PowerBI 3

UNIT OBJECTIVES

* Introduction to Power BI and its Ecosystem
* Explore the Power BI Interface
* Discuss Real-World Applications

Unit 1

Lesson 1

## What is Power BI?

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* What Power BI is and how it fits with Microsoft Fabric
* The key differences between Power BI Desktop and Power BI service
* How to get started with Power BI step-by-step

### **Introducing PowerBI**

Power BI is Microsoft’s Business Intelligence tool used for connecting, transforming,

and visualizing data. The purpose is to turn raw data into meaningful insights through

interactive dashboards and reports.

**Real-World Examples:**

* Retail: Sales dashboards showing performance by region.
* HR: Attrition dashboards tracking employee trends.
* Finance: Budget vs. Actual spend monitoring.

 **Power BI and Microsoft Fabric**

**What is Microsoft Fabric?**

Microsoft Fabric is an end-to-end analytics platform that brings together all the data tools an organization needs — from data movement, storage, and engineering to real-time analytics and BI reporting.

**Key Components of Fabric**

* Fabric unifies different workloads under one umbrella:
* **Data Factory** → For data integration and pipelines.
* **Synapse Data Engineering** → For big data analytics with Spark.
* **Synapse Data Science** → For advanced ML & AI experiments.
* **Synapse Data Warehousing** → For enterprise-scale relational analytics.
* **Real-Time Analytics** → For streaming data insights.
* **Power BI** → For visualization, storytelling, and sharing insights.

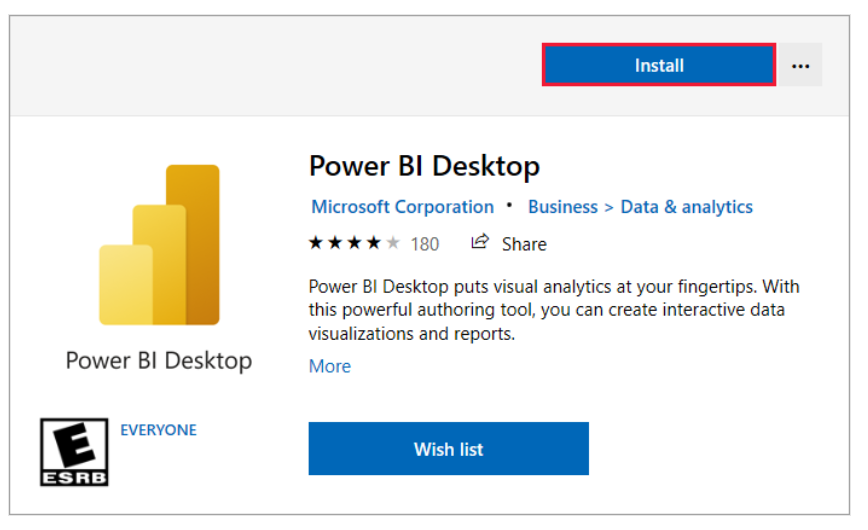
Power BI is the front door of Microsoft Fabric — its what business users see, but under the hood, Fabric connects everything else.

**Why this matters:**

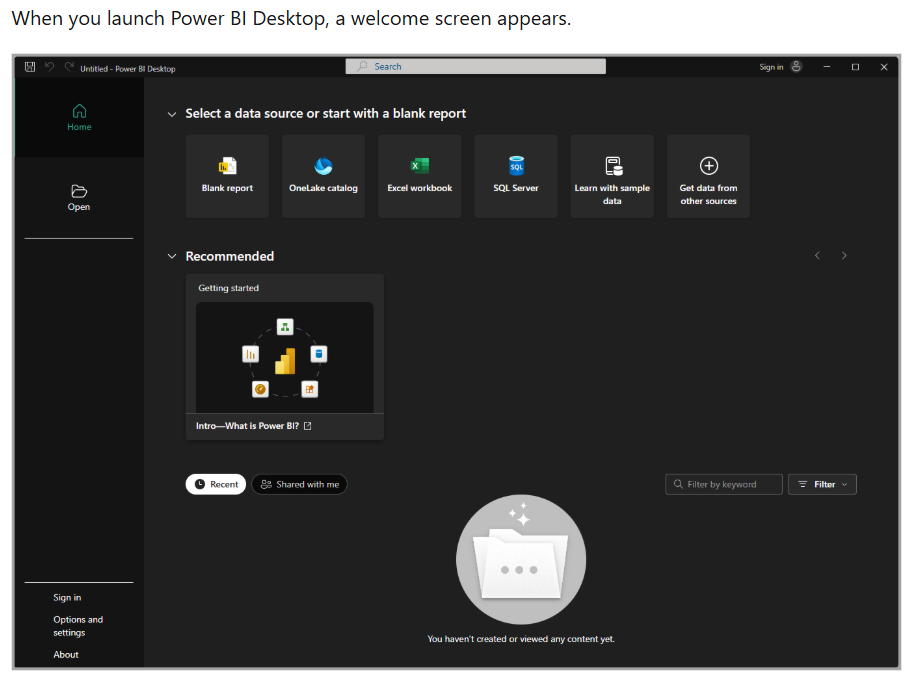
* Power BI isn’t just a reporting tool anymore — it’s part of a much larger data ecosystem.
* With Fabric, your reports can connect directly to OneLake (the unified data lake of Fabric) instead of depending only on Excel, SQL, or CSV files.
* This makes solutions scalable, cloud-native, and enterprise-ready.

**Installing PowerBI**

After you've landed on the **Power BI Desktop** page of the Microsoft Store, select **Install**



When you launch this Screen appears



Connecting to PowerBI datasources

**UNIT 2**

**Connecting to PowerBI data sources**

Lesson 1

Introductions to data Connections 11

Lesson 2

Connecting to File-based data sources 13

Lesson 3

Connecting to Databases 21

Lesson 4

Connecting to Cloud & Online Services 45

Lesson 5

Connecting to Web & API’s 45

UNIT OBJECTIVES

* Explain the different categories of data sources available in Power BI.
* Connect Power BI Desktop to file-based data sources such as Excel, CSV, JSON, and PDF.
* Connect to Microsoft Fabric using Lakehouse, Warehouse, and One Lake Data Hub.
* Apply best practices when connecting to data, including selecting relevant tables, setting data types, and handling credentials securely.

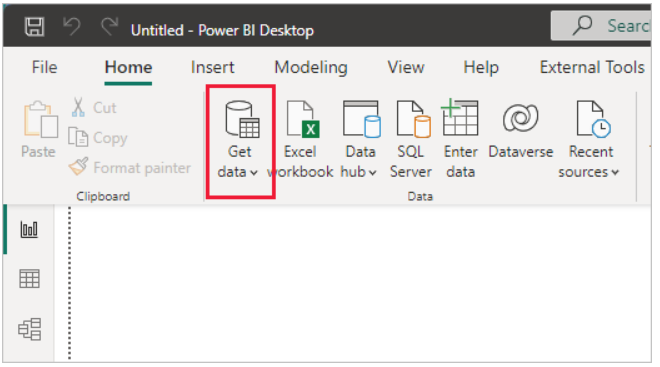
1. **What is a Data Source in Power BI?**

In Power BI, a data source is the location from which your report retrieves its data. Data sources can be:

* + - * A file on your computer (Excel, CSV, JSON)
      * A database (SQL Server, Oracle, MySQL, etc.)
      * A cloud service (Azure, SharePoint, OneDrive, Salesforce, etc.)
      * An online feed (Web, OData, APIs)
      * Microsoft Fabric Lakehouses and Warehouses

1. **The “Get Data” Button**

All connections start with the Get Data button on the Home ribbon in Power BI Desktop.



Since this data source is an Excel file, select **Excel** from the **Get Data** window, then select the **Connect** button.

1. **Common Connection Scenarios**

**a) Files**

* Useful for individual analysis or small projects
* Examples: Excel workbooks, CSV exports from a system

**b) Databases**

* Used for large, structured datasets stored in systems like SQL Server or Oracle
* Offers options for Import, DirectQuery, or Live Connection

**c) Cloud & Online Services**

* Connect directly to services like SharePoint, Dynamics, or Salesforce
* Ideal for teams and organizations storing data in the cloud

**d) Microsoft Fabric**

* Power BI integrates natively with **OneLake** in Fabric
* You can connect to Lakehouses and Warehouses for enterprise-scale analytics

**e) Web & APIs**

* Import public data tables from websites (e.g., population stats, stock prices)
* Advanced users can connect to APIs via OData feeds or custom queries

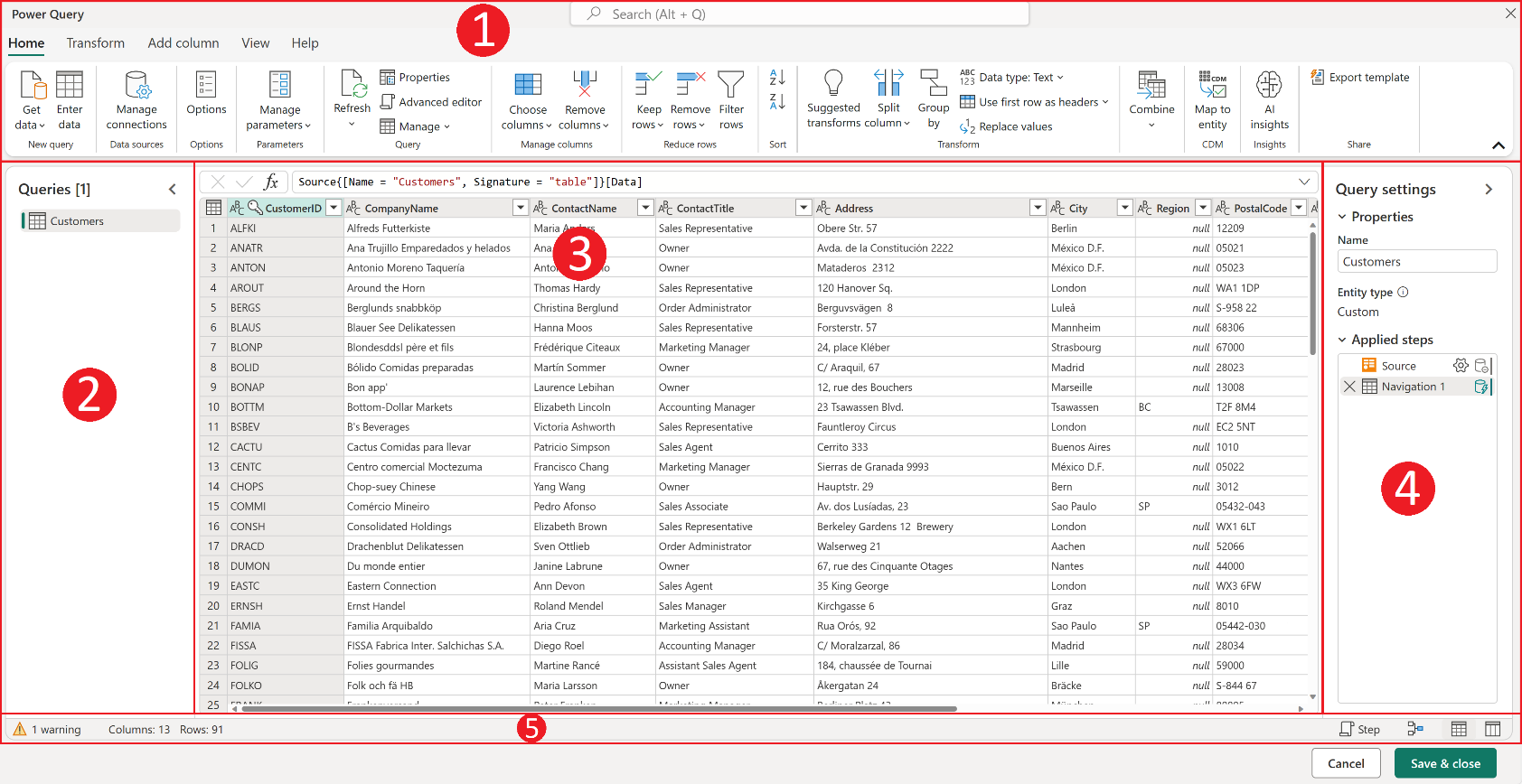
### **What is Power Query?**

Power Query is a tool inside Power BI used to clean, shape, and transform data before analysis. Think of it as a “data cleaning kitchen” before serving your dashboard.

**Opening Power Query**

* + - * Load the dataset: Apocalypse Food Prep → Purchase Tracker sheet.
* In Power BI Desktop → Home → Get Data → Excel → Select file → Load.
* Click Transform Data → Power Query Editor opens**.**

**Exploring Power Query Editor**

The Power Query editor represents the Power Query user interface. In this user interface, you can add or modify queries, manage queries by grouping or adding descriptions to query steps, or visualize your queries and their

### **Exploring the Power Query Editor**

* Ribbon → tools for transforming data.
* Queries Pane → shows "Purchase Tracker".
* Data Preview Grid → shows rows from the Excel sheet.
* Applied Steps Pane → logs every action (like a recipe).

**2. Importing & Transforming Data**

**Basic Cleanup**

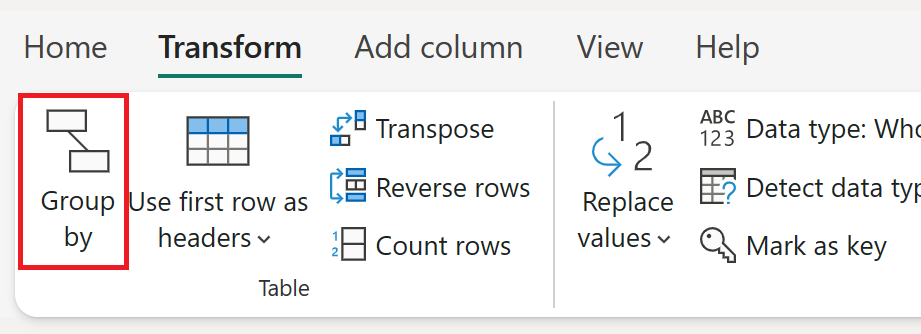
Home → Remove Top Rows → Delete the blank/title rows at the top.

A screenshot of a computer

AI-generated content may be incorrect.

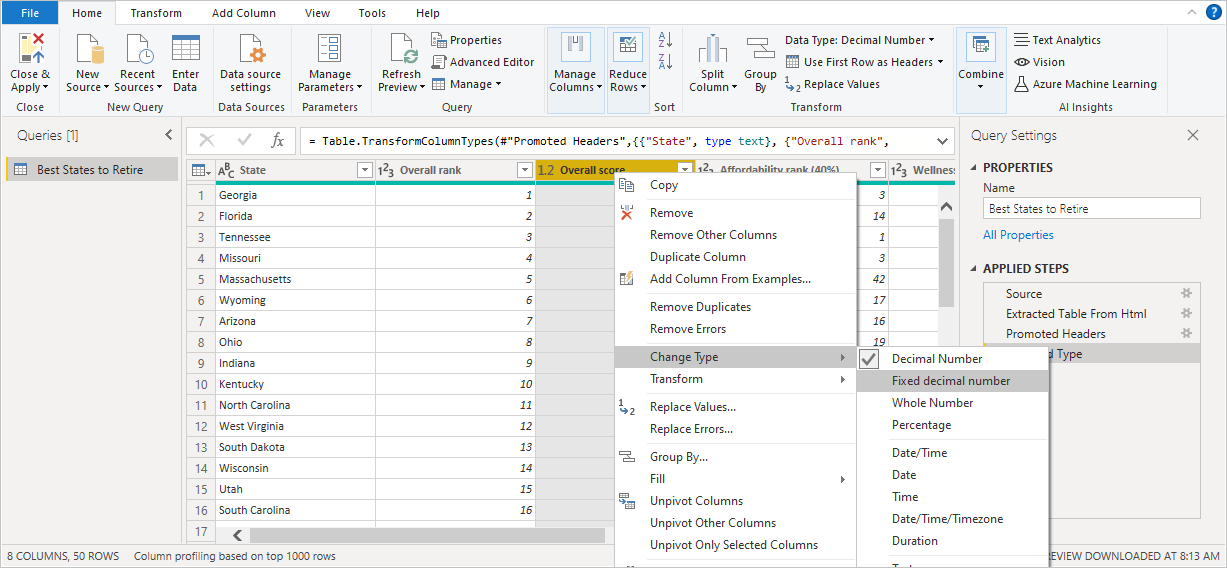
**Steps to use the First Row as Headers**

* In Power Query Editor, go to the Home tab on the ribbon.
* Click Use First Row as Headers under the Transform group.
* The first row will move into the header position, making your dataset cleaner and easier to work with.



**Steps to Change a Column’s Data type**

* 1. In the Power Query Editor, select the column you want to adjust.
* In our dataset, for example, the Product Cost values are numeric.
  1. Go to the Data Type dropdown (on the Home tab, top menu).
  2. Choose the correct data type from the list. Common ones include:
* Decimal Number → for prices, amounts, or values with decimals.
* Whole Number → for counts or quantities.
* Date → for date columns like Date\_Purchased.
* Text → for product names or store names.
  1. Once applied, you’ll see a new step added in the Applied Steps pane called Changed Type.



**3. Changing Column Types in Power Query**

Every column in your dataset needs the correct data type so Power BI knows how to handle it. For example:

* Dates must be stored as Date to create timelines.
* Numbers must be Whole Numbers or Decimals to allow calculations.
* Text fields (like Store or Product Name) must be text, so they’re not treated as numbers.

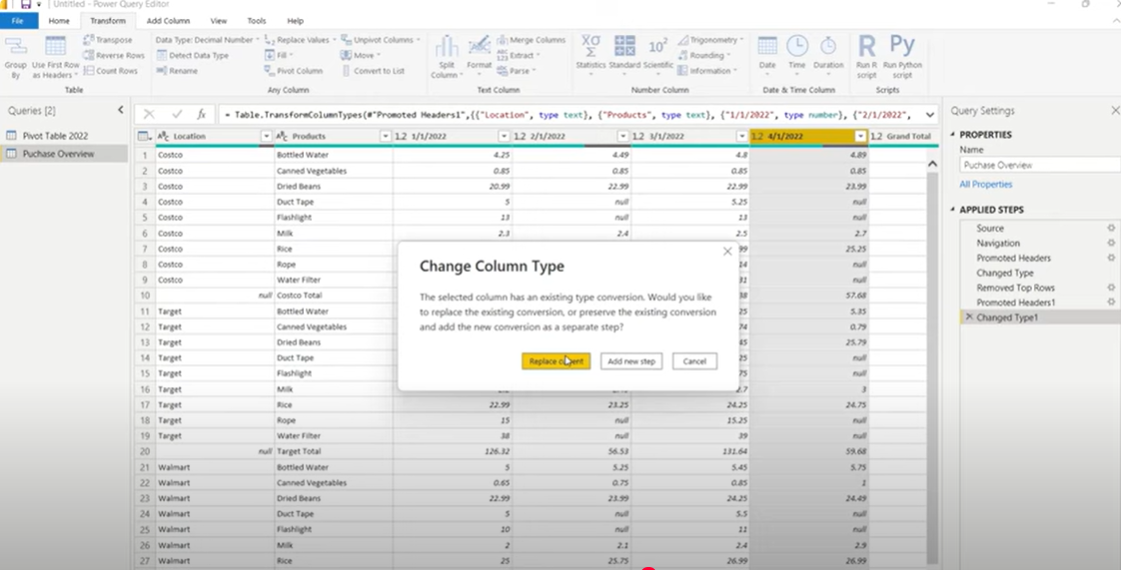
If the data type is wrong, you may get errors or misleading results in your visuals.

**Steps to Change a Column’s Data Type:**

1. Select the column you want to update.

In our example, we’re updating Product Cost columns to Decimal Number.

1. On the Home tab, click the **Data Type dropdown.**
2. Select the correct type (Date, Decimal, Whole Number, Text, etc.).



1. Filtering Data in Power Query

Filtering helps you focus only on the rows you need for analysis. This is especially useful when:

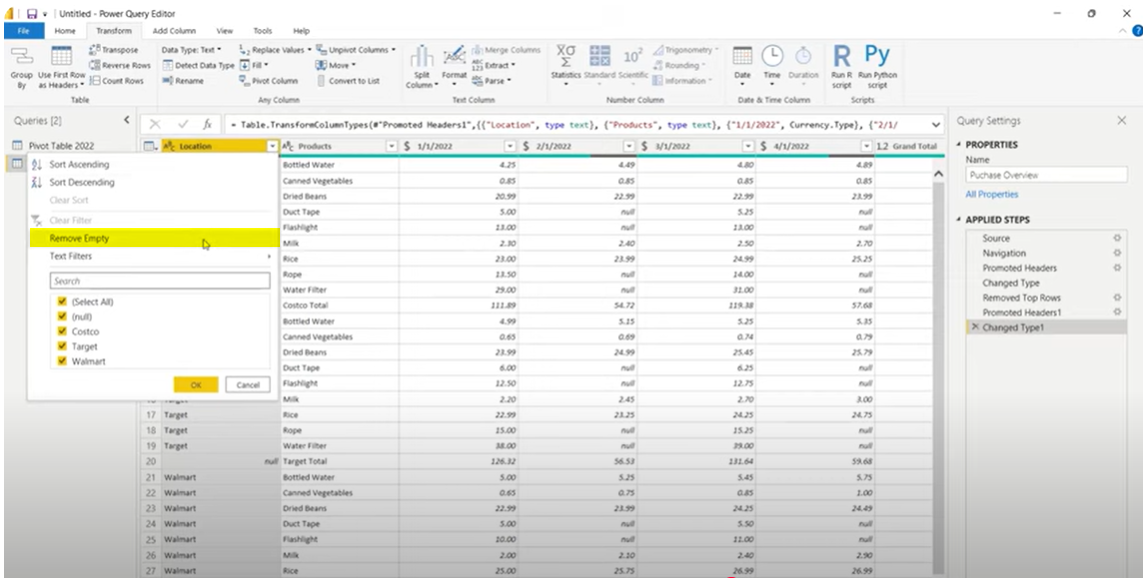
* You want to remove irrelevant values (e.g., Grand Total rows).
* You want to work with data from specific categories (e.g., only one store).
* You want to clean out blank or null entries.

Steps to Filter Data in Power Query:

1. Select the column you want to filter.

In this example, the column is Location.

1. Click the dropdown arrow next to the column name.
2. A filter menu appears (like in Excel). From here, you can:
   * + - * Select/deselect items.
         * Remove blanks (null values).
         * Apply text filters (e.g., contains, begins with)
3. Click OK to apply the filter.



Text Filter: Does Not Contain

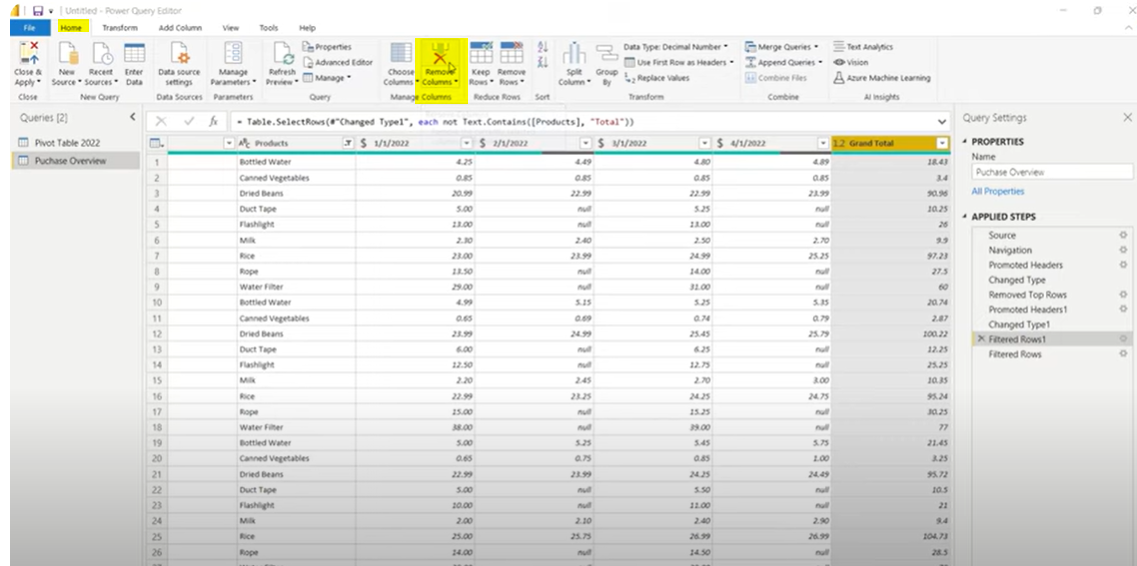
A screenshot of a computer

AI-generated content may be incorrect.

**Removing Unnecessary Columns**

After cleaning the rows, the next step is to simplify the dataset by keeping only the columns you need.

* Select the unwanted columns (e.g., Grand Total).
* On the Home tab, click Manage Columns → Remove Columns.



**Pivoting and Unpivoting Columns**

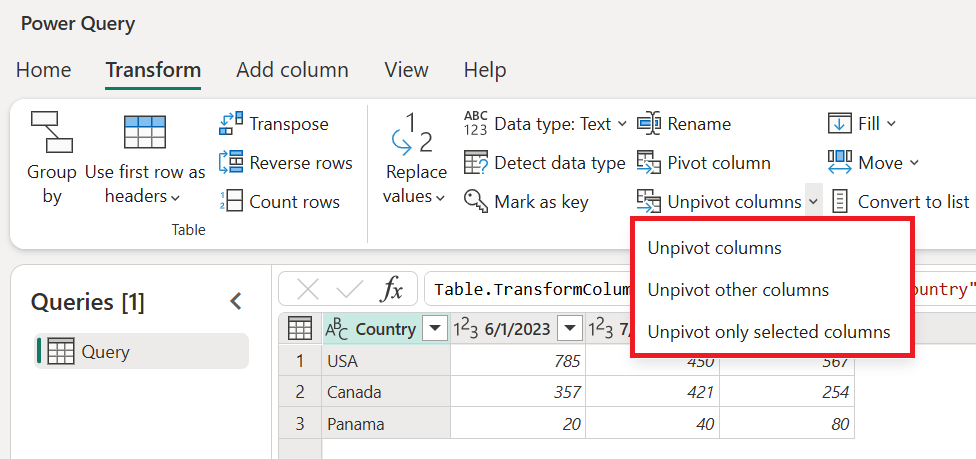
* + - Unpivot → turns columns into rows (useful when dates or categories are spread across columns, but you need them stacked in one column)
    - Pivot → turns rows into columns (useful for creating summary tables).

**Unpivoting Columns**

1. Select the columns you want to keep as identifiers (Location and Products).
2. On the Transform tab, click Unpivot Columns.
3. The date columns are transformed into two new fields:

* Attribute → holds the original column names (dates).
* Value → holds the corresponding product cost.

Now your dataset is tidy: each row represents a purchase with Location, Product, Date, and Value.



**Renaming Columns for Clarity**

To make the dataset easier to understand, rename $ Value → **Product Cost** and **Location** → **Store.** Clear

names improve readability and ensure consistency when building visuals in Power BI.

Unit 4

Lesson 1

## How to Create and Manage Relationships in PowerBI

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* Understand what relationships are in Power BI.
* Create and edit relationships between tables.
* Manage cardinality and cross-filtering settings.

### Dataset: Apocalypse Food prep- Relationship Tutorial

**What is a Relationship?**

* A connection between two tables, based on a common column (key).
* Example: Sales table (ProductID) linked to Products table (ProductID).

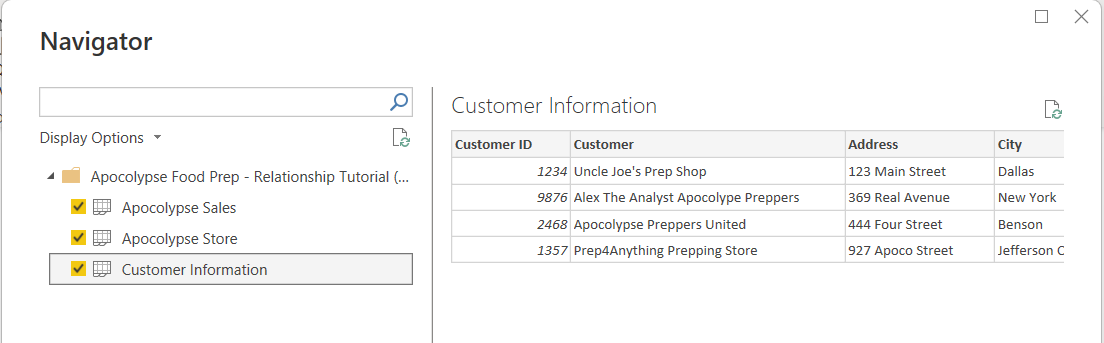
**Why Relationships Matter**

* They allow you to build reports that combine fields from different tables.
* Prevent duplicate data and keep your model efficient.

### **Creating Relationships**

**Step 1: Import Data**

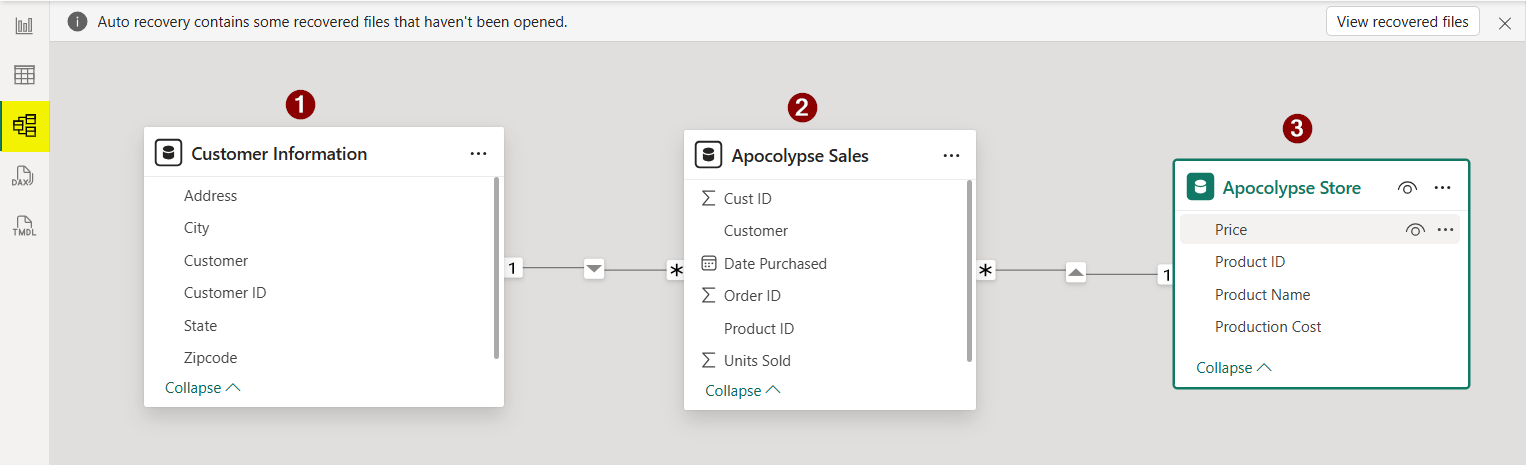
* From the Home tab → Get Data → Excel, select your file.
* In the **Navigator window**, tick all three sheets
* Customer Information
* Apocolypse Sales
* Apocolypse Store
* Use Click Load to bring them into Power BI.



**Step 2: Switch to Model View**

* On the left sidebar, click the Model icon to open the model view.
* Arrange the tables so they are easy to see:

1. Customer Information on the left.
2. Apocolypse Sales in the center.
3. Apocolypse Store on the right.



**Step 3: Create Relationships**

Power BI often detects relationships automatically, but you can also create them manually.

The model now shows:

* One-to-Many (1:\*) between Customer Information → Apocolypse Sales.
* One-to-Many (1:\*) between Apocolypse Store → Apocolypse Sales.

This is a **star schema:** Sales is the fact table in the middle, connected to Customer and Store dimension tables.

**Tip for learners:** Always check that relationships match the real-world logic (e.g., one customer can place many sales, one store product can appear in many sales)

**Editing Relationships and Understanding Cardinality**

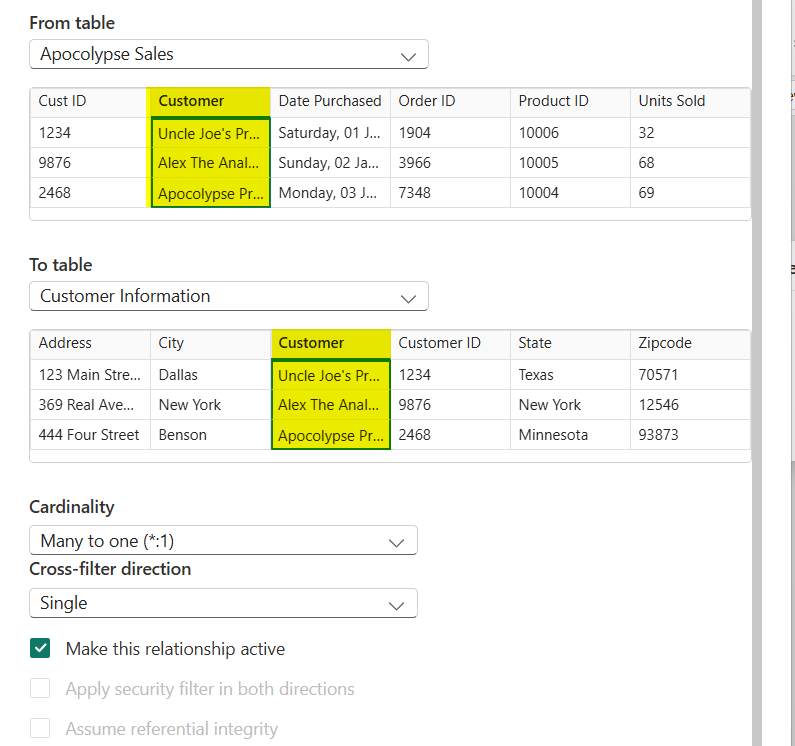
When Power BI detects relationships, it sometimes selects the wrong column to join tables. In this case, Power BI automatically used the Customer field to link the Apocolypse Sales and Customer Information tables. However, using names is risky because they can repeat or be misspelled.

**Step 1: Adjusting the Relationship Fields**

* Open the relationship settings by double-clicking the line connecting the two tables.
* Change the relationship so it uses **Customer ID** instead of Customer Name.
* Select **Cust ID** in Apocolypse Sales and **Customer ID** in Customer Information.
* Click **OK** to apply.

This ensures the relationship is based on a unique identifier (Customer ID), which is more reliable for analysis.

This ensures the relationship is based on a unique identifier (Customer ID), which is more reliable for analysis.



**Understanding Cardinality**

Cardinality defines how data in one table relates to data in another.

1. **One-to-Many (1:\*)** → Most common.
   * + - Example: One customer can make many sales.

* Power BI uses this when linking dimension tables (like Customer Information) to fact tables (like Apocolypse Sales).

1. **One-to-One (1:1)** → Used when both tables have unique matching records.

* Example: Employee table linked to Badge table, one record each.

1. **Many-to-One (\*:1)** → The same as One-to-Many but viewed from the opposite direction (depending on which table you start from).

1. **Many-to-Many (:)** → Both tables contain duplicate values for the key.

* Usually avoided because it can cause ambiguous results.

💡 Tip: Use One-to-Many relationships wherever possible for clean, reliable data models.

A screenshot of a computer

AI-generated content may be incorrect.

Understanding Cross Filter Direction (Example: Sales and Store)

Cross-filtering determines how filters flow between related tables in Power BI — it defines whether one table can influence another when you create visuals.

**Single Direction (Recommended Default)**

In a Single cross-filter, the filter flows from the “one” side to the “many” side of the relationship.

Example: Filtering by Customer Name affects Sales, but not vice versa.

Both Directions (Bidirectional Filtering)

Filters move both ways — each table can affect the other.

Useful for lookup or bridge tables but can slow performance or create ambiguity.

***Example:*** *When to Use Bidirectional Filtering*

**Aggregated Reporting Across Both Tables**

Suppose you have visuals showing:

* **Total Sales by Store Location**, and
* **Stores by Total Sales Performance** (e.g., Top 5 and Bottom 5 stores).

In this case, you want **filters to flow both ways** — from Sales to Store and from Store to Sales — so that:

* When a user clicks on a store in the Store Performance chart, the Sales by Location chart updates automatically.
* Likewise, when a user selects a region or sales record in the Sales chart, it highlights or filters the related store.

This bidirectional setup keeps both visuals synchronized and provides a more interactive dashboard experience.

Unit 5

Lesson 1

## How to use Dax in PowerBI

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* + Understand what DAX is, why it’s used in Power BI, and how to create basic measures and calculated columns using DAX formulas.

### **What is Dax?**

DAX (Data Analysis Expressions) is the formula language used in Power BI, Power Pivot, and Analysis Services. It allows you to create new information from your existing data model — such as totals, averages, counts, percentages, and more complex metrics.

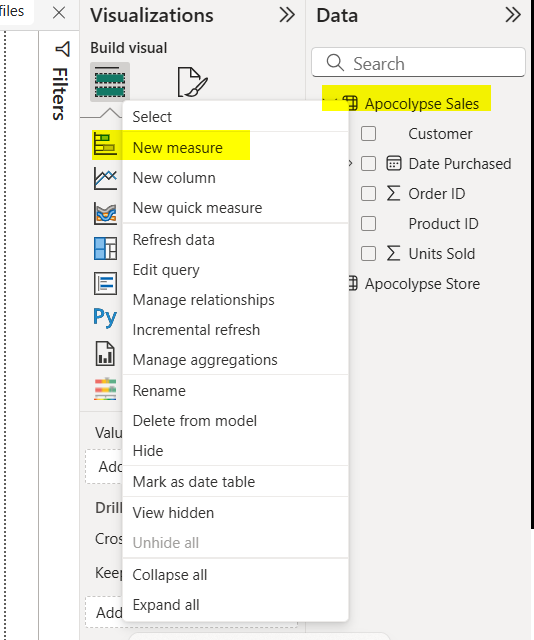
Think of DAX like Excel formulas but designed for relational data and filter contexts in Power BI.

**Basic DAX Functions Every Beginner Should Know**

|  |  |  |
| --- | --- | --- |
| Function | Description | Example |
| SUM() | Adds up all numbers in a column | SUM('Sales'[Revenue])` |
| AVERAGE() | Calculates the mean | AVERAGE('Sales'[Revenue]) |
| COUNT () | Counts number of rows with values | COUNT ('Sales'[Order ID]) |
| DISTINCTCOUNT() | Counts unique values | DISTINCTCOUNT('Sales'[Customer ID]) |
| MIN() / MAX() | Finds the smallest or largest value | MAX('Sales'[Price]) |
| DIVIDE() | Safe division (avoids divide by zero error) | `DIVIDE([Total Sales], [Total Orders])` |
|  |  |  |

**Step 1: Creating a New Measure**

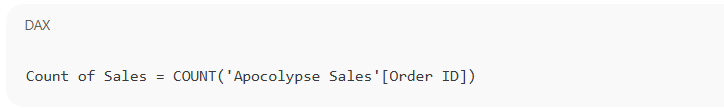
* 1. **Load the Dataset**
* Make sure the Apocolypse Sales and Apocolypse Store tables are already loaded into Power BI.
  1. **Open the Report View**
* On the left sidebar, click the Report icon (the one with the bar chart).
* This is where you build your visuals and create DAX calculations.
  1. **Select the Table for the Measure**
* In the Data pane, right-click on Apocolypse Sales.
* Choose New Measure from the dropdown menu (as shown in the screenshot).
  1. **A Formula Bar Appears**
* At the top of the report canvas, you’ll now see a formula bar.
* This is where you’ll type DAX formulas (Data Analysis Expressions) to create calculated metrics like totals, averages, and percentages.



**A screenshot of a computer

AI-generated content may be incorrect.**

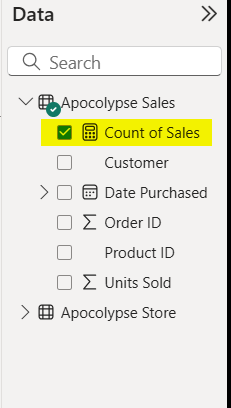
**Step 2: Writing Your First DAX Measure**



1. Press **Enter** to create the measure.

* Power BI will validate your formula and automatically add the new measure to the **Apocolypse Sales** table in the **Data pane.**

1. Notice that a **calculator icon** (🧮) appears next to your new measure — this indicates it’s a DAX measure rather than a column.
2. The name “Count of Sales” is just the label for your measure — you can rename it at any time to make it clearer.

**A screenshot of a computer

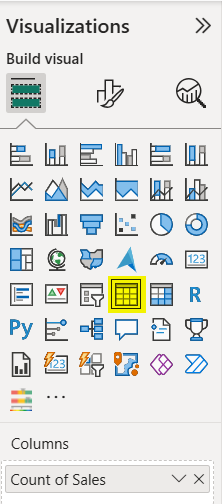
AI-generated content may be incorrect.**

Lesson: Wrangling

**Step 3: Visualizing the DAX Measure in a Table**

1. **Select the Table Visual**

* In the Visualizations pane, click on the Table icon (highlighted in the image).
* This will add a blank table visual to your report canvas.

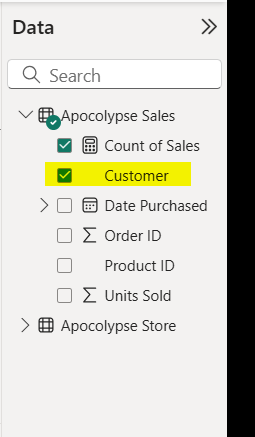
****

1. **Add Your Measure**

* Select the Count of Sales measure from the Apocolypse Sales table into the Values section of the visual.
* You’ll now see the total number of sales (in this case, 74) displayed in the table.

1. **Add a Dimension (Customer)**

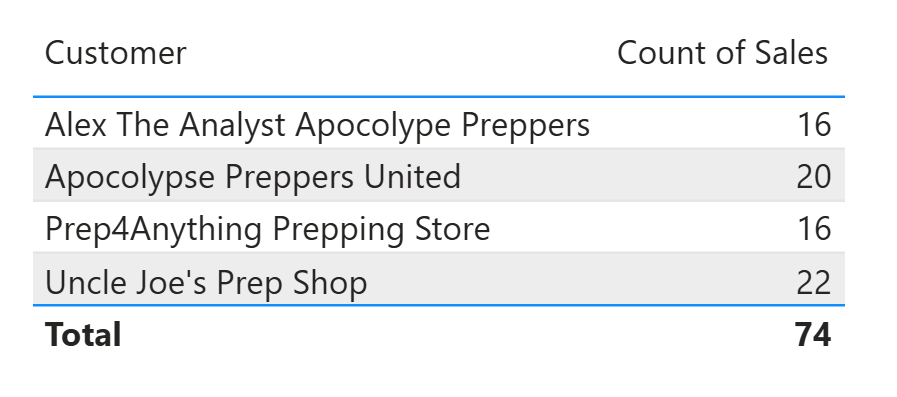
* In the Data pane, check the box next to Customer under Apocolypse Sales.

****

The visual will now display each customer alongside their corresponding count of sales.

1. **Analyze the Results**

* You can now see which customer made the most purchases.
* For example, in this case, Uncle Joe’s Prep Shop had the highest number of sales (22).

****

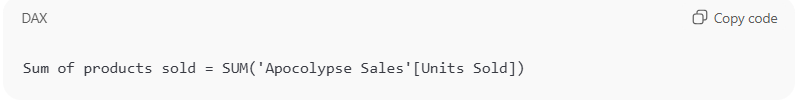
💡 Tip:

Table visuals are great for summarizing and comparing DAX measures across different categories (like customers, products, or regions).

**Step 4: Creating a SUM Measure for Total Products Sold**

1. **Create a New Measure**

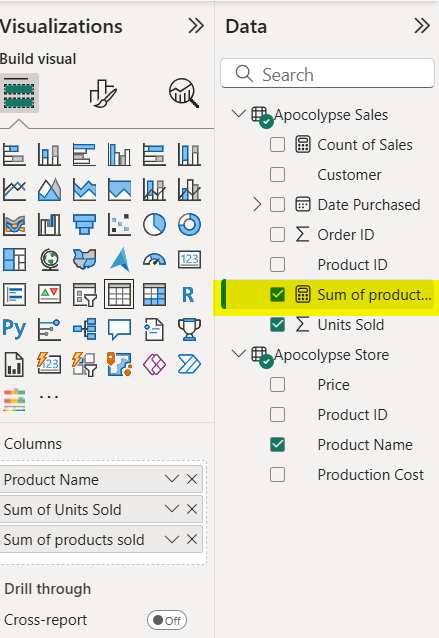
* Go to the Apocolypse Sales table again.
* Right-click and select New measure.
* In the formula bar, type the following DAX expression:



* Press **Enter** to create the measure.

1. **Add a Matrix Visual**

* In the Visualizations pane, select the Matrix icon.
* Select Product Name from the Apocolypse Store table into the Rows field.
* Then Select both Sum of Units Sold and Sum of products sold into the Values field.

****

1. **Analyze Your Matrix**

* The matrix now displays each product alongside its Sum of Units Sold and Sum of products sold.
* You can see the total at the bottom (in this example, 613 units in total).

**A screenshot of a graph

AI-generated content may be incorrect.**

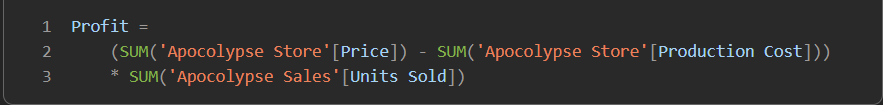
💡 Tip: Even though both columns show the same total here, this exercise demonstrates how to create and compare multiple measures — a core skill in DAX reporting.

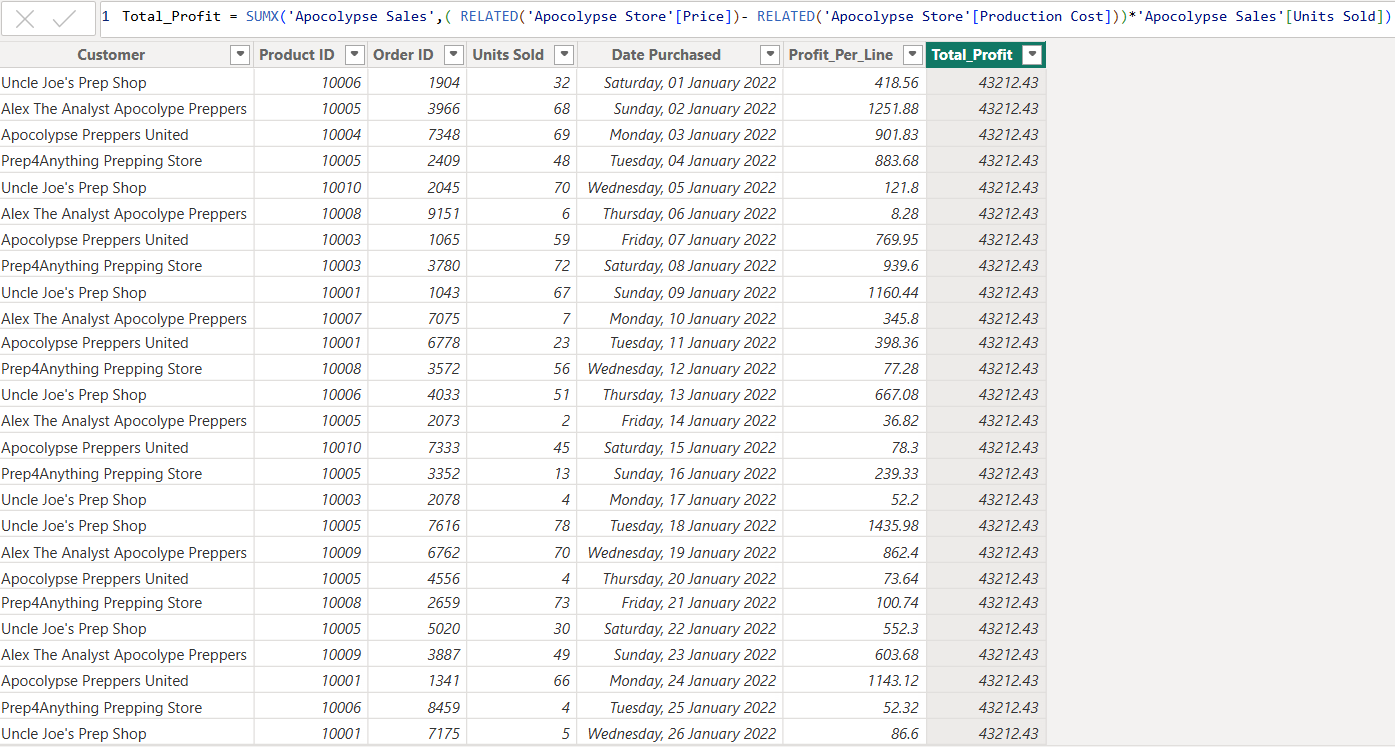
**We are now looking at the difference between Sum and Sumx**

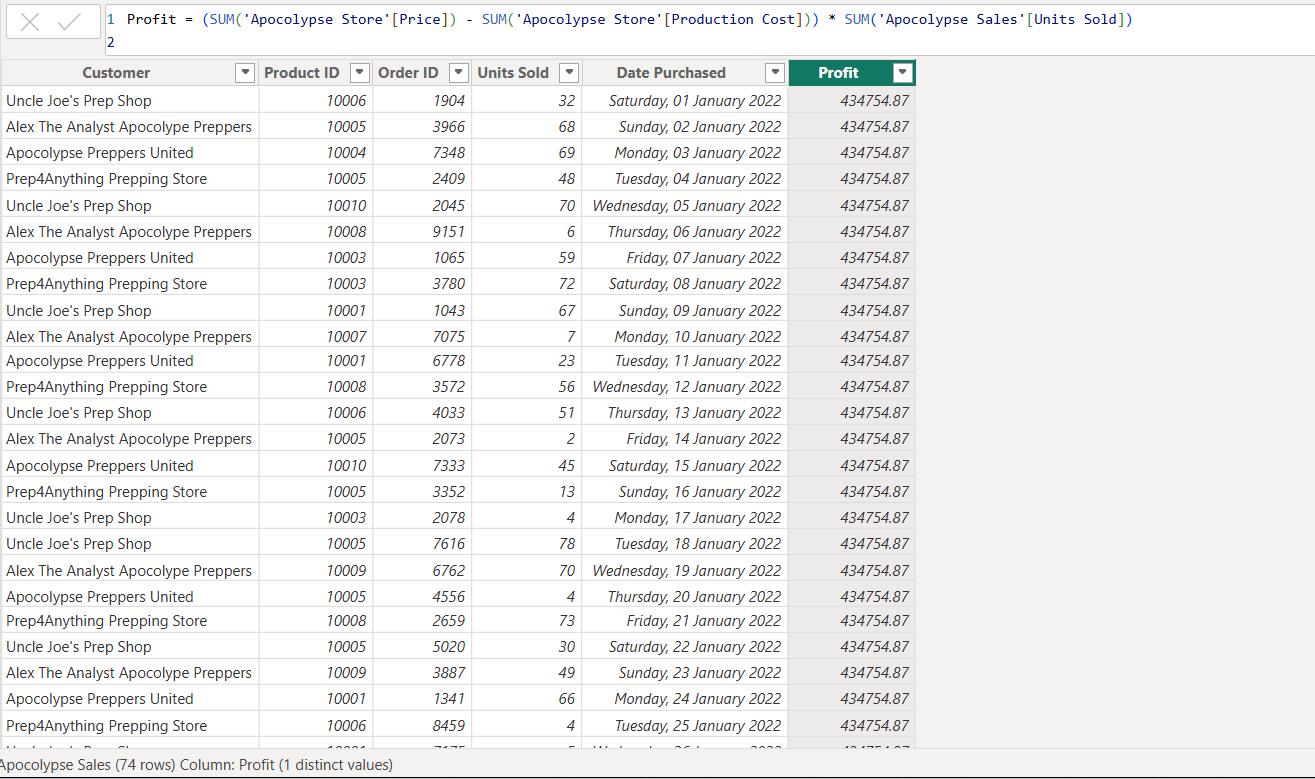
**Step 5: Profit per line (column) + Total Profit (SUMX measure)**

**A) Create a row-level profit column (in Apocolypse Sales)**

1. Go to **Data view** → select **Apocolypse Sales** → **New column**.
2. Name it **Profit** and use:





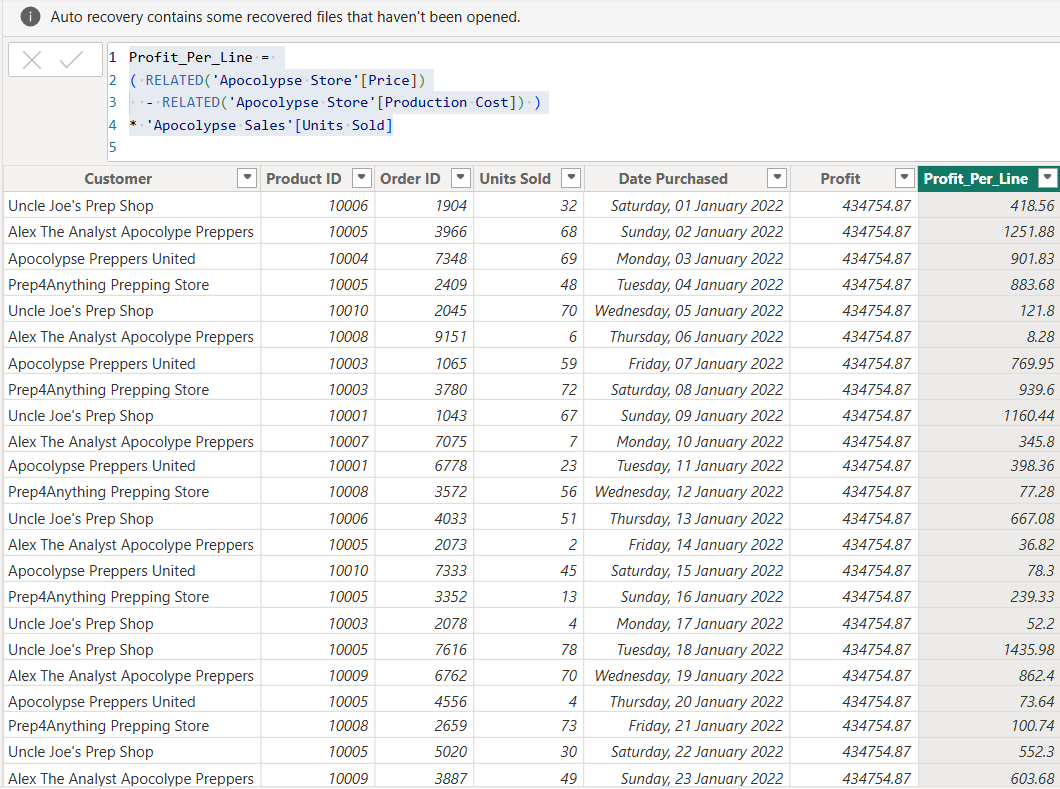


**B) Create a SUMX measure for total profit (filters aware)**

1. A screen shot of a computer

   AI-generated content may be incorrect.In **Apocolypse Sales**, **New measure** → name **Total Profit**:

* SUMX iterates the Sales table, calculating row profit in the current filter context and summing it.



🧮 **Difference Between SUM and SUMX in Power BI**

| **Function** | **What It Does** | **When to Use** | **Example** |
| --- | --- | --- | --- |
| **SUM** | Adds up **a single column** of numbers. It doesn’t evaluate any expressions per row — it just totals one field. | When you need a simple column total. | SUM('Apocolypse Sales'[Units Sold]) — totals all units sold. |
| **SUMX** | Works **row by row** across a table (like Excel’s =SUMPRODUCT) — it **evaluates an expression for each row**, then sums the results. | When you need to multiply or calculate something *before* summing (e.g., Price × Quantity, Profit per product). | SUMX('Apocolypse Sales', 'Apocolypse Sales'[Price] \* 'Apocolypse Sales'[Units Sold]) — calculates total sales amount. |

A yellow square with black corners

AI-generated content may be incorrect.LESSON SUMMARY

You should now be able to:

* SUM is **fast and simple**, but it only works on a **single column** — it doesn’t perform row-by-row calculations.
* SUMX is **more flexible**, allowing you to **evaluate an expression for each row** before summing

Unit 6

Lesson 1

## Creating Visualizations in Power BI

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* Understand what data visualizations are and why they’re essential in Power BI



### Introduction to Visualizations

Visualizations are the heart of Power BI — they turn raw data into meaningful insights that can be easily understood. Instead of scanning through rows of numbers, visuals help you see patterns, trends, and relationships at a glance.

With Power BI, you can create a wide variety of visuals such as bar charts, pie charts, tables, maps, and KPIs to tell your data story. Each visualization helps answer a different type of question — for example:

* Which products are performing best?
* How are sales trending over time?
* Which customers contribute the most revenue?

Creating effective visuals is not just about displaying data — it’s about communicating insights clearly and interactively. Power BI allows you to customize visuals, apply filters, and connect multiple visuals together to create a dynamic, story-driven dashboard.

**Creating a Stacked Bar Chart in Power BI**

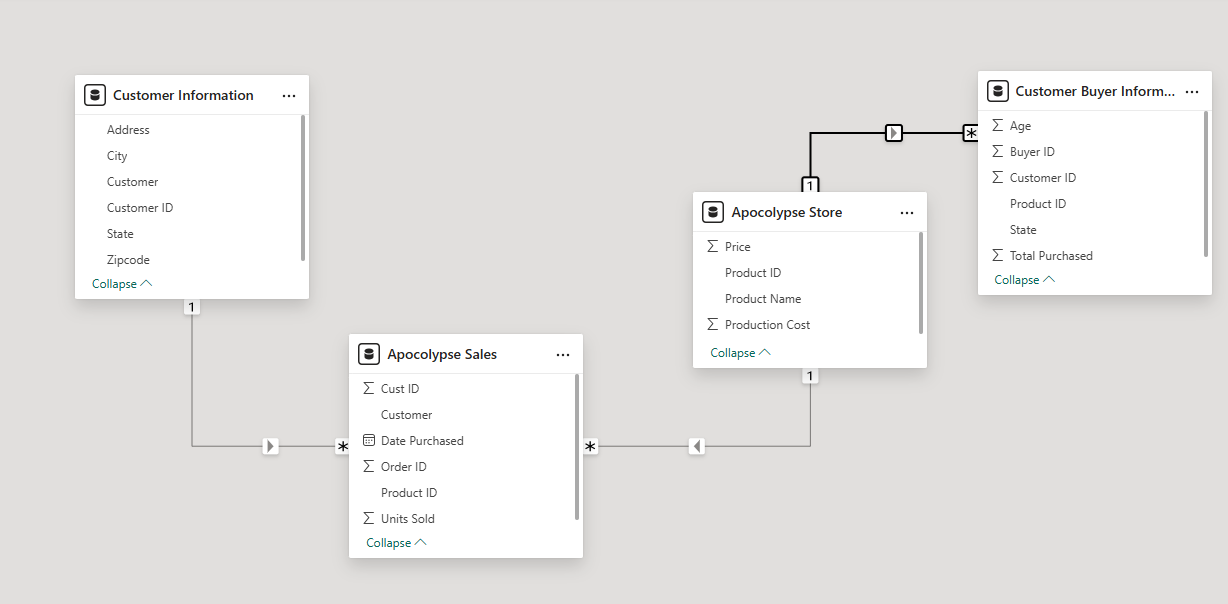
A stacked bar chart is one of the most popular visuals in Power BI. It allows you to compare values across categories and see how each part contributes to the total — all in a single view.

In this example, we’ll visualize the **total Units Sold** for each **Product Name** using a stacked bar chart.

🧭 **Step-by-Step Guide**

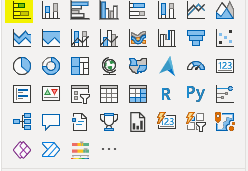
1. Open the Dataset
2. Open your Power BI file titled “**Apocalypse Food Prep – Visualizations Tutorial**”.
3. Tick All Columns
4. Make sure all columns from the dataset are selected so that Power BI recognizes all available fields.
5. Go to Model View
6. Select the Model View (relationship icon on the left sidebar).
7. Verify that all your tables are connected

You should see a relationship between Product ID in Apocalypse Store and Product ID Purchased in Customer Buyer Information.

****

**Building the Visualization**

* + 1. **Select a Stacked Bar Chart**
    2. In the **Visualizations pane**, click on **the Stacked Bar Chart icon.**

****

* + 1. Select Product Name (from Apocalypse Store) to the Y-axis.
    2. Select Units Sold (from Apocalypse Sales) to the X-axis.
    3. You now have a simple bar chart showing the Sum of Units Sold by Product Name.

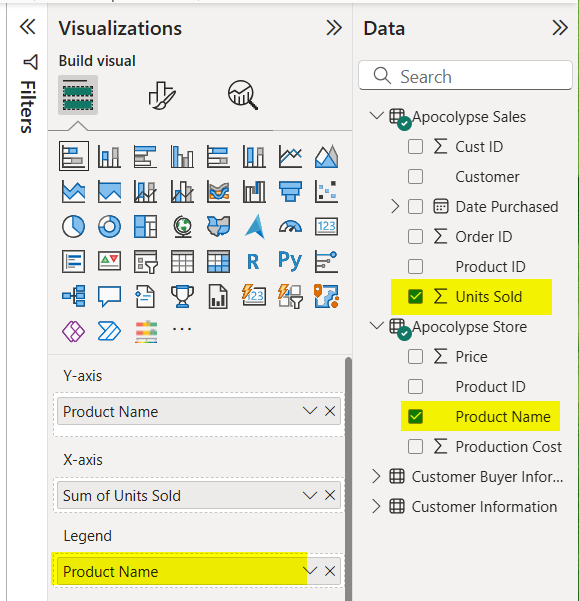
A graph with blue lines

AI-generated content may be incorrect.

**Add Color and Detail**

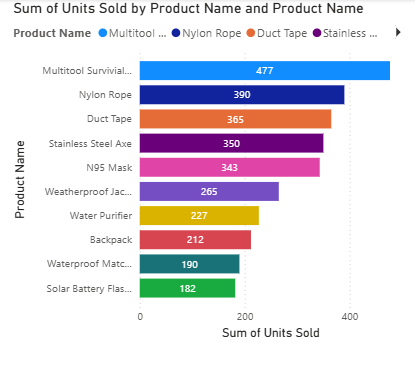
**Add a Legend**

1. To make the chart multi-colored, drag Product Name again to the **Legend section**.



1. Each bar now represents a product with its own color.
2. **Format the Chart**
3. Click the paint roller icon in the Visualizations pane to open the Format Your Visual options.
   * + 1. A yellow rectangle with a brush and a black text

          AI-generated content may be incorrect.
4. Scroll down and turn Data Labels → On.
5. Adjust label position, color, and font size for better readability.



**Section 2: Creating a 100% Stacked Column Chart**

A 100% stacked column chart allows you to compare the relative percentage contribution of each category to the total instead of showing raw numbers. It’s a great way to understand proportions and distribution within your dataset.

In this example, we’ll visualize the percentage of total units sold by each customer.

1. **Select the Chart Type**

From the Visualizations pane, choose the 100% Stacked Column Chart icon.

1. **Add Data Fields**
   * Under Customer Information, select Customer.
   * Under Apocalypse Sales, select Units Sold.

Power BI will automatically plot the Sum of Units Sold for each customer as a percentage of the total.

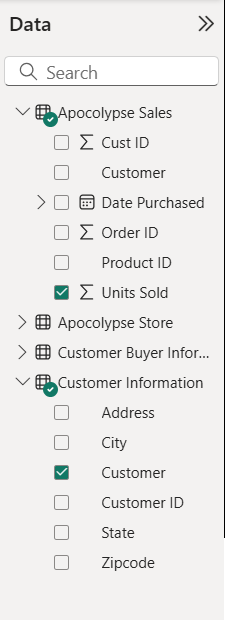
1. **Customize the Chart**

* Add a clear title, e.g., “Sum of Units Sold by Customer”.
* Adjust the colors and labels by selecting the paint roller icon to open the Format Visual pane.
* Turn Data Labels → On to display the percentage values directly on the bars.

**Result**

You’ll now see a visual that shows how much each customer contributes to total sales, expressed as a percentage.

This type of chart helps you quickly identify which customers make up the largest share of your total sales — providing insights into customer performance and contribution.

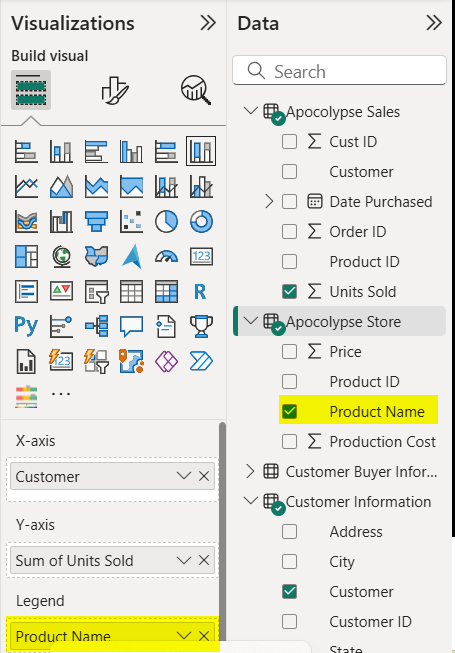
 A screenshot of a graph

AI-generated content may be incorrect.

1. **Add the Product Breakdown**

To see which products make up each customer’s total, drag **Product Name (from Apocalypse Store**) into the Legend area.

Each color in the bar now represents a different product — showing how much of each customer’s purchases come from that specific product.

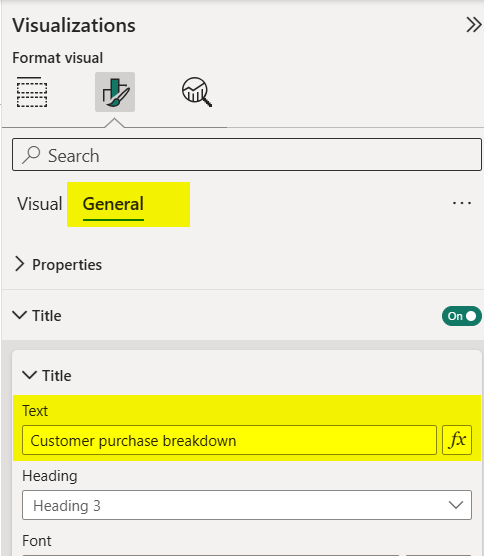


1. **Format Your Chart**

* Add Data Labels
* Click the chart.
* Go to the Format Visual tab (paint roller icon).
* Turn Data Labels → On.
* This displays percentage values for each product segment within the total.

1. **Edit the Chart Title**

* Under Visual → General → Title, type:
* Customer Purchase Breakdown
* Adjust Axis Titles (Optional)
* Under X-axis, rename the title to: Customer

A screenshot of a graph

AI-generated content may be incorrect.

## Learning Assessment

1. Models always contain data.

*Determine whether this statement is true or false.*

True

 False

1. What are types of models used in SAP Analytics Cloud? Note: there are two correct answers for this question.

*Choose the correct answers.*

* 1. Planning
  2. Story specific
  3. Analytic

 D Application specific

1. Data wrangling involves manipulating the data to make it accurate and to match your business needs.

*Determine whether this statement is true or false.*

True

 False

1. What are two possible types of dimensions in SAP Analytics Cloud?

*Choose the correct answers.*

* 1. Live
  2. Ad hoc
  3. Public
  4. Private

1. A private dimension can be shared by several models.

*Determine whether this statement is true or false.*

True

 False

1. What dimension types can you add when you create a model in SAP Analytics Cloud? Note: there are three correct answers for this question.

*Choose the correct answers.*

* 1. Account
  2. Version
  3. Generic
  4. Organization

 E Date

1. The Account dimension can only have one hierarchy.

*Determine whether this statement is true or false.*

True

 False

1. The New Model requires an Account dimension.

*Determine whether this statement is true or false.*

True

False

False

X

That's incorrect. Live models do not contain data.

1. What are types of models used in SAP Analytics Cloud? Note: there are two correct answers for this question.

*Choose the correct answers.*

* 1. Planning

X

* 1. Story specific
  2. Analytic

X

 D Application specific

That's correct! Planning model and Analytic model are model types used in SAP Analytics Cloud. Models are always public, they are not story or applicaton specific.

1. Data wrangling involves manipulating the data to make it accurate and to match your business needs.

*Determine whether this statement is true or false.*

True

X

 False

That's correct! Data wrangling involves manipulating the data to make it accurate and to match your business needs.

1. What are two possible types of dimensions in SAP Analytics Cloud?

*Choose the correct answers.*

* 1. Live
  2. Ad hoc
  3. Public

X

* 1. Private

X

That's correct! The two possible types of dimensions in SAP Analytics Cloud are Public and Private.

1. A private dimension can be shared by several models.

*Determine whether this statement is true or false.*

True

False

X

That's incorrect. Only public dimension can be shared by several models.

1. What dimension types can you add when you create a model in SAP Analytics Cloud? Note: there are three correct answers for this question.

*Choose the correct answers.*

* 1. Account

X

* 1. Version
  2. Generic

X

* 1. Organization

X

 E Date

That's correct! When creating a model, you can add account, generic, and organization dimension types. Version and date dimensions are added by the system.

1. The Account dimension can only have one hierarchy.

*Determine whether this statement is true or false.*

True

False

X

That's incorrect. The Account dimension can have multiple hierarchies.

Unit 2: Learning Assessment - Answers

1. The New Model requires an Account dimension.

*Determine whether this statement is true or false.*

True

False

X

Incorrect. The Account dimension is optional in the New Model.

Watermark Sample

Watermark Sample

**UNIT 3**

**Integrating SAP Analytic Cloud with SAP Systems**

Lesson 1

Integrating with SAP S/4HANA 55

Lesson 2

Integrating with SAP HANA Cloud 61

Lesson 3

Integrating with SAP BusinessObjects

Exercise 2: Create an Analytic Model from a Universe

66

71

Lesson 4

Integrating with SAP Data Warehouse Cloud 87

Lesson 5

Integrating with SAP Business Warehouse

Exercise 3: Access SAP BW Live Data

89

95

UNIT OBJECTIVES

* Access data from S/4HANA
* Access data from SAP HANA Cloud
* Access data from SAP BusinessObjects Universe
* Access data from SAP Data Warehouse Cloud
* Access data from SAP Business Warehouse

Watermark Sample

Watermark Sample

Unit 3

Lesson 1

## Integrating with SAP S/4HANA

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* Access data from S/4HANA

### SAP S/4HANA Integration

SAP S/4HANA is a complete enterprise resource planning (ERP) system with built-in intelligent technologies, including AI, machine learning, and advanced analytics. It helps companies adopt new business models, manage business change at speed, orchestrate internal and external resources, and use the predictive power of AI. With SAP Analytics Cloud's ability to connect to SAP S/4HANA, S/4HANA customers can leverage their on- premise system with SAP Analytics Cloud's cloud-based modeling and analytics capabilities.

The Integration with an S/4HANA system can be done in two ways:

* As a live data connection: no data will be stored on the cloud.
* As an import data connection: data will be stored in the cloud and updated periodically. NOTE: When using SAC as the interface for planning activities, data must be imported from S/4HANA.

SAP S/4HANA Live Data Connection

You can create live data connections to SAP S/4HANA systems using the Direct, Tunnel, and Cloud connection types. You can build stories based on released CDS views or leverage queries created by you.



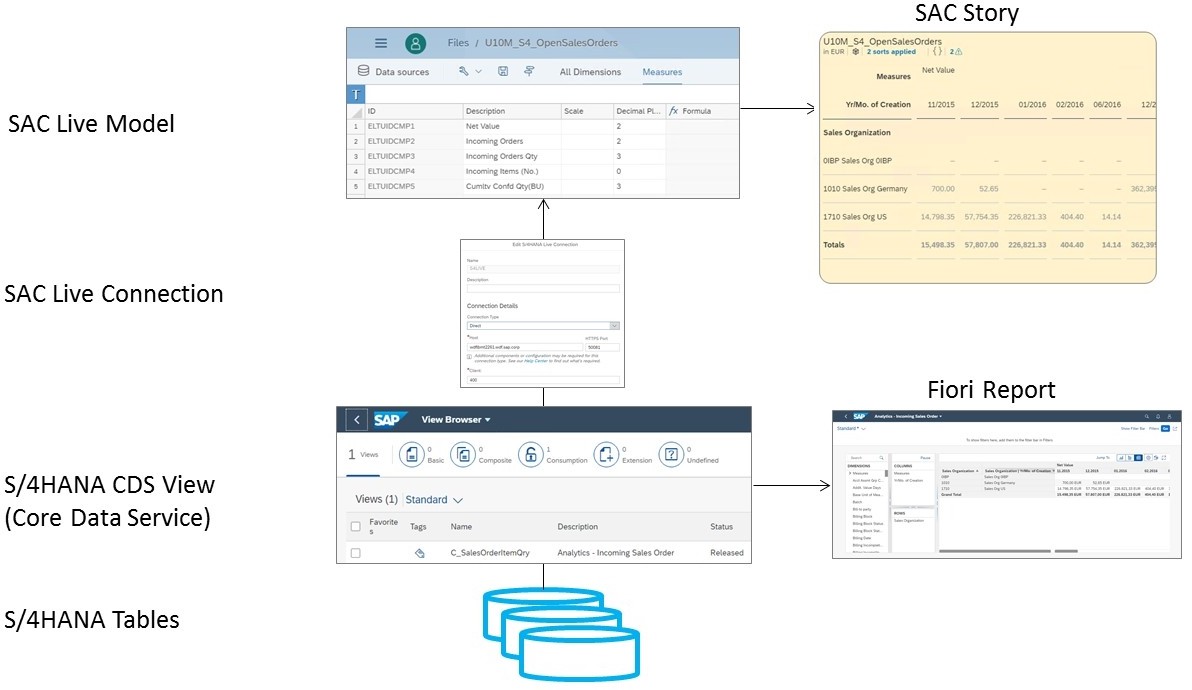


Figure 23: SAP S/4HANA Live Data Flow

Watermark Sample

In the figure *SAP S/4HANA Live Data Flow* we illustrate the data flow from S/4HANA to SAC. The *View Browser* Fiori app is used to access CDS views of all kinds.





Note:

Use the *Access Live SAP S/4HANA Data with a Model* procedure to access the View Browser app.

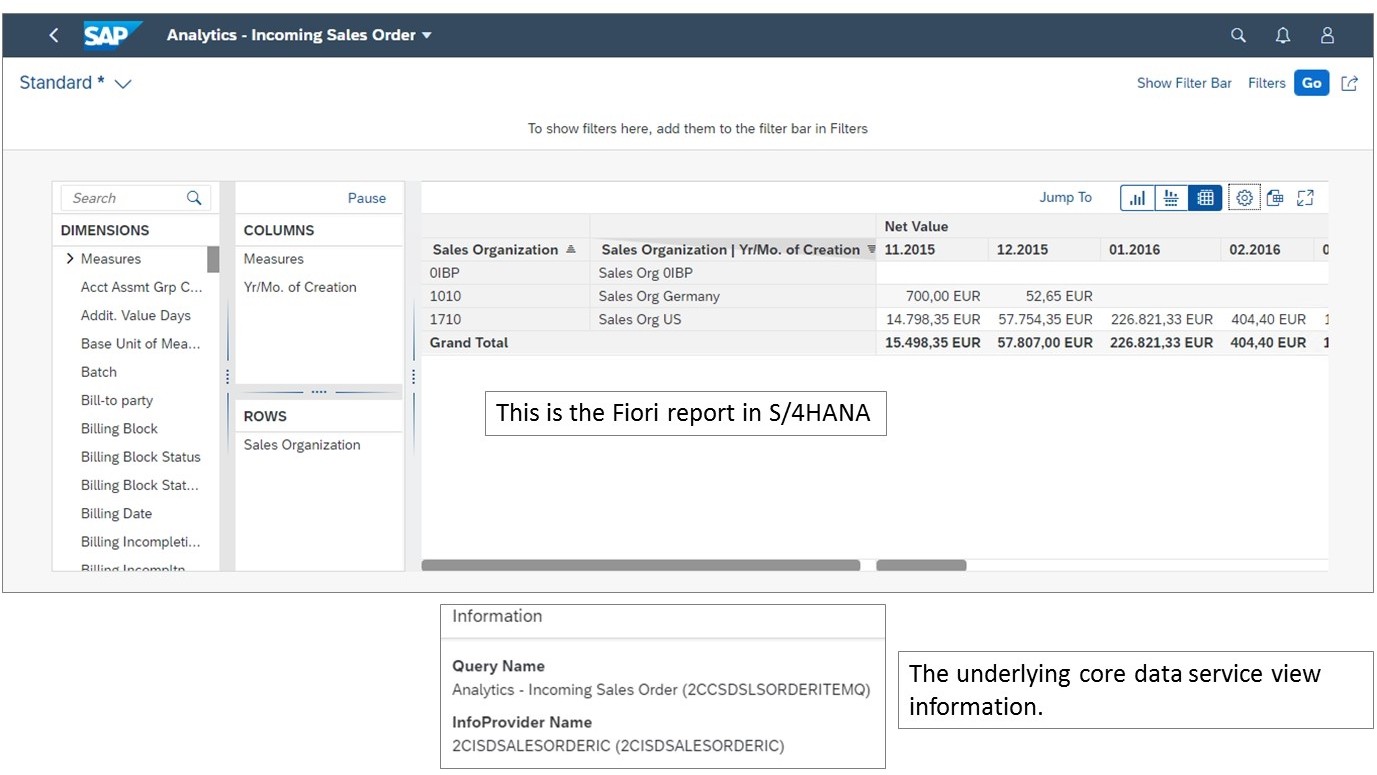


Figure 24: SAP S/4HANA Source View

Import Data Connection

Data can also be imported from SAP S/4HANA data into SAP Analytics Cloud. This import is necessary to carry out planning processes in SAC.

Importing data from SAP S/4HANA has the following prerequisites:

* On-Premise Edition is SAP S/4HANA 1610 with SAP NW release 7.51 SP2.
* SAP Analytics Cloud Edition is the latest version.

First, Master Data must be imported from S/4HANA to SAC, as illustrated in the figure *SAP S/4HANA Data Flow for Master Data*.

Watermark Sample



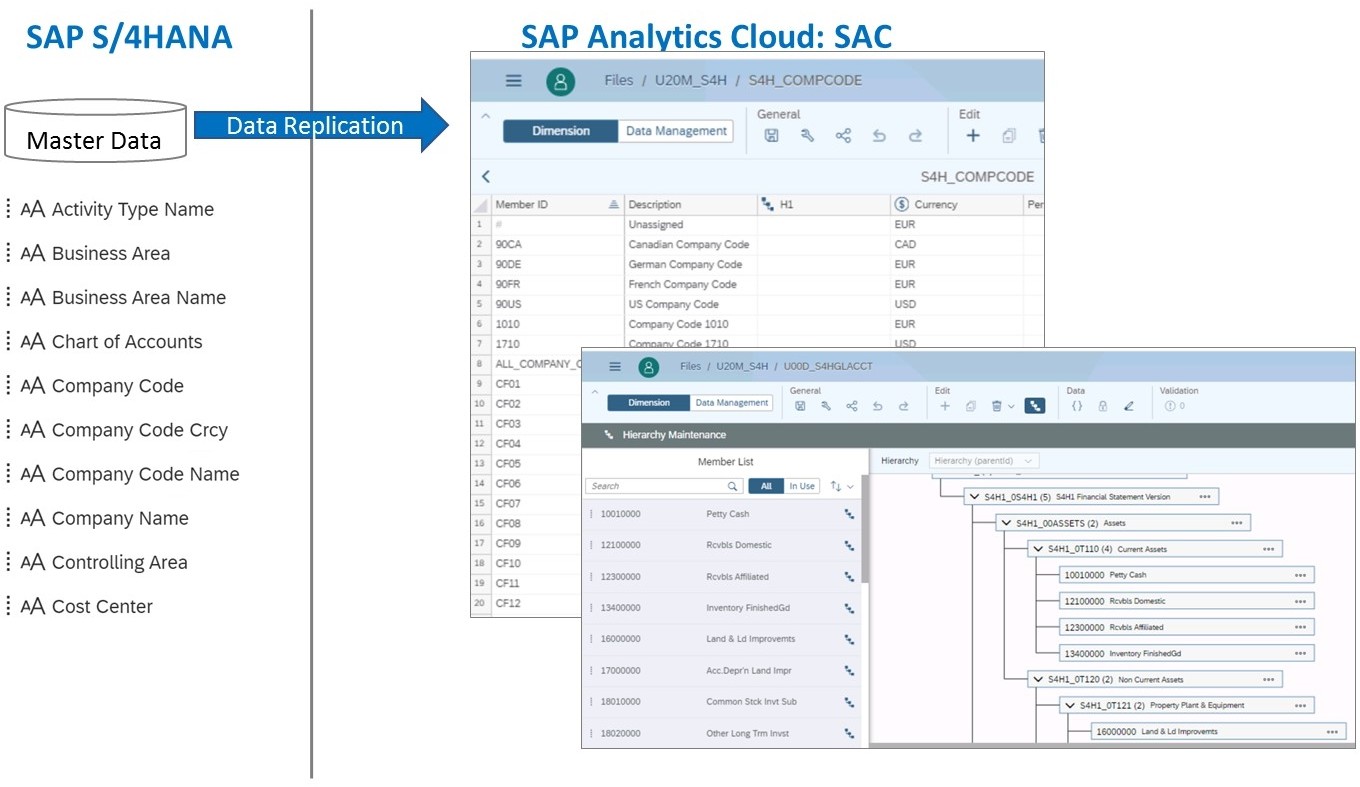


Figure 25: SAP S/4HANA Data Flow for Master Data

There are some unique considerations when importing Master Data from S/4HANA:

* + SAP S/4HANA has unique tables for master data. There are separate tables for cost centers, company codes, and GL accounts for example.
  + The master data includes member IDs, such as cash 10010000 accounts, descriptions like cash equivalent, properties such as balance sheet type accounts, and hierarchies. These properties may need to be imported separately from the ID/description data.
  + If you need to import SAP S/4HANA hierarchies, you must first import all the members of the hierarchy, then import the hierarchy nodes and leaves, and finally, import the hierarchy node text.
  + Setting up the import process is a one time event.
  + The import process can be scheduled to run periodically after the initial set up.
  + Not all SAP Analytics Cloud dimensional data needs to be imported. It can also be maintained in SAP Analytics Cloud, for example, in cases where the data is static.

Watermark Sample

### Data Analyzer



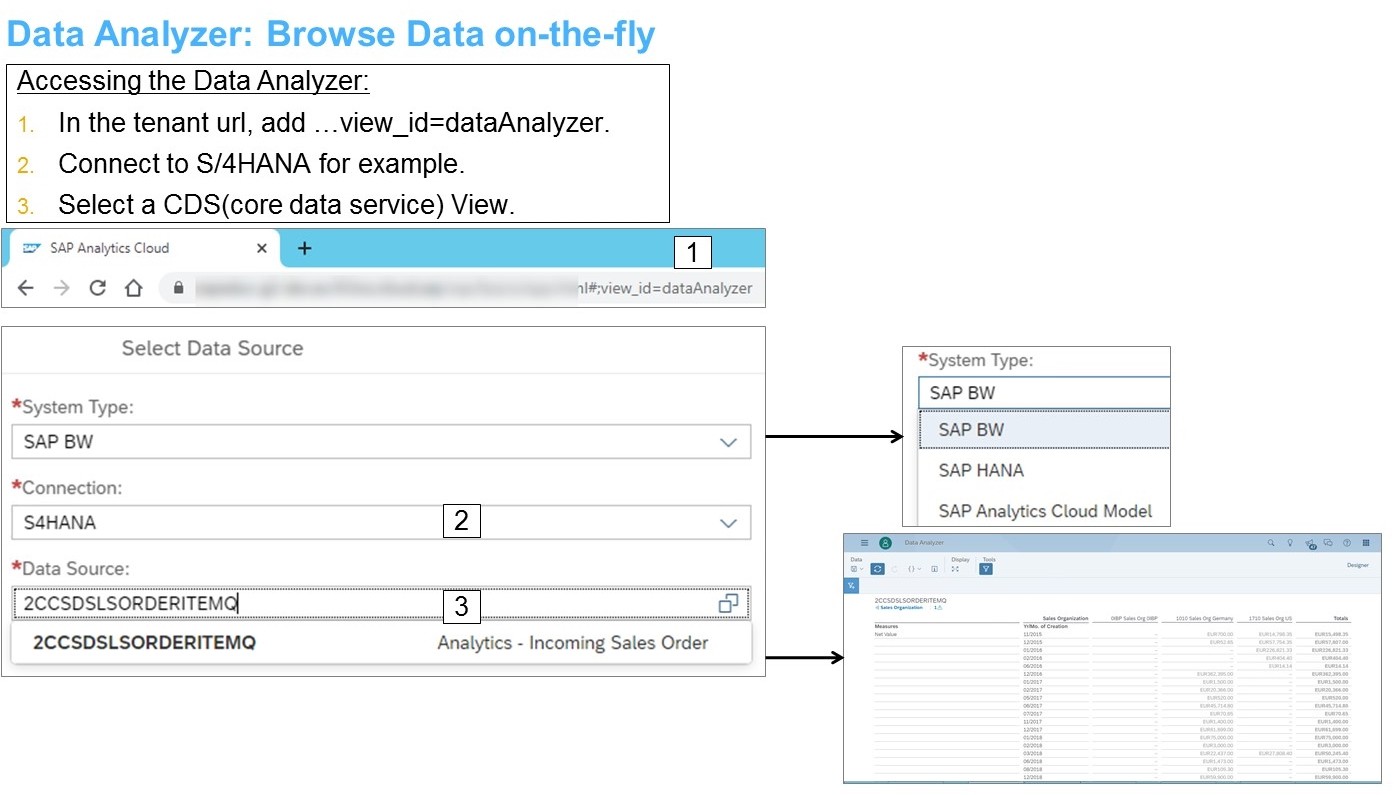


Figure 26: Data Analyzer - Access

Data analyzer is a predefined ready-to-run service for SAP Business Warehouse (SAP BW) queries, SAP HANA Live views, and SAP Analytics Cloud models for ad-hoc analysis. It uses SAP BW live connections, SAP HANA views and models created in SAP Analytics Cloud. All SAP BW queries, SAP HANA Live views and models can be accessed directly in the Select Data Source dialog and no additional model is created.

Data Analyzer - User Interface



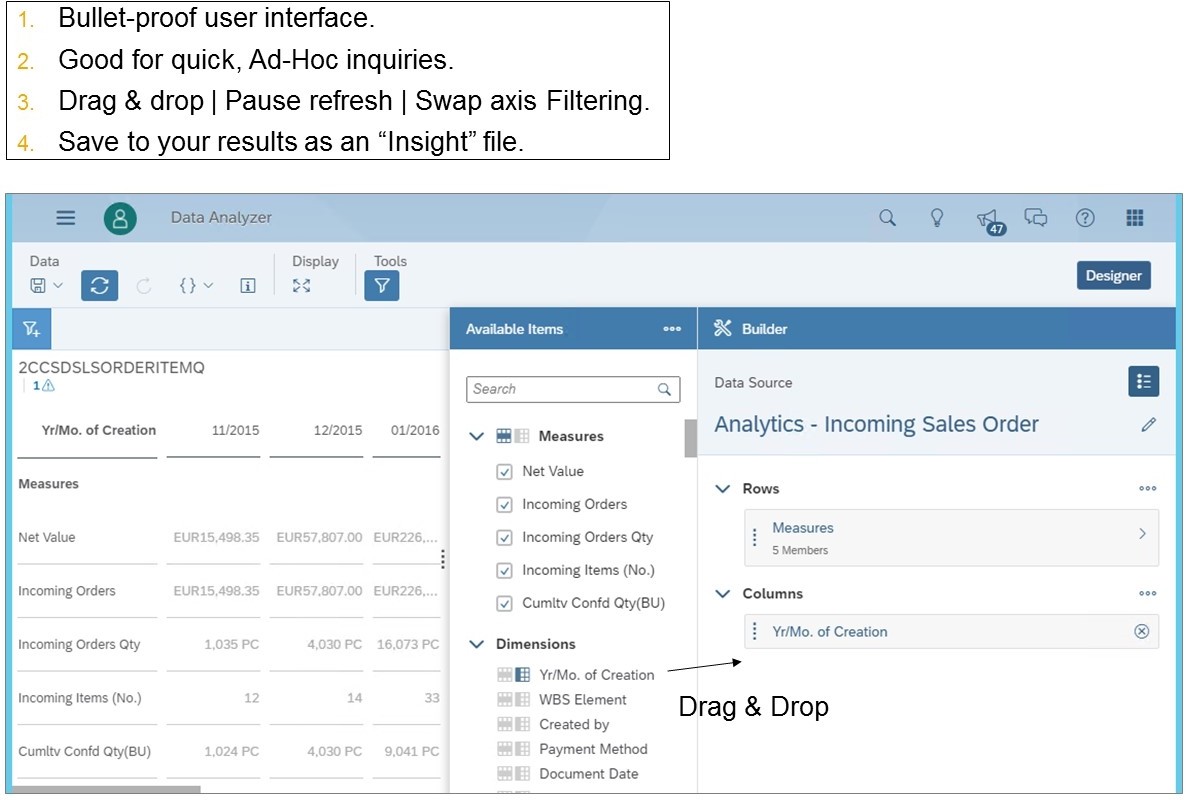


Figure 27: Data Analyzer - User Interface

Data analyzer contains a table, a filter area, and a builder panel with navigation capabilities to add and remove dimensions and measures from the table. In addition, you find a menu bar

Watermark Sample

with a Refresh option and an Edit Prompts dialog (in case your datasource is designed for setting prompts).

Table formulas are available but calculated measures are not.

After you have drilled down to the data details according to your needs and analyzed your data, you might want to save this insight. For this, choose Save in the upper left corner of data analyzer. In the Save Insight dialog, select the file location (public or private) for your insight and enter a name and description for it.



Note:

You can also test a CDS view from S/4HANA. In the data analyzer, connect to BW

→ S/4 → Log in with wss4 / Welcome1 → search for journal → If prompted, use company code 1010. In Fiori, this is the Journal Entry Analyzer. Launch Fiori from the SACP30 folder in the N drive with user s4h00-00 / Welcome1. It can be accessed with the Query Browser. The technical name of the view is C\_GLLineItemsQ0001.

To Access Live SAP S/4HANA Data with a Model

You need to access live data from SAP S/4HANA using a delivered core data services view.

1. Navigate to *Application (N:)* → *SAC* → *SACP30* and double-click the *Fiori\_Launchpad*

shortcut.

1. If necessary, log on as follows:

|  |  |
| --- | --- |
| Field | Value |
| User: | **SACP30-##** |
| Password: | **Welcome1** |

1. In the upper right corner, choose the  down arrow and choose *Query Design*.
2. In the main workspace area, select the *View Browser* tile and search for

**C\_SalesOrderItemQry**.

1. On the far right, choose the *>* button and then choose *Show Content*. The query runs and prompts for an Exchange Rate and Display Currency.
2. In the *Display Currency* field, enter **EUR**. Note the data results in the table.
3. Choose the *SAC* tab in Chrome to return to SAP Analytics Cloud.
4. Create a model from a live datasource.
   1. From the Navigation Bar choose  *Modeler*.
   2. Choose *From a Data Source*.
5. Configure the live data connection to the query using the following information:

|  |  |
| --- | --- |
| Field | Value |
| System Type: | SAP BW |

Watermark Sample

|  |  |
| --- | --- |
| Field | Value |
| Connection: | SAP S/4HANA if necessary, log on with the following credentials:  User: **WSS4H** Password: **Welcome1**  | |
| Datasource: | **2CCSDSLSORDERITEMQ** |

1. Choose *Live Data connection* on the right, under *Connect to live data*.
2. Configure the information above and choose *OK*.
3. Save the model as **U##M\_S4\_OpenSalesOrders**.
4. Create a story using the new model with a table on a canvas page. Use **EUR** as the variable value for *Display Currency*.
   1. From the Navigation Bar choose *Stories*.
   2. Choose *Add a Canvas Page*.
   3. Choose *Table*.
   4. In the *Select Model* dialog, choose *Select other model* and choose your new

*U##M\_S4\_OpenSalesOrders* model.

* 1. Enter **EUR** for the *Display Currency* variable and choose *Set*. Optionally, you can choose the *Automatically open prompt when story opens* check box.

You see the same data in the table as in the query preview from Step 6. You can use the model in a story as with any other model, including adding calculated measures.

LESSON SUMMARY

You should now be able to:

* Access data from S/4HANA

Watermark Sample

Unit 3

Lesson 2

## Integrating with SAP HANA Cloud

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* + Access data from SAP HANA Cloud

### SAP HANA Cloud Integration Scenario

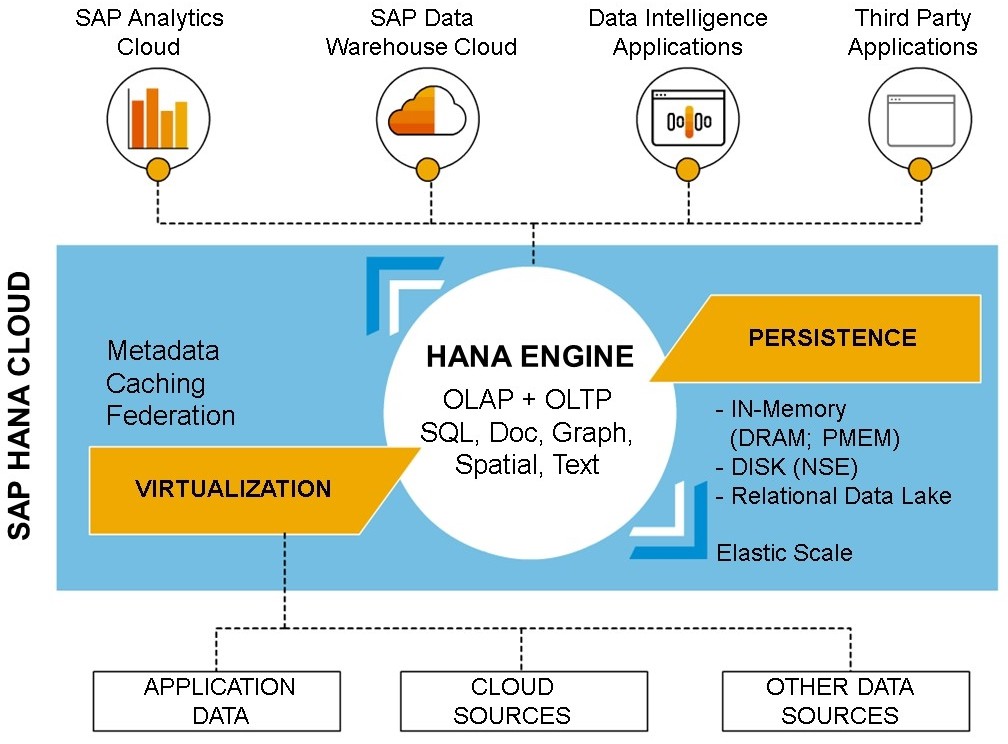
SAP HANA Cloud is a database-as-a-service that offers the power and performance of SAP HANA natively in the cloud, including full capabilities to process various data types, federate data storage, and run powerful applications. It offers a relational data lake for large-scale data storage and processing. Using SAP HANA Cloud, you can set up and run a cloud-based database service easily, and bind it to applications running on SAP BTP or elsewhere. You can also access SAP HANA Cloud using a variety of programming languages and interfaces, and build applications and data models.





Figure 28: SAP HANA Cloud Overview

### SAP HANA Cloud Live Data Connection

In SAP Analytics Cloud, you can create live data connections to SAP HANA Cloud using Live Connection with Connection Type “SAP HANA Cloud”. SAP Analytics Cloud can delegate model management to SAP HANA Cloud by building SAC models using HANA Calculation Views as a data source. Calculation Views allows users to define more advanced slices on the data available in the SAP HANA database. As a live data connection, no data will be stored on the cloud.

Watermark Sample





Figure 29: SAP HANA Cloud Hybrid Deployment

### Use Cases for SAP HANA Cloud

SAP HANA Cloud is central to SAP’s strategy of providing a next-generation digital platform that can power both existing and new applications, either on-premise or cloud or hybrid.

These applications can be either SAP or non-SAP applications.

There are many use cases and scenarios for SAP HANA Cloud. These include the following:

* To provide the database for next generation applications that require super-fast performance on huge data volumes at scale.
* To power data warehouses including SAP Data Warehouse Cloud and custom data warehouses. Also, to power analytics, including SAP Analytics Cloud, by connecting to, and coordinating distributed data into a single, consistent data model.
* To extend the data storage and processing capacities of on-premise applications, for example when large numbers of new users are on-boarded.
* To extend the functionality of existing SAP applications using cloud-based services.

### SAP HANA Cloud in a Hybrid Landscape

SAP HANA Cloud can play a key role in a hybrid landscape. As customers move more of their applications to the cloud, they often continue to run on-premise applications with their on- premise databases, side-by-side.

Watermark Sample



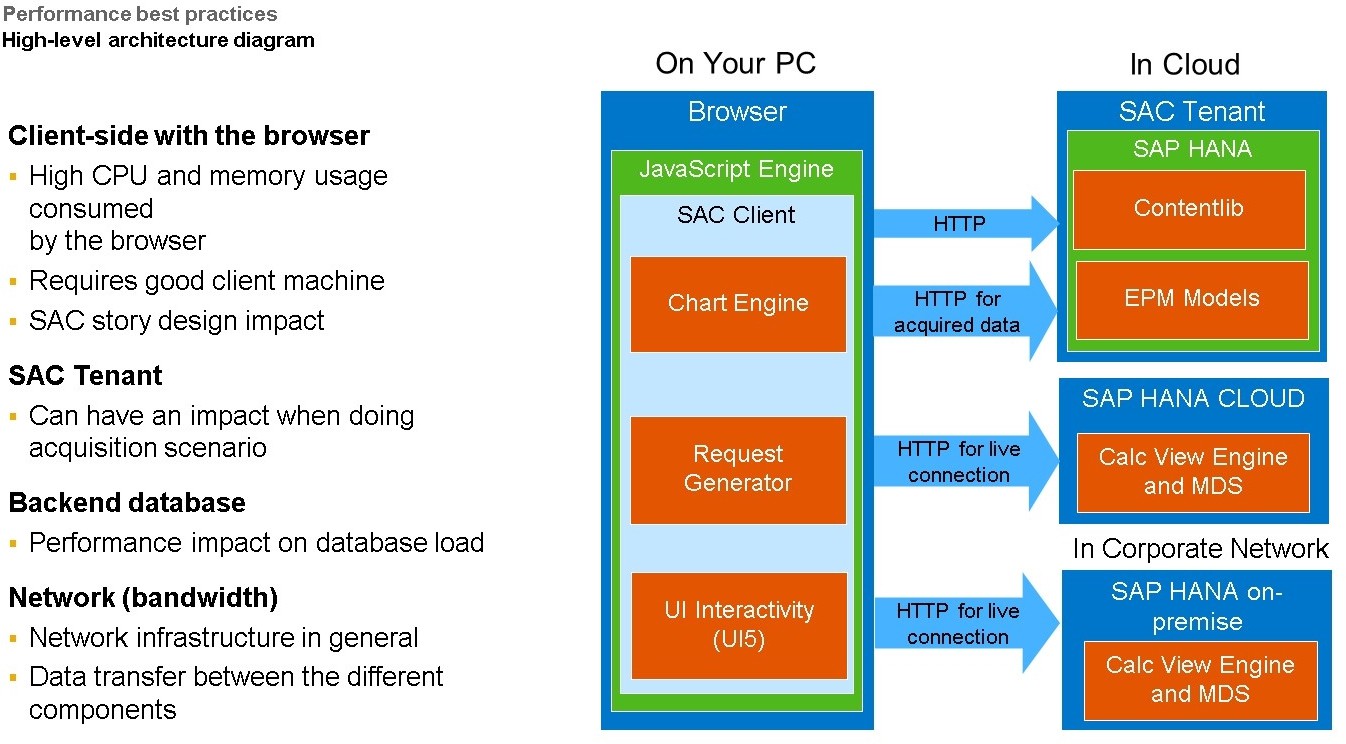


Figure 30: HANA as a backend database

### SAP HANA Multi-Dimensional Services

SAP HANA Multi-Dimensional Services (MDS) is used to process multidimensional queries including aggregation, transformation and calculations in SAP Analytics Cloud, SAP HANA Cloud and SAP HANA on-premise.

SAP Analytics Cloud(SAC) models based on HANA Calculation View can improve Stories performance by delegating and moving data intensive calculations to SAP HANA Cloud. In this case, SAP Analytics Cloud is the application layer in the Future approach scenario as specified in diagram. By pushing down data intensive calculations to the SAP HANA Cloud and only returning the result, this would optimized the performance of the SAC Tenant.

To Access Live SAP HANA Cloud Data



Note:

This procedure is only for the instructor. (SSG setup instructions required prior to demonstration)

You need to access live data from SAP HANA Cloud using a Calculation View.

* + 1. Start the *SACDEMO* instance in *HANA Cloud Central*.
       1. Open Google Chrome and enter the *HANA Cloud Central* URL: https://hana- cockpit.cfapps.eu11.hana.ondemand.com/hcs/sap/hana/cloud/index.html#/org/ 3e841fc3-64b2-4651-b26f-d505a5ab9d19/space/1b1f3f94-1547-4be4-

be3c-6c86d0a426e3/databases? databaseguid=32c217f9-64c8-4417-815d-527939a69c61

* + - 1. Sign in to the default identity provider with the User and Password as specified in the SSG.
      2. The SACDEMO should be running. If not, choose ...Actions → Choose Start.
    1. Access the Business Application Studio.

Watermark Sample

1. Open a new Google Chrome window, enter URL: https:// hana4sac.eu11cf.applicationstudio.cloud.sap/index.html
2. Next to SACMS1\_DEV, choose  *Start* → Wait a few seconds until it is *RUNNING*.
   * 1. Click on the *SACMS1\_DEV* space.
     2. From the *WORKSPACE* on the left, navigate: db → scr → Calculation View.
     3. Double-click *CV\_inventory\_analysis.hdbcalculationview*.
     4. Right-click the Aggregation node → select *Data Preview*.

|  |
| --- |
|  |

|  |
| --- |
| Note:  If you see an error in the Business Application Studio: *Failed to get Cloud Foundry token. Please log back in to re-authenticate and reopen the editor.* On the lower right, choose the hyperlink: *Log in to Cloud Foundry.* The link https:// api.cr.eu11.hana.ondemand.com appears. Press Enter. Enter user and password provided in the SSG. Select the default organization *KTE\_HANA\_hana4sac* and space *SACDC*. Re-open the *Business Application Studio* editor. Re-open the *Business Application Studio* editor. |

* + 1. Now it is time to create an SAC Model to access this data. Go to the SAC browser with the A00 user and password provided in the SSG.
    2. On the upper left, choose the *Expand Navigation Bar* icon.
    3. Create a model from a live data source.
       1. From the Navigation Bar choose *Modeler* icon.
       2. Choose *Live Data Model*.
    4. Select the live data connection and inventory calculation view.
       1. Select System Type *SAP HANA* → Select Connection *HANACLOUD*.
       2. Select Data Source: *CV\_inventory\_analysis*.

Watermark Sample

|  |
| --- |
| Note:  If the underlying calculation view contains a time field, the time dimension in SAC can be configured and thereby allow automatic time determination in stories. If the underlying calculation view contains a latitude and longitude field, the location dimension in SAC can be configured and thereby allow geo mapping. |

* + 1. Save the model as **U##M\_SHC\_Inventory\_Analysis**.
    2. Create a story with a table to view the data.
       1. Go to Actions → Story → Canvas.
       2. Choose Table.
       3. Add Product to the rows.

|  |
| --- |
|  |

* + 1. Close the extra tabs for SAP HANA Cloud.

LESSON SUMMARY

You should now be able to:

* Access data from SAP HANA Cloud

Watermark Sample

Unit 3

Lesson 3

## Integrating with SAP BusinessObjects

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* Access data from SAP BusinessObjects Universe

### SAP BusinessObjects Integration Scenario

The SAP BusinessObjects Business Intelligence platform is a flexible and scalable solution for delivering information to end users, in multiple forms including dashboards and interactive reports, via any web application—intranet, extranet, Internet, or corporate portal. The platform delivers tangible benefits extending across and beyond the organization, as an integrated suite for reporting, analysis, and information delivery. It also provides a solution for increasing end-user productivity and reducing administrative efforts. For example, it is used to distribute weekly sales reports, to provide customers with personalized service offerings, or to integrate critical information in corporate portals.

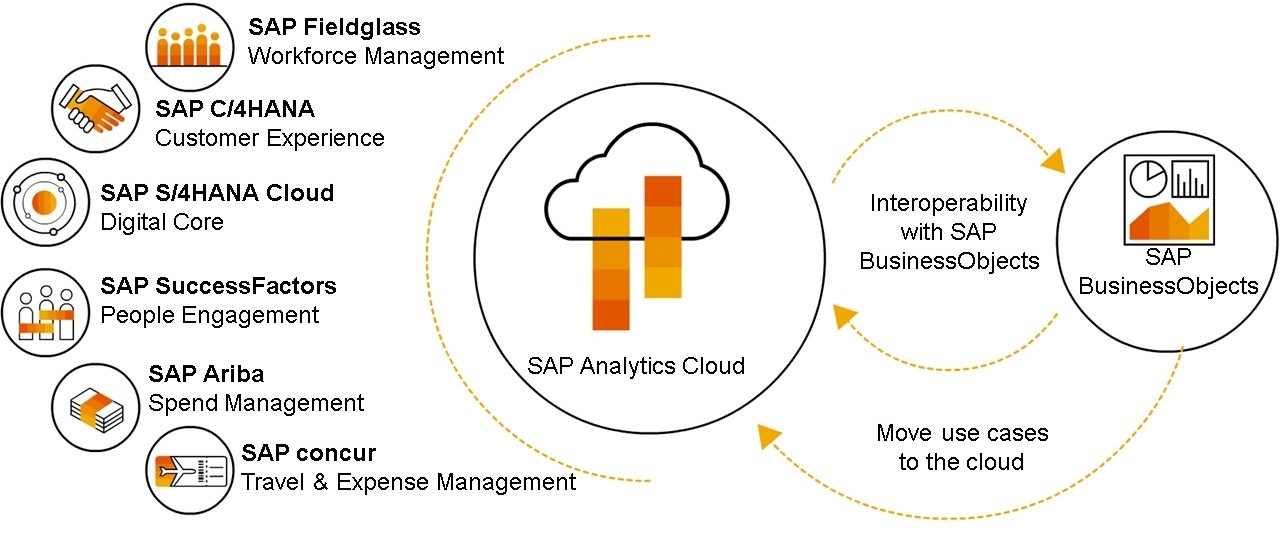
SAP BusinessObjects Business Intelligence on-premise solutions work with SAP Analytics Cloud in a Hybrid Analytics scenario. In the long term, SAP has strategically positioned SAP Analytics Cloud as the Augmented Analytics tool. Meanwhile, SAP continues to deliver innovations with SAP BusinessObjects 4.3 as part of a hybrid Analytics configuration. This configuration allows SAP Analytics Cloud to access on-premise Universes and Web Intelligence Documents as a data source.





Figure 31: SAP BusinessObjects Integration scenario

### SAP BusinessObjects Enterprise Live Data Connection

SAP BOE Live Data Connect enables a live connection to on-premise data sources and allows customers to reuse existing data, queries, and security without moving the data to cloud. BI

4.3 allows customers to use both universes and Web Intelligence data models to build SAC Stories.

Watermark Sample

### SAP BusinessObjects, Private Cloud Edition

SAP will continue to enhance SAP BusinessObjects BI Suite (BI 4.3) in the subsequent releases until the end of 2027, providing one of the longest maintenance timelines in the industry. The product will then evolve into a managed cloud subscription only, as part of the SAP BusinessObjects Enterprise, Private Cloud Edition (BOE PCE). BOE PCE allows customer options to shift their on-premise workloads to a managed cloud environment and will continue to evolve and be supported beyond 2027. In addition, BOE PCE also provides the option to leverage existing on-premise investments from the managed cloud environment:

SAP BusinessObjects BI 4.3 allows administrators to easily onboard on-premise users to the cloud and to keep them synchronized with their identity provisioning system.

### Live Data Connection - BOE PCE Solution

The Live Data Connection for BusinessObjects Enterprise, Private Cloud Edition includes Universes, Calculation Views, ABAP CDS Views, and BW queries. To support hybrid scenarios, Live Data connection connects to both on-premise as well as cloud data.

BOE PCE runs on the same software used by on-premise BI 4.3 with identical user interfaces, but runs as a managed private cloud solution hosted by SAP or Hyperscaler partners (AWS, AZURE, GCP). To connect SAP Analytics Cloud to BusinessObjects Private Cloud Edition the following components are required to be deployed in managed cloud environments:

* The SAP Cloud Connector
* The SAC Agent
* BusinessObjects Live Data Connect

These required components are now included as an optional deployment with the Private Cloud Edition SKUs. They are considered optional and not provisioned by default as some customers may have them deployed as part of SAP S/4HANA. The SAP Cloud Architect and Advisor (CAA) provides the technical requirements to each BOE PCE solution.

### Live Data Connection - SAP Universes

Live Data connection can connect to a SAP Business Objects Universe model and be used to create stories.

Universe is a an organized collection of metadata objects that enable business users to analyze and report on corporate data in a nontechnical language. These objects include dimensions, measures, and attributes. The metadata object layer, called the business layer, is built on a relational database schema or an OLAP cube. The objects map directly to the database structures via SQL or MDX expressions. A universe includes connections identifying the data sources so queries can be run on the data.

Watermark Sample



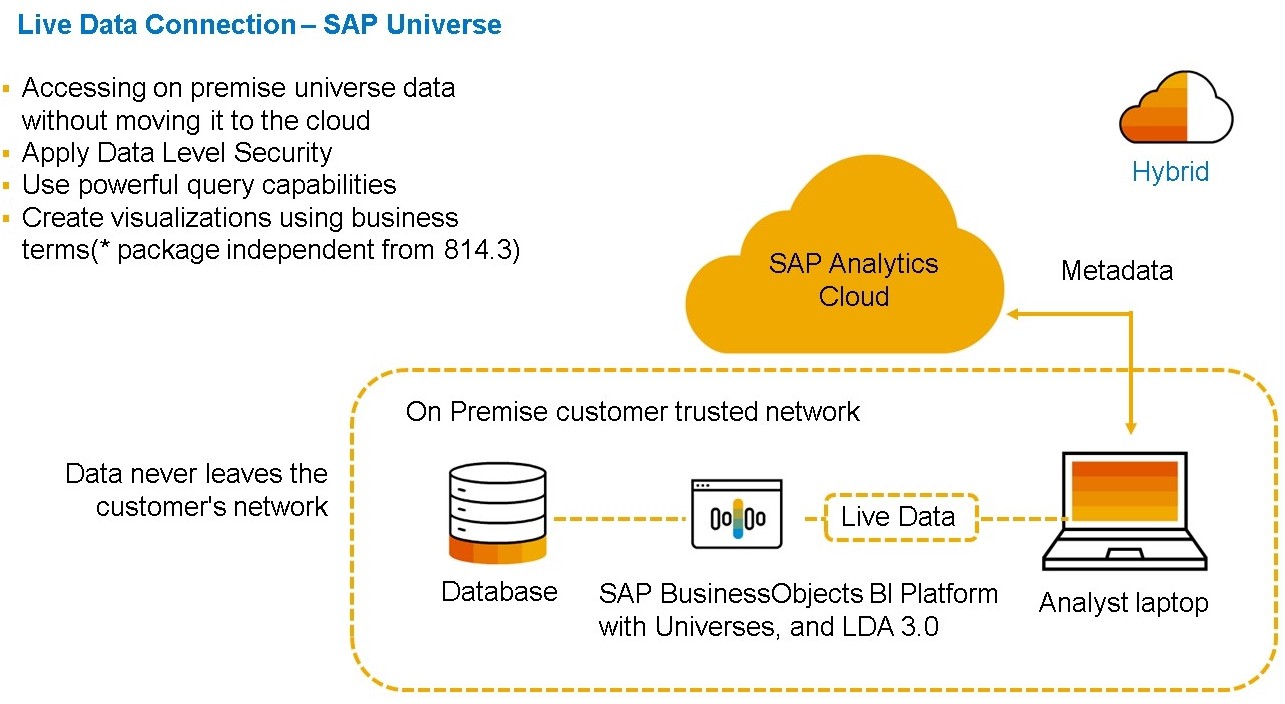


Figure 32: BI 4.3 SAP Universe

### SAP Universe Integration

SAP Analytics Cloud can be integrated with many SAP systems, including the SAP BusinessObjects BI platform.

The SAP BusinessObjects universe is one of the BI platform's semantic layer options and is able to connect to almost any datasource, both SAP and non-SAP. With SAP Analytics Cloud's ability to connect to universes, BI platform customers can leverage their on-premise BusinessObjects system with SAP Analytics Cloud's cloud-based modeling and analytics capabilities.

Most data preparation should be done within the universe, including data optimization and data enhancement, before using the universe in SAP Analytics Cloud.

The Integration of an SAP universe can be done in two ways:

* As a live data connection: no data will be stored on the cloud.
* As an import data connection: data will be stored in the cloud and updated periodically. Requirements for Live Data Connection
* SAP BusinessObjects Live Data Connect
* For more information on the live data connection to SAP universes, see: https:// help.sap.com/viewer/00f68c2e08b941f081002fd3691d86a7/release/en-US/ 3600be69a97a4b2888e73dcebbfa8ace.html

This URL must be running (in the Training Desktop) to use the universe live connection: https://wdflbmt7326.wdf.sap.corp:10443/sap/boc/ina/GetServerInfo

Requirements for Data Import

* SAP BTP Connectivity Cloud Connector
* SAP Analytics Cloud Agent
* For more information on the import data connection to SAP universes, see: https:// help.sap.com/viewer/00f68c2e08b941f081002fd3691d86a7/release/en-US/ 221a85298685468f8ab6efbb91e13d05.html

Watermark Sample

General Considerations

According to the connection type, decide if you need direct (browser) or path (reverse proxy). If using a reverse proxy, the name of the path prefix must match that of the path statement as defined in the Apache httpd.conf file.

Be aware that not all universe functionality is supported in the final SAP Analytics Cloud model.

Integration Check

To check whether the integration of a universe meets your expectations, see: https:// wiki.scn.sap.com/wiki/display/BOC/SAP+Analytics+Cloud+Support

+Matrix#SAPAnalyticsCloudSupportMatrix-LiveUniverseSupport

Create the Universe Connection

To connect to a universe, first create the appropriate connection. Once you have the connection, create a universe query to identify the data required by the SAP Analytics Cloud model.



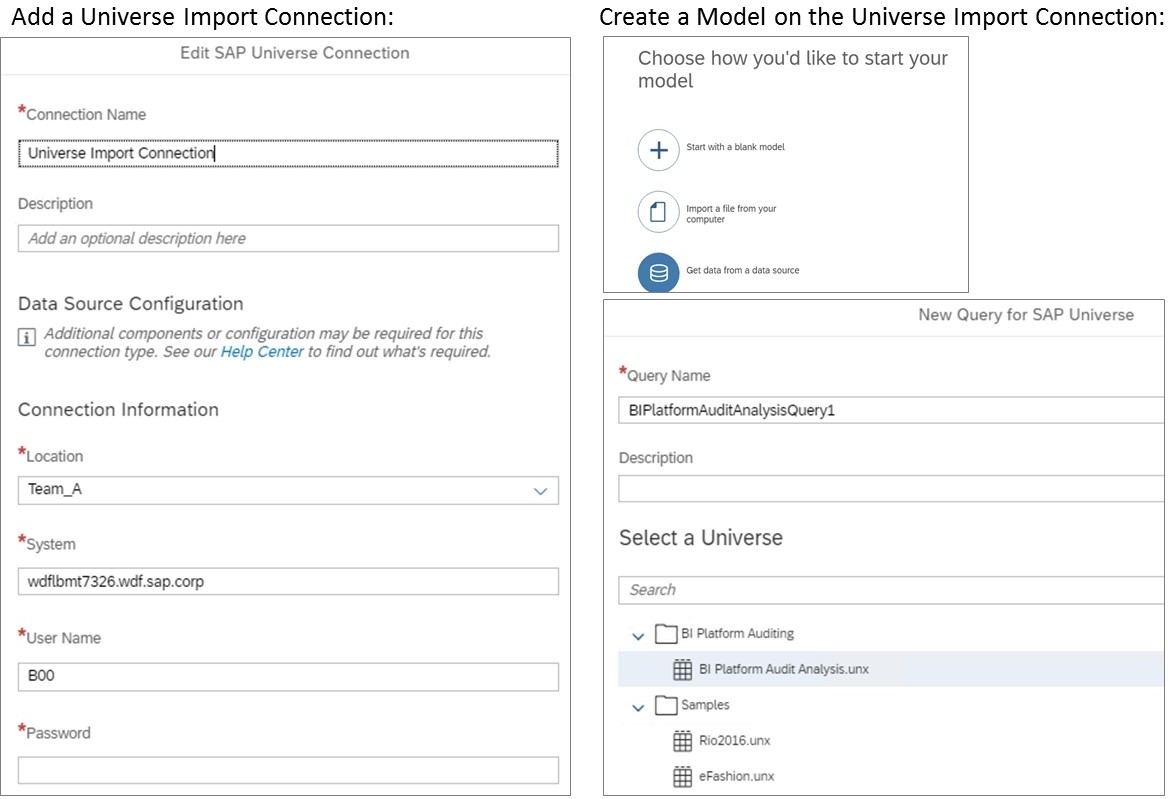


Figure 33: Create the Universe Connection

Watermark Sample



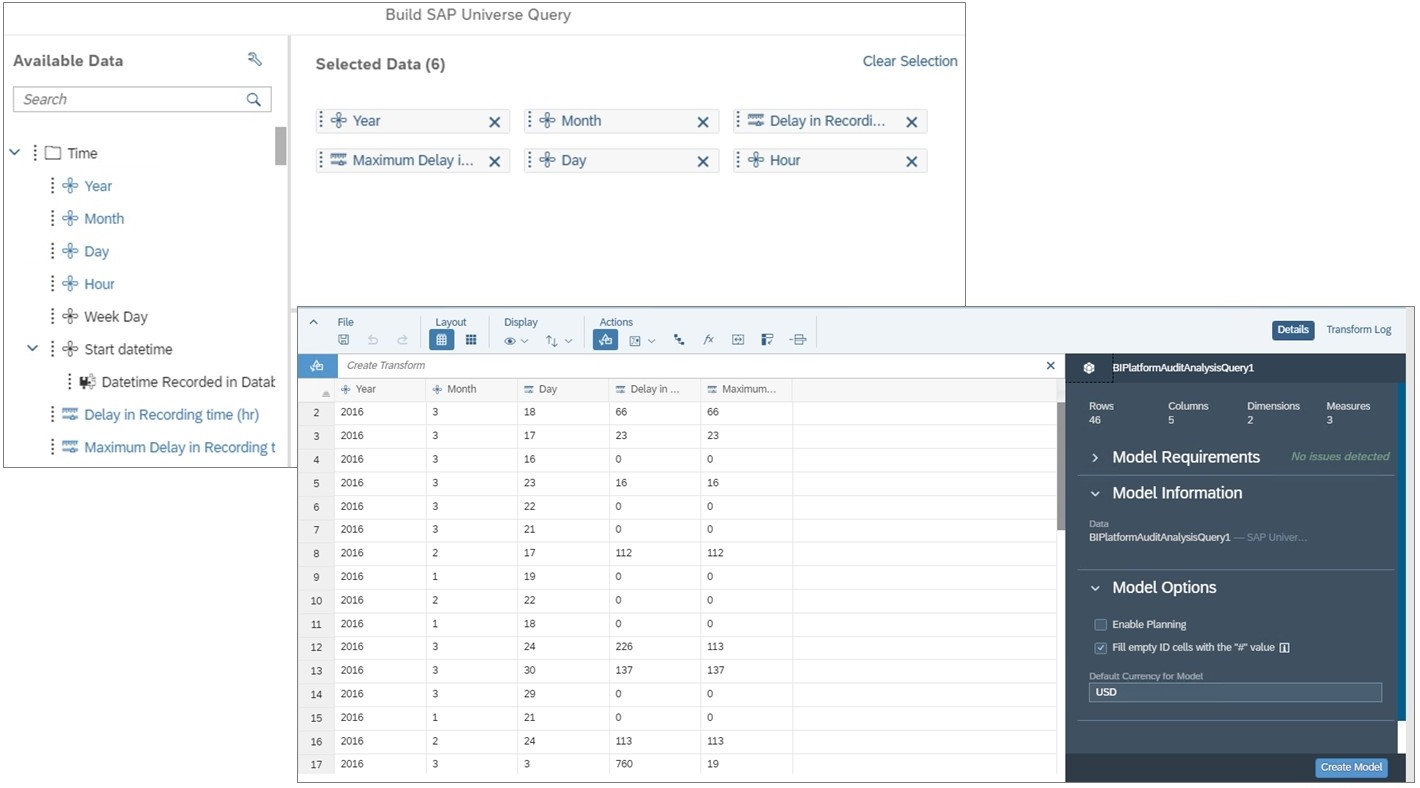


Figure 34: Create the Universe Query

Watermark Sample

Unit 3

Exercise 2

## Create an Analytic Model from a Universe

You need to acquire and wrangle weekly data from a universe into an analytic New Model. The weekly data should be organized in a time hierarchy with a 4-4-5 pattern. The data only needs to be analyzed therefore, a planning model is not needed.

Task Flow

In this exercise, you will perform the following tasks:

* Create an Analytic New Model
* Enable a week-based data pattern of 4-4-5
* Allow data for 2014 to 2023 with a week granularity
* Import data from a universe
* Wrangle data by concatenating the month and year data for the Date dimension
* Create model-specific calculations
* Create a Story using your model



Note:

The universe import connection is required for this exercise:

* UNV Import\_A (or B)
* The connection must be shared with the A and B teams
* System: wdflbmt7326.wdf.sap.corp
* User: SAC, Password: Welcome1

The instructor needs to set up the Team in the Cloud Connector as noted in the system-setup-guide for this exercise to work.

Task 1: Log on to SAP Analytics Cloud

1. Log on to SAP Analytics Cloud. Go to the next task if you are already logged on and you have created the course files (step 2).
   * User: **A##** or **B##**

## is your 2-digit group number, and the letter is what your instructor assigned to you.

* + Password: **Welcome1**

Task 2: Create an Analytic New Model

1. Create an analytic New Model and add public dimensions and measures.

Watermark Sample

1. Turn off the *Planning Capabilities* of the model by changing it to an analytic model.
2. Continuing in the *Model Preferences*, enable the week-based date pattern in the date settings.
3. Allow data records from 2014 to 2023. Set the granularity to *Week*.
4. Add the following existing public dimensions to the model:

|  |
| --- |
| Dimensions to Add |
| STATE (this should be uppercase) |
| CITY (this should be uppercase) |
| LINES (this should be uppercase) |

1. Add the *Revenue, Quantity*, and *Margin* measures to the model.

|  |  |
| --- | --- |
| Field | Value |
| Name | **Revenue** |
| Unit Type | **Currency** |
| Currency: Fixed | USD |

|  |  |
| --- | --- |
| Field | Value |
| Name | **Quantity** |
| Unit Type | **Unit** |
| Fixed Unit | **PC** |

|  |  |
| --- | --- |
| Field | Value |
| Name | **Margin** |
| Unit Type | **Currency** |
| Currency: Fixed | USD |

1. Go to the *Model Preferences* and confirm that you have changed to an *Analytic* type model.
2. Save the model as follows:

|  |  |
| --- | --- |
| Field | Value |
| Name | **U##M\_Analytic\_Universe** |
| Description | **U##M Analytic Model for Universe Data** |

Task 3: Populate the Model with Data from a Query from a Universe

1. In the upper left of the modeler, choose *Data Management* from the drop-down below

*Workspace*.

Watermark Sample

1. Import data from an SAP Universe.
2. Configure the new query as follows:

|  |  |
| --- | --- |
| Field | Value |
| Connection | UNV Import\_A (or B) |
| Universe | eFashion.unx (under *Samples*) |

The Build SAP Universe Query dialog opens.

1. Select the following fields:

|  |
| --- |
| Fields to Select |
| Year (the Fiscal Period description will be selected automatically) |
| Week (the Year/week description will be selected automatically) |
| State |
| City |
| Lines |
| Sales revenue |
| Quantity sold |
| Margin |

1. Add a filter for weeks 1 through 52 only.

|  |
| --- |
| Note:  The source system has test data in week 53, and we do not want to include it in this scenario. |

Task 4: Set Up the Import Job to Wrangle and Map the Query Data to the Model

1. Begin the set up of the import job.
2. Create a new column called *WK* by inserting a 0 for weeks before week 10 as follows:

[WK] = if([Week] < 10, concatenate(0, [Week], ""), [Week])

The new *WK* column appears in the data table. You can also build the formula via auto- complete (hint: start with the if statement).

1. Concatenate *Year* and *WK* without a separator:

Concatenate[Year],[WK] using ""

The *Year\_WK* column appears.

1. Click *Next* to map the data.
2. Map the *Year\_WK* column to the *Date* dimension. Mapping is complete. The date format is set to YYYYWW.
3. Click *Next* to map the properties for private dimensions.

Watermark Sample

|  |
| --- |
| Note:  Our model does not include any private dimensions, so this step will be blank. |

1. Click *Next* to review the import.

The import simulation will run and you should not see any messages or exceptions. If you do, go back to the prepare data step and correct the issue. Also, check your mapping.

1. Import the query data to the model. Over 11000 rows have been imported.
2. View the data.

Task 5: Add Calculated Measures to the Model

1. Add the *Price* calculated measure as follows:

|  |  |
| --- | --- |
| Field | Value |
| Name | **Price** |
| Description | **Price** |
| Aggregations (all blank) |  |
| Units & Currencies: |  |
| Unit Type: Currency | **Currency**: USD |
| Formatting - Scale: |  |
| Formatting - Decimal Places: | **1** (this is the default for the story) |
| Formula | **Revenue / Quantity** |

In the Preview area, you can see that price was calculated according to your formula.

1. Add the *Gross Margin Percentage* calculated measure as follows:

|  |  |
| --- | --- |
| Field | Value |
| Name | **GM\_PCT** |
| Description | **Gross Margin Percentage** |
| Units & Currencies: |  |
| Unit Type: | **None** |
| Scale | **Percent** |
| Formatting - Decimal Places: | **1**(this is the default for the story) |
| Formula | **Margin / Revenue** |

In the Preview area, you can see that the margin % was calculated according to your formula.

Watermark Sample

|  |
| --- |
|  |
| Figure 35: New Model Result |

1. Save the model.
2. Remain in the Modeler page.

Task 6: Build a Story with your Universe Model

Create a story in SAP Analytics Cloud with charts and tables.



Note:

If you see a message regarding *Optimized View Mode Enabled,* select *Don't show this message again*.

1. Use the Actions option to create a story using a canvas page (from the Model).
2. Add a table based on the *U##M Analytic Model*.

A table is added to the story and it is already assigned to your model.

1. Configure the table as follows:

|  |  |
| --- | --- |
| Axis | Value |
| Rows | *Lines, State* |
| Columns | *Revenue, Margin, Price, GM\_PCT* |

Your values may vary vs. the table below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Lines | State | Rev | Margin | Price | GM% |
| Access... | Calif... | 1869006 | 711895 | 153 | 38 |
| ... |  |  |  |  |  |

1. Set the style to 0 decimals for *Revenue* and *Margin*.
2. Confirm that the table setting for *Optimized Presentation* is off.

Watermark Sample

The data in the table is more compact with *Optimized Presentation* turned off.

1. Add date into the rows and expand it to level 5 in order to validate the 4-4-5 week pattern. Jan and Feb should have 4 weeks but Mar should have 5.
2. Save your story as follows:

|  |  |
| --- | --- |
| Field | Value |
| *Name* | **U##S\_Universe** |
| *Description* | **U## Universe Story** |

Here's an example. Your values may vary.

|  |
| --- |
|  |

1. Return to the SAC Home page. You have completed this exercise.

Watermark Sample

Unit 3

Solution 2

## Create an Analytic Model from a Universe

You need to acquire and wrangle weekly data from a universe into an analytic New Model. The weekly data should be organized in a time hierarchy with a 4-4-5 pattern. The data only needs to be analyzed therefore, a planning model is not needed.

Task Flow

In this exercise, you will perform the following tasks:

* + Create an Analytic New Model
  + Enable a week-based data pattern of 4-4-5
  + Allow data for 2014 to 2023 with a week granularity
  + Import data from a universe
  + Wrangle data by concatenating the month and year data for the Date dimension
  + Create model-specific calculations
  + Create a Story using your model



Note:

The universe import connection is required for this exercise:

* UNV Import\_A (or B)
* The connection must be shared with the A and B teams
* System: wdflbmt7326.wdf.sap.corp
* User: SAC, Password: Welcome1

The instructor needs to set up the Team in the Cloud Connector as noted in the system-setup-guide for this exercise to work.

Task 1: Log on to SAP Analytics Cloud

1. Log on to SAP Analytics Cloud. Go to the next task if you are already logged on and you have created the course files (step 2).
   * User: **A##** or **B##**

## is your 2-digit group number, and the letter is what your instructor assigned to you.

* + Password: **Welcome1**

1. From your training remote desktop, launch Google Chrome.

Watermark Sample

1. Enter the URL (provided by your instructor) for the SAC tenant you will use in class.
2. Your instructor has assigned you to a group, either A or B, and also given you a 2-digit group number. Log on to SAP Analytics Cloud using the credentials above.

Task 2: Create an Analytic New Model

1. Create an analytic New Model and add public dimensions and measures.
   1. From the Navigation Bar choose *Modeler*.
   2. From the *Create New* area click the + above *Model*.
   3. Ensure the *New Model* (Recommended) radio button is selected and click *Select*.
2. Turn off the *Planning Capabilities* of the model by changing it to an analytic model.
   1. From the *General* area of the modeling toolbar, click the  *Model Preferences*

button.

* 1. In the *General Settings* area, click the  next to *Planning*.
  2. Move the slider for *Planning Capabilities* to the  left to turn off the planning capabilities.

1. Continuing in the *Model Preferences*, enable the week-based date pattern in the date settings.
   1. In the *Model Preferences*, go to *Date Settings*.
   2. Enable Week-Based Date Pattern: .
   3. Choose OK.
2. Allow data records from 2014 to 2023. Set the granularity to *Week*.
   1. On the left, select the *Date* dimension.
   2. On the right in the *Dimension Details*, set the *From/To Year* to **2014**.
   3. Under Granularity, select *Week*.
   4. Click the < *Back* button in the upper left of the modeling window (in the light blue menu area, above the column headings).
3. Add the following existing public dimensions to the model:

|  |
| --- |
| Dimensions to Add |
| STATE (this should be uppercase) |
| CITY (this should be uppercase) |
| LINES (this should be uppercase) |

* 1. Go to the Model Structure workspace if needed.
  2. From the *Edit* area of the modeling toolbar, click the + and choose *Add existing dimensions*.

Watermark Sample

* 1. In the *Add existing dimensions* dialog box, choose *Organization* → *CITY*. Click *Add*.
  2. From the *Edit* area of the modeling toolbar, click the + and choose *Add existing dimensions*.
  3. In the *Add existing dimensions* dialog box, choose *Generic* → *LINES and STATE* and click *Add*.

1. Add the *Revenue, Quantity*, and *Margin* measures to the model.

|  |  |
| --- | --- |
| Field | Value |
| Name | **Revenue** |
| Unit Type | **Currency** |
| Currency: Fixed | USD |

|  |  |
| --- | --- |
| Field | Value |
| Name | **Quantity** |
| Unit Type | **Unit** |
| Fixed Unit | **PC** |

|  |  |
| --- | --- |
| Field | Value |
| Name | **Margin** |
| Unit Type | **Currency** |
| Currency: Fixed | USD |

* 1. From the left side of the modeler click +*Add* in the *Measures* area.
  2. On the right side of the model, in the *Measure Details* pane, enter the information in the table above.
  3. Repeat step a and b for the other two measures.

1. Go to the *Model Preferences* and confirm that you have changed to an *Analytic* type model.
2. Save the model as follows:

|  |  |
| --- | --- |
| Field | Value |
| Name | **U##M\_Analytic\_Universe** |
| Description | **U##M Analytic Model for Universe Data** |

Task 3: Populate the Model with Data from a Query from a Universe

1. In the upper left of the modeler, choose *Data Management* from the drop-down below

*Workspace*.

1. Import data from an SAP Universe.

Watermark Sample

1. From the *Import Jobs* area, click the  *Import Data* button and choose *Data source*.
2. Select *SAP Universe* as the data source.
3. Configure the new query as follows:

|  |  |
| --- | --- |
| Field | Value |
| Connection | UNV Import\_A (or B) |
| Universe | eFashion.unx (under *Samples*) |

* 1. Select the connection specified above and click *Next*.
  2. Select the universe specified above and click *Create*. The Build SAP Universe Query dialog opens.

1. Select the following fields:

|  |
| --- |
| Fields to Select |
| Year (the Fiscal Period description will be selected automatically) |
| Week (the Year/week description will be selected automatically) |
| State |
| City |
| Lines |
| Sales revenue |
| Quantity sold |
| Margin |

* 1. Drag (or double-click) the fields into the *Selected Data* area as in the table.
  2. Stay in the dialog.

1. Add a filter for weeks 1 through 52 only.
   1. Drag *Week* into the *Filter* area.
   2. Select an operator of *Less than or equal to* and type **52** in the operand field.
   3. Choose *Create*.

|  |
| --- |
| Note:  The source system has test data in week 53, and we do not want to include it in this scenario. |

Task 4: Set Up the Import Job to Wrangle and Map the Query Data to the Model

1. Begin the set up of the import job.

Watermark Sample

* 1. In the *Import Jobs* area, click *Set Up Import* in the *Last Known Status* column.
  2. This will take several seconds. After 1 minute or so, you may need to  reload the SAC browser or click the *Refresh* hyperlink until *Set Up Import* appears.

1. Create a new column called *WK* by inserting a 0 for weeks before week 10 as follows:

[WK] = if([Week] < 10, concatenate(0, [Week], ""), [Week])

* 1. From the *Actions* area of the *Prepare Data* toolbar, click the  *Toggle Custom Expression Editor* button.
  2. In row 1, enter **[WK] = if([Week] < 10, concatenate(0, [Week], ""), [Week])**.
  3. Click the  *Execute* button (or press enter).
  4. Close the Custom Expression Editor.

The new *WK* column appears in the data table. You can also build the formula via auto- complete (hint: start with the if statement).

1. Concatenate *Year* and *WK* without a separator:

Concatenate[Year],[WK] using ""

* 1. Highlight the *Year* column.
  2. Choose → Create a Transform.
  3. Choose *Concatenate*.
  4. In the formula bar, click on Column, enter **W** → Select *WK*.
  5. In the formula bar, click on *value* (in the double quotes) and leave it blank (no separator is needed).
  6. Press *Enter*.

The *Year\_WK* column appears.

1. Click *Next* to map the data.
2. Map the *Year\_WK* column to the *Date* dimension.
   1. From the *Source Columns* on the left side, drag *Year\_WK* to the *Drop a column* for the

*Date* dimension in the *Unmapped* area in the center of the screen.

Mapping is complete. The date format is set to YYYYWW.

1. Click *Next* to map the properties for private dimensions.

|  |
| --- |
| Note:  Our model does not include any private dimensions, so this step will be blank. |

1. Click *Next* to review the import.

Watermark Sample

The import simulation will run and you should not see any messages or exceptions. If you do, go back to the prepare data step and correct the issue. Also, check your mapping.

1. Import the query data to the model.
   1. On the lower right, click *Run Import*.
   2. From the *Run Import* dialog box, click *Finish*. Over 11000 rows have been imported.
2. View the data.
   1. Go to the Model Structure workspace.
   2. Click the  foundation view to see the data.

Task 5: Add Calculated Measures to the Model

1. Add the *Price* calculated measure as follows:

|  |  |
| --- | --- |
| Field | Value |
| Name | **Price** |
| Description | **Price** |
| Aggregations (all blank) |  |
| Units & Currencies: |  |
| Unit Type: Currency | **Currency**: USD |
| Formatting - Scale: |  |
| Formatting - Decimal Places: | **1** (this is the default for the story) |
| Formula | **Revenue / Quantity** |

* 1. Go to the *Calculations* workspace.
  2. Next to *Calculated Measures*, choose *+*.
  3. Enter the data provided in the table above.
  4. Enter the formula in the canvas near the top middle of the page.

In the Preview area, you can see that price was calculated according to your formula.

1. Add the *Gross Margin Percentage* calculated measure as follows:

|  |  |
| --- | --- |
| Field | Value |
| Name | **GM\_PCT** |
| Description | **Gross Margin Percentage** |
| Units & Currencies: |  |
| Unit Type: | **None** |
| Scale | **Percent** |
| Formatting - Decimal Places: | **1**(this is the default for the story) |

Watermark Sample

|  |  |
| --- | --- |
| Field | Value |
| Formula | **Margin / Revenue** |

* 1. Go to the *Calculations* workspace if needed.
  2. Next to *Calculated Measures*, choose +.
  3. Enter the data provided in the table above.
  4. Enter the formula in the canvas near the top middle of the page.

In the Preview area, you can see that the margin % was calculated according to your formula.

|  |
| --- |
|  |
| Figure 35: New Model Result |

1. Save the model.
2. Remain in the Modeler page.

Task 6: Build a Story with your Universe Model

Create a story in SAP Analytics Cloud with charts and tables.



Note:

If you see a message regarding *Optimized View Mode Enabled,* select *Don't show this message again*.

1. Use the Actions option to create a story using a canvas page (from the Model).
   1. In the ribbon, choose 
   2. Choose *Story* → *Canvas*.
2. Add a table based on the *U##M Analytic Model*.

Watermark Sample

a) Choose the  *Table* icon.

A table is added to the story and it is already assigned to your model.

1. Configure the table as follows:

|  |  |
| --- | --- |
| Axis | Value |
| Rows | *Lines, State* |
| Columns | *Revenue, Margin, Price, GM\_PCT* |

* 1. For the Rows choose *+Add Measures/Dimension*.
  2. Choose *State* and *Lines*.
  3. For the Columns, hover over *Measures* and select the  filer icon.
  4. Select *Margin, Price,* and *GM\_PCT*.
  5. Enlarge the table as needed.

Your values may vary vs. the table below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Lines | State | Rev | Margin | Price | GM% |
| Access... | Calif... | 1869006 | 711895 | 153 | 38 |
| ... |  |  |  |  |  |

1. Set the style to 0 decimals for *Revenue* and *Margin*.
   1. Highlight the *Revenue* header in the table.
   2. Go to the  panel on the upper right.
   3. Scroll down if needed and set *Decimal Places* to **0**.
   4. Highlight the *Margin* header in the table.
   5. Scroll down if needed and set *Decimal Places* to **0**.
2. Confirm that the table setting for *Optimized Presentation* is off.
   1. Go to the  pane.
   2. Deselect  *Optimized Presentation*.

The data in the table is more compact with *Optimized Presentation* turned off.

1. Add date into the rows and expand it to level 5 in order to validate the 4-4-5 week pattern.
   1. In the *Builder* on the right, under *Rows* choose *+Add Measures/Dimension*s.
   2. Select *Date*.
   3. In the table, right click on *Date* and choose *Drill* → *Level 5*. Jan and Feb should have 4 weeks but Mar should have 5.
2. Save your story as follows:

Watermark Sample

|  |  |
| --- | --- |
| Field | Value |
| *Name* | **U##S\_Universe** |
| *Description* | **U## Universe Story** |

Here's an example. Your values may vary.

|  |
| --- |
|  |

1. Return to the SAC Home page. You have completed this exercise.

Watermark Sample

LESSON SUMMARY

You should now be able to:

* Access data from SAP BusinessObjects Universe

Watermark Sample

### SAP Data Warehouse Cloud

As part of SAP HANA Cloud Services, SAP Analytics Cloud (SAC) and SAP Data Warehouse Cloud (DWC) are an essential part of the SAP strategy. The combination of both is a powerful tool that is designed for complex analytics scenarios based on hybrid system architectures.

There are several ways to visualize data from DWC in SAC. Each SAP Data Warehouse cloud contains an SAP Analytics Cloud, embedded version. With this embedded version you can directly access the data that is available in the warehouse. If you already have an SAC system, you can access DWC data via a live connection. To do so, an SAC connection must be created and SAC must be added as a trusted origin. For more information, use this link: https:// blogs.sap.com/2020/05/19/how-to-connect-sap-analytics-cloud-and-sap-data-warehouse- cloud/.



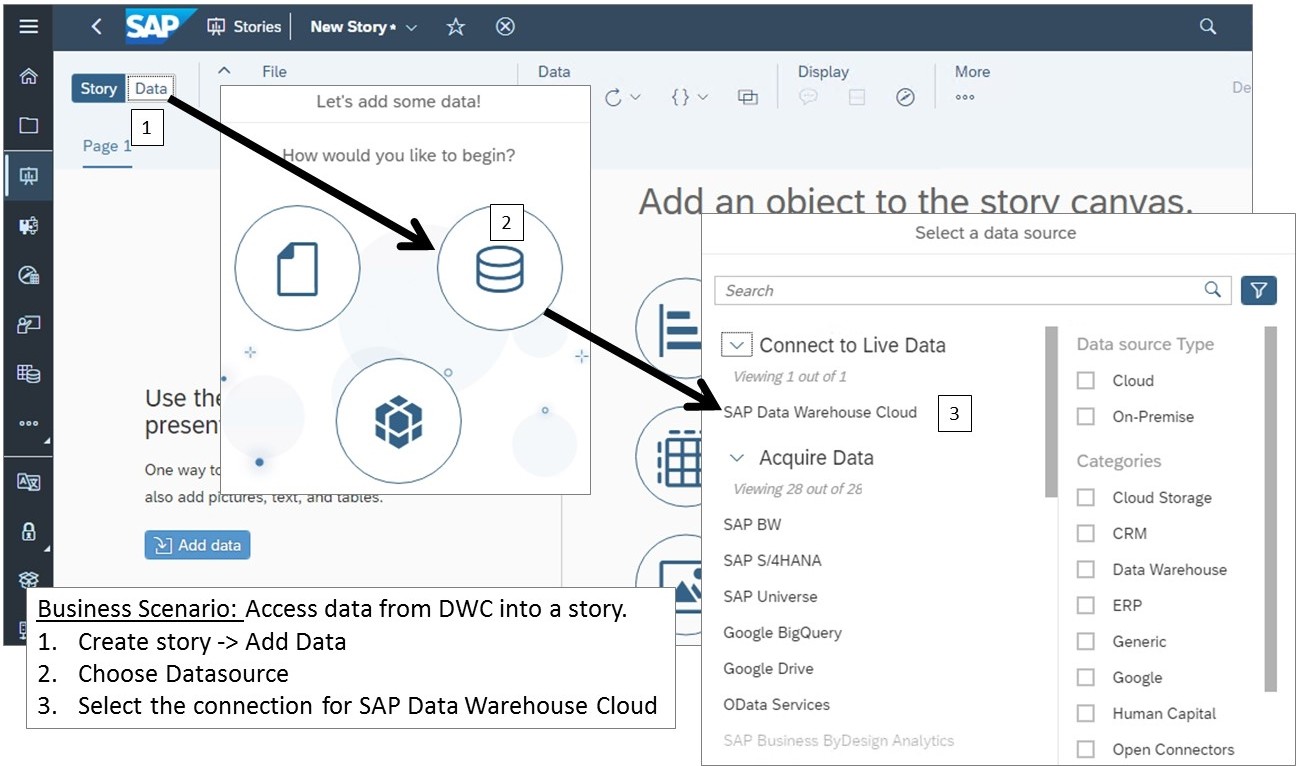


Figure 36: Connect to DWC from an SAC Story

It is important to know that it is not required to create a data model in SAC. SAP Analytics Cloud can directly access analytical datasets within SAP Data Warehouse Cloud and use it as data source.

Watermark Sample



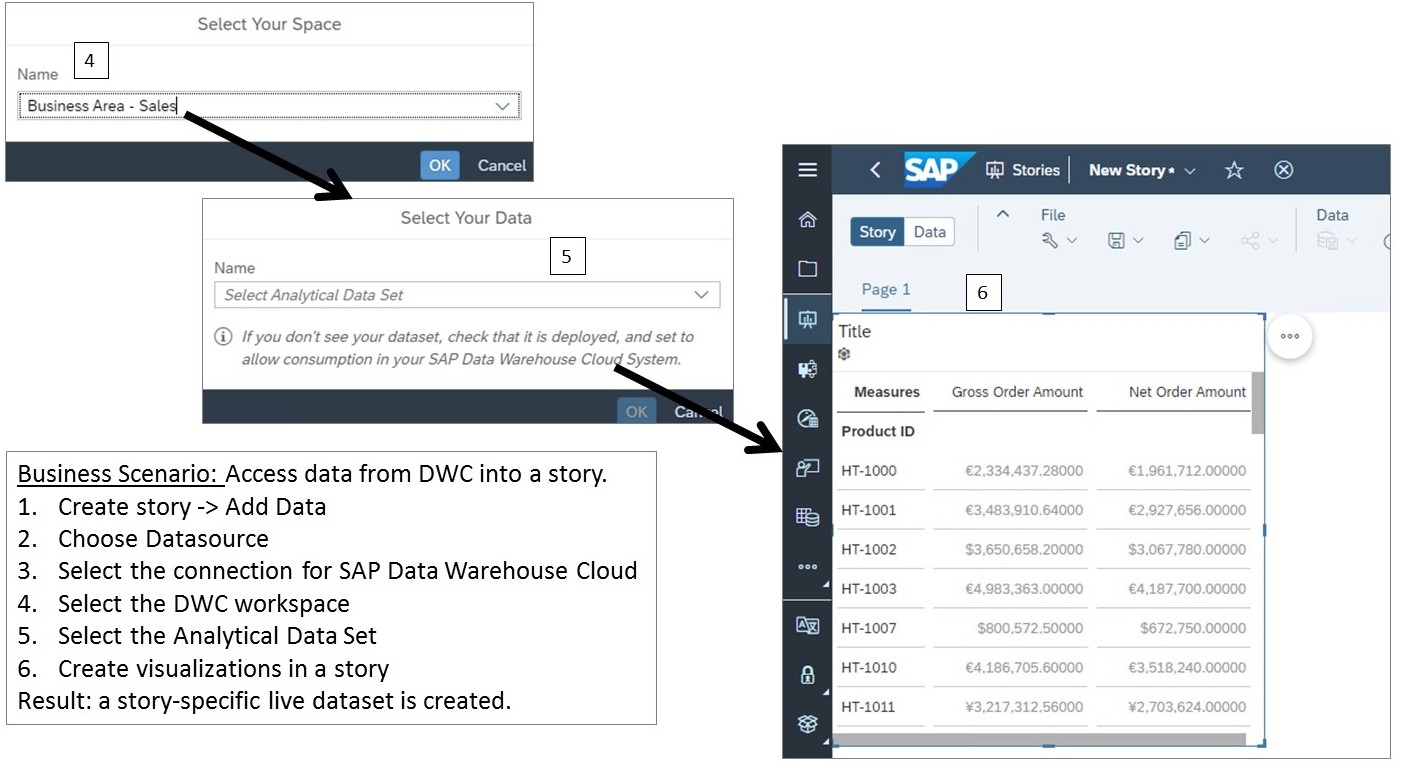


Figure 37: Display Live Data in an SAC Story for DWC



Note:

As of September 2022, only live connections to DWC are available.

Access Data from SAP Data Warehouse Cloud

1. Create a connection to DWC.
2. In DWC, set the SAC tenant url as a trusted origin.
3. In SAC:
   * Create a story
   * Go to the Data tab → Select Datasource → Select the SAP Data Warehouse Cloud live connection. (in the training system the DWC connection should already be set up. If not, use Host: sapeducdwc.eu10.hcs.cloud.sap Port: 443 Authentication Method: SAML Single Sign On.
   * Log into DWC if needed (Note for instructor: see the SSG for the user & password).
   * Select a DWC workspace (such as Business Area Sales in the training server) → Select a DWC Analytical Data Set (such as Sales Country in the training server).
   * Add a table and configure it with product id in the columns.
   * Save the story.

LESSON SUMMARY

You should now be able to:

* Access data from SAP Data Warehouse Cloud

Watermark Sample

### SAP Business Warehouse Integration

SAP Analytics Cloud and SAP BW, SAP BW/4HANA

As with SAP BusinessObjects universes and SAP S/4HANA, SAP Analytics Cloud integrates seamlessly with SAP Business Warehouse (SAP BW) to make the most of your existing investments. And the Integration of an SAP universe can be done in two ways:

* + As a live data connection: no data will be stored on the cloud.
  + As an import data connection: data will be stored in the cloud and updated periodically.

Import Data Connection

When using an imported data connection to SAP BW, you have the following advantages:

* + Uses existing SAP BW queries to securely import your data into the cloud.
  + Applies SAP Analytics Cloud native data security concepts.
  + Leverages the calculation engine of SAP HANA to allow for additional analytic use cases.

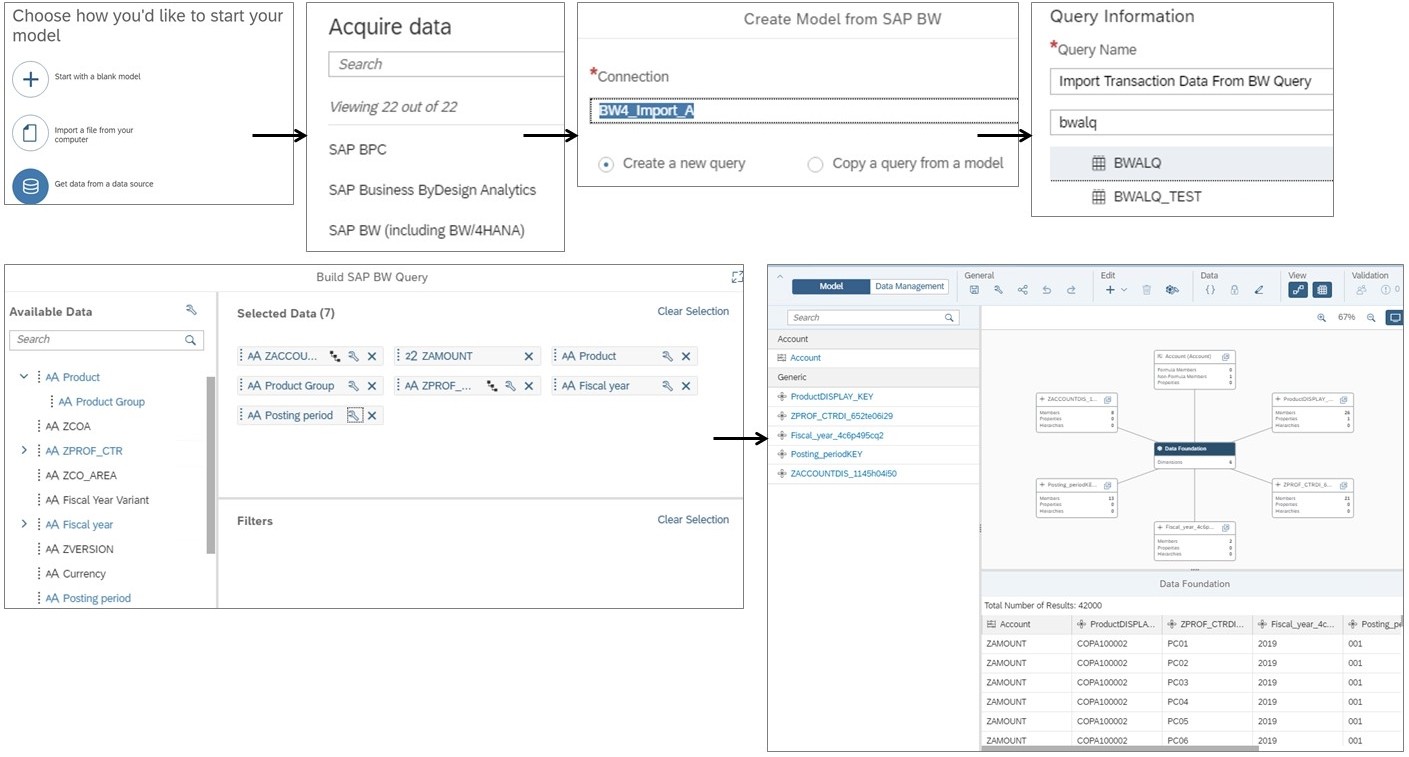
The process of importing data from SAP BW to SAC is illustrated in the figure *SAP BW Import Workflow*.

Watermark Sample





Figure 38: SAP BW Import Workflow

Importing data from SAP BW has the following prerequisites:

* The SAP Business Warehouse (BW) system is version 7.3x or higher, or SAP BW/4HANA system, SP4 or higher.
* SAP BTP Connectivity Cloud Connector is installed.
* SAP Analytics Cloud agent is installed.

Live Data Connection



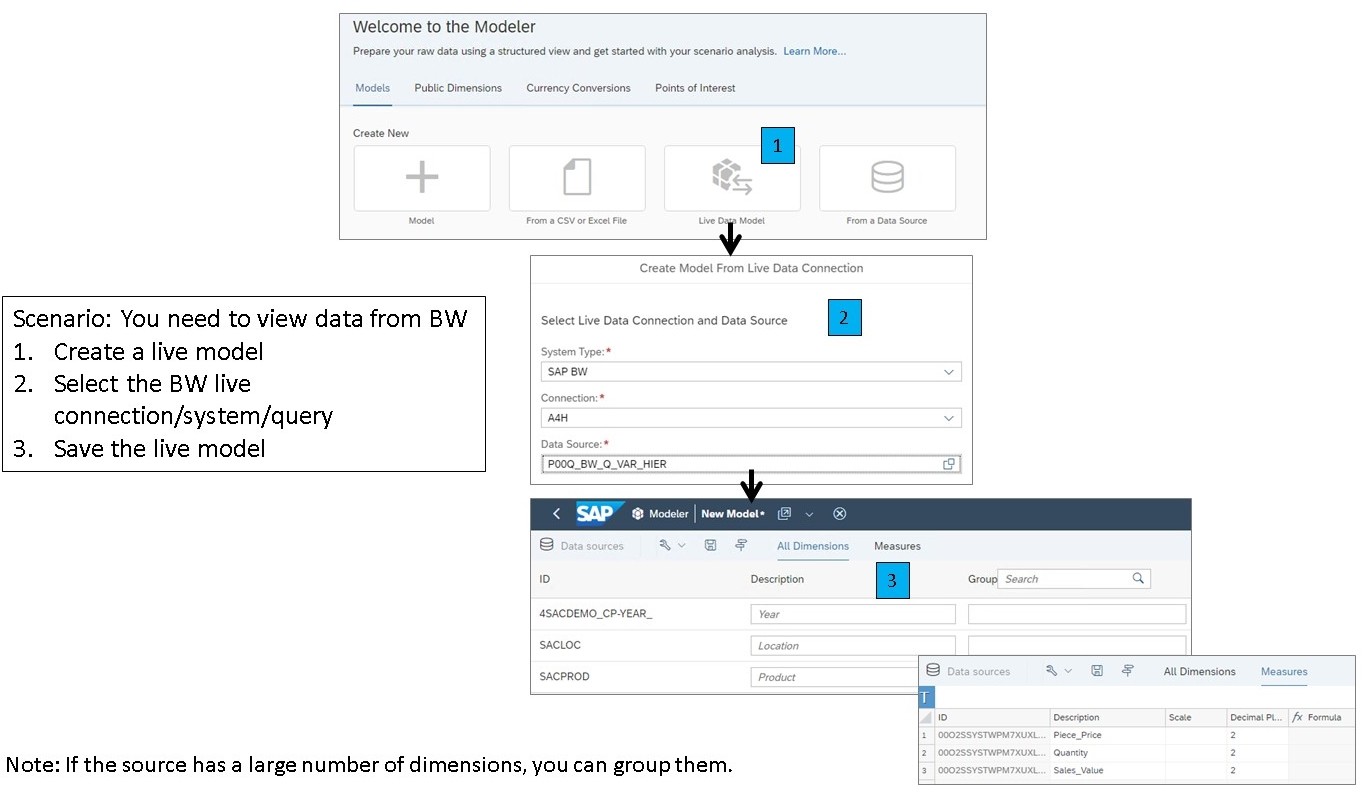


Figure 39: BW Live Model Workflow

When using a live data connection to SAP BW, you have the following advantages:

Watermark Sample

* + Accesses SAP BW features that cannot be accessed through standard SQL or MDX query interfaces.
  + Leverages the SAP BW metadata without additional modeling.
  + Re-uses existing authorization concepts.
  + Connects to SAP BW, as well as generated SAP HANA views.



Note:

For more information on supported features and required updates for the integration of SAP BW with SAP Analytics Cloud, see: SAP Note - 2541557



Note:

When creating linked dimensions for SAP BW data sources, you can choose to link on matching hierarchies.

Supported SAP BW Elements in Live Models

The following SAP BW elements are supported in SAC stories:

* + Time-dependent hierarchies
  + BW Variables
  + BW Variants and Personalization
  + BW Structures
  + Parallel execution of BW queries



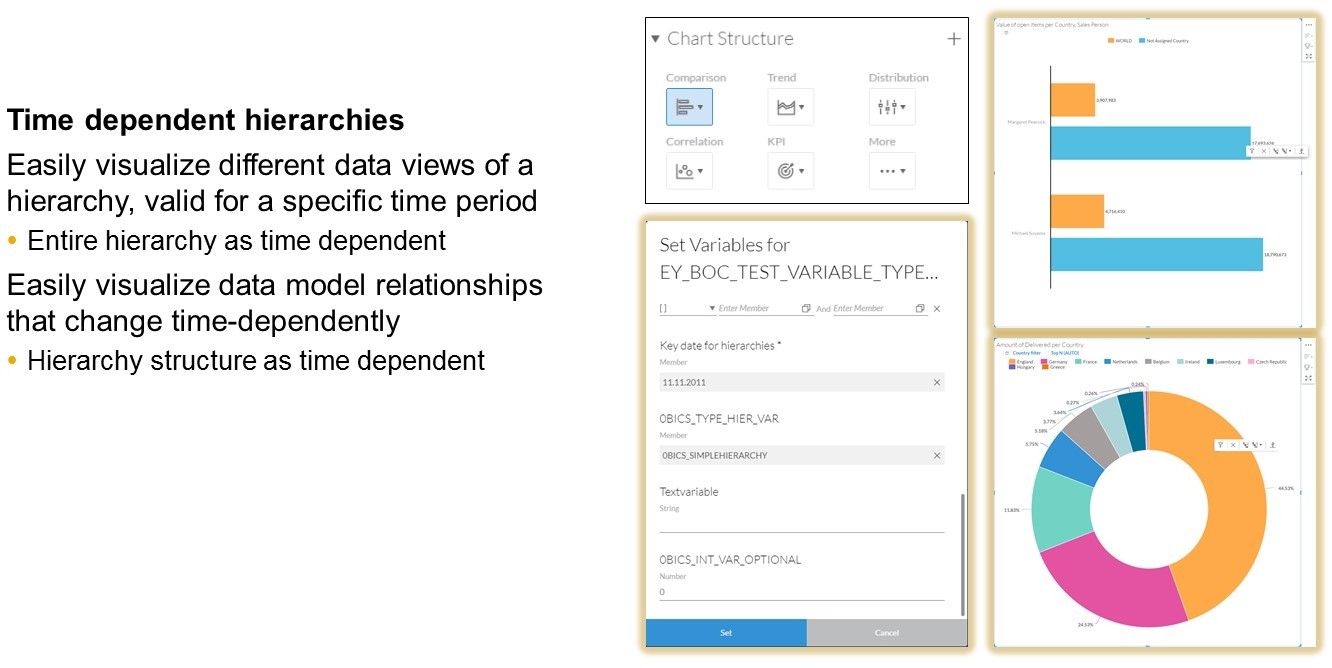
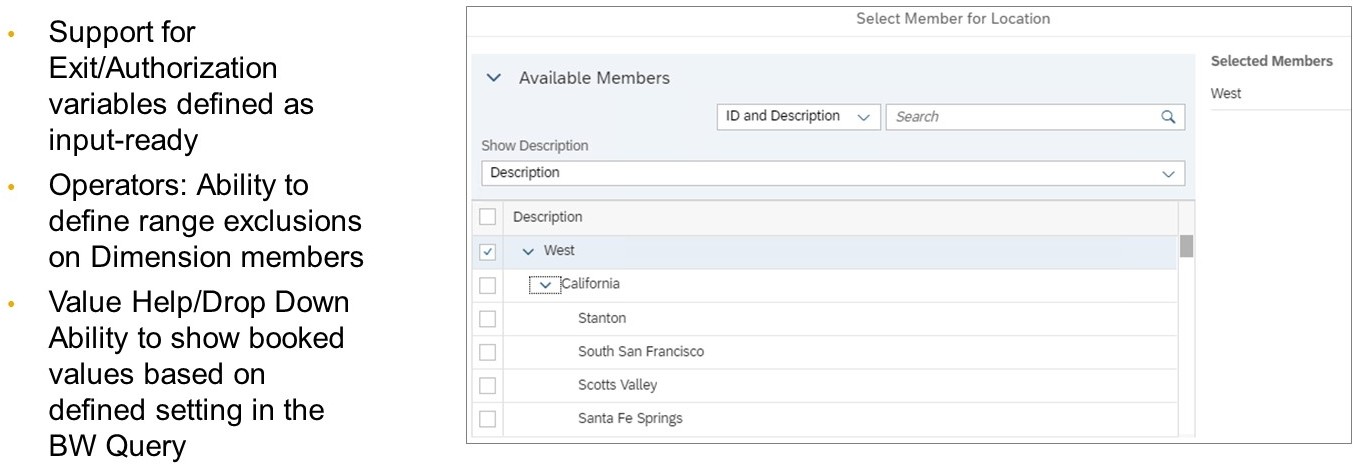
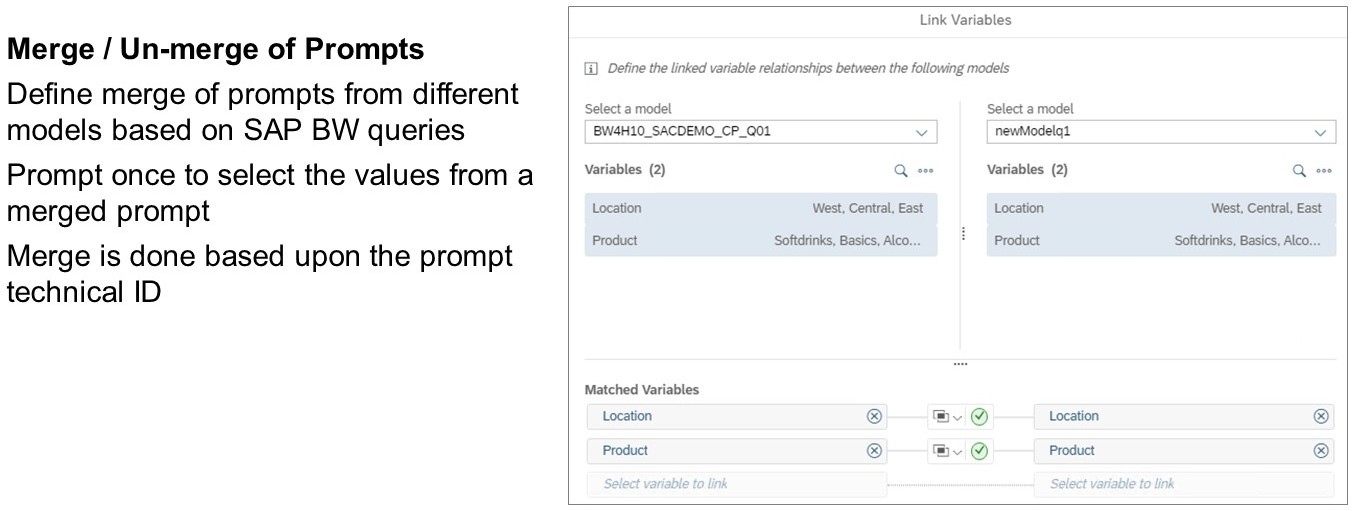


Figure 40: BW Hierarchy Support



Watermark Sample





Figure 41: BW: Merging of Prompts



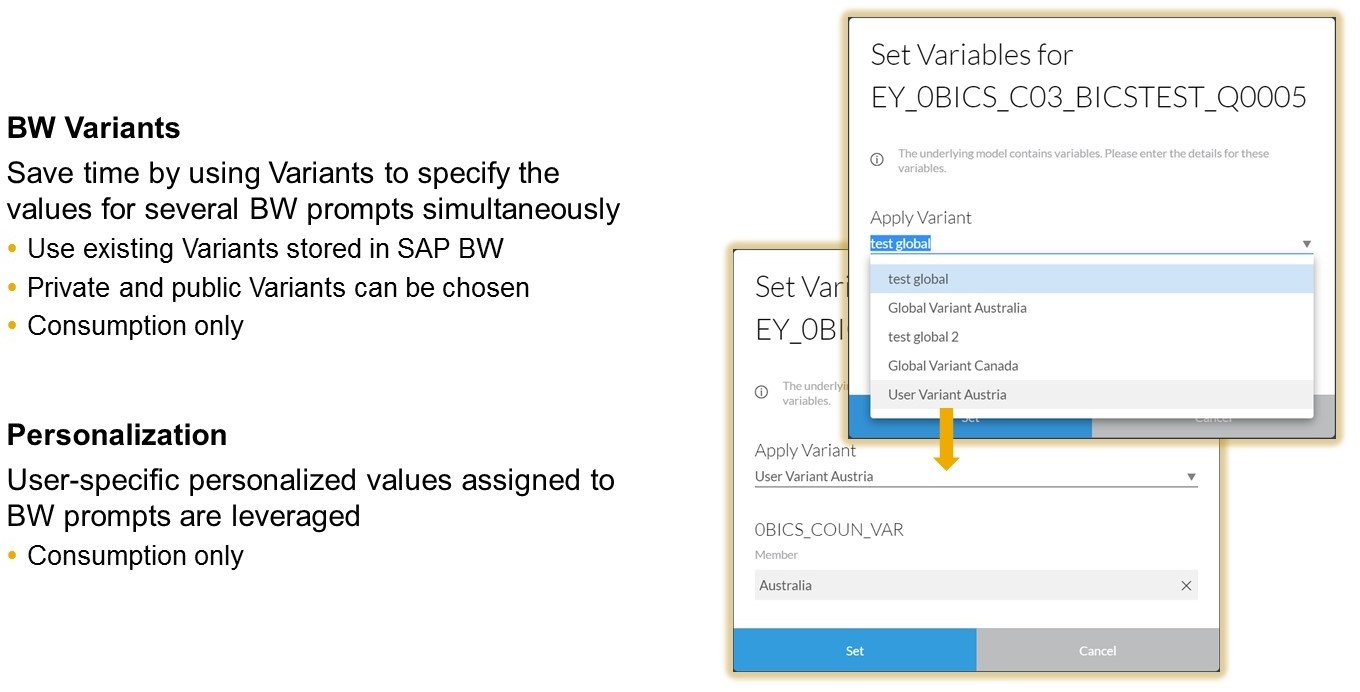


Figure 42: BW Variants and Personalization Support



Figure 43: BW Variable Support

Watermark Sample



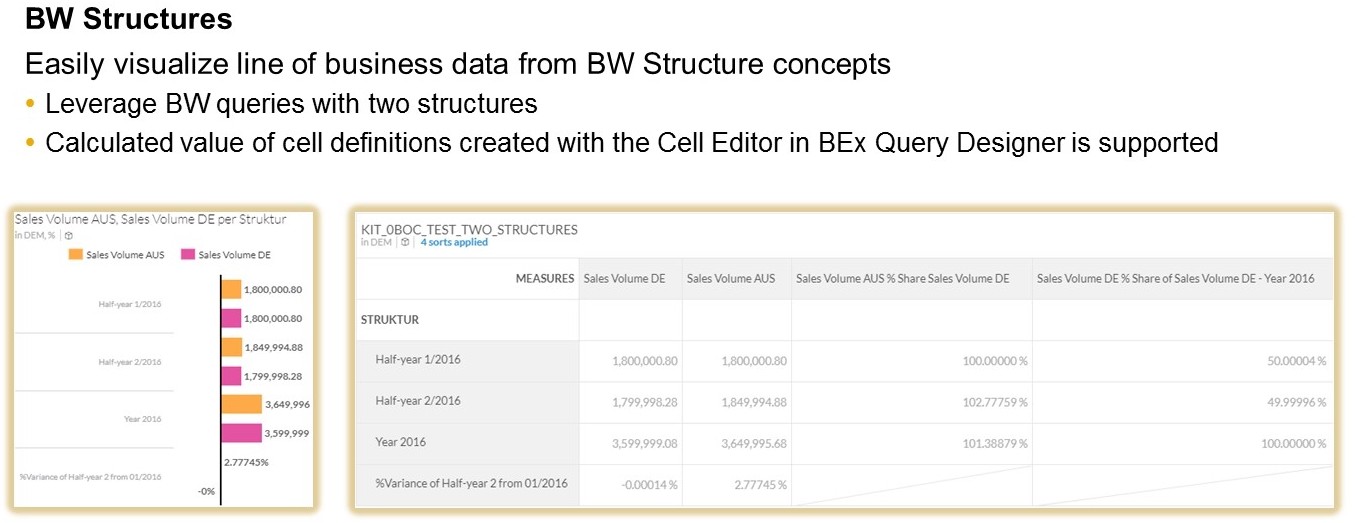


Figure 44: BW Structures Support

In addition, live connectivity makes it easy to visualize line-of-business data from SAP BW structure concepts, as shown in the figure, Live Data Connectivity to SAP BW: BW Structures Support.



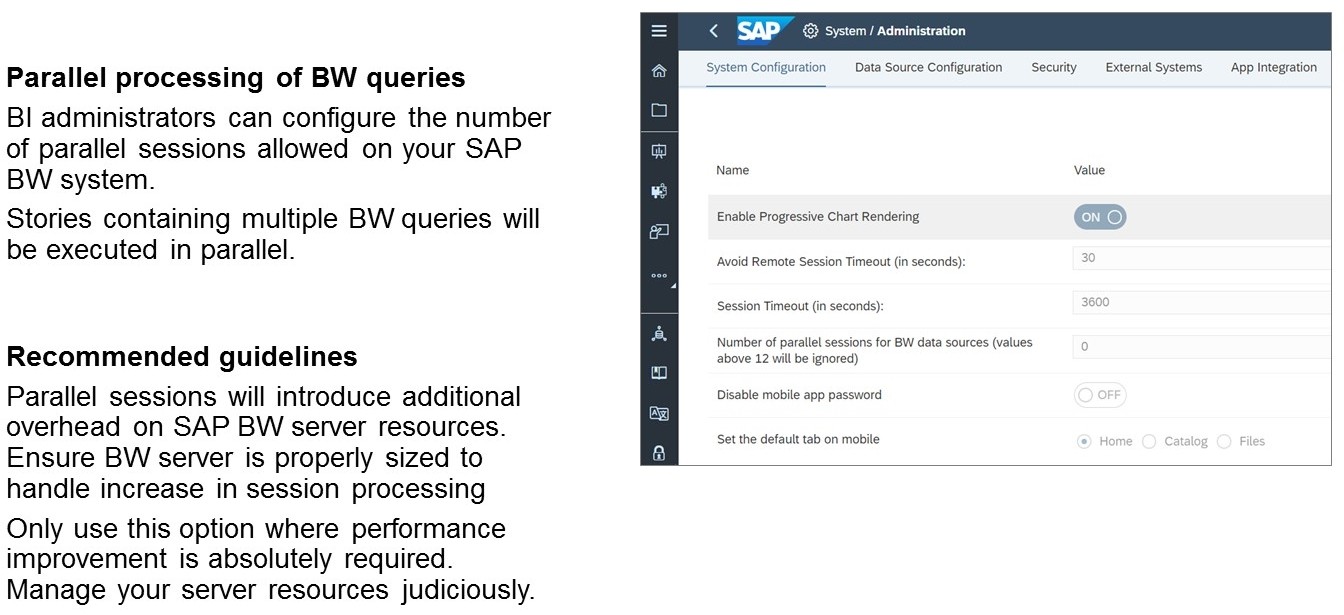


Figure 45: Parallel Processing To Enhance Story Performance



Note:

Display and navigation attributes are not fully supported. The end user must manually change the attribute column in SAC as a property classification.

For more information on SAP BW connections, see: https://help.sap.com/viewer/ 00f68c2e08b941f081002fd3691d86a7/release/en-US/ 5dd668839c9f44599d61e6f0016dd553.html

Watermark Sample

Watermark Sample

Unit 3

Exercise 3

## Access SAP BW Live Data

Your company has been using BW for years and now you are implementing SAC. You would like to display some of you BW Queries in SAC Stories using live connections. You need to use the BW data in multiple stories so you will need to create a live model. For another BW Query, you will use data analyzer in more of an Ad-Hoc scenario.

Task Flow

In this exercise, you will perform the following tasks:

* Create an SAP Analytics Cloud model based on an SAP BW live connection and an SAP BW query with variables and a hierarchy.
* Create a story based on the model.
* Use the SAP BW variables to make your selections in the story.
* Use Data Analyzer to analyze the data in a BW query with two structures.

Task 1: Log on to SAP Analytics Cloud

1. Log on to SAP Analytics Cloud.
   * User: **A##** or **B##**.

## is your 2-digit group number, and the letter is what your instructor assigned to you.

* + Password: **Welcome1**.

Task 2: Create a Model for the SAP BW Query with a Hierarchy and Variables

1. Create a model with a live connection using the SAP BW query P00Q\_BW\_Q\_VAR\_HIER, as follows:

|  |  |
| --- | --- |
| Field | Value |
| *System Type* | **SAP BW** |
| *Connection* | **A4H** |
| If needed:*User* | **WSBW** |
| If needed:*Password* | **Welcome1** |
| *Data Source* | **P00Q\_BW\_Q\_VAR\_HIER** |

A modeler screen displays with the following two tabs:

* + All Dimensions
  + Measures

Watermark Sample

The Measure tab will display all measures that are part of the query, in this case, *Quantity, Piece Price*, and *Sales Value*.

For measures, you can change the following:

* Descriptions
* Scale to be displayed as default (thousands, millions, and so on)
* The decimal place defaults

|  |
| --- |
| Note:  This is preferable at the model level if measure reporting has a specific format universally (as in financial reporting), as opposed to always changing scale and decimals at the story and visualization level. |

1. Choose *All Dimensions*.

You will see various dimensions listed, as well as the SAP BW structure that will be part of the model.

|  |
| --- |
| Note:  For dimensions, you can change the following:   * Descriptions * Create semantic groups of dimensions for easier navigation during story creation. This is advantageous in the case of SAP BW queries that have a large number of dimensions to choose from. |

1. Save the model as follows:

|  |  |
| --- | --- |
| Field | Value |
| *Name* | **U##M\_BW\_HIER\_VAR** |
| *Description* | **U## Model for BW Query Hier Var** |

Task 3: Build a Story with Live SAP BW Connections

Create a story in SAP Analytics Cloud with charts and tables.



Note:

If you see a message regarding *Optimized View Mode Enabled,* select *Don't show this message again*.

1. Use the Actions option to create a story using a responsive page from the Model. The Set Variable prompt opens.
2. In the Set Variable dialog, make your selections as follows:

Watermark Sample

|  |
| --- |
| Selections |
| Automatically open prompt when story opens |
| Apply Variant: None |
| Location: |
| **West** |
| Product: |
| **Softdrinks** |
| **Basics** |
| **Alcohol** |
| **Juices** |

1. Add a table to the story. Put *Product* into the row axis. The new table appears in the left lane.
2. Delete the right lane so that you are only working with one lane.
3. Add the title Sales Overview and center it on the page.
4. Add a chart to display quantity and sales value based on product dimension from your U##M\_BW\_HIER\_VAR model.
5. Configure the chart as follows:

|  |  |
| --- | --- |
| Field | Value |
| Chart type | Combination Column & Line |
| Column Axis | Quantity |
| Line Axis | Sales\_Value |
| Dimensions | Product |

A chart depicting quantity and sales value based on the product dimension is created.

1. Drill up and down the hierarchy.

|  |
| --- |
|  |

1. Save your story as follows. Press Ctrl\_S.

Watermark Sample

|  |  |
| --- | --- |
| Field | Value |
| *Name* | **U##S\_BW** |
| *Description* | **U## BW Story** |

1. Return to the SAC Home page.

Task 4: Use Data Analyzer to Analyze the SAP BW Query with Two Structures

1. In the Navigation bar, choose the  *Data Analyzer.*

The *Welcome to Data Analyzer* page opens.

1. Access the *P00Q\_BW\_Q\_STRUCTURES* BW Query as follows:

|  |  |
| --- | --- |
| Field | Value |
| *System Type* | **SAP BW** |
| *Connection* | **A4H** |
| *User* (if necessary) | **WSBW** |
| *Password* (if necessary) | **Welcome1** |
| *Data Source* | **P00Q\_BW\_Q\_STRUCTURES** |

1. Use the Designer to change the dimensions displayed in the table to *Product* and the Structure.
2. Click *Designer* to close the *Builder* options.
3. Use the Filter to restrict the data displayed in the table to only the Central *Region*.
4. Use the Filter to restrict the Structure to *Q1 2023, Q2 2023,* and *Change Q2 vs. Q1 (%)*.

|  |
| --- |
|  |

1. Save the Insight as follows:

Watermark Sample

Table 2:

|  |  |
| --- | --- |
| Field | Value |
| Name | **U##I\_Product\_Sales** |
| Description | **U## Product Sales** |

1. Return to the SAC Home page. You have completed this exercise.

Watermark Sample

Unit 3

Solution 3

## Access SAP BW Live Data

Your company has been using BW for years and now you are implementing SAC. You would like to display some of you BW Queries in SAC Stories using live connections. You need to use the BW data in multiple stories so you will need to create a live model. For another BW Query, you will use data analyzer in more of an Ad-Hoc scenario.

Task Flow

In this exercise, you will perform the following tasks:

* Create an SAP Analytics Cloud model based on an SAP BW live connection and an SAP BW query with variables and a hierarchy.
* Create a story based on the model.
* Use the SAP BW variables to make your selections in the story.
* Use Data Analyzer to analyze the data in a BW query with two structures.

Task 1: Log on to SAP Analytics Cloud

1. Log on to SAP Analytics Cloud.

* User: **A##** or **B##**.

## is your 2-digit group number, and the letter is what your instructor assigned to you.

* Password: **Welcome1**.

1. From your training remote desktop, launch Google Chrome.
2. Enter the URL (provided by your instructor) for the SAC tenant you will use in class.
3. Your instructor has assigned you to a group, either A or B, and also given you a 2-digit group number. Log on to SAP Analytics Cloud using the provided credentials.

Task 2: Create a Model for the SAP BW Query with a Hierarchy and Variables

1. Create a model with a live connection using the SAP BW query P00Q\_BW\_Q\_VAR\_HIER, as follows:

|  |  |
| --- | --- |
| Field | Value |
| *System Type* | **SAP BW** |
| *Connection* | **A4H** |
| If needed:*User* | **WSBW** |
| If needed:*Password* | **Welcome1** |
| *Data Source* | **P00Q\_BW\_Q\_VAR\_HIER** |

Watermark Sample

* 1. From the Navigation Bar choose  *Modeler*.
  2. Choose *Live Data Model*.
  3. Choose the predefined SAP BW connection and credentials as shown above.
  4. In the *Data Source* field, enter **P00Q**.
  5. Choose the *P00Q\_BW\_Q\_VAR\_HIER Query*.
  6. Choose *OK*.

A modeler screen displays with the following two tabs:

* All Dimensions
* Measures

The Measure tab will display all measures that are part of the query, in this case, *Quantity, Piece Price*, and *Sales Value*.

For measures, you can change the following:

* Descriptions
* Scale to be displayed as default (thousands, millions, and so on)
* The decimal place defaults

|  |
| --- |
| Note:  This is preferable at the model level if measure reporting has a specific format universally (as in financial reporting), as opposed to always changing scale and decimals at the story and visualization level. |

1. Choose *All Dimensions*.

You will see various dimensions listed, as well as the SAP BW structure that will be part of the model.

|  |
| --- |
| Note:  For dimensions, you can change the following:   * Descriptions * Create semantic groups of dimensions for easier navigation during story creation. This is advantageous in the case of SAP BW queries that have a large number of dimensions to choose from. |

1. Save the model as follows:

|  |  |
| --- | --- |
| Field | Value |
| *Name* | **U##M\_BW\_HIER\_VAR** |

Watermark Sample

|  |  |
| --- | --- |
| Field | Value |
| *Description* | **U## Model for BW Query Hier Var** |

1. Choose *Save*.
2. Enter the provided data.
3. Choose *OK*.
4. Stay on the Modeler page.

Task 3: Build a Story with Live SAP BW Connections

Create a story in SAP Analytics Cloud with charts and tables.



Note:

If you see a message regarding *Optimized View Mode Enabled,* select *Don't show this message again*.

1. Use the Actions option to create a story using a responsive page from the Model.
   1. In the ribbon, choose 
   2. Choose *Story* → *Responsive*. The Set Variable prompt opens.
2. In the Set Variable dialog, make your selections as follows:

|  |
| --- |
| Selections |
| Automatically open prompt when story opens |
| Apply Variant: None |
| Location: |
| **West** |
| Product: |
| **Softdrinks** |
| **Basics** |
| **Alcohol** |
| **Juices** |

* 1. Make your selections as shown above.
  2. Choose *Set*.

1. Add a table to the story. Put *Product* into the row axis.
   1. In the ribbon (or ...More menu), choose .

Watermark Sample

* 1. In *Builder* → *Row* axis, choose *+Add Measures/Dimensions* → Select *Product*. The new table appears in the left lane.

1. Delete the right lane so that you are only working with one lane.

a) To remove the right lane, click *More Actions…* for the lane and choose *Remove*.

1. Add the title Sales Overview and center it on the page.
2. Double-click on *Click to enter title* and enter **Sales Overview** into the lane title.
3. On the upper right, choose *Designer* if needed.
4. Go to the *Styling* pane on the upper right.
5. Format the title so that it is centered on the page by setting the *Alignment* to *Center*. If needed, double-click *Sales Overview* to access the Header style.
6. Add a chart to display quantity and sales value based on product dimension from your U##M\_BW\_HIER\_VAR model.
7. Choose the  *Chart* icon in the ribbon (or ...More menu).
8. Select the *U##M\_BW\_HIER\_VAR* model if needed.
9. Configure the chart as follows:

|  |  |
| --- | --- |
| Field | Value |
| Chart type | Combination Column & Line |
| Column Axis | Quantity |
| Line Axis | Sales\_Value |
| Dimensions | Product |

1. Choose the *Comparison* drop down and select *Combination Column & Line Chart*.
2. For the column axis, choose *+Add Measure*.
3. Choose *Quantity*.
4. For the line axis, choose *+Add Measure* and select *Sales\_Value*.
5. Under *Dimensions*, choose *Product*.
6. Enlarge the chart as desired.

A chart depicting quantity and sales value based on the product dimension is created.

1. Drill up and down the hierarchy.
2. Choose the *Alcohol* bar.
3. Click the down arrows to see the products that are part of the *Alcohol* product group.
4. Click the *Beer* column and use the up arrows to return to the product group level of granularity.

Watermark Sample

|  |
| --- |
|  |

1. Save your story as follows. Press Ctrl\_S.

|  |  |
| --- | --- |
| Field | Value |
| *Name* | **U##S\_BW** |
| *Description* | **U## BW Story** |

1. Return to the SAC Home page.

Task 4: Use Data Analyzer to Analyze the SAP BW Query with Two Structures

1. In the Navigation bar, choose the  *Data Analyzer.*

The *Welcome to Data Analyzer* page opens.

1. Access the *P00Q\_BW\_Q\_STRUCTURES* BW Query as follows:

|  |  |
| --- | --- |
| Field | Value |
| *System Type* | **SAP BW** |
| *Connection* | **A4H** |
| *User* (if necessary) | **WSBW** |
| *Password* (if necessary) | **Welcome1** |
| *Data Source* | **P00Q\_BW\_Q\_STRUCTURES** |

* 1. In the *Start New* area, click *From a Data Source*.
  2. Provide the information as shown above and choose **OK**.

1. Use the Designer to change the dimensions displayed in the table to *Product* and the Structure.
   1. Click *Designer* in the upper right corner.
   2. From the *Available Items* area, drag *Product* to the *Rows* area, positioning it above

*Location*. (*Product* appears grayed-out, but you can move it to the Rows.)

* 1. Remove *Location* from the *Rows*.

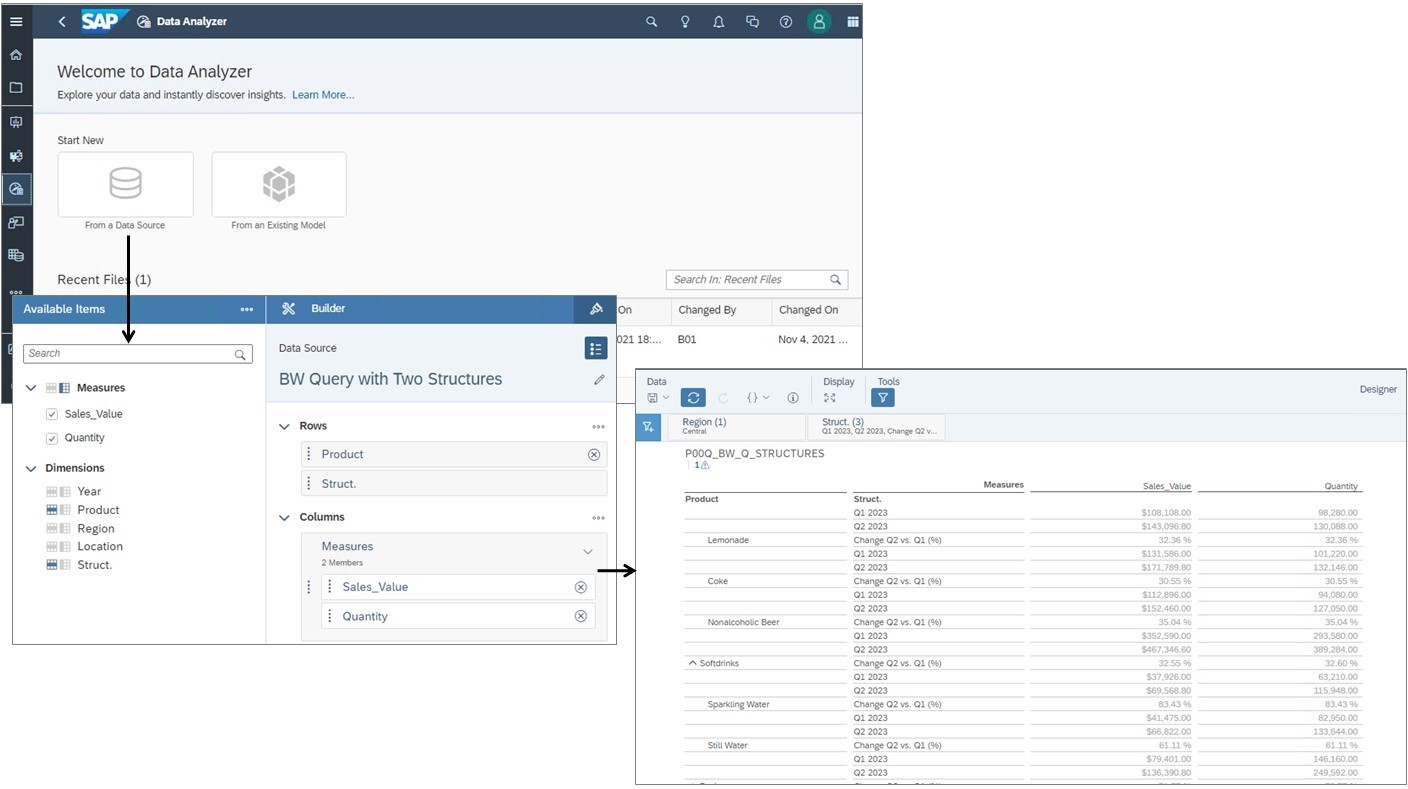
1. Click *Designer* to close the *Builder* options.
2. Use the Filter to restrict the data displayed in the table to only the Central *Region*.

Watermark Sample

* 1. In the filter bar, on the upper left click the  *Add Story Filter/Prompt* button.
  2. Choose *Region*.
  3. Choose *Central* and click *OK*.

1. Use the Filter to restrict the Structure to *Q1 2023, Q2 2023,* and *Change Q2 vs. Q1 (%)*.
   1. In the filter bar, click the  *Add Story Filter/Prompt* button.
   2. Choose *Struct*.
   3. Choose *Q1 2023, Q2 2023,* and *Change Q2 vs. Q1 (%)*, then click *OK*.

|  |
| --- |
|  |



1. Save the Insight as follows: Table 2:

|  |  |
| --- | --- |
| Field | Value |
| Name | **U##I\_Product\_Sales** |
| Description | **U## Product Sales** |

1. Return to the SAC Home page. You have completed this exercise.

Watermark Sample

LESSON SUMMARY

You should now be able to:

* + Access data from SAP Business Warehouse

## Learning Assessment

Watermark Sample

1. What is required to create an import data connection to an SAP BusinessObjects universe?

*Choose the correct answer.*

* 1. ODBC Connector
  2. SAP BTP Connectivity Cloud Connector
  3. JDBC Connector

 D Generic Connector

1. When importing data from SAP S/4HANA, dimension members are imported before any hierarchy data.

*Determine whether this statement is true or false.*

True

 False

1. With a live data connection to SAP BW, authorizations must be duplicated into SAP Analytics Cloud.

*Determine whether this statement is true or false.*

True

 False

1. What is the source when accessing data from SAP HANA Cloud into SAP Analytics Cloud?

*Choose the correct answer.*

* 1. A core data services view
  2. A calculation view
  3. A composite view
  4. A responsive view

Watermark Sample

1. What is a pre-requisite to access data from SAP Data Warehouse Cloud into SAP Analytics Cloud?

*Choose the correct answer.*

* 1. Create a model
  2. Add SAP Analytics Cloud as a trusted origin
  3. Import data

## Learning Assessment - Answers

Watermark Sample

1. What is required to create an import data connection to an SAP BusinessObjects universe?

*Choose the correct answer.*

* 1. ODBC Connector
  2. SAP BTP Connectivity Cloud Connector

X

* 1. JDBC Connector

 D Generic Connector

That's correct! The SAP BTP Connectivity Cloud Connector is required to create an import data connection to an SAP BusinessObjects universe.

1. When importing data from SAP S/4HANA, dimension members are imported before any hierarchy data.

*Determine whether this statement is true or false.*

True

X

 False

That's correct! When importing data from SAP S/4HANA, dimension members are imported before any hierarchy data.

1. With a live data connection to SAP BW, authorizations must be duplicated into SAP Analytics Cloud.

*Determine whether this statement is true or false.*

True

False

X

That's incorrect. With a live data connection to SAP BW, SAP Analytics Cloud re-uses existing SAP BW authorizations.

Watermark Sample

1. What is the source when accessing data from SAP HANA Cloud into SAP Analytics Cloud?

*Choose the correct answer.*

* 1. A core data services view
  2. A calculation view

X

* 1. A composite view

 D A responsive view

That's correct! Calculation views in SAP HANA Cloud are used to provide data into SAP Analytics Cloud.

1. What is a pre-requisite to access data from SAP Data Warehouse Cloud into SAP Analytics Cloud?

*Choose the correct answer.*

* 1. Create a model
  2. Add SAP Analytics Cloud as a trusted origin

X

 C Import data

That's correct! SAP Analytics Cloud must be added as a trusted origin. Models are not required. Importing data is not required.

Watermark Sample

**UNIT 4**

**Using Templates and Explorer**

### Lesson 1

Using Templates and Analyzing with Explorer

Exercise 4: Create a Story from a Template and Use Explorer

112

115

UNIT OBJECTIVES

* Use templates for building stories
* Use Explorer for creating dynamic visualizations

Watermark Sample

Unit 4

Lesson 1

## Using Templates and Analyzing with Explorer

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* Use templates for building stories
* Use Explorer for creating dynamic visualizations

### Use a Template for a Story

You can apply formatting to your story by using a template, which provides predefined layouts and placeholders for objects to help you build a story. When creating a new story, you can choose a template as a starting point. You can also apply a template to an existing story

by clicking  *Layouts* in the main toolbar.

There are three categories of templates to choose from, depending on the type of story you are trying to create:

* Monitor - These layouts can help you create a story ideal for monitoring you data or create a boardroom monitor. The layouts combine tiles such as maps, content, and keyfigures.
* Report - These layouts can help you create a story ideal for reporting. The layouts combine tiles such as charts, tables, and annotations.
* Present - These layouts can help you create a story ideal for presentations. The layouts provide the formatting and elements necessary to create a slide deck.

You can also create your own templates simply by saving a story as a Template. Saving the story as a template removes all data and converts charts, tables, maps, input controls, and value driver trees into empty placeholders. All grid pages and custom formatting are removed.

### Explore Your Data

You can start creating a story for your data by experimenting with filters and charts in the explorer.

In the explorer, you see a faceted view of your data, which you can manipulate to generate charts for your story pages. When you select measures and dimensions in the upper-pane, the visualization in the lower-pane updates in real time. You can filter dimensions by selecting individual members, and the visualization changes immediately to show you the filtered result.

Initially, the visualization type is chosen automatically based on the selected data, but you can change it to any of the types supported for your data. When using the table visualization, the behavior is the same as when using tables in stories.