

DATA SOCIETY:

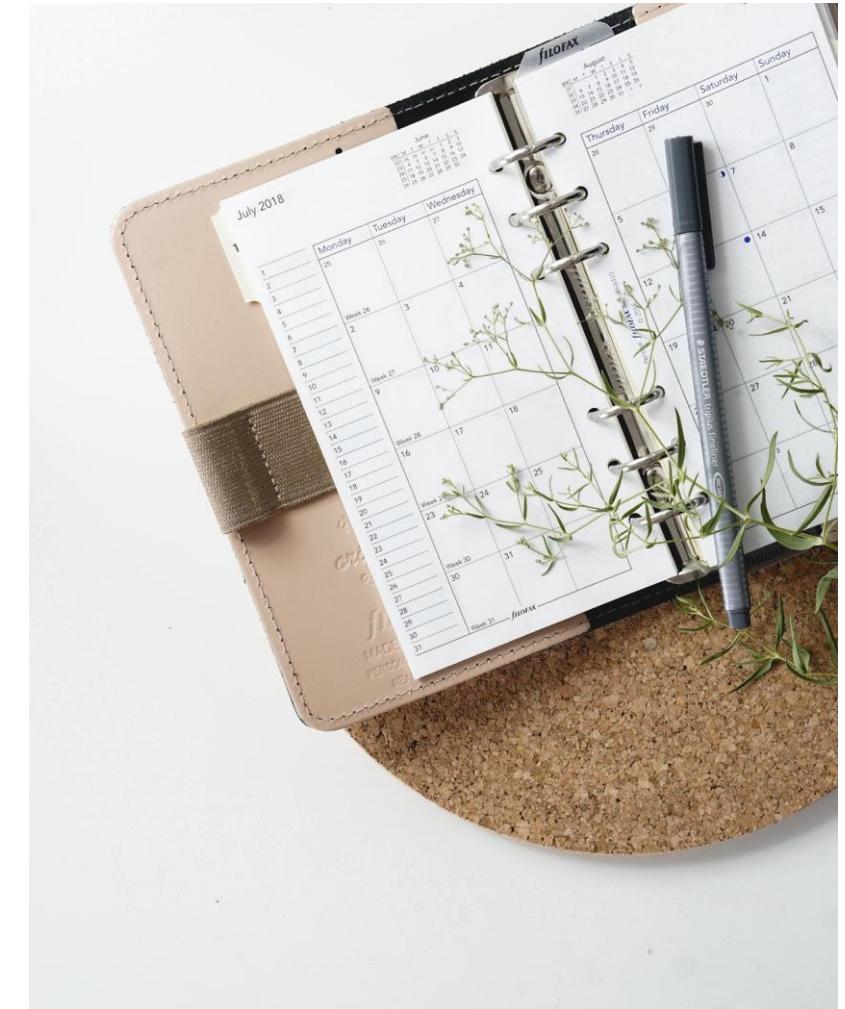
Power BI Intermediate

Day 1



Welcome!

- Hours and breaks
 - 4 days
 - 11 am - 2 pm
 - 1-2 short breaks each class
- Materials
 - PDF slide decks, Datasets, lab and exercise files



Best practices for virtual classes

- Find a quiet place, free of as many distractions as possible. Headphones are recommended.
- Remove or silence alerts from cell phones, e-mail pop-ups, etc.
- Participate in activities and ask questions. This will be interactive!



Who we are

Data Society's mission is to integrate Big Data and machine learning best practices across entire teams and empower professionals to identify new insights.

We provide:

- High-quality data science training programs
- Customized executive workshops
- Custom software solutions and consulting services

Since 2014, we've worked with thousands of professionals to make their data work for them.



Pre-Work for PowerBI

If needed, review the following:

- [Overview of Power BI Desktop](#) - 4 minutes
- [Getting started with Power BI Desktop](#) - 9 minutes
- [Connect to data sources in Power BI Desktop](#) - 8 minutes

Setting Expectations

- This course is for users who have a pre-existing working knowledge of Power BI with analytical experience using Power BI Desktop and service
- Each lesson will be accompanied by student exercise and knowledge checks
- This is **not** a DAX-centric course but is designed for Business Intelligence report development
- Students are advised to explore the additional resources to further skill development beyond this course

Introduction to the Data Sets



For the lab exercises, we will be using the Power Cycle data set. This is a Global fictitious Cycle Retailer and your Job will be to use the data set to complete the lab based on the instructions provided in the lab resource materials.

Fact Tables (Sales Table)

Australia
Canada
France
Germany
UK
US
Power Cycle Target

Dimension Tables

Product Category
Product Sub-Category
Product Sales Region
Salesperson

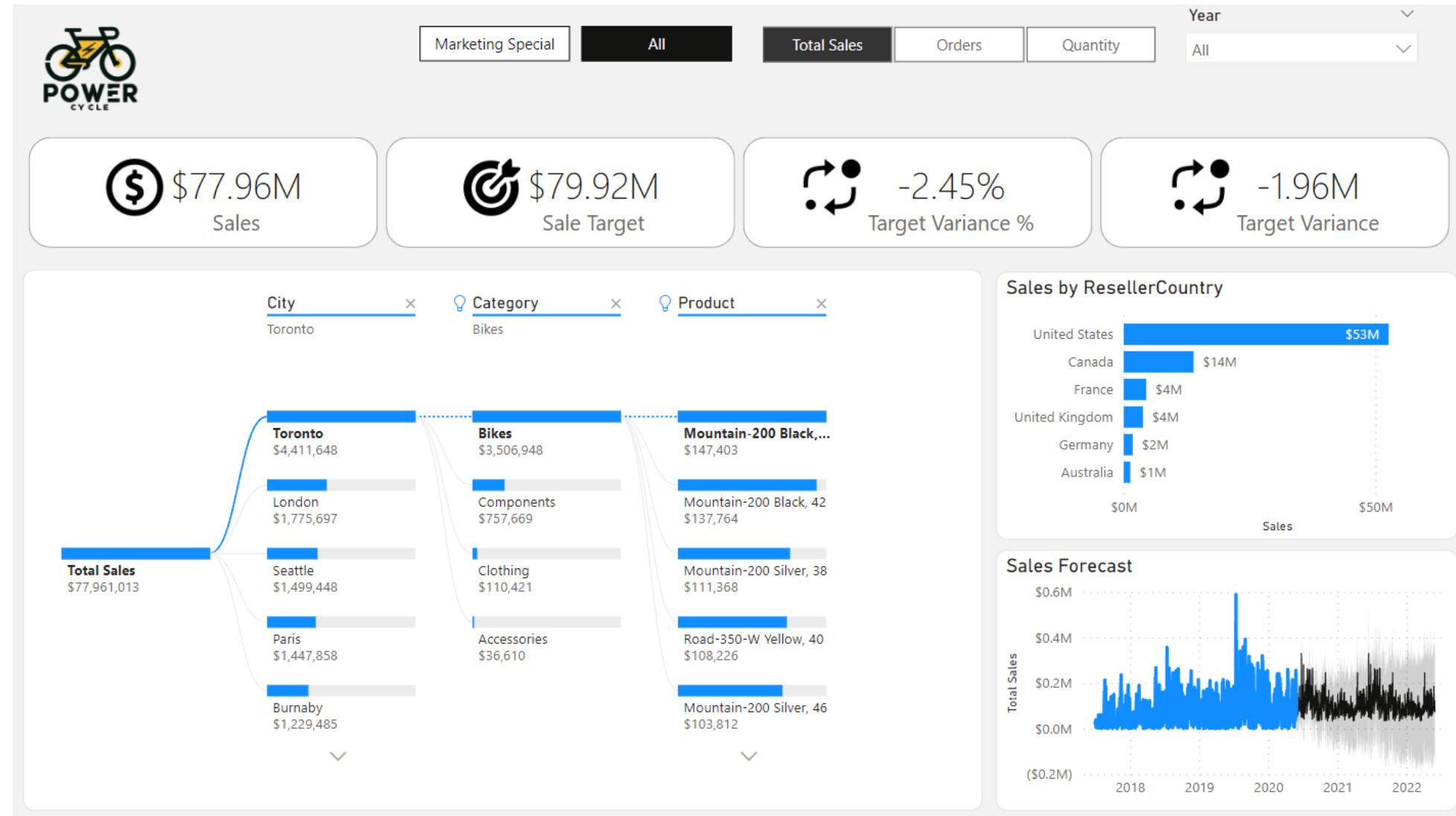
Course Name Here

- 📁 Lab Resources ←
- 📁 Lab Solutions ←
- 📁 My Solutions ←
- 📁 Power Cycle Dataset
- 📁 RLS Data

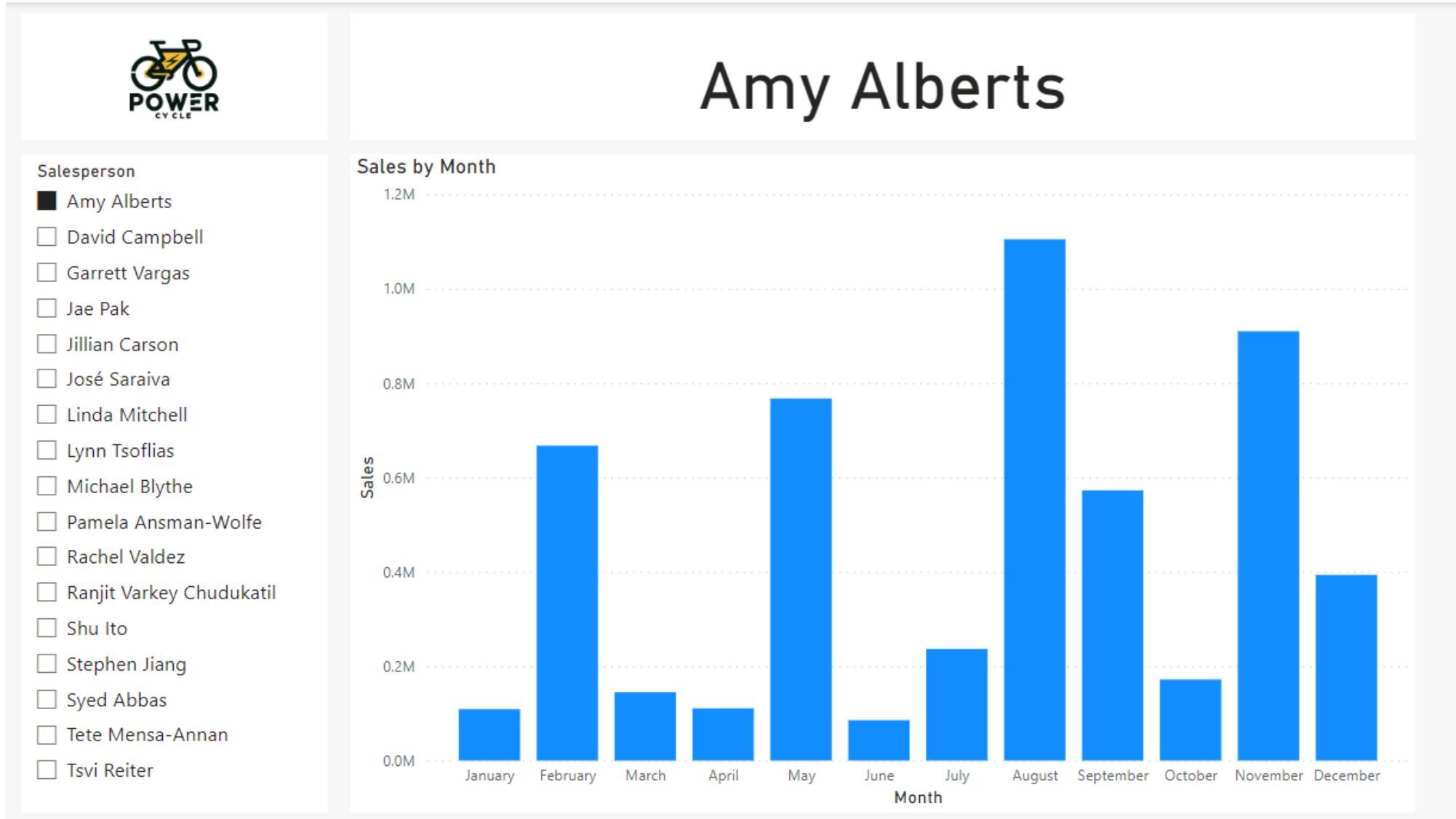
- 🔗 Lab 1 Data Import & Transformation Part 1.pdf
- 🔗 Lab 2 Data Import & Transformation Part 2.pdf
- 🔗 Lab 3 Model Data in Power BI.pdf
- 🔗 Lab 4 Create Model Calculations using DAX.pdf
- 🔗 Lab 5 Advanced Calculate Function and DAX Patterns.pdf
- 🔗 Lab 6 DAX Time Series.pdf
- 🔗 Lab 7 Visualizing Data in Power BI Desktop.pdf
- 🔗 Lab 8 - Row Level Security.pdf

- 🔗 Lab 2 (Start Up) Data Import & Transformation Part 2.pbix
- 🔗 Lab 3 (Start Up) Model Data in Power BI.pbix
- 🔗 Lab 4 (Start Up) Create Model Calculations using DAX.pbix
- 🔗 Lab 5 (Start Up) Advanced Calculate Function and DAX Patterns.pbix
- 🔗 Lab 6 (Start Up) DAX Time Series.pbix
- 🔗 Lab 7 (Start Up) Visualizing Data in Power BI Desktop.pbix
- 🔗 Lab 8 (Start Up) - RLS .pbix

Final Report



Employee RLS Page

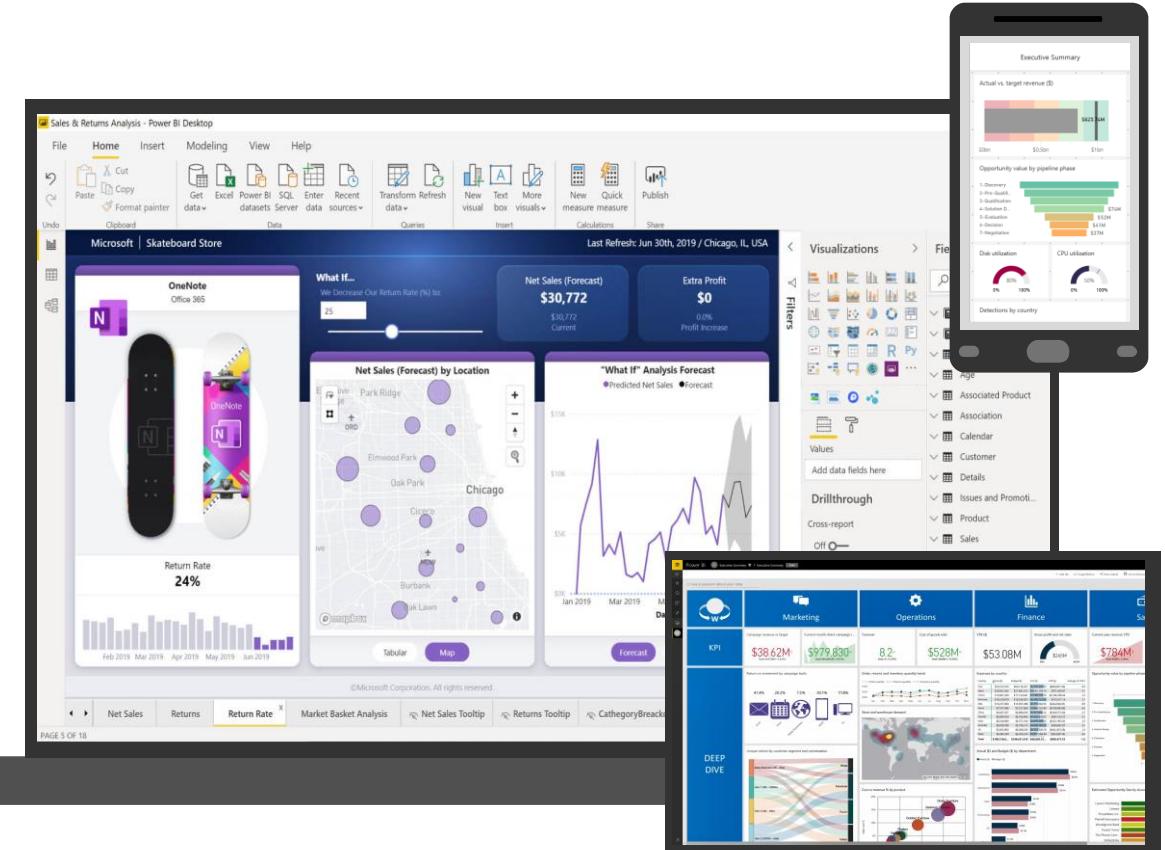


Power BI Overview

Connect to all your data and get a consolidated view across your business through a single pane of glass

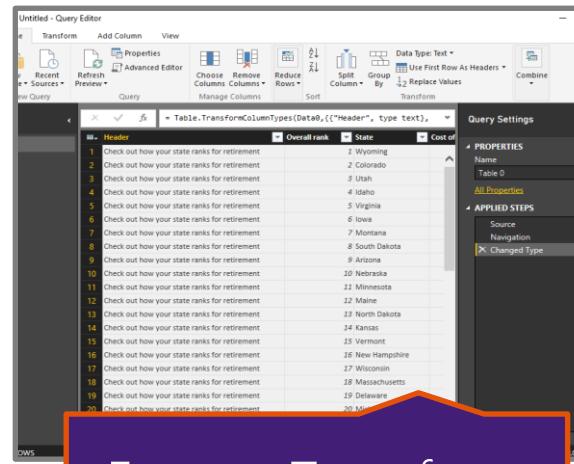
Create ad-hoc analysis, live dashboards and interactive reports that are easy to consume on the web and across mobile devices

Build smart apps by infusing insights from your data and drive action with the power of Microsoft Power Platform

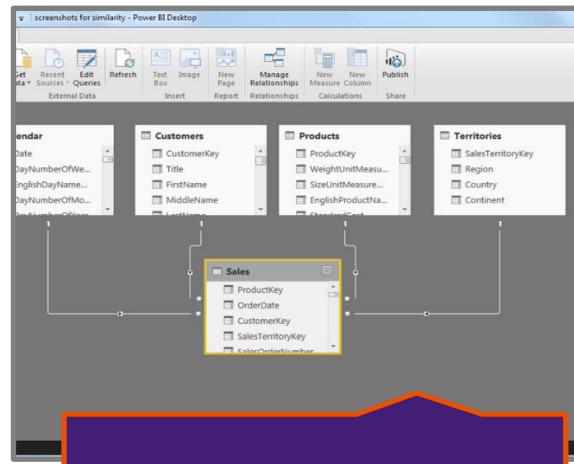


4 Layers of Power BI

Model Developer

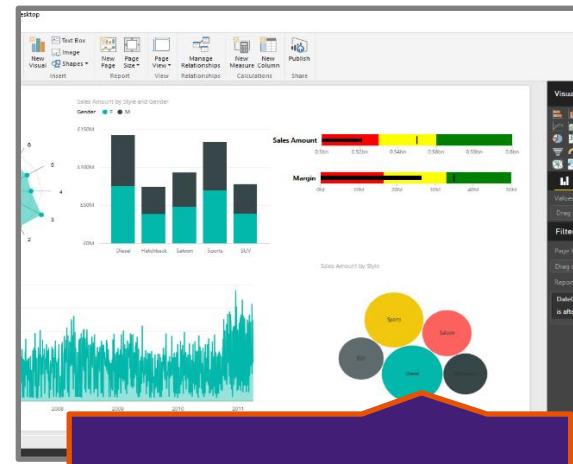


Extract, Transform
and Load (ETL)

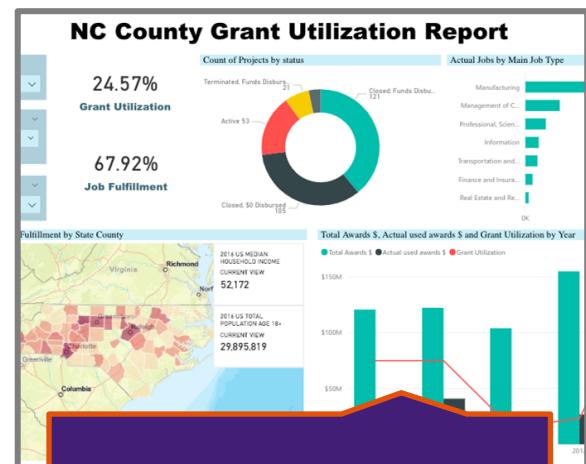


Data Modeling

Report Developer



Report Design



Web Portal

Introduction to the Lab Exercises

1. The data set is from a fictitious cycling retailer with an international footprint. Your goal as the analyst is to use Power BI and DAX calculations to answer key questions about the business including sales trends, top performing resellers, actual sales vs. budget and segmentation.
2. There will be a total of 6 Lab to be completed. Each Lab will build upon the previous one
3. All you've been given is a collection of raw csv files containing POS sales records, along with details about the company's products, customers, locations and employees.

*This data is adapted from Microsoft Adventure works data pack and is for informational purposes only. These samples are provided "as is" without warranty of any kind. The example companies, organizations, products, domain names, email addresses, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, person, place, or event is intended or should be inferred.

Lesson 1

Getting data from different sources

After completing this module, students will be able to:

- Identify and retrieve data from different data sources
- Understand the connection methods and their performance implications
- Connecting to Sharepoint and Power BI services
- Understanding storage modes

Source Types

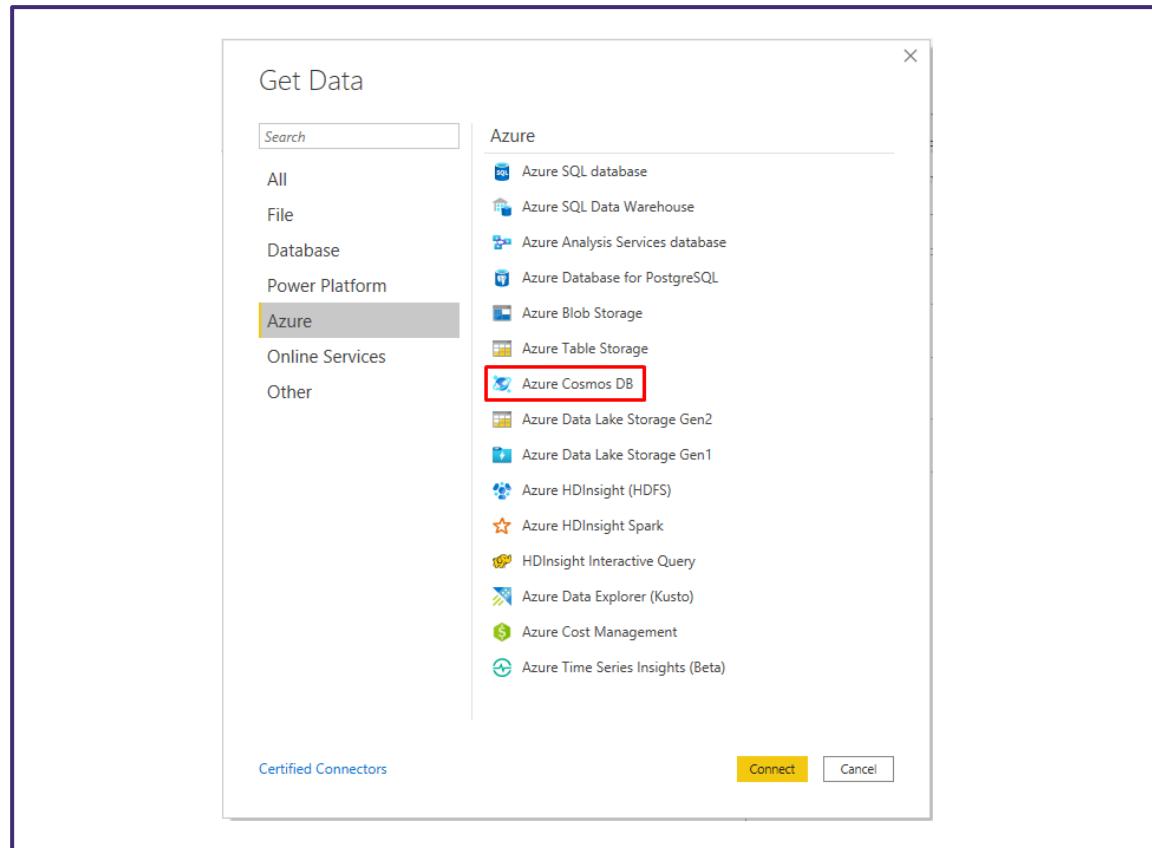
- Flat File
- Relational Data Sources
- Non-SQL Data (unstructured)
- Online Services
- API's



Get data from relational data sources



Unstructured Data Source



The screenshot shows the Power Query Editor interface titled 'Module 3 - Power Query Editor'. The 'File' tab is selected, and the 'Close & Apply' button is highlighted with a red box. The 'Queries [9]' pane on the left lists several queries: Category, Sales, Territory, Order, Budget, Country, New Country, Product, and Employee Data. The 'Employee Data' query is currently selected. The main pane displays a table with columns: Department (highlighted in yellow), Extension, and Position Title. The data shows marketing and operations roles with their respective counts and titles. A formula bar at the top indicates the current step: `= Table.TransformColumnTypes(#"Promoted Headers",{{"Department", Text}})`.

Department	Extension	Position Title
MARKETING	425	Marketing Advisor
MARKETING	206	Marketing Advisor
MARKETING	207	Brand Manager
MARKETING	349	Senior Brand Manager
MARKETING	425	Marketing - Coordina...
MARKETING	210	Marketing - Coordina...
MARKETING	208	Marketing Consultant
MARKETING	249	Marketing Consultant
OPERATIONS	425	Supervisor

Getting Data From SharePoint

The screenshot shows the Power BI desktop interface. On the left, a navigation pane lists 'Anonymous', 'Windows', and 'Microsoft account'. A red box highlights the 'Sign in' button in the Microsoft account section. In the center, a 'SharePoint' connection dialog is open, showing the URL <https://themeasuredproduct.sharepoint.com/>. A red box highlights the 'Sign in' button. Below the dialog are 'Back', 'Connect', 'Cancel', and 'Extract Table Using Examples' buttons. To the right, the 'Navigator' window is open, displaying a hierarchical list of SharePoint sites and lists. The 'BudgetRequests' list is selected, indicated by a checked checkbox. The 'BudgetRequests' table is shown in a preview pane with the following data:

FileSystemObjectType	Id	ServerRedirectedEmbedUri	ServerRedirectedEmbed
0	1		null
0	2		null
0	3		null
0	4		null
0	5		null

At the bottom right of the Power BI interface are 'Load', 'Transform Data', and 'Cancel' buttons.

Get data from Analysis Services

An analytical data engine that lets you digest data from multiple data sources and create calculations on the fly.

SQL Server Analysis Services database

Server ⓘ

asazure://westus.asazure.windows.net/azureanalysisservicesiketest

Database (optional)

adventureworks

Import

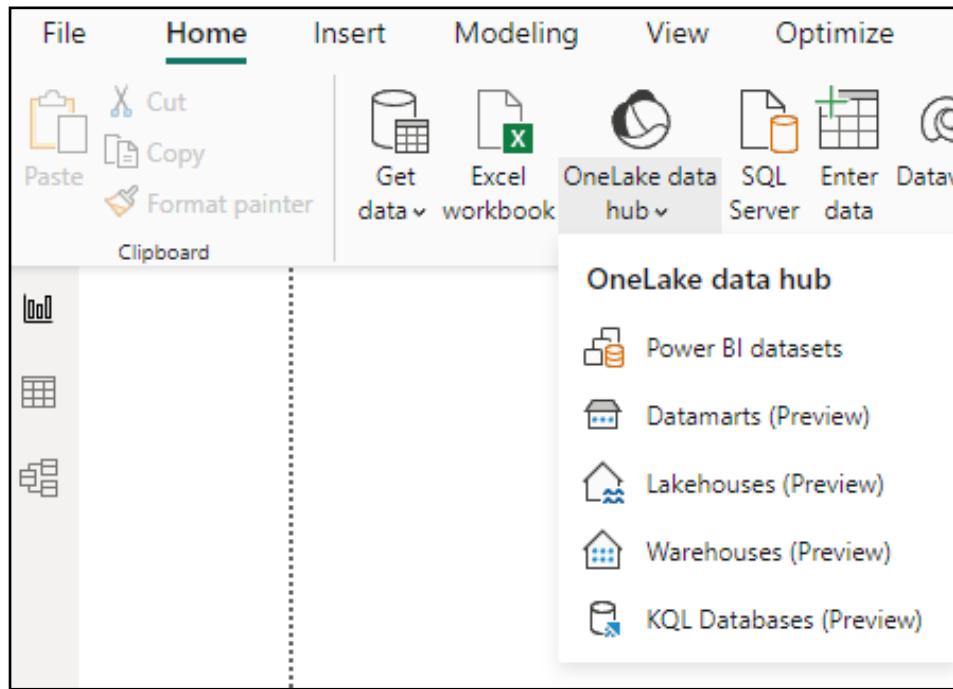
Connect live

▷ MDX or DAX query (optional)

OK

Cancel

Power BI Published Dataset



The screenshot shows a search results dialog for the 'OneLake data hub'. The title bar says 'OneLake data hub'. The main area displays a list of datasets with columns for Name, Refreshed, Location, Endorsement, and Sensitivity. The 'Endorsement' column includes a 'Promoted' badge for one item. The 'Connect' and 'Cancel' buttons are at the bottom right.

Name	Refreshed	Location	Endorsement	Sensitivity
Lab 6 -Advanced Visualization...	9/17/23, 10:56:37 PM	Power BI Intermediate Tr...	Promoted	-
Lab 7 -RLS	8/17/23, 3:04:07 PM	Power BI Intermediate Tr...	-	-
Purview Hub	9/17/23, 11:27:12 PM	Admin monitoring	-	-
Sales Report	9/12/23, 2:50:06 PM	Smart Dolphins Demo	-	-
CRM Analytics for Dynamics 3...	9/5/23, 11:52:58 PM	Agile CRM Analytics for ...	-	-
VAS_API 8.7.23	8/11/23, 5:26:57 PM	My Workspace	-	-
NC Grants Lab 5 Complete	7/14/23, 4:07:22 PM	Power BI Training	-	-

Web Services & API's

From Web

Basic Advanced

URL

A_C http://api.openmetrolinx.com/OpenDataAPI/api/v1/Gtfs/Feed/VehiclePosit

OK Cancel

= Json.Document(Web.Contents("http://api.openmetrolinx.com/OpenDataAPI/api/v1/Gtfs/Feed/VehiclePosition?key=30020173"))

header Record

entity List

trip.trip_id entity.vehicle.trip.route_id entity.vehicle.trip.direction_id entity.vehicle.trip.start_time entity.vehicle.trip.start_date entity.vehicle.trip.schedule_relationship entity.vehicle.position

Query Settings

Properties

Name: VehiclePosition?key=30020173

All Properties

Applied Steps

Source
Converted to Table
Expanded header
Expanded entity
Expanded entity1
Expanded entity.vehicle
Expanded entity.vehicle.trip
Expanded entity.vehicle.vehicle
Expanded entity.vehicle.position
Changed Type

trip.trip_id	entity.vehicle.trip.route_id	entity.vehicle.trip.direction_id	entity.vehicle.trip.start_time	entity.vehicle.trip.start_date	entity.vehicle.trip.schedule_relationship	entity.vehicle.position
1 00	09231223-41	0	6:30:00 AM	20230918	SCHEDULED	
2 11	09231223-41	1	6:55:00 AM	20230918	SCHEDULED	
3 20	09231223-56	1	7:00:00 AM	20230918	SCHEDULED	
4 20	09231223-41	0	7:00:00 AM	20230918	SCHEDULED	
5 31	09231223-56	0	7:10:00 AM	20230918	SCHEDULED	
6 40	09231223-33	0	7:25:00 AM	20230918	SCHEDULED	
7 50	09231223-56	1	7:25:00 AM	20230918	SCHEDULED	
8 51	09231223-41	1	7:25:00 AM	20230918	SCHEDULED	

Power BI Desktop Data Storage Modes



Import

- Dataset is less than 1GB (after compression) & fast performance
- Source data does not change frequently
- No restrictions on Power Query, data modeling, and DAX functions



DirectQuery

- Dataset is too large to be stored in-memory
- Source data changes frequently and reports must reflect changes
- Company policy states that data can only be accessed from the original source



Dule (Composite Model)

- Boost performance by setting appropriate storage for each table
- Combine a DirectQuery model with additional imported data
- Create a single model from two or more DirectQuery models



Live Connection

- Create one dataset that serves as a central source of truth
- Analyst teams can create different reports from the same source
- Multi-developer teams where one user builds the model and another works on visualization

Implications of using DirectQuery

Benefits

- Frequently changing data
- Need near real-time
- Large data volumes
- Multi-dimensional data

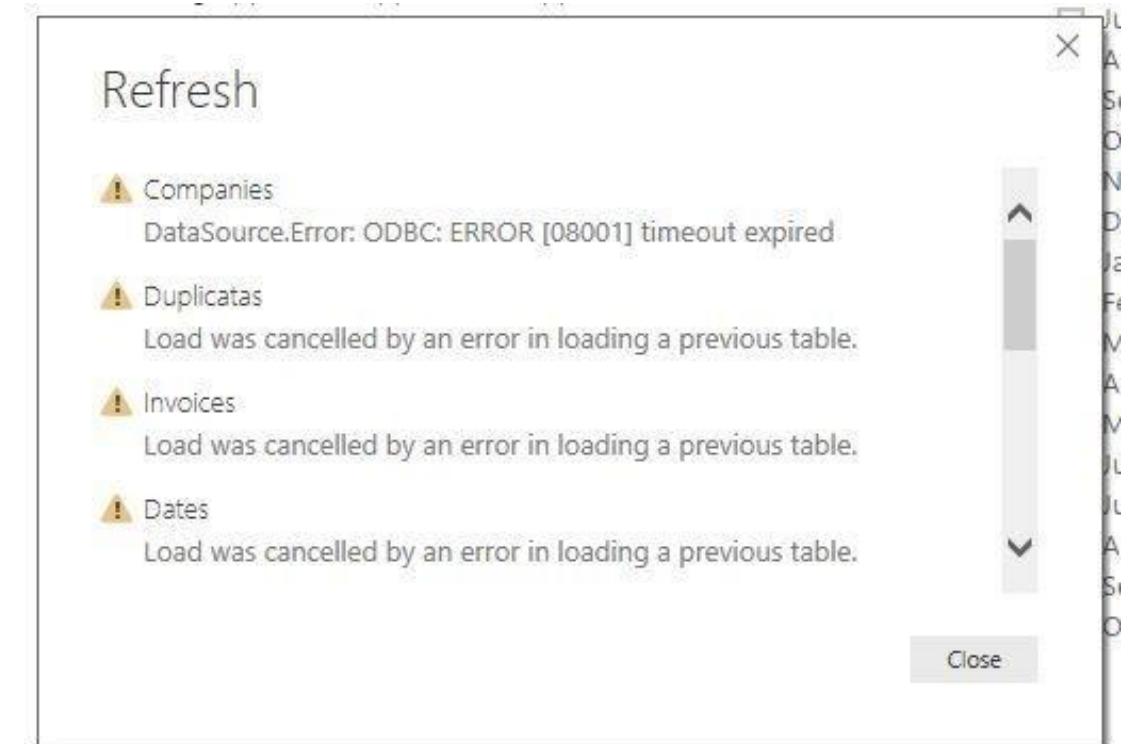
Limitations

- Performance: Dependent on the underlying data source.
- Security: Understand how data moves between source and destination.
- Modeling: Some modeling capabilities are limited or aren't supported.
- Transformation: Some data transformation techniques are limited.

Dealing with Import Errors

When loading your data, you may encounter the following errors:

- *Query Timeout*
- *Couldn't find data formatted as a table*
- *Could not find file*
- *Data type errors*



Chat Question

What are some common data import errors do you encounter when creating reports?



Optimize Query Performance



Performance in Power Query depends heavily on the performance at the data source.

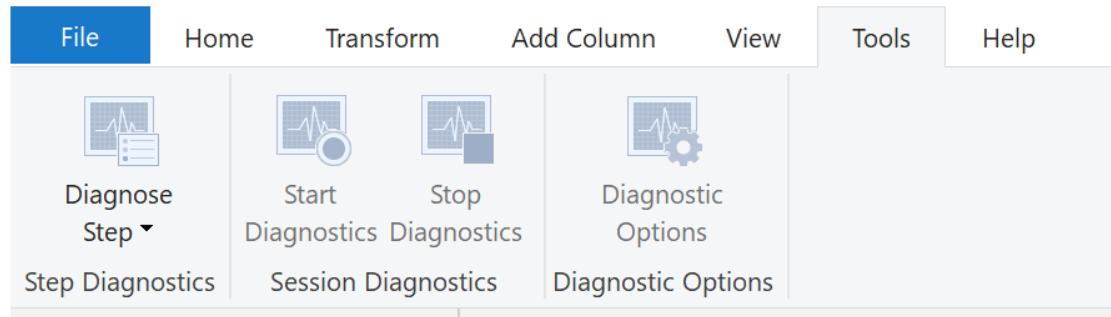


Follow performance tuning guidelines of the source product.



Some performance tuning can be done in Power BI.

Fix Performance Issues



A ^B Id	A ^B Query	A ^B Step	A ^B Exclusive Duration	A ^B Category	A ^B Data Source Kind
1.1	Product	Changed Type	0.00:00:00.0457545	Evaluator	null
1.2	Product	Source	0.00:00:01.9567741	Evaluator	null
Diagnostics_2020-0...					

KC 1



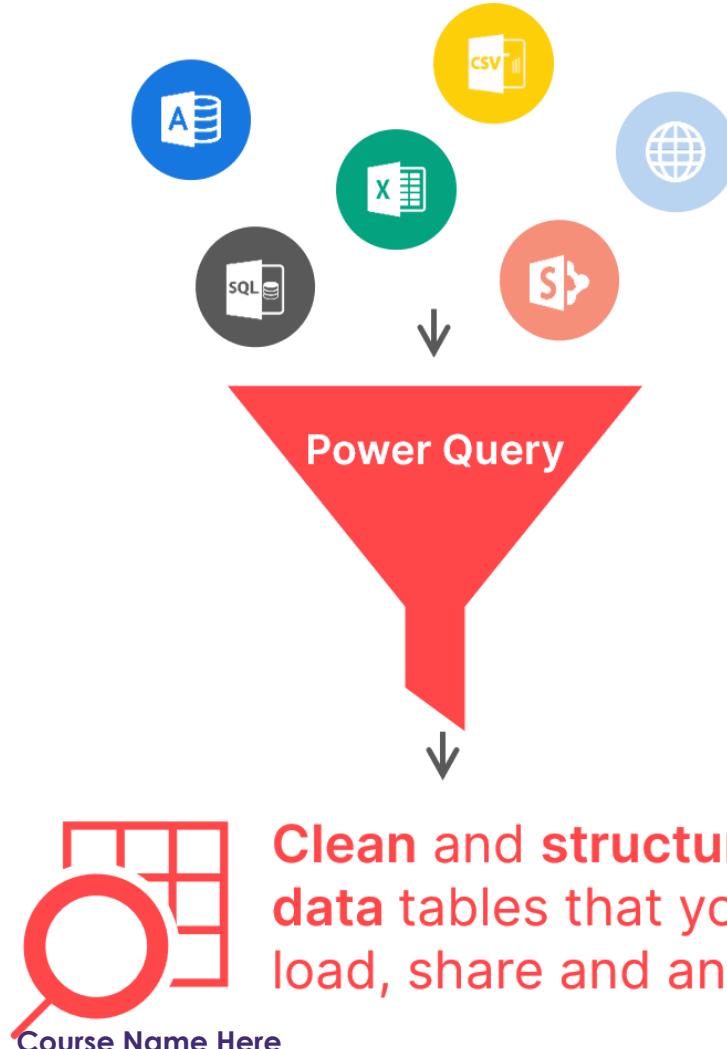
Lesson 2

Clean Transform, Load the Data

After completing this module, students will be able to:

- Apply data shape transformations
- Enhance the structure of the data
- Profile and examine the data

Power Query Editor



Before

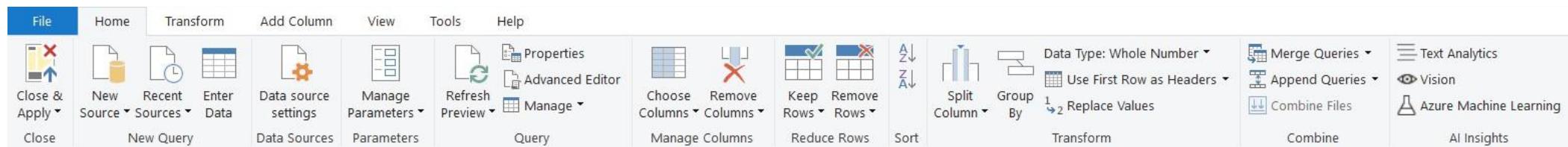
	Date	Customer	Prod 1	Kg 1	Bag 1	Prod 2	Kg 2	Bag 2
1	1/1/2019	A	Rice	10	1	Potato	100	10
2	1/2/2019	A	Potato	20	2	Rice	200	20
3	1/3/2019	B	Tomarto	30	3	Beans	300	30
4	1/4/2019	C	Beans	40	4	Beans	400	40

After

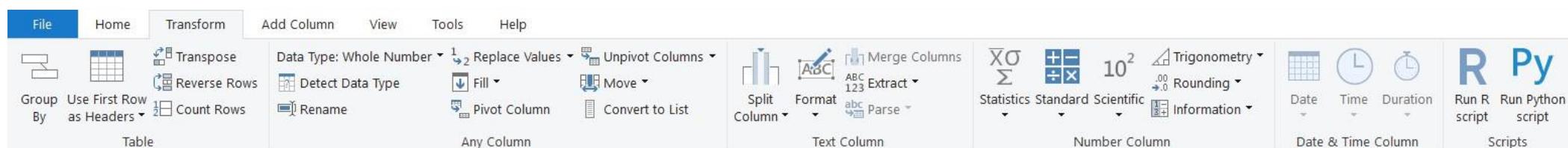
	Date	Customer	Product	Kg	Bag
1	2019-01-01	A	Potato	100	10
2	2019-01-01	A	Rice	10	1
3	2019-01-02	A	Potato	20	2
4	2019-01-02	A	Rice	200	20
5	2019-01-03	B	Beans	300	30
6	2019-01-03	B	Tomarto	30	3
7	2019-01-04	C	Beans	440	44

Common transformations

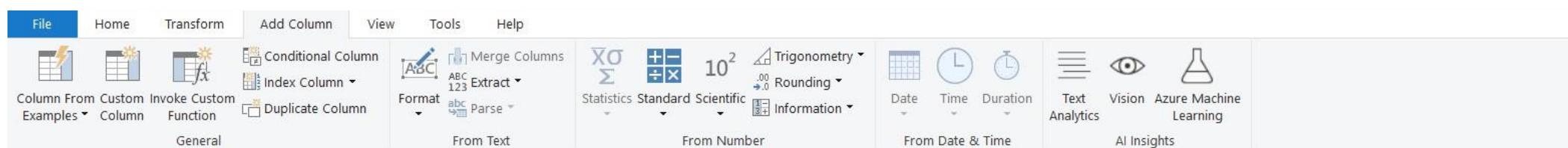
The **HOME** tab includes general settings and common table transformation tools



The **TRANSFORM** tab includes tools to modify existing columns (splitting/grouping, transposing, extracting text, etc.)



The **ADD COLUMN** tools create new columns (based on conditional rules, text operations, calculations, dates, etc.)



Power Query Interface

The screenshot illustrates the Microsoft Power Query interface with several key components highlighted by red boxes and arrows:

- Queries List (Left):** A sidebar titled "Queries [13]" lists all available queries. One query, "Targets", is selected and highlighted with a red box. A red arrow points from the text "Queries are listed and available for selection, viewing, and shaping" to this list.
- Ribbon (Top):** The ribbon bar contains various tabs like File, Home, Transform, Add Column, View, Tools, and Help. A red box highlights the "Transform" tab, and another red arrow points from the text "In the ribbon, buttons are active to interact with the data in the query" to the ribbon area.
- Center Pane (Main Area):** The main workspace displays the data from the selected query, "Targets". It shows a table with columns EmployeeID, Date, and Target. Below the table, there are three small charts: a bar chart for EmployeeID, a bar chart for Date, and a histogram for Target. A red box highlights the table area, and a red arrow points from the text "In the center pane, data from the selected query is displayed and available for shaping" to the table.
- Query Settings (Right):** A floating window titled "Query Settings" lists the properties and applied steps for the selected query. A red box highlights the "APPLIED STEPS" section, which includes steps like "Source", "Navigation", "Changed Type", etc., and a step labeled "Replaced Errors" which is also highlighted with a red box. A red arrow points from the text "The Query Settings window appears, listing the query's properties and applied steps" to this window.

Course Name Here

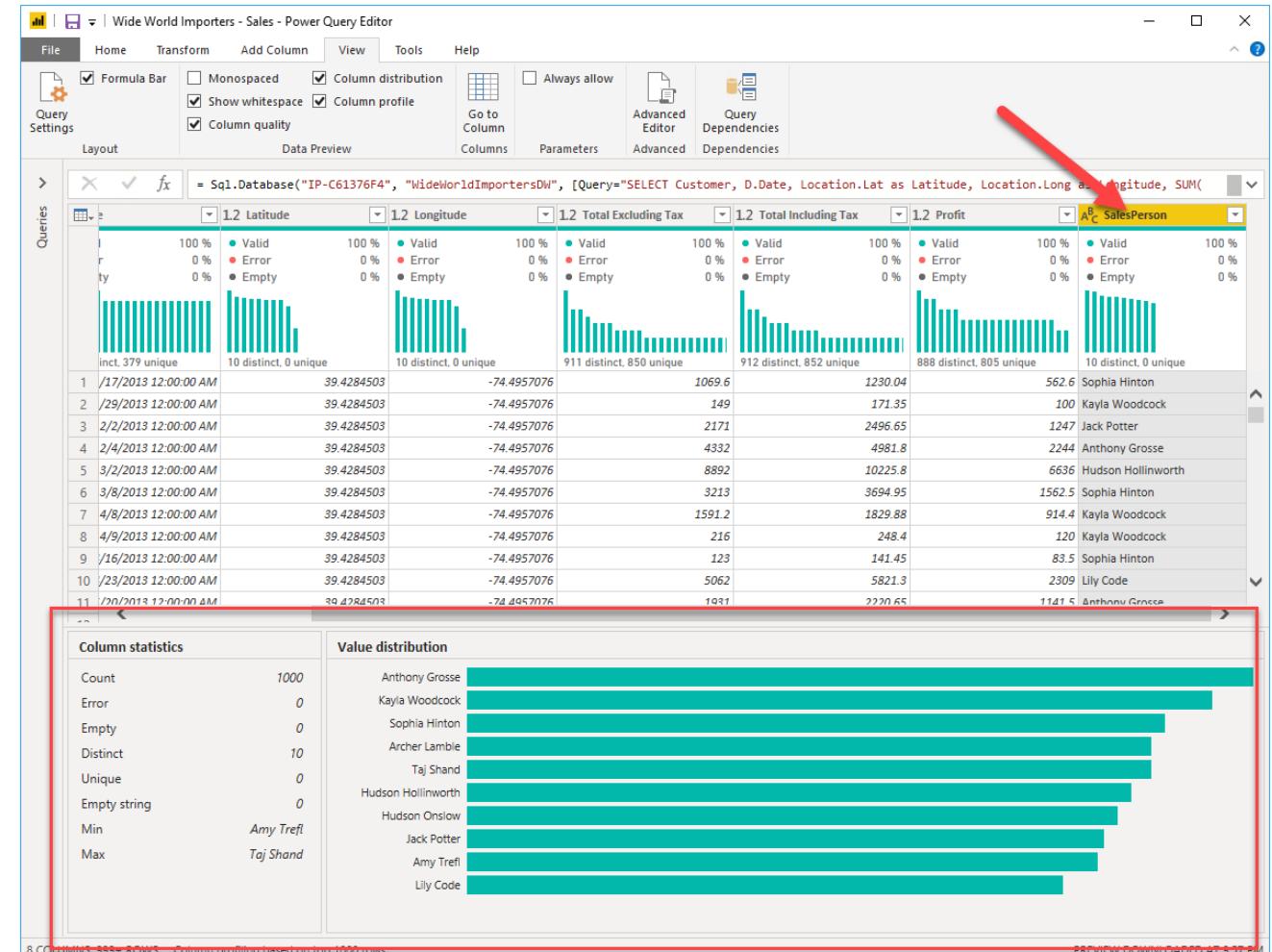
3 COLUMNS, 918 ROWS Column profiling based on top 1000 rows

PREVIEW DOWNLOADED AT 3:32 PM DATA SOCIETY © 2023 31

Find data anomalies and data statistics

Data profiling entails studying data nuances, detecting anomalies, exploring underlying structures, and querying statistics like row counts and value distributions. This is crucial for simplifying data interaction and streamlining the development of report elements on the frontend.

Note: Data Preview is based on the first 1,000 rows



Combine multiple queries into one

Append

Concatenate rows from three or more tables into a single table.

Two tables Three or more tables

Available tables

Production Suppliers
Sales Customers
HR Employees

Add >>

Tables to append

Production Suppliers
Sales Customers
HR Employees

OK

Merge

Select a table and matching columns to create a merged table.

Sales Orders

orderid	custid	empid	orderdate	requireddate	shippeddate	shipperid	freight	shipname
10248	85	5	7/4/2014	8/1/2014	7/16/2014	3	32.38	Ship to 85-B
10249	79	6	7/5/2014	8/16/2014	7/10/2014	1	11.61	Ship to 79-C
10250	34	4	7/8/2014	8/5/2014	7/12/2014	2	65.83	Destination SCQ
10251	84	3	7/8/2014	8/5/2014	7/15/2014	1	41.34	Ship to 84-A

Sales OrderDetails

orderid	productid	unitprice	qty	discount
10248	11	14.00	12	0
10248	42	9.80	10	0
10248	72	34.80	5	0
10249	14	18.60	9	0
10249	51	42.40	40	0

Join Kind

Left Outer (all from first, matching from second)

Use fuzzy matching to perform the merge

Fuzzy matching options

✓ The selection matches 830 of 830 rows from the first table.

OK Cancel

Splitting Columns

The screenshot illustrates the process of splitting a single column into multiple columns in Power Query. On the left, a single column named "Reseller CityState" is shown. This column contains a list of city-state pairs like "Rhodes-New South Wales". A red box highlights this column. A large green arrow points from the original column to the result on the right. The result shows three new columns: "Reseller", "Reseller CityState.1", and "Reseller CityState.2". The "Reseller" column lists resellers such as "Nationwide Supply". The "Reseller CityState.1" column lists cities: "Rhodes", "Newcastle", "Darlinghurst", "Sydney", "Melbourne", "Matraville", "North Sydney", "Seaford", and "North Ryde". The "Reseller CityState.2" column lists states: "New South Wales", "Australia", "New South Wales", "Australia", "New South Wales", "Australia", "Victoria", "New South Wales", and "New South Wales". The Power Query ribbon at the top shows the "Text Column" tab selected, and the "Split Column" icon is highlighted with a red box.

To split columns in Power Query, first, select the column you want to divide. Then, use the "Split Column" function and specify whether you want to split by a delimiter (like a comma or space) or by a fixed number of characters.

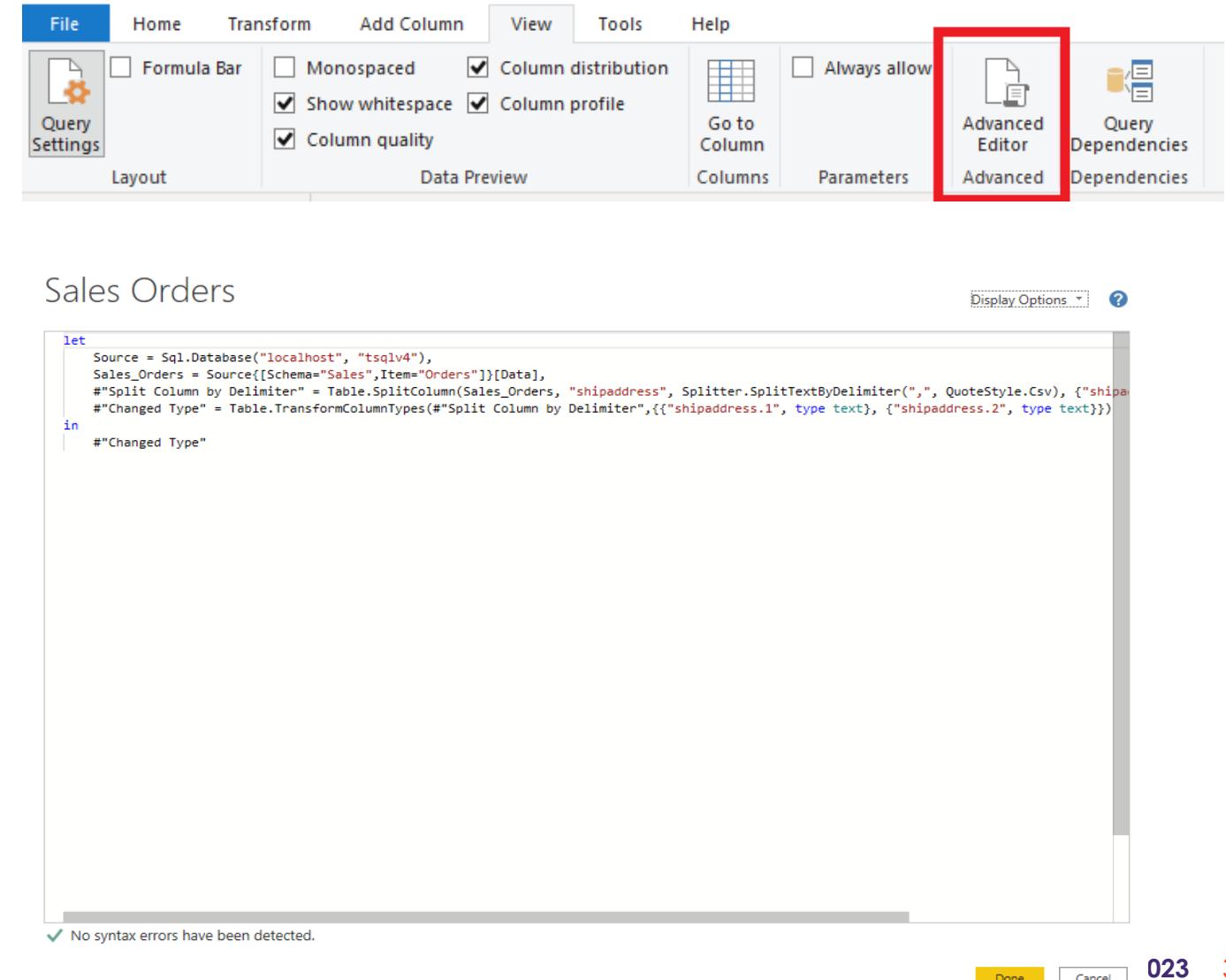
Unpivot columns or pivot

	A ^B _C Category Name	A ^B _C Subcategory Name
1	Bikes	Mountain Bikes
2	Bikes	Road Bikes
3	Bikes	Touring Bikes
4	Clothing	Bib-Shorts
5	Clothing	Caps
6	Clothing	Gloves
7	Clothing	Jerseys
8	Clothing	Shorts
9	Clothing	Socks
10	Clothing	Tights
11	Clothing	Vests
12	Accessories	Bike Racks
13	Accessories	Bike Stands
14	Accessories	Bottles and Cages

	1.2 Bikes	1.2 Components	1.2 Clothing	1.2 Accessories
1	3	14	8	12

Use Advanced Editor to modify M code

When shaping data in Power Query, each action creates a step in the process that can be reordered, deleted, or modified. While the graphical interface is commonly used, Power Query operates with the underlying M language.



The screenshot shows the Power Query Advanced Editor interface. At the top, there's a ribbon with tabs: File, Home, Transform, Add Column, View (which is selected), Tools, and Help. Below the ribbon is a toolbar with icons for Query Settings, Formula Bar, Monospaced, Column distribution, Show whitespace, Column profile, Column quality, Go to Column, Always allow, Advanced Editor (which is highlighted with a red box), Query Dependencies, and Advanced Dependencies. The main area is titled "Sales Orders" and contains M code:

```
let
    Source = Sql.Database("localhost", "tsqlv4"),
    Sales_Orders = Source{[Schema="Sales",Item="Orders"]}[Data],
    #"Split Column by Delimiter" = Table.SplitColumn(Sales_Orders, "shipaddress", Splitter.SplitTextByDelimiter(", ", QuoteStyle.Csv), {"shipaddress.1", "shipaddress.2"}),
    #"Changed Type" = Table.TransformColumnTypes(#"Split Column by Delimiter",{{"shipaddress.1", type text}, {"shipaddress.2", type text}})
in
    #"Changed Type"
```

At the bottom of the editor, a status bar says "No syntax errors have been detected." There are "Done" and "Cancel" buttons at the bottom right.

Data reduction best practices

- ✓ Remove unnecessary columns
- ✓ Remove unnecessary rows
- ✓ Group by and summarize
- ✓ Optimize column data types
- ✓ Preference for custom columns
- ✓ Disable Power Query load
- ✓ Disable auto date/time
- ✓ Switch to Mixed mode (Composite model)



Power BI Options & Settings

Options

GLOBAL

- Data Load
- Power Query Editor
- DirectQuery
- R scripting
- Python scripting
- Security**
- Privacy
- Regional Settings
- Updates
- Usage Data
- Diagnostics
- Preview features**
- Auto recovery
- Report settings

CURRENT FILE

- Data Load**
- Regional Settings**
- Privacy
- Auto recovery
- Published dataset settings
- Query reduction
- Report settings

Preview Features

- Shape map visual [Learn more](#)
- Spanish language support for Q&A [Learn more](#)
- Q&A for live connected Analysis Services databases [Learn more](#)
- Connect to external datasets shared with me [Learn more](#) | [Share feedback](#)
- Modern visual tooltips [Learn more](#) | [Share feedback](#)
- Sparklines [Learn more](#)
- Metrics visual [Learn more](#)
- Quick measure suggestions [Learn more](#) | [Share feedback](#)
- Field parameters [Learn more](#)
- Enhanced row-level security editor [Learn more](#)
- On-object interaction [Learn more](#) | [Share feedback](#)
- Enable setting sensitivity label on exported PDF [Learn more](#)
- Dynamic format string for measures [Learn more](#)
- Save to OneDrive and SharePoint [Learn more](#)
 - Share to OneDrive and SharePoint [Learn more](#)
- Power BI Project (.pbip) save option [Learn more](#)
- New card visual [Learn more](#)

Data Load

Type Detection

- Detect column types and headers for unstructured sources

Relationships

- Import relationships from data sources on first load (i)
 - Update or delete relationships when refreshing data (i)
 - Autodetect new relationships after data is loaded (i)
- [Learn more](#)

Time intelligence

- Auto date/time (i) [Learn more](#)

Background Data

- Allow data previews to download in the background

Parallel loading of tables (i)

Maximum number of concurrent jobs [Learn more](#)

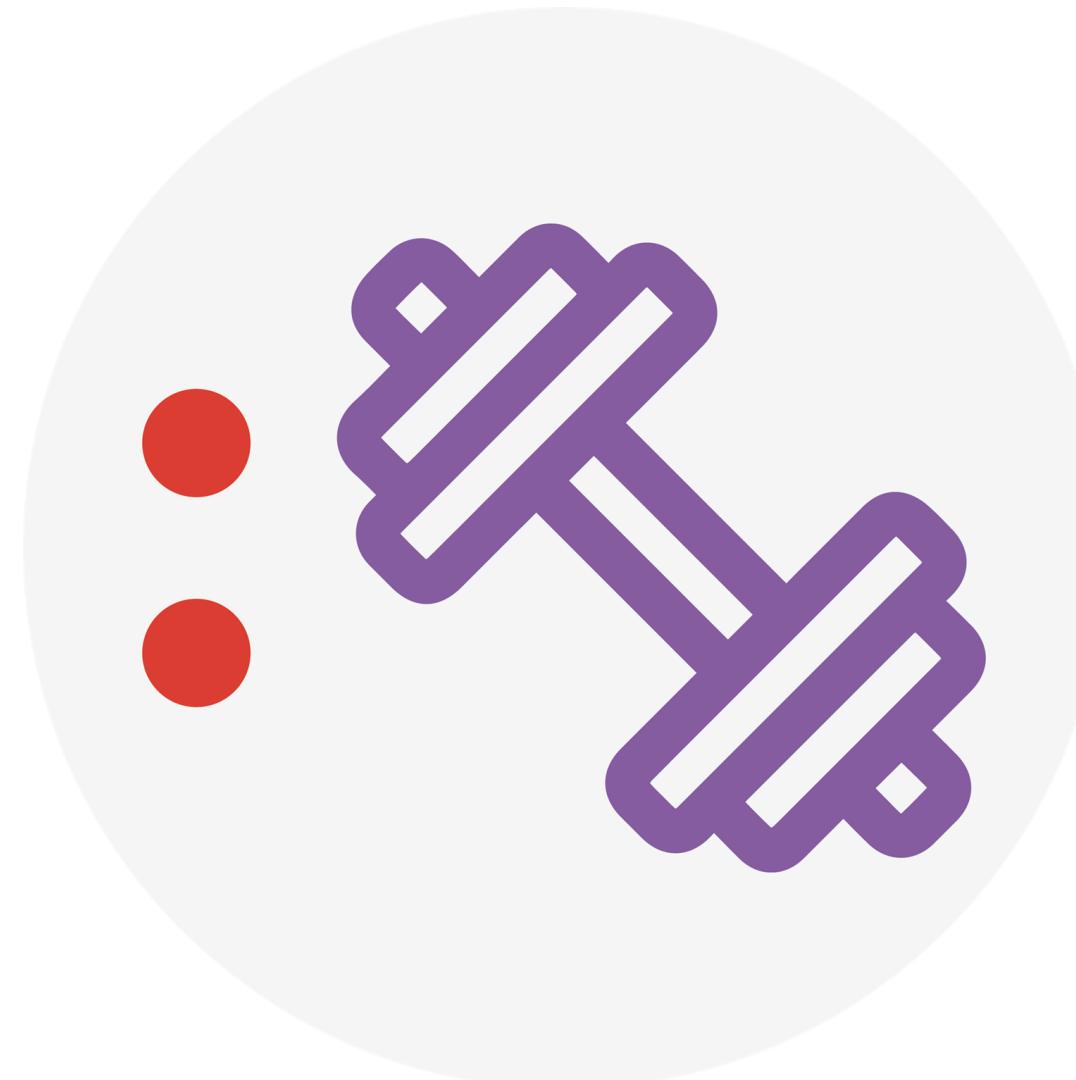
- Default
- One (disable parallel loading)
- Custom

Regional Settings

Locale for import

Locale determines the regional settings used to interpret numbers, dates, and time in imported text for this file.

Live Demo



KC2



DATA^SOCIETY:

Power BI Intermediate

Day 2



Lesson 3

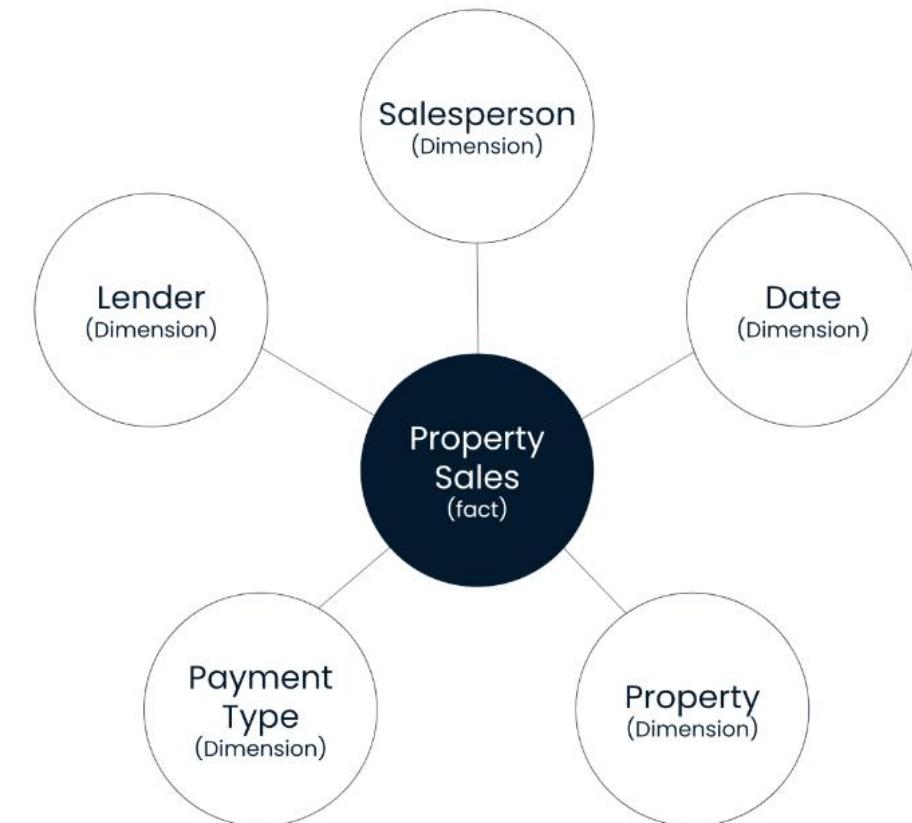
Design a Data Model

After completing this module, students will be able to:

- Define relationships and their cardinality
- Implement Dimensions and Hierarchies
- Understanding filter propagation

STAR SCHEMA

- **Fact tables** contain observational or event data values: sales orders, product counts, prices, transactional dates and times, and quantities.
- **Dimension tables** contain the details about the data in fact tables: products, locations, employees, and order types. These tables are connected to the fact table through key columns. Dimension tables are used to filter and group the data in fact tables



Snowflake Schema

Snowflake schema allows relationships between dimensions and differs from the star schema in that it has more hierarchies between tables.

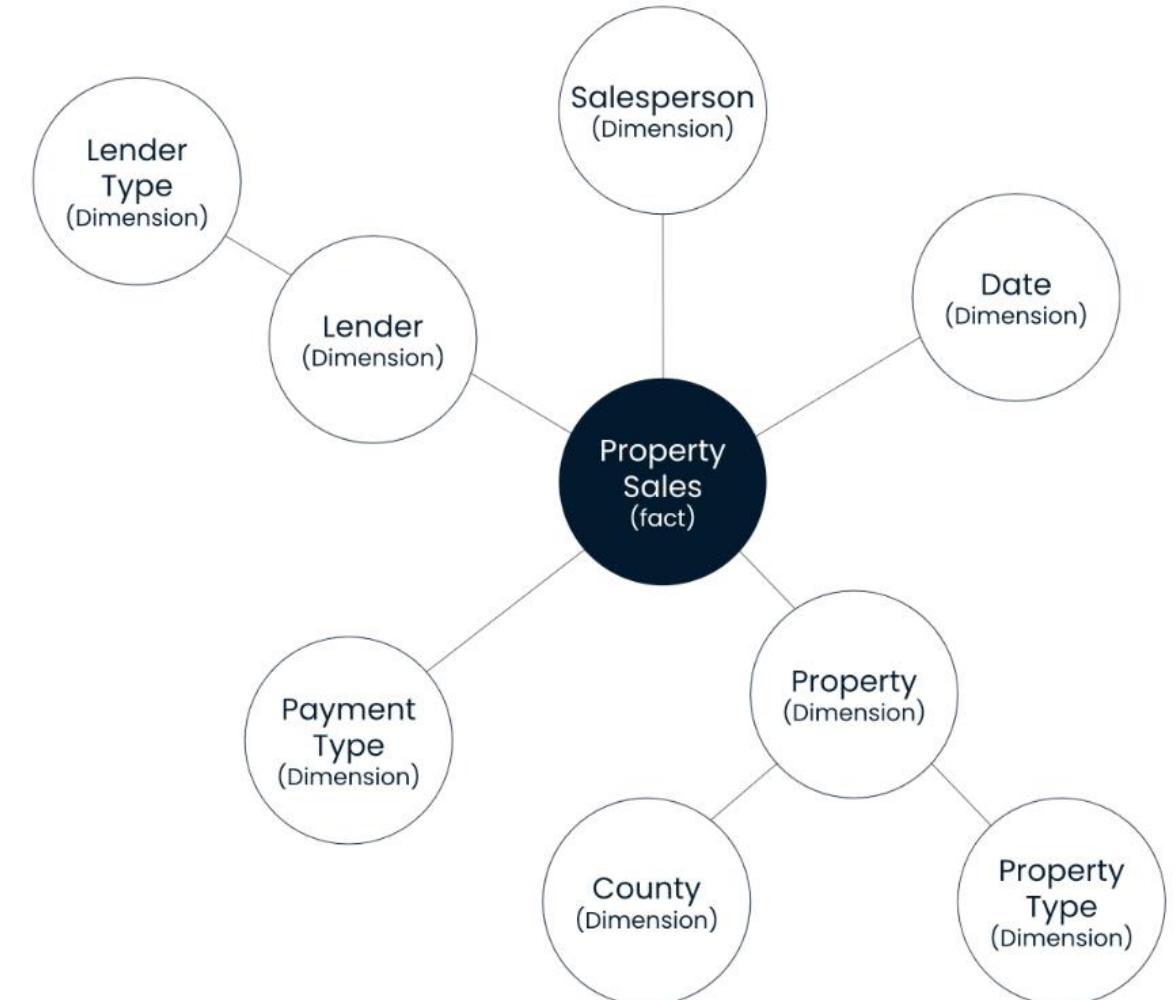
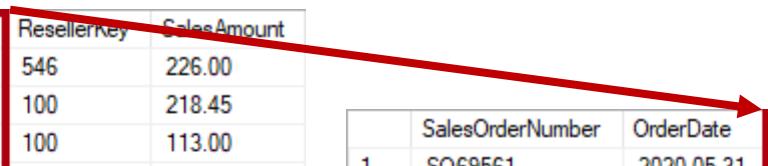


Table Normalization

	SalesOrderNumber	OrderDate	ProductKey	ResellerKey	SalesAmount
1	SO69561	2020-05-31	594	546	226.00
2	SO69560	2020-05-30	513	100	218.45
3	SO69560	2020-05-30	594	100	113.00
4	SO69539	2020-05-28	243	529	858.90
5	SO69539	2020-05-28	378	529	1466.01
6	SO69541	2020-05-28	594	661	113.00
7	SO69542	2020-05-28	243	317	1717.80
8	SO69544	2020-05-28	243	666	3435.60
9	SO69545	2020-05-28	378	436	5864.04
10	SO69532	2020-05-27	594	312	113.00
11	SO69532	2020-05-27	513	312	436.90
12	SO69533	2020-05-27	594	476	226.00



	SalesOrderNumber	OrderDate	ProductKey	Product	Category	Color	Size	ResellerKey	SalesAmount
1	SO69561	2020-05-31	594	Mountain-500 Silver, 48	Bikes	Silver	48	546	226.00
2	SO69560	2020-05-30	513	ML Mountain Frame-W - Silver, 46	Components	Silver	46	100	218.45
3	SO69560	2020-05-30	594	Mountain-500 Silver, 48	Bikes	Silver	48	100	113.00
4	SO69539	2020-05-28	243	HL Road Frame - Red, 44	Components	Red	44	529	858.90
5	SO69539	2020-05-28	378	Road-250 Black, 52	Bikes	Black	52	529	1466.01
6	SO69541	2020-05-28	594	Mountain-500 Silver, 48	Bikes	Silver	48	661	113.00
7	SO69542	2020-05-28	243	HL Road Frame - Red, 44	Components	Red	44	317	1717.80
8	SO69544	2020-05-28	243	HL Road Frame - Red, 44	Components	Red	44	666	3435.60
9	SO69545	2020-05-28	378	Road-250 Black, 52	Bikes	Black	52	436	5864.04
10	SO69532	2020-05-27	594	Mountain-500 Silver, 48	Bikes	Silver	48	312	113.00
11	SO69532	2020-05-27	513	ML Mountain Frame-W - Silver, 46	Components	Silver	46	312	436.90
12	SO69533	2020-05-27	594	Mountain-500 Silver, 48	Bikes	Silver	48	476	226.00

COMPARE FACT AND DIMENSION TABLES

Characteristic	Dimension table	Fact table
Model purpose	Stores business entities	Stores events or observations
Table structure	Includes a key column and descriptive columns for filtering and grouping	Includes dimension key columns and numeric measure columns that can be summarized
Data volume	Typically, contains fewer rows (relative to fact tables)	Can contain numerous rows
Query purpose	To filter and group	To summarize

Relationship & Cardinality

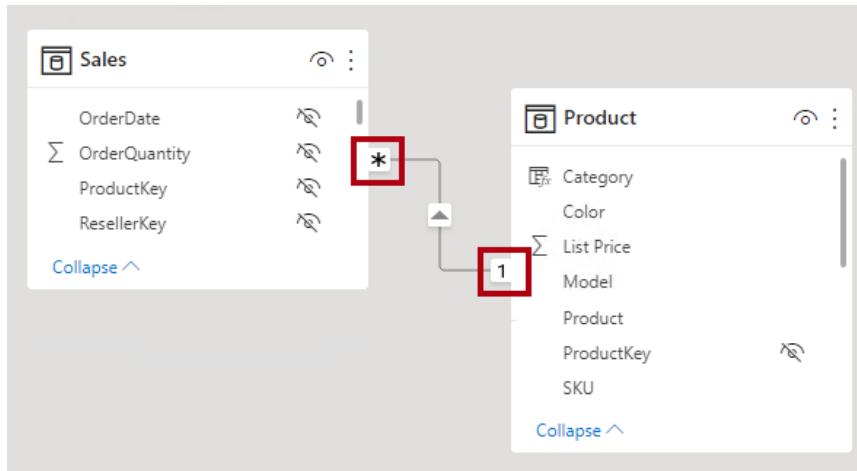
Each model relationship is defined by a cardinality type. There are four cardinality-type options, representing the data characteristics of the "from" and "to" related columns. The "one" side means the column contains unique values; the "many" side means the column can contain duplicate values.

1. One-to-one (1:1)

2. One-to-many (1:*) Many-to-one (*:1)

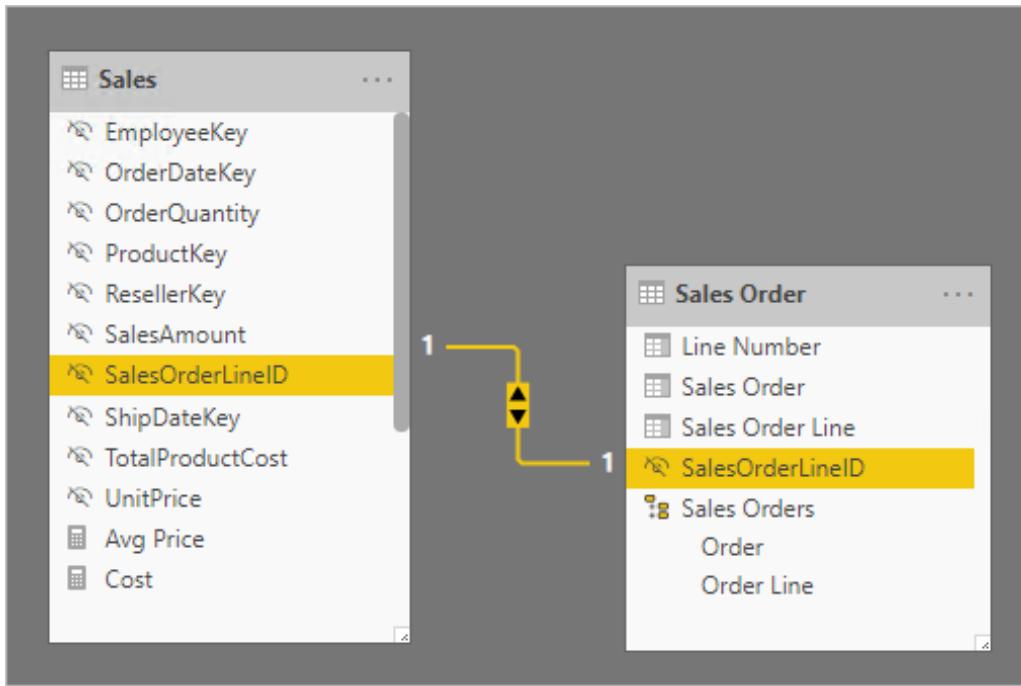
3. Many-to-many (*:*)

One-to-many (1:*) Many-to-one (*:1)



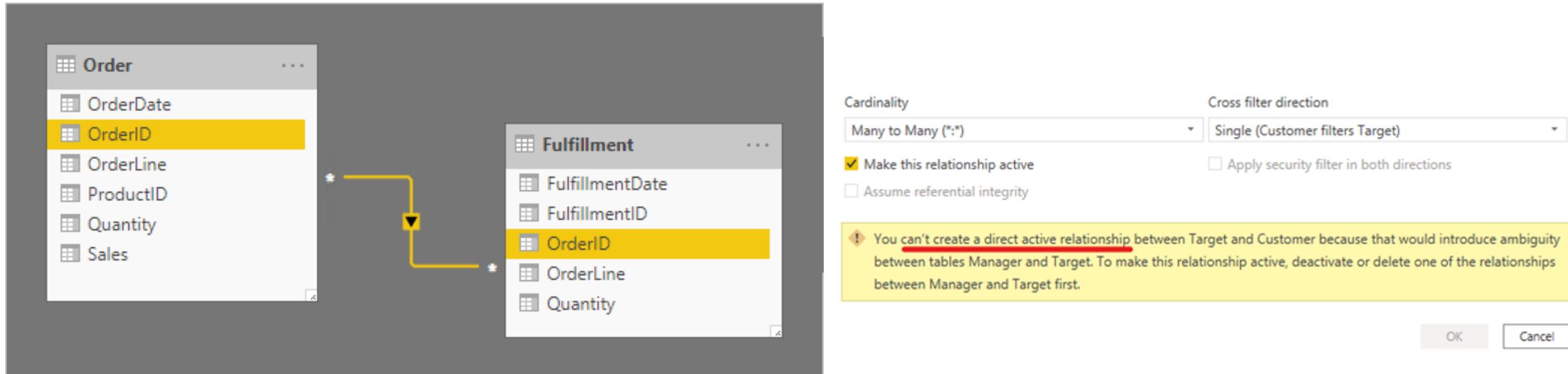
The one-to-many and many-to-one cardinality options are essentially the same, and they're also the most common cardinality types meaning there should be a single instance of each primary key while potentially accommodating multiple instances of each foreign key.

One-to-one (1:1)



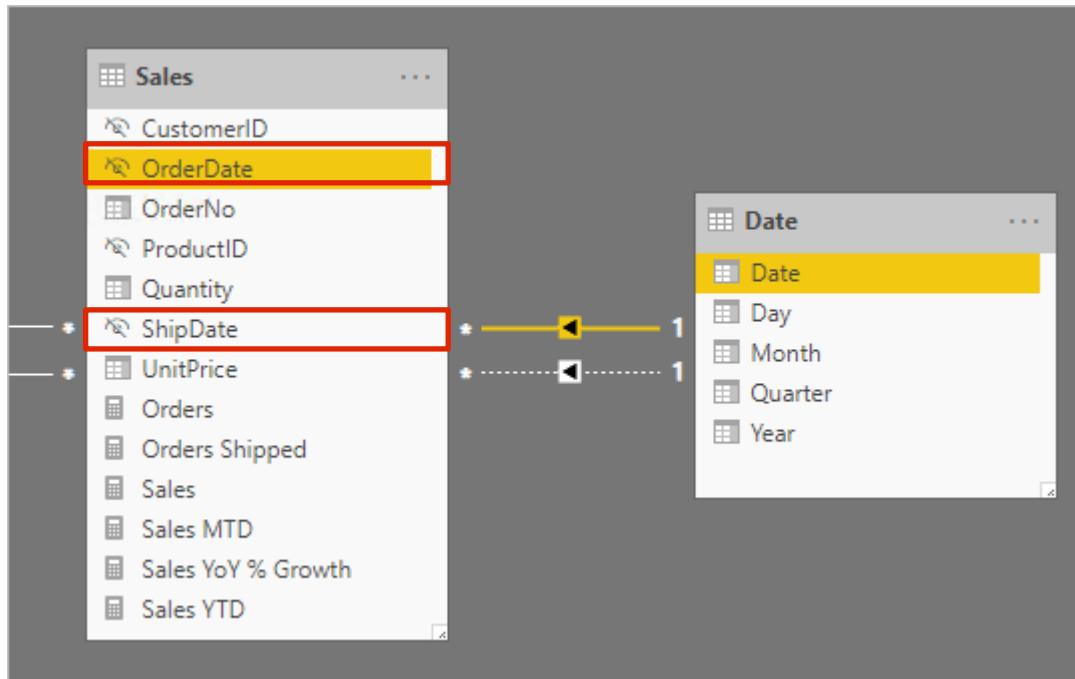
Maintaining one-to-one relationships may lead to redundant data storage since the tables involved often contain overlapping information. This redundancy can increase memory usage and make the data model more challenging to manage.

Many-to-many (*:*)



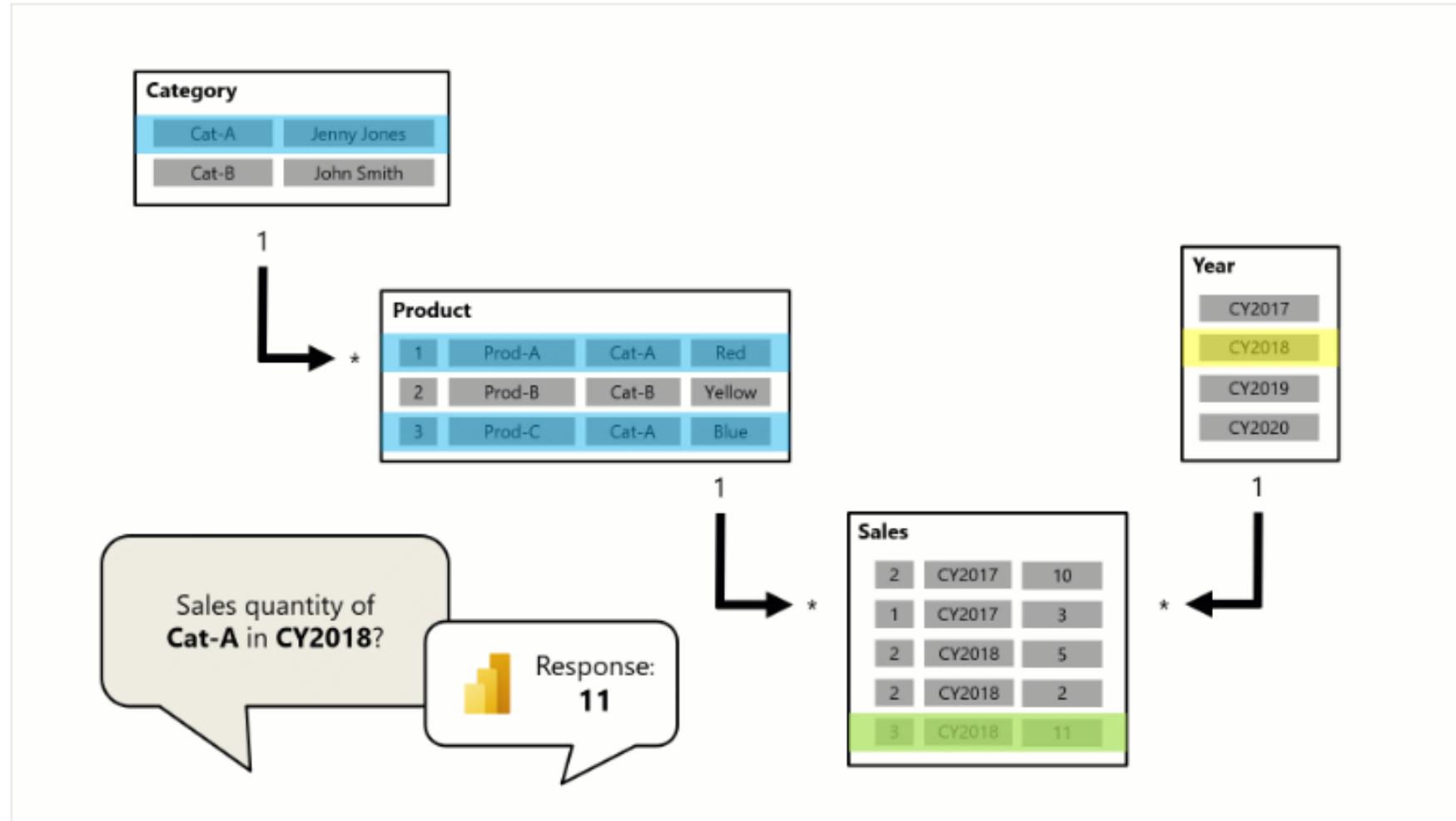
A many-to-many relationship means both columns can contain duplicate values. This cardinality type is infrequently used. It's typically useful when designing complex model requirements.

Inactive Relationships



Inactive relationships in Power BI are relationships between tables that are temporarily turned off for filtering and calculations to avoid conflicts in data analysis. They can be activated as needed for specific tasks.

Filter Propagation



CROSS FILTER DIRECTION

Create relationship

Select tables and columns that are related.

Sales

SalesOrderID	OrderDate	Sort_of_Sales	Freight	Freight_1	ProductID	OrderQty	SpecialOfferID
52242	Sunday, July 7, 2013	Internet	0.1248	0.1248	870	1	
52592	Sunday, July 14, 2013	Internet	0.1248	0.1248	870	1	
52694	Tuesday, July 16, 2013	Internet	0.1248	0.1248	870	1	

DatesTable

DateCol	Year	Month	Week of Year	Day Name
Tuesday, May 31, 2011	2011	5	23	Tuesday
Tuesday, June 7, 2011	2011	6	24	Tuesday
Tuesday, June 14, 2011	2011	6	25	Tuesday

Cardinality

Many to one (*:1)

Cross filter direction

Both

Make this relationship active

Apply security filter in both directions

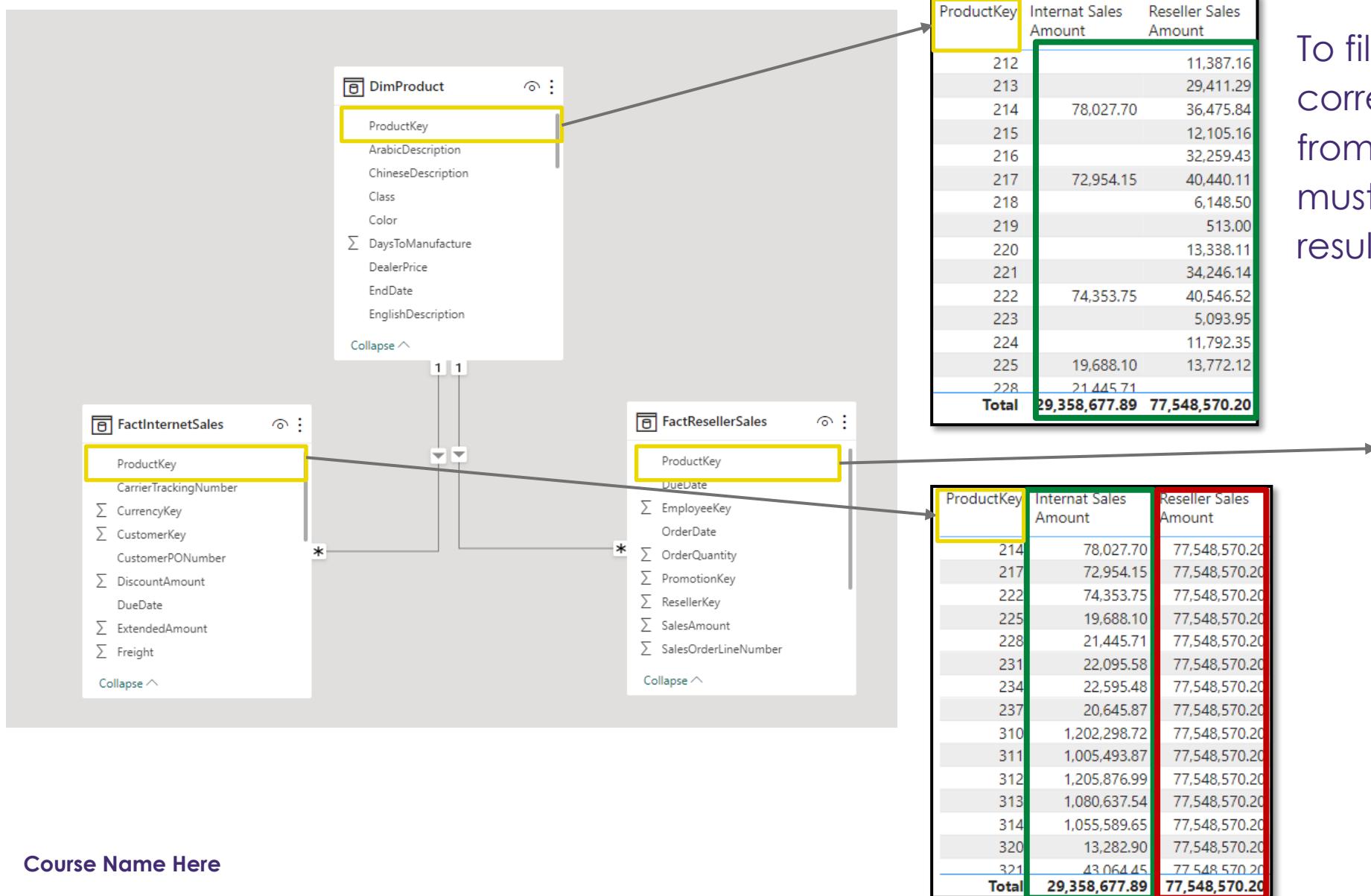
Assume referential integrity

OK Cancel

Cardinality type	Cross filter options
One-to-many (or Many-to-one)	Single Both
One-to-one	Both
Many-to-many	Single (Table1 to Table2) Single (Table2 to Table1) Both

Each model relationship is defined with a cross-filter direction. Your setting determines the direction(s) that filters will propagate. The possible cross-filter options are dependent on the cardinality type.

Filter Flow Example

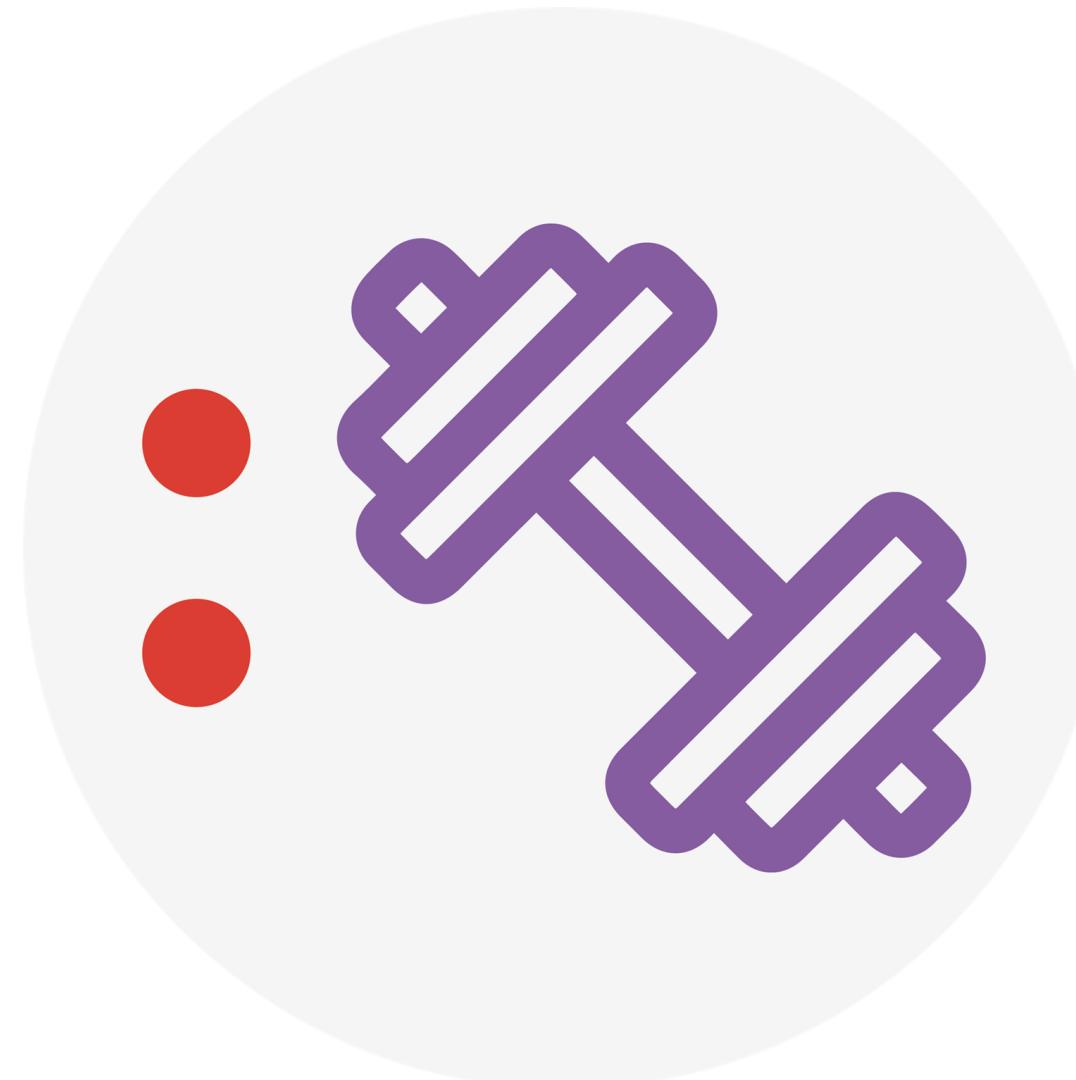


DATA MODEL BEST PRACTICES

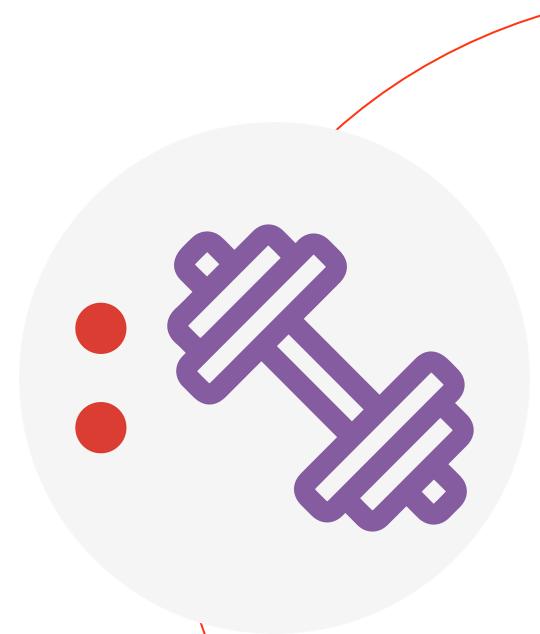
- ✓ Use a star schema with
- ✓ Contain relationships with one way filters (vs. bidirectional)
- ✓ Contain tables that each serve a specific purpose, including data (fact) tables and lookup (dim) tables
- ✓ Only include the data you need for analysis (no redundant or unnecessary records or fields)
- ✓ Split out individual date and time components from DateTime fields



Live Demo



Exercise 1



KC3



Lesson 4

Create Model Calculations using DAX

After completing this module, students will be able to:

- Understand DAX functions
- Creating a calculated Data table using DAX
- Use DAX for formulas and expressions
- Create calculated tables and measures
- Build dynamic measures

DAX (Data Analysis Expressions)

DAX can be used to create three types of calculations:

- Calculated table
- Calculated column
- Measure
- DAX can also be used to define row-level security (RLS) rules



Common DAX Functions

Math & Stats

Common Examples:

- SUM
- AVERAGE
- MAX/MIN
- DIVIDE
- COUNT/COUNTA
- COUNTROWS
- DISTINCTCOUNT

- SUMX
- AVERAGEX
- MAXX/MINX
- COUNTX

Logic

Common Examples:

- IF
- IFERROR
- AND
- OR
- NOT
- SWITCH
- TRUE
- FALSE

Text

Common Examples:

- CONCATENATE
- FORMAT
- LEFT/MID/RIGHT
- UPPER/LOWER
- PROPER
- LEN
- SEARCH/FIND
- REPLACE
- REPT
- SUBSTITUTE
- TRIM
- UNICHAR

Filter

Common Examples:

- CALCULATE
- FILTER
- ALL
- ALLEXCEPT
- RELATED
- RELATEDTABLE
- DISTINCT
- VALUES
- EARLIER/EARLIEST
- HASONEVALUE
- HASONFILTER
- ISFILTERED
- USERELATIONSHIP

Time Intelligence

Common Examples:

- TOTALMTD
- TOTALQTD
- TOTALYTD
- SAMEPERIODLASTYEAR
- PARALLELPERIOD
- DATESBETWEEN
- DATEADD
- DATESINPERIOD

Common DAX Operators

Arithmetic Operator	Meaning	Example
+	Addition	$2 + 7$
-	Subtraction	$5 - 3$
*	Multiplication	$2 * 6$
/	Division	$4 / 2$
\wedge	Exponent	$2 \wedge 5$

Comparison Operator	Meaning	Example
=	Equal to	[City] = "Boston"
>	Greater than	[Quantity] > 10
<	Less than	[Quantity] < 10
\geq	Greater than or equal to	[Unit_Price] \geq 2.5
\leq	Less than or equal to	[Unit_Price] \leq 2.5
\neq	Not equal to	[Country] \neq "Mexico"

Text/Logical Operator	Meaning	Example
&	Concatenates two values to produce one text string	[City] & " " & [State]
&&	Create an AND condition between two logical expressions	([State] = "MA") && ([Quantity] > 10)
(double pipe)	Create an OR condition between two logical expressions	([State] = "MA") ([State] = "CT")
IN	Creates a logical OR condition based on a given list (using curly brackets)	'Store Lookup'[State] IN { "MA", "CT", "NY" }

Standard Aggregation Measures

Typically, numeric columns are summarized using aggregation functions.

Common examples of summarizing:

- Sum
- Count
- Average
- Min and Max

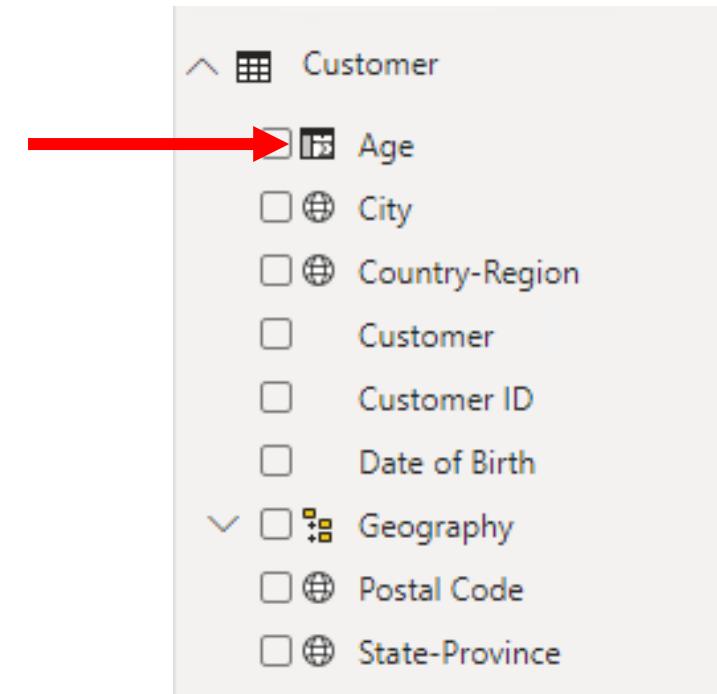
Column references always enclose the column name in square brackets

```
Total Sale = sum('Sales By Country Files'[Sales])
```

```
Sale Quantity = Average('Sales By Country Files'[Quantity])
```

DAX Calculated column

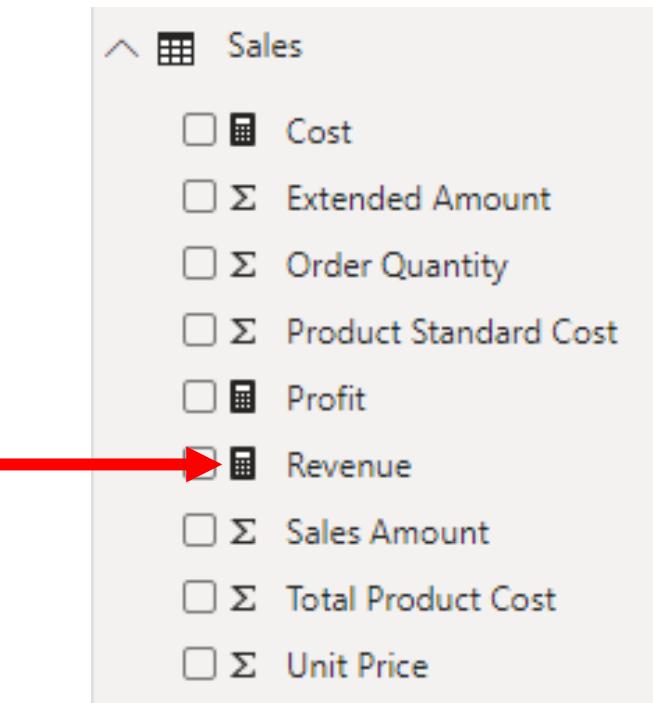
- A **Calculated column** adds a new column to a table
- A formula is evaluated for each row in the table
- The formula must return a single value
- It is only evaluated when the model is refreshed*
- It increases the storage size of the model*



*Except when added to a DirectQuery table

DAX Calculated measures

- A Measure summarizes model data
- The formula must return a single value
- It is evaluated at query-time
- Results are never stored in the model



Calculated columns vs. measures

There are differences between calculated column and measure

	Calculated Columns	Measures
Purpose	Extend table	Summarize model data
Evaluation	Row context at data refresh-time	Filter context at query-time
Storage	Stores value in each table row	Never stores value
Visual use	Filter, group, or summarize	Designed to summarize

Implicit & Explicit Measures

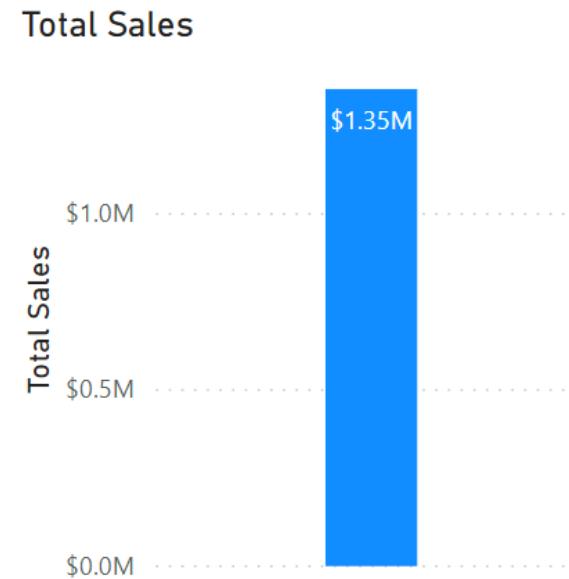
	Benefits	Limitations
Implicit Measures  Order Quantity	<ul style="list-style-type: none">Accessible to non-technical users.Enables self-service analytics.Easy to modify and adjust.	<ul style="list-style-type: none">Limited flexibility for complex calculations.Limited control and customizationMay lack granularity and detailed data analysis capabilities.
Explicit Measures  Profit	<ul style="list-style-type: none">Greater flexibility and customizationRequires deeper understanding of DAX and the data model.Can be used in Analyze with Excel export option	<ul style="list-style-type: none">Requires advanced technical knowledge of DAX.Time-consuming to create and manage complex calculations.Excessive use can impact report performance.

Understand context when using DAX

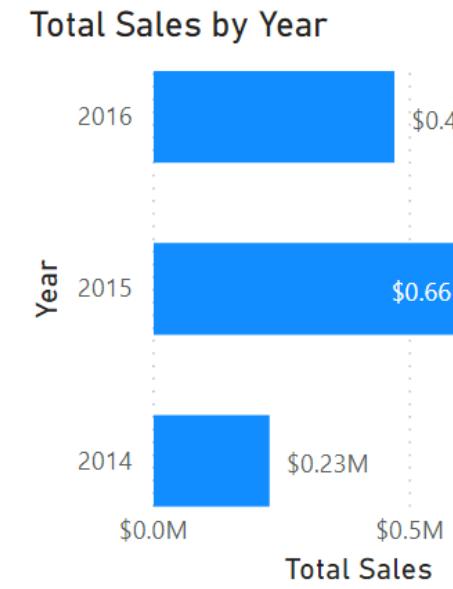
How context affects DAX measures is a difficult concept to comprehend. The ensuing visuals will demonstrate how context affects DAX measures so you can see how they interact together.

The following three visuals use the exact same DAX measure: Total Sales.

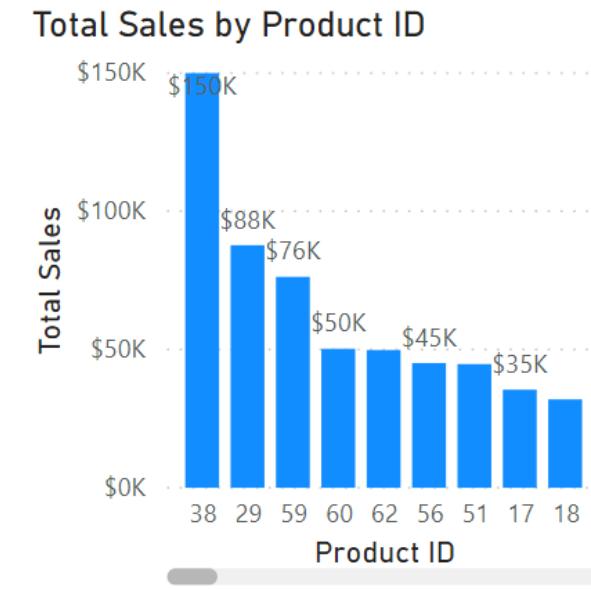
Total Sales



Total Sales by Year



Total Sales by Product ID



DAX Formatting

From

```
USSales = CALCULATE([Total Sales],FILTER(Reseller,Reseller[ResellerCountry]= "United States"))
```



To

```
USSales =  
CALCULATE (  
    [Total Sales],  
    FILTER ( Reseller, Reseller[ResellerCountry] = "United States" )  
)
```

Use **shift + enter** to split out and indent each component of your DAX formulas to make them more readable



<https://www.daxformatter.com/>

Formula Commenting

```
Sales Previous Year % Change =  
  
var CY = [Total Sales] // Current Year Sales ←  
var PY = [Sales Last Year] // Last Years Sales ←  
RETURN  
  
DIVIDE(CY-PY,PY)
```

Comment Type	Marker
Single Line Comment	-- or //
Multi Line Comment	/* ... */

Adding comments to your code can assist fellow users in understanding your script, and this practice proves especially beneficial for intricate queries featuring numerous lines and nested functions.

Using Variables

Variables can be defined in an expression to make writing DAX easier

Main benefits:

- Improves readability of a formula
- Improves performance when an expression is used multiple times
- Allows testing portions of a complex formula, by returning only a variable for review

```
VAR = <name> RETURN <result_expression>
```

Example:

```
VAR RevenuePriorYear = CALCULATE( [Revenue],  
SAMEPERIODLASTYEAR('Date'[Date]) )
```

RETURN

```
DIVIDE( [Revenue Measure] -RevenuePriorYear,  
RevenuePriorYear )
```

DAX Quick Measures

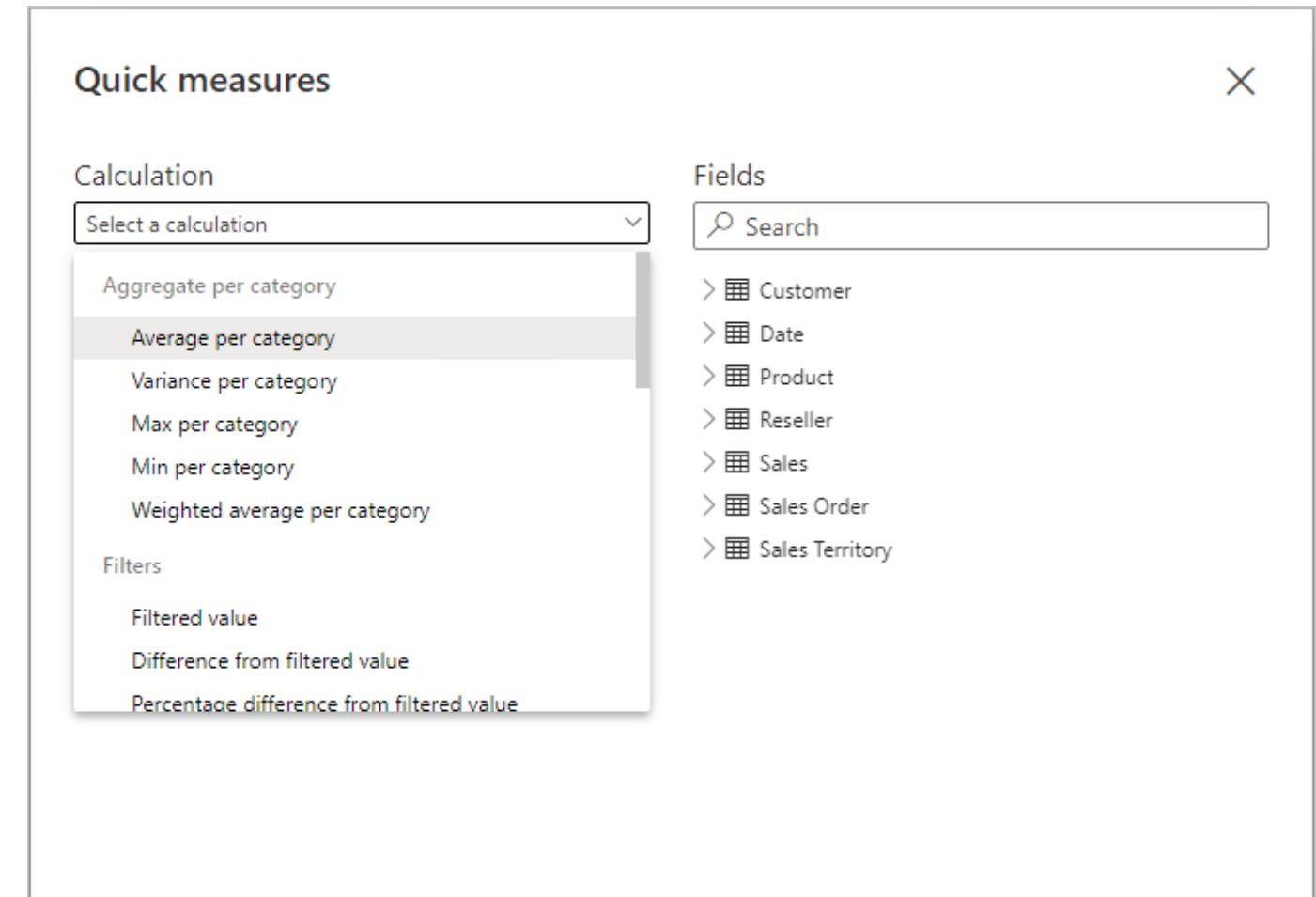
Power BI Desktop supports creating **Quick Measures**

Benefits:

- Quick and easy
- Common calculation templates are available
- Automatically creates the DAX expression
- No need to understand any DAX

Once created, they are like any other explicit measure

- To modify the measure, simply edit the formula



DAX Quick Suggestion AI generation (New)

Quick measure suggestions assist in creating DAX measures using natural language instead of templates or writing DAX from scratch.

Quick measures

Select a calculation to create a measure or describe the measure you need and we'll generate suggestions in DAX, which you can customize later.

Calculations Suggestions

Sales amount for California in 2020

Generate

Suggested measures

Total sales amount where state-province is California and year of date is 2020

Preview value

\$1,785,099.77

DAX ⓘ

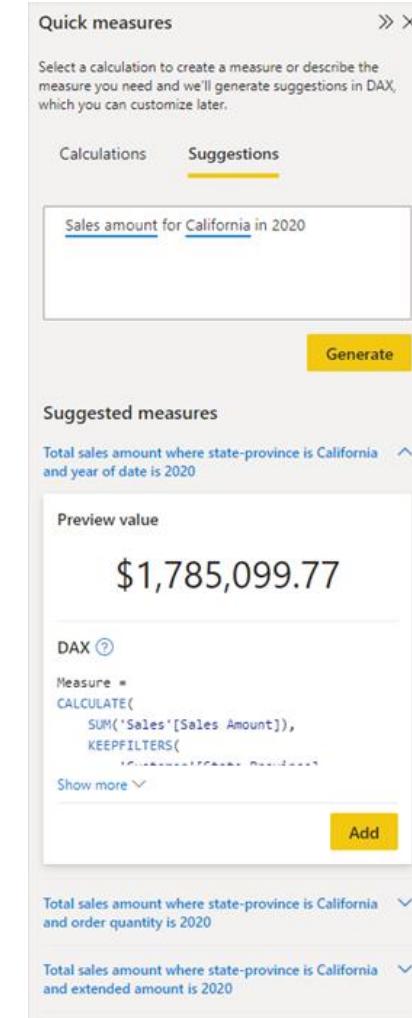
Measure =
CALCULATE(
SUM('Sales'[Sales Amount]),
KEEPFILTERS(
))

Show more ▾

Add

Total sales amount where state-province is California and order quantity is 2020

Total sales amount where state-province is California and extended amount is 2020



Quick measures

Select a calculation to create a measure or describe the measure you need and we'll generate suggestions in DAX, which you can customize later.

Calculations Suggestions

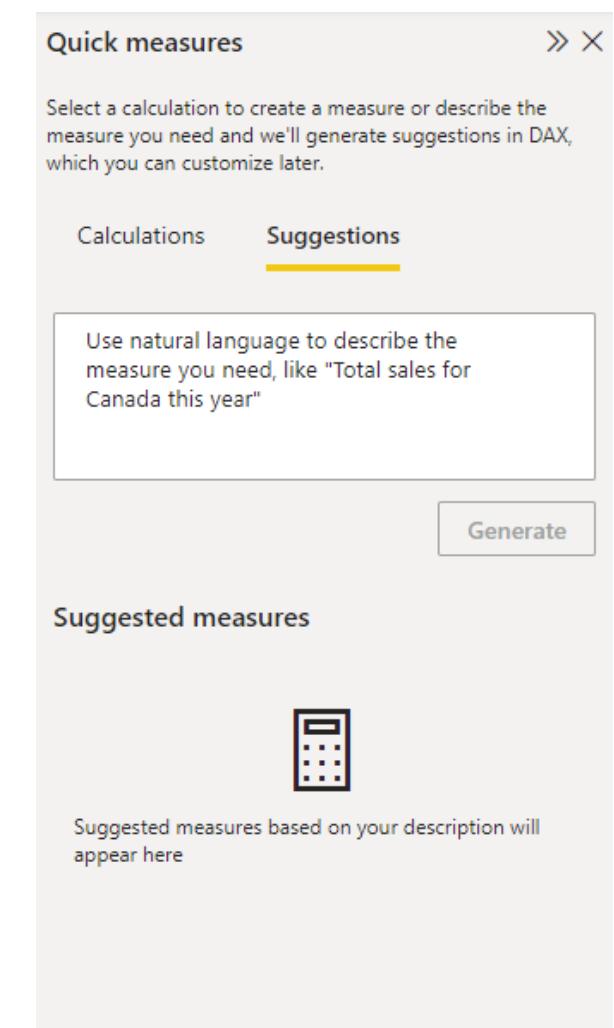
Use natural language to describe the measure you need, like "Total sales for Canada this year"

Generate

Suggested measures



Suggested measures based on your description will appear here



Iterator functions

Iterator functions enumerate all rows of a given table and evaluate a given expression for each row

- They provide flexibility over how calculations summarize data
- All iterator functions require Table and Expression
- The expression must return scalar or single value
- Iterator functions are easily identified by the appended “X”

For example:

- Common aggregation: SUMX, AVERAGEX, COUNTX, MINX, and MAXX
- Special: CONCATENATEX, and RANKX

Iterator functions

Example

```
Measure =  
SUMX(  
    <Table>,  
    <Expression>  
)
```

```
Revenue =  
SUMX(  
    Sales,  
    Sales[Order Quantity] * Sales[Unit Price]  
)
```

Iterator functions example

OrderKey	Order Quantity	Unit Price	
5	2	\$5.19	10.38
5	4	\$20.19	80.76
5	1	\$419.46	419.46
5	1	\$874.79	874.79
6	1	\$809.76	809.76
6	1	\$714.70	714.70
6	2	\$714.70	1,429.40
6	4	\$5.19	...

$\sum 208,202.17$

Automatic Date Table

By default, Power BI automatically generates a concealed date table for any table that includes a Date or **DateTime** column on one side of a relationship. These automatically generated calendars encompass all dates up to the end of the year, irrespective of the actual date range within the table.

The screenshot shows the Power BI Data View. On the left, the Sales table is listed with various columns: CategoryID, Customer ID, Customer Name, Discount%, EmployeeID, LocationID, Order Date, Order ID, Product ID, Quantity, Row ID, Segment, and Ship Date. The 'Order Date' and 'Ship Date' columns are highlighted with red boxes. A red curly brace on the right side groups these two columns together, indicating they are related to the automatically generated date table on the right.

Date	Day	MonthNo	Month	QuarterNo	Quarter	Year
1/01/2017 00:00:00	1	1	January	1	Q1	2017
1/02/2017 00:00:00	2	1	January	1	Q1	2017
1/03/2017 00:00:00	3	1	January	1	Q1	2017
1/04/2017 00:00:00	4	1	January	1	Q1	2017
1/05/2017 00:00:00	5	1	January	1	Q1	2017
1/06/2017 00:00:00	6	1	January	1	Q1	2017
1/07/2017 00:00:00	7	1	January	1	Q1	2017
1/08/2017 00:00:00	8	1	January	1	Q1	2017
1/09/2017 00:00:00	9	1	January	1	Q1	2017
1/10/2017 00:00:00	10	1	January	1	Q1	2017
1/11/2017 00:00:00	11	1	January	1	Q1	2017
1/12/2017 00:00:00	12	1	January	1	Q1	2017

Automatically created a hidden date table in the background containing all these columns

PROS & CONS AUTOMATIC DATE TABLES

PROS



- Automatically generated
- No code or additional table required
- Built-in date Hierarchies aren't automatically generated

Cons



- Hidden background table that not be modified
- Inflates the Power BI file size due to every date field in your table having a auto-date table
- Not a centralized date dimension table

Date Dimension Table Requirements Checklist

- ✓ It should encompass all days corresponding to the years presented in your fact tables.
- ✓ It is necessary to designate at least one field as a Date or DateTime data type.
- ✓ Duplicate dates or datetime values are not permissible.
- ✓ If utilizing a time component within a date column, all times must be uniform (e.g., 12:00 PM).
- ✓ While not mandatory, it is recommended to label it as a date table for optimal use.

CALENDAR Function

- Returns a table with one column of all dates between start and end date
- Dates can also be referenced from date fields in other tables as seen in example 2

```
=CALENDAR(StartDate,EndDate)
```

Example 1: CALENDAR DATE(2020,01,01), DATE (2023,12,31)

Example 2:

```
CALENDAR  
DATE( YEAR ( MIN (Sales [Order Date] )), 1,1)  
DATE( YEAR ( MAX (Sales [Order Date] )), 12,31)
```



Date
1/1/2020 12:00:00 AM
1/2/2020 12:00:00 AM
1/3/2020 12:00:00 AM
1/4/2020 12:00:00 AM
1/5/2020 12:00:00 AM
1/6/2020 12:00:00 AM
1/7/2020 12:00:00 AM
1/8/2020 12:00:00 AM
1/9/2020 12:00:00 AM
1/10/2020 12:00:00 AM
1/11/2020 12:00:00 AM
1/12/2020 12:00:00 AM
1/13/2020 12:00:00 AM
1/14/2020 12:00:00 AM
1/15/2020 12:00:00 AM
1/16/2020 12:00:00 AM
1/17/2020 12:00:00 AM
1/18/2020 12:00:00 AM
1/19/2020 12:00:00 AM
1/20/2020 12:00:00 AM
1/21/2020 12:00:00 AM
1/22/2020 12:00:00 AM

CALENDARAUTO Function

Returns a table with one column of dates based on a fiscal year end month. The Range of dates is calculated automatically based on data in the model

Example 1: CALENDARAUTO(4)

Example 2:

Calendar Table =

```
VAR MinYear = YEAR( MIN( 'Country'[Total Sales]))  
VAR MaxYear= YEAR( MAX('Country'[Total Sales]))  
  
RETURN  
  
FILTER(CALENDARAUTO(),  
YEAR( [Date] ) >= MinYear&&YEAR( [Date] ) <= MaxYear)
```



Date
5/1/2014 12:00:00 AM
5/2/2014 12:00:00 AM
5/3/2014 12:00:00 AM
5/4/2014 12:00:00 AM
5/5/2014 12:00:00 AM
5/6/2014 12:00:00 AM
5/7/2014 12:00:00 AM
5/8/2014 12:00:00 AM
5/9/2014 12:00:00 AM
5/10/2014 12:00:00 AM
5/11/2014 12:00:00 AM
5/12/2014 12:00:00 AM
5/13/2014 12:00:00 AM

The Date Dimension Table

Date	Year	Month	Month Short Name	Month Long Name	Quater	Day of the week	WeekNum
7/1/2020	2020	7	Jul	July	Q3	Wed	27
7/2/2020	2020	7	Jul	July	Q3	Thu	27
7/3/2020	2020	7	Jul	July	Q3	Fri	27
7/4/2020	2020	7	Jul	July	Q3	Sat	27
7/5/2020	2020	7	Jul	July	Q3	Sun	27
7/6/2020	2020	7	Jul	July	Q3	Mon	28
7/7/2020	2020	7	Jul	July	Q3	Tue	28
7/8/2020	2020	7	Jul	July	Q3	Wed	28

Column Name	Fonction
DAX Defined Date	Dates = CALENDAR(DATE(2011, 5, 31), DATE(2022, 12, 31))
DAX Auto Date	Dates = CALENDARAUTO()
DAX Year	Year = YEAR(Dates[Date])
DAX Month	MonthNum = MONTH(Dates[Date])
DAX Week number	WeekNum = WEEKNUM(Dates[Date])
DAX Day of the week	DayoftheWeek = FORMAT(Dates[Date], "DDDD")

Formatting your date

By using the DAX **FORMAT** function to specify date/time formatting you can specify the format output you would like for your fields.

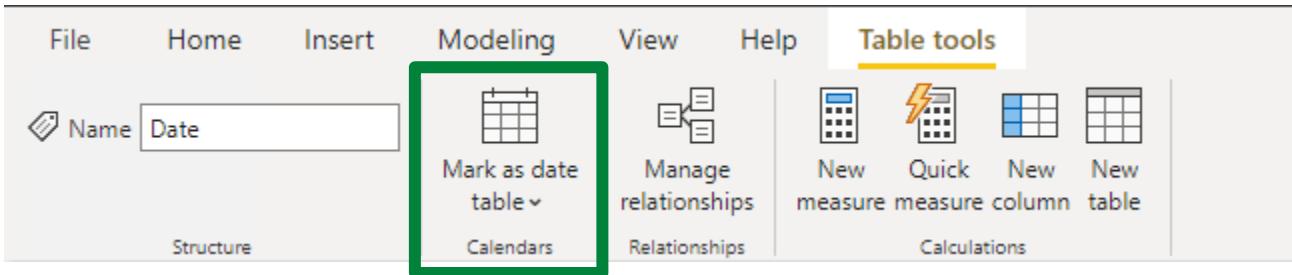
```
FORMAT(<value>, <format_string>[, <locale_name>])
```

```
Short Month Name = FORMAT('Table'[Date], "mmm")
```

Date	Short Month Name
5/1/2014 12:00:00 AM	May
5/2/2014 12:00:00 AM	May
5/3/2014 12:00:00 AM	May
5/4/2014 12:00:00 AM	May
5/5/2014 12:00:00 AM	May
5/6/2014 12:00:00 AM	May
5/7/2014 12:00:00 AM	May

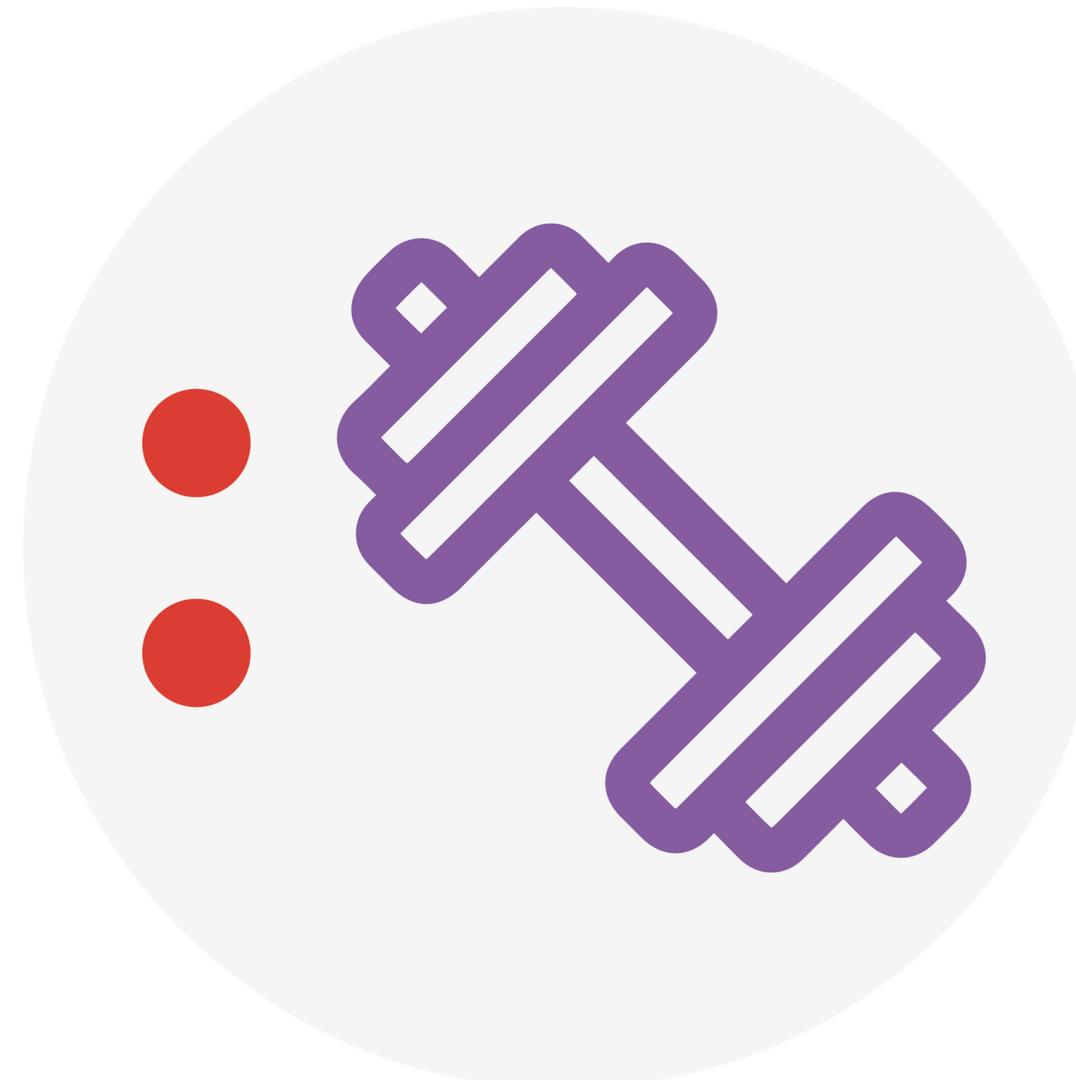
Code	Description
dddd	Display the full day name
ddd	Display the short day name
mmmm	Show full month name
mmm	Show abbreviated month name
mm	Display the month as a two digit number
q	Display the quarter of the year
yyyy	Show the year as a four digit number

Marking Date Dimension Table

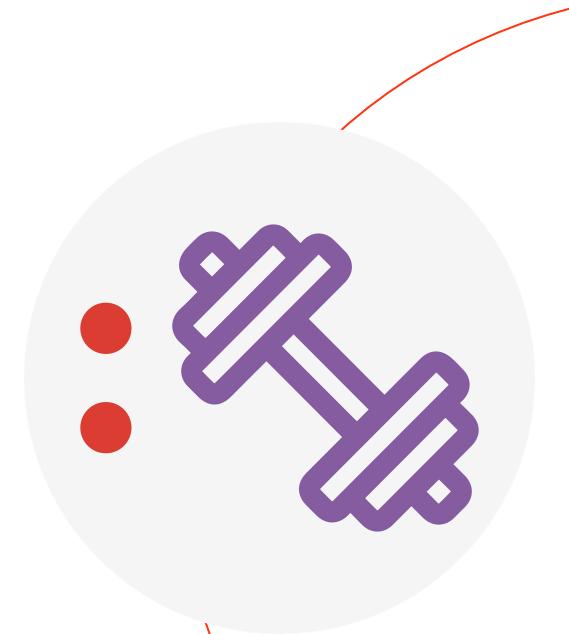


- To use the Time Intelligence functions, a table must be marked as a date table
 - It also allows the model tables to use a non-date type key
 - The date key can be in ISO format, for example, 20210630
- Multiple model tables can be marked as date tables

Live Demo



Exercise 2



KC4



DATA^SOCIETY:

Power BI Intermediate

Day 3



Lesson 5

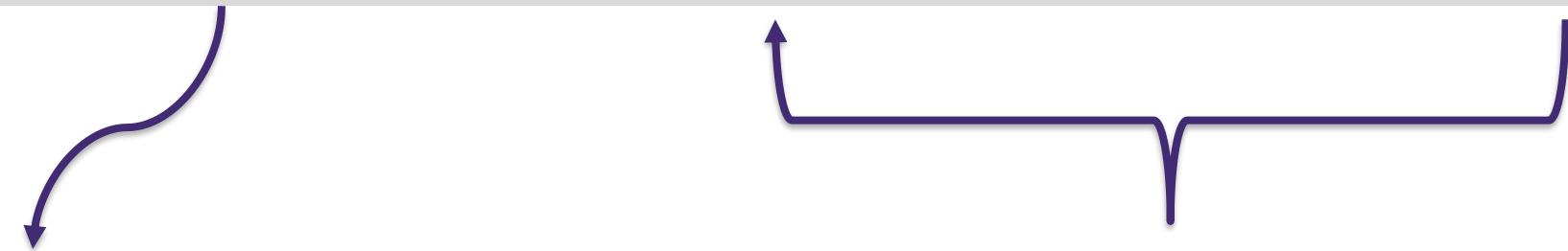
CALCULATE Function & Modifiers

After completing this module, students will be able to:

- Use the Calculate function
- Use filter modifier functions within Calculate
- Examine filter context

DAX Calculate Function

```
CALCULATE ( <expression> , [[<filter1>], <filter2>]...)
```



Name of an existing measure, or a DAX formula for a valid measure

Example

[Total Sales] existing measure

SUM(Regionsales[Sales]) a DAX formula

List of simple Boolean (True/False) filter expressions.
Note you cannot create filters based on a measure

Example

Product[color] = "Blue"

Calendar[Year] > 2021

DAX Calculate Function

Boolean expression filters must evaluate as TRUE or FALSE
Each filter:

- Can reference only a single column
- Can not reference measures
- Can not use functions that scan or return table

1

```
Revenue Red =  
CALCULATE([Revenue], 'Product'[Color] = "Red")
```

2

```
Revenue Red =  
CALCULATE(  
    [Revenue],  
    FILTER(  
        'Product',  
        'Product'[Color] = "Red"  
    )  
)
```

Table Expression Filters

Table expression filters apply a table object as a filter

Each filter:

- Can be a reference to model table
- Can be a DAX function that returns a table object
- If columns are not in filter context, new filters are added to filter context
- If columns are already in filter context, existing filters are overwritten

```
Revenue High Margin Products =  
CALCULATE(  
    [Revenue],  
    FILTER(  
        'Product',  
        'Product'[List Price] > 'Product'[Standard Cost] * 2  
    )  
)
```

CALCULATE Filter Modifiers

When using CALCULATE, it is possible to pass in filter modification functions

Filter modifier functions include:

- **REMOVEFILTERS** – Remove filters from all tables, a single table or column(s)
- **ALL** – Remove filters from a single table or column(s)
- **ALLEXCEPT** – Remove filters from all columns of a single table, except those explicitly passed in
- **ALLNONBLANKROW** – From the parent table of a relationship, returns all rows but the blank row, or all distinct values of a column but the blank row, and disregards any context filters that might exist

CALCULATE Filter Modifier example

New filter added

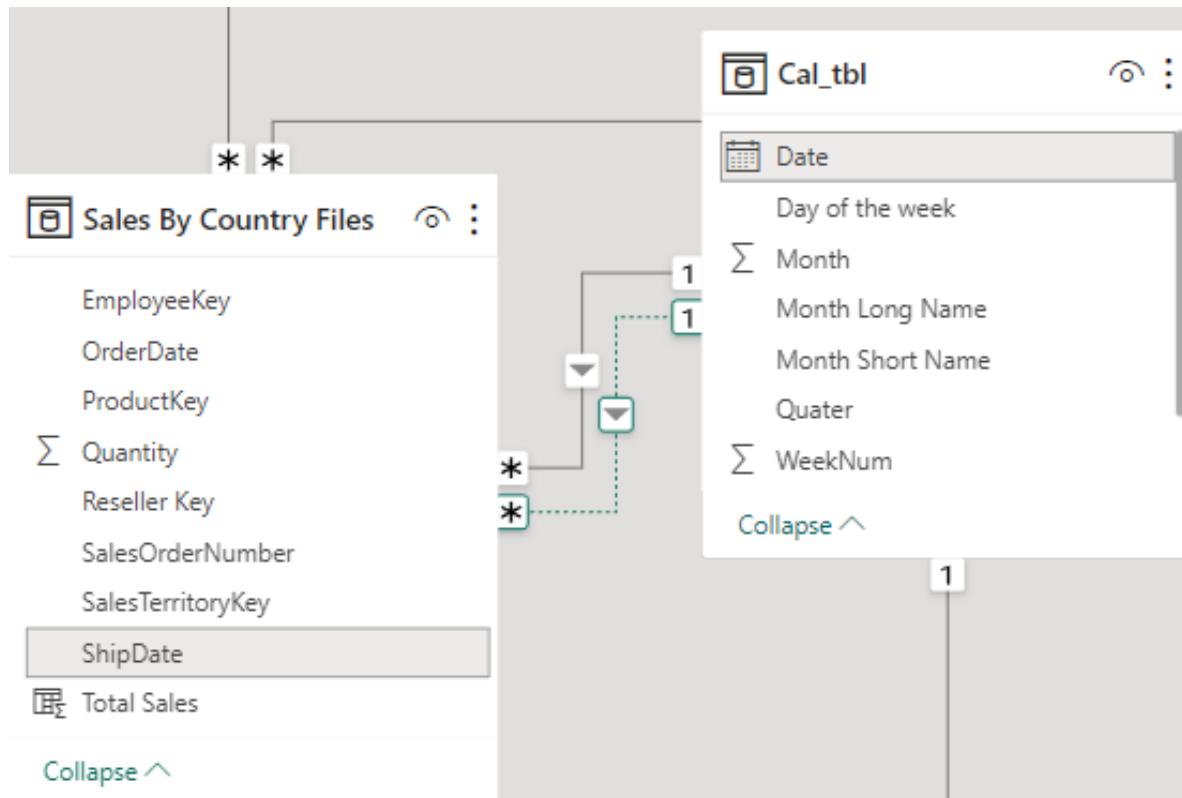
Region	Revenue	Revenue Red
Australia	\$10,655,335.96	\$2,681,324.79
Canada	\$16,355,770.46	\$3,573,412.99
Central	\$7,909,009.01	\$1,585,997.34
France	\$7,251,555.65	\$1,051,014.15
Germany	\$4,878,300.38	\$670,607.30
Northeast	\$6,939,374.48	\$1,876,016.33
Northwest	\$16,084,942.55	\$2,292,905.61
Southeast	\$7,879,655.07	\$1,457,221.07
Southwest	\$24,184,609.60	\$5,345,637.47
United Kingdom	\$7,670,721.04	\$1,063,753.75
Total	\$109,809,274.20	\$21,597,890.81

Existing filter overwritten

Color	Revenue	Revenue Red
Black	\$38,236,124.06	\$21,597,890.81
Blue	\$9,602,850.97	\$21,597,890.81
Grey		\$21,597,890.81
Multi	\$649,030.25	\$21,597,890.81
NA	\$1,099,303.91	\$21,597,890.81
Red	\$21,597,890.81	\$21,597,890.81
Silver	\$19,777,339.95	\$21,597,890.81
Silver/Black	\$147,483.91	\$21,597,890.81
White	\$29,745.13	\$21,597,890.81
Yellow	\$18,669,505.22	\$21,597,890.81
Total	\$109,809,274.20	\$21,597,890.81

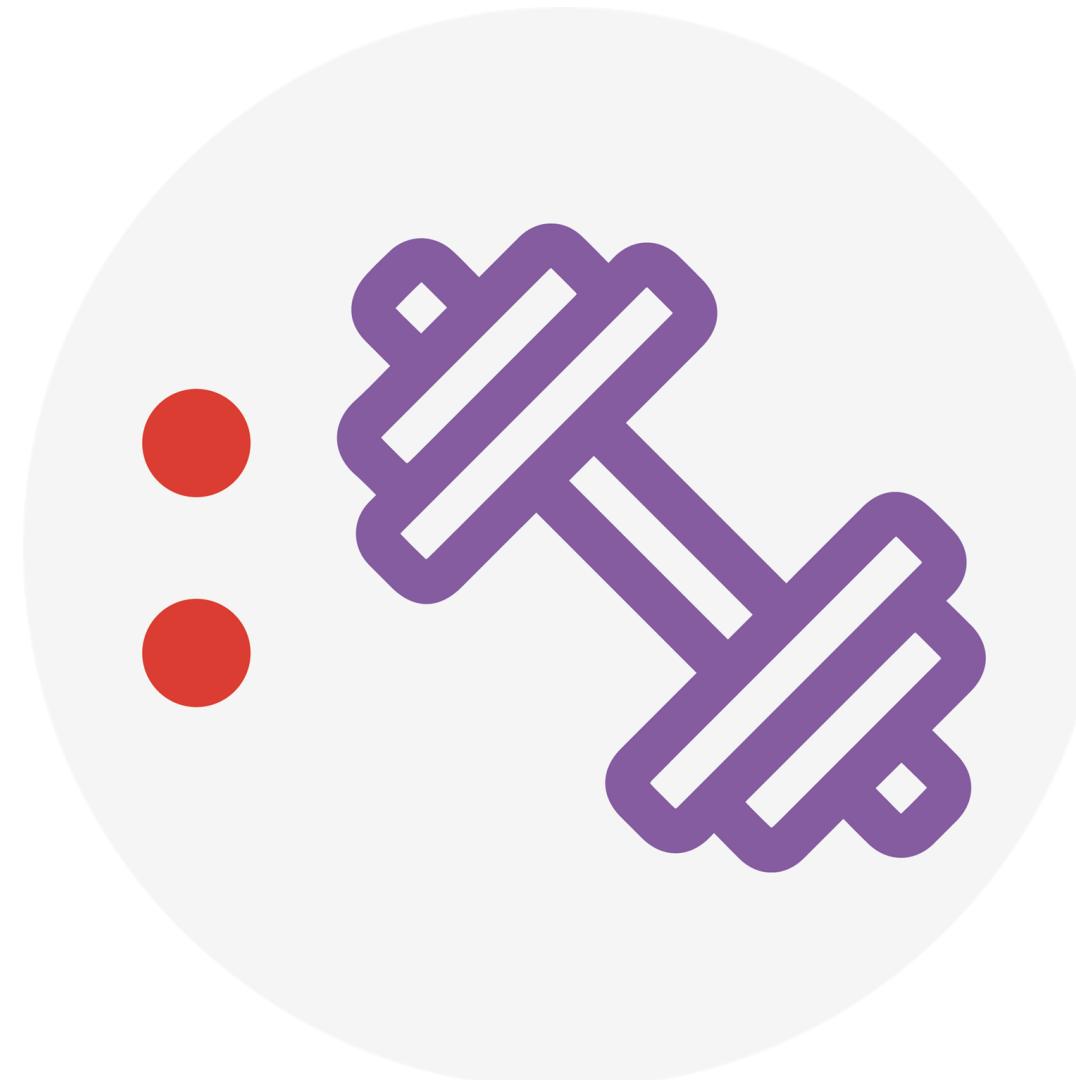
Relationship Modifiers

Use the USERRELATIONSHIP function to make an inactive relationship active during the evaluation of the CALCULATE function

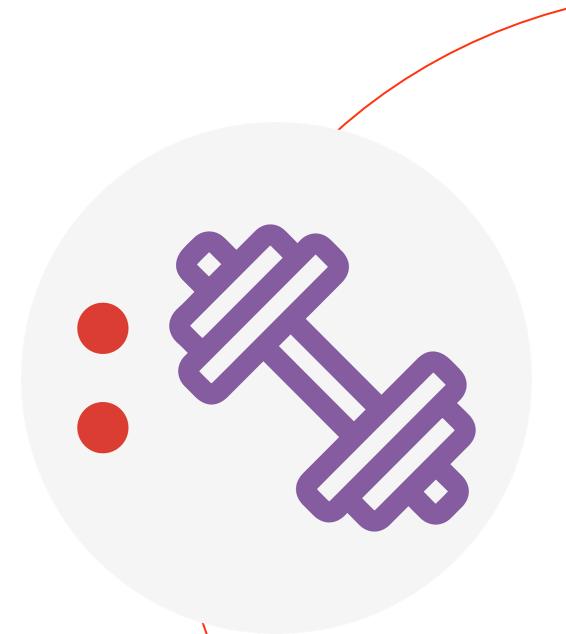


```
Revenue Shipped =  
CALCULATE(  
    [Revenue],  
    USERELATIONSHIP(  
        'Date'[DateKey],  
        'Sales'[ShipDateKey]  
    )  
)
```

Live Demo



Exercise 3



KC5



Lesson 6

Time Intelligence DAX Patterns

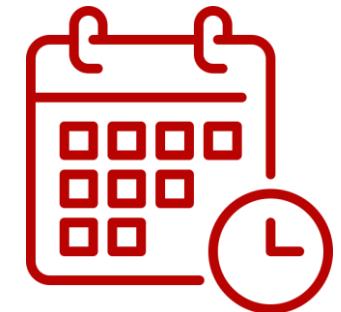
After completing this module, students will be able to:

- Define time intelligence
- Use time intelligence functions
- Understand common time intelligence patterns

Define Time Intelligence

Time Intelligence relates to calculations over dates, months, quarters, or years

- It involves modifying the filter context for date filters
- The model requires a date table
- The date table can be created using the CALENDAR or CALENDARAUTO functions



Define Time Intelligence

Example

Month	Revenue	Revenue YTD
2019 Jan	\$889,902.00	\$889,902.00
2019 Feb	\$837,304.45	\$1,727,206.45
2019 Mar	\$900,089.70	\$2,627,296.15
2019 Apr	\$6,366,809.65	\$8,994,105.80
2019 May	\$5,140,577.65	\$14,134,683.45
2019 Jun	\$3,801,978.60	\$17,936,662.05
2019 Jul	\$3,224,809.45	\$21,161,471.50
2019 Aug	\$2,705,752.60	\$23,867,224.10
2019 Sep	\$2,732,810.85	\$26,600,034.95
2019 Oct	\$6,104,340.55	\$32,704,375.50
2019 Nov	\$5,294,826.15	\$37,999,201.65
2019 Dec	\$4,271,109.60	\$42,270,311.25
Total	\$42,270,311.25	\$42,270,311.25

Use Time Intelligence functions

DAX has many inbuilt Time Intelligence functions

- For example, TOTALYTD and PREVIOUSYEAR
- It is possible to replace them using CALCULATE
- However, that requires more work and could be slower for Power BI to evaluate

Common Time Intelligence Patterns

- Cumulative totals
- Period comparisons
- New customer orders by month
- Stock on-hand value

Summarization over time

- DATESYTD – Returns a single-column table of dates
- TOTALYTD – Evaluates expression for year-to-date (YTD)
- DATESBETWEEN – Returns single-column table with range dates
- DATESINPERIOD – Returns single-column table with range of dates

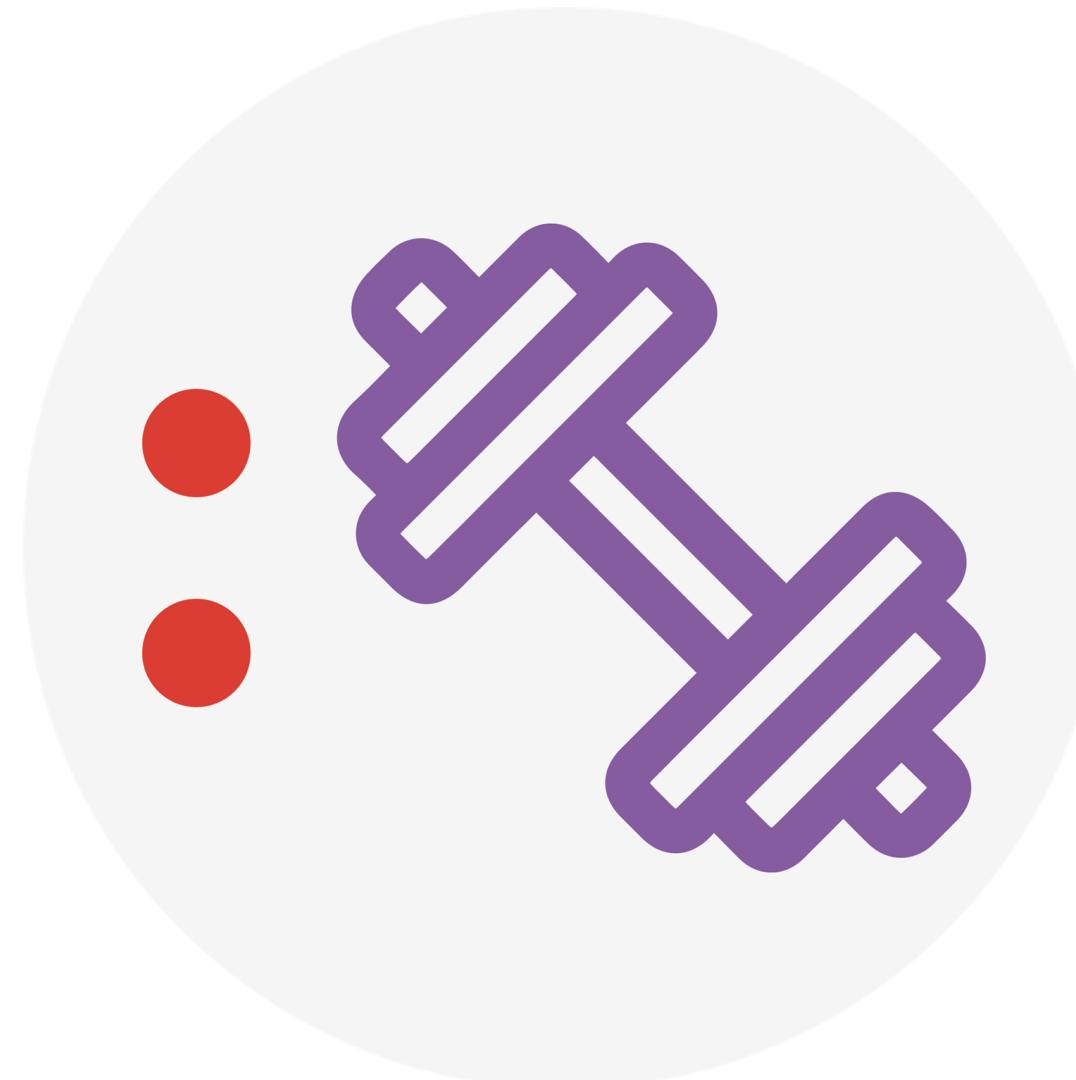
Shifting time periods

- DATEADD – Returns single-column table of dates shifted by interval
- PARALLELPERIOD – Returns single-column table of parallel dates
- SAMEPERIODLASTYEAR – Returns dates shifted back one year
- Many others – NEXTDAY, NEXTMONTH, NEXTQUARTER, NEXTYEAR, PREVDAY, PREVMONTH, PREVQUARTER ,PREVYEAR

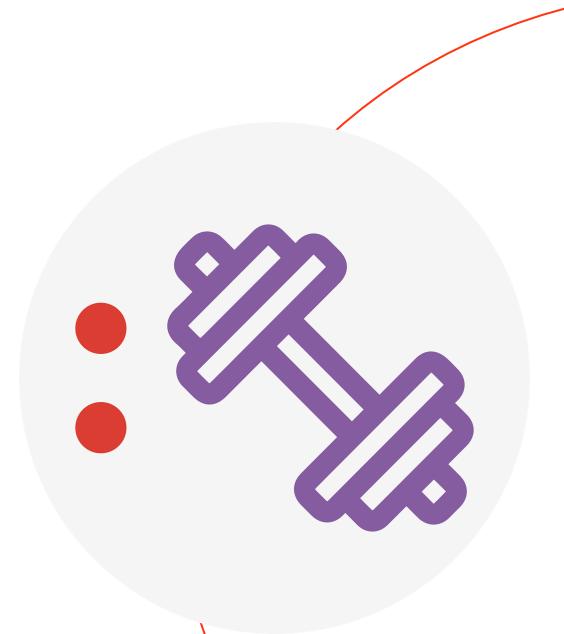
Shifting time periods

- DATEADD – Returns single-column table of dates shifted by interval
- PARALLELPERIOD – Returns single-column table of parallel dates
- SAMEPERIODLASTYEAR – Returns dates shifted back one year
- Many others – NEXTDAY, NEXTMONTH, NEXTQUARTER, NEXTYEAR, PREVDAY, PREVMONTH, PREVQUARTER ,PREVYEAR

Live Demo



Exercise 4



KC6



DATA SOCIETY:

Power BI Intermediate

Day 4



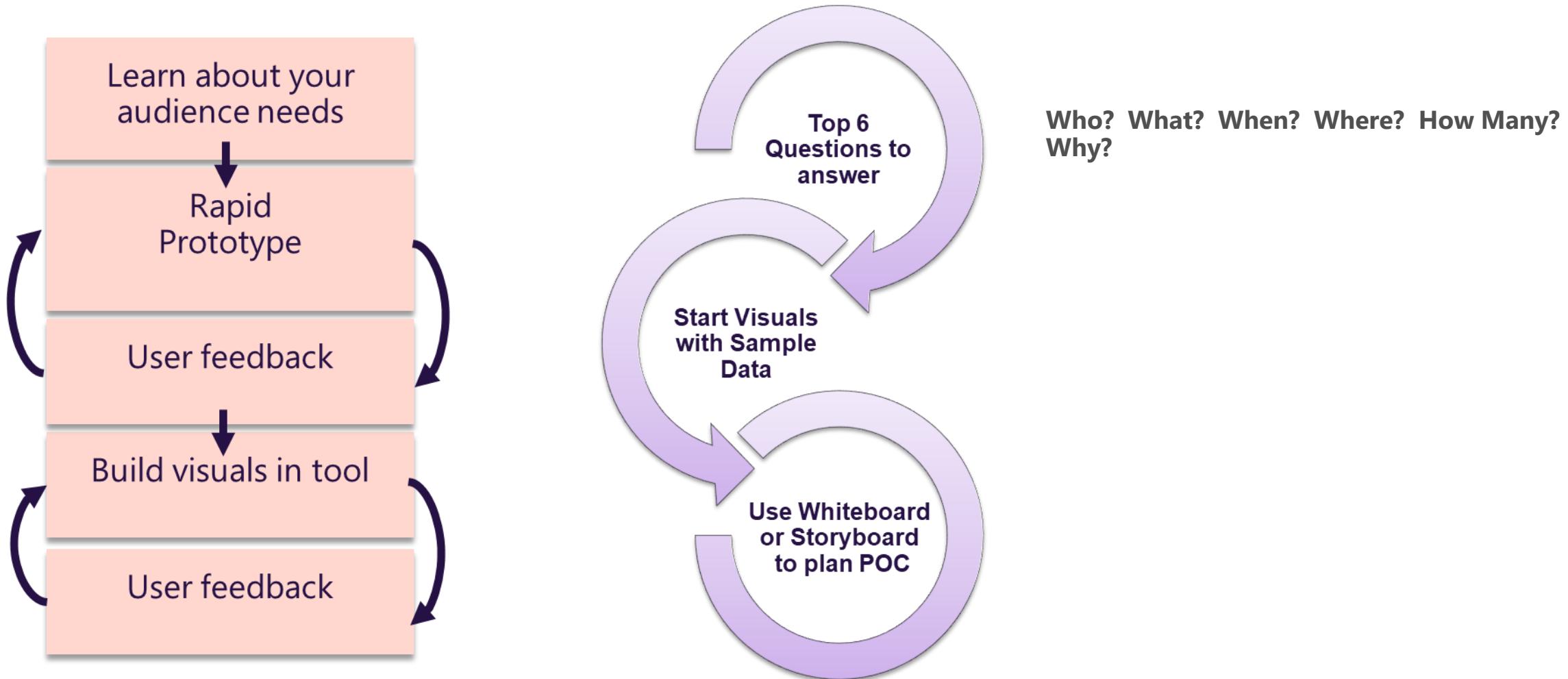
Lesson 7

Advanced Report Design Principles

After completing this module, students will be able to:

- Understand the importance of data storytelling
- Plan report layout, user interface and flow
- Utilize best practices for custom report navigation, report tooltips and drill-through options

Agile Report Creation



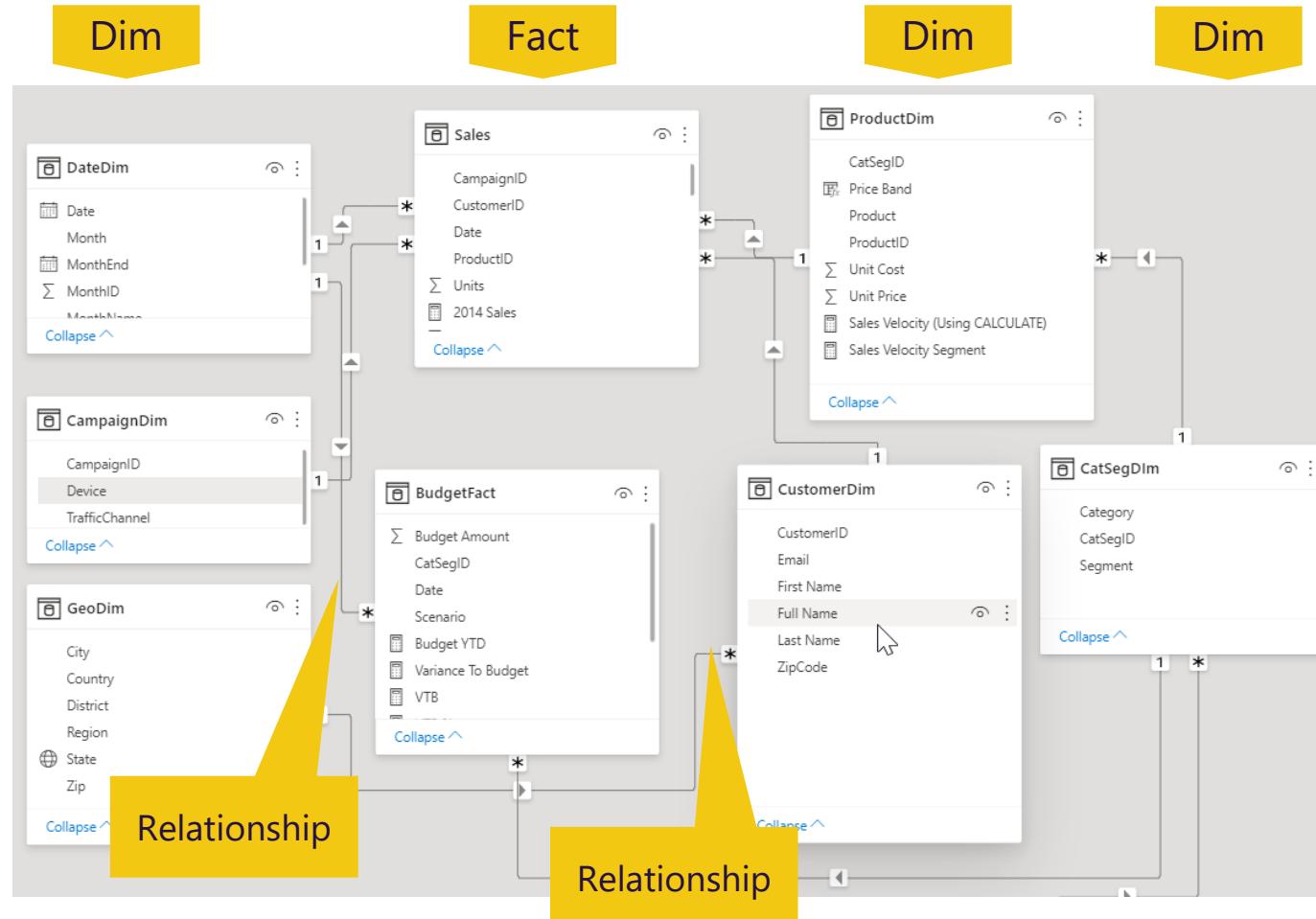
Convert the Story to a Date Model

List your big questions:

- What is my Total Sales for a Selected Year and Region?
- How is my Total Sales doing Year Over Year?
- How are my Units trending for various States in my region?
- How is my Sales doing by Channel, Device, Category for selected Year?
- Which categories are performing best to worst by Total Sales ?

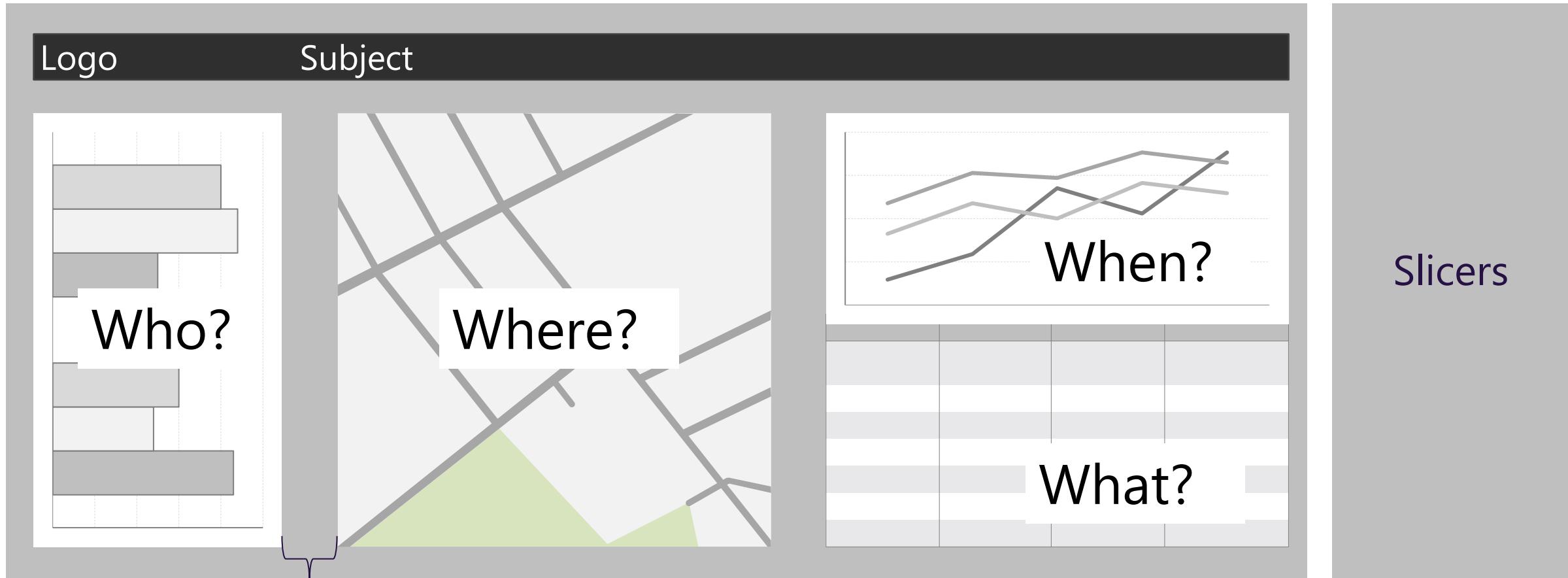


Convert Story to a Data Model



- Measures (e.g. Units or Sales) live in **fact** tables
- Descriptive **attributes** (e.g. Campaign, Customer name) live in **dimension** tables
- **Relationships** tie the data together so you can slice your measures by your attributes

Use Storyboarding to Design Report Layout

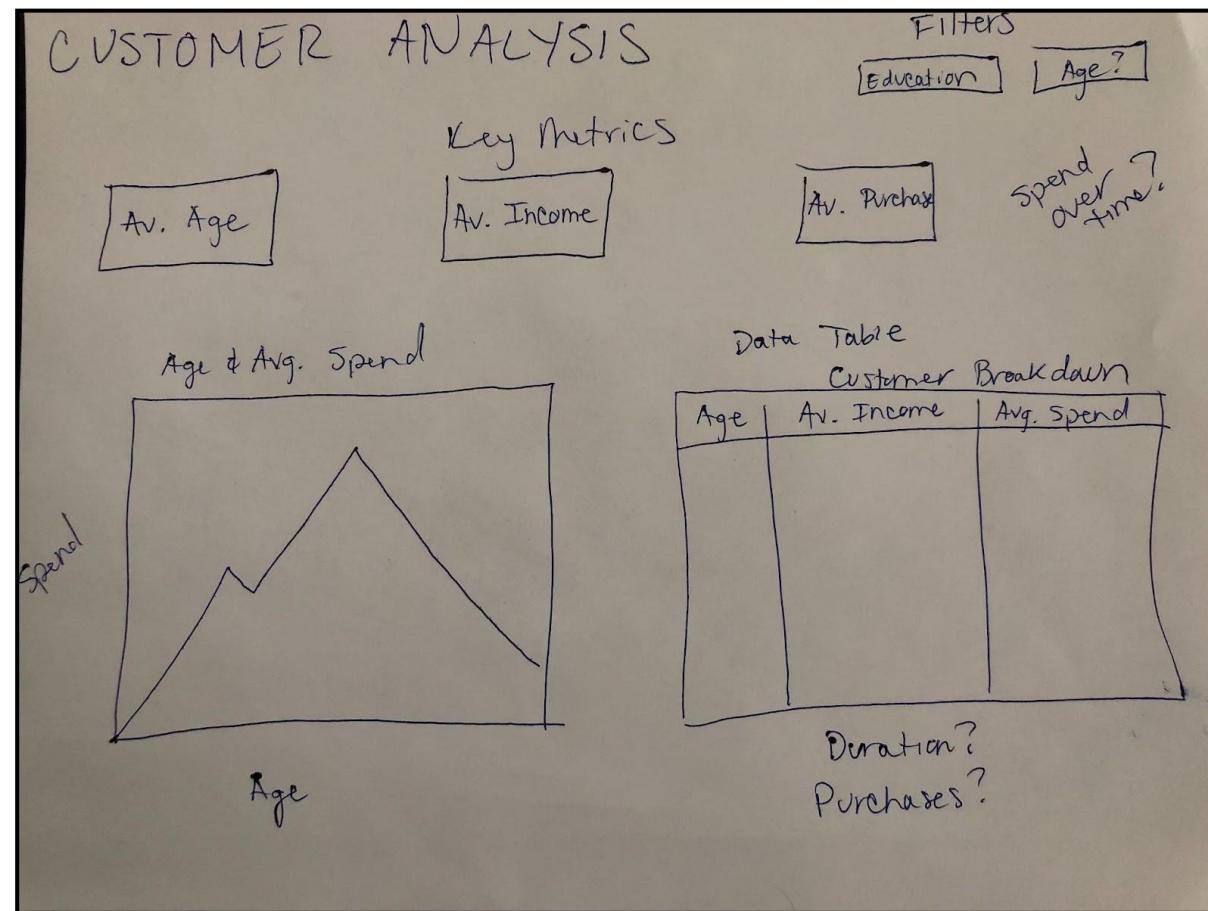


Control this space. If all objects tell the same story, use the same gap space. If some objects tell a different story, make the gap bigger

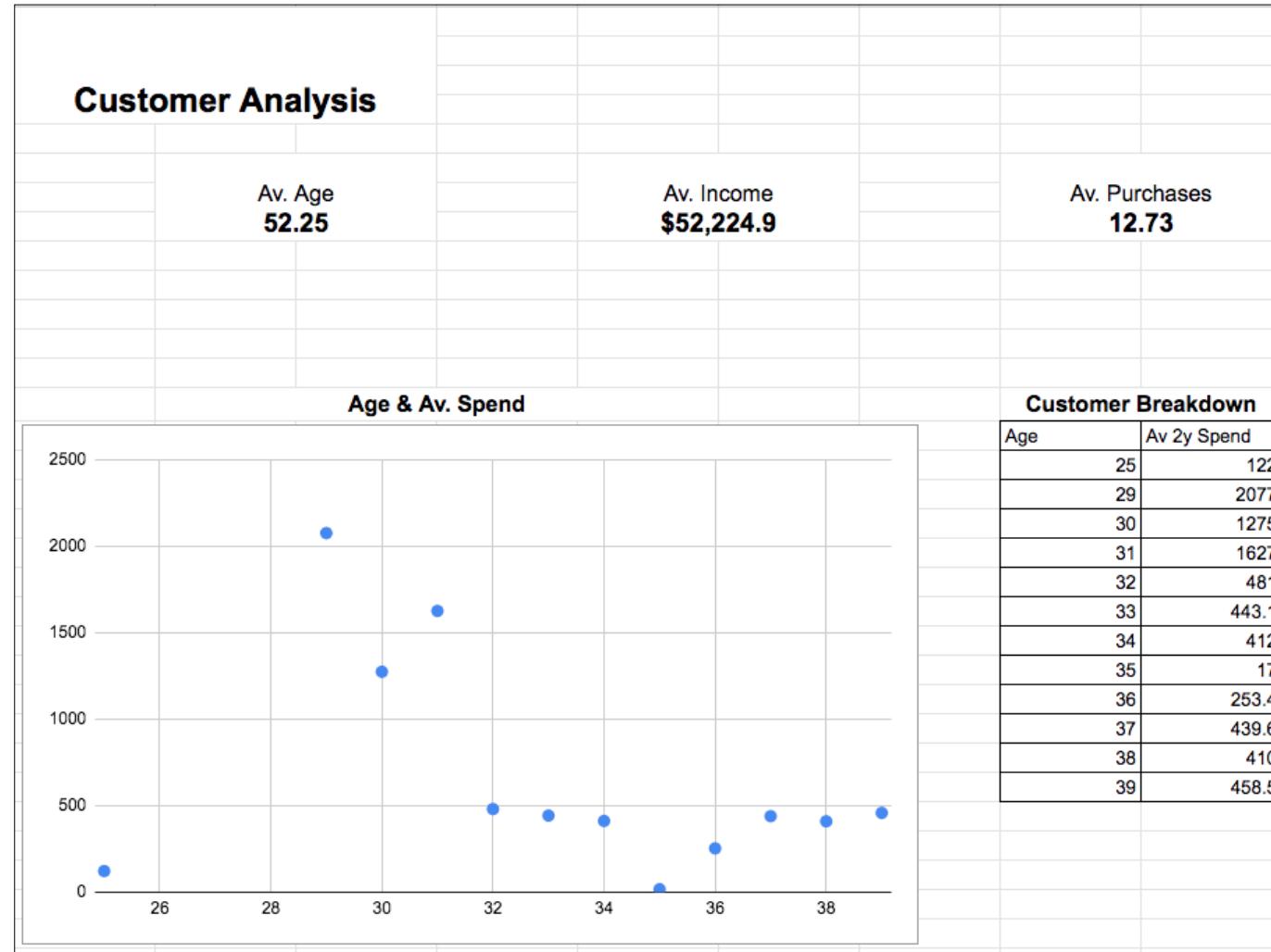
Visual Selection

Comparison	Data Over Time	Correlation	Distribution	Part-to-whole	Ranking
 Bar Chart	 Bar Chart	 Bubble Chart	 Bubble Chart	 Donut	 Ord. Column
 Grouped Bar	 Line Chart	 Column Line	 Grouped Bar	 Stacked Bar	 Ordered Bar
 Line Chart	 Stacked Bar	 Scatterplot		 Treemap	 Ribbon
 Bubble Chart	 Area Chart				 Decomp. Tree
 Area Chart	 Stacked Area				 Funnel
 Stacked Bar	 Bubble Chart				
 Ribbon	 Waterfall				
 Shape Map					

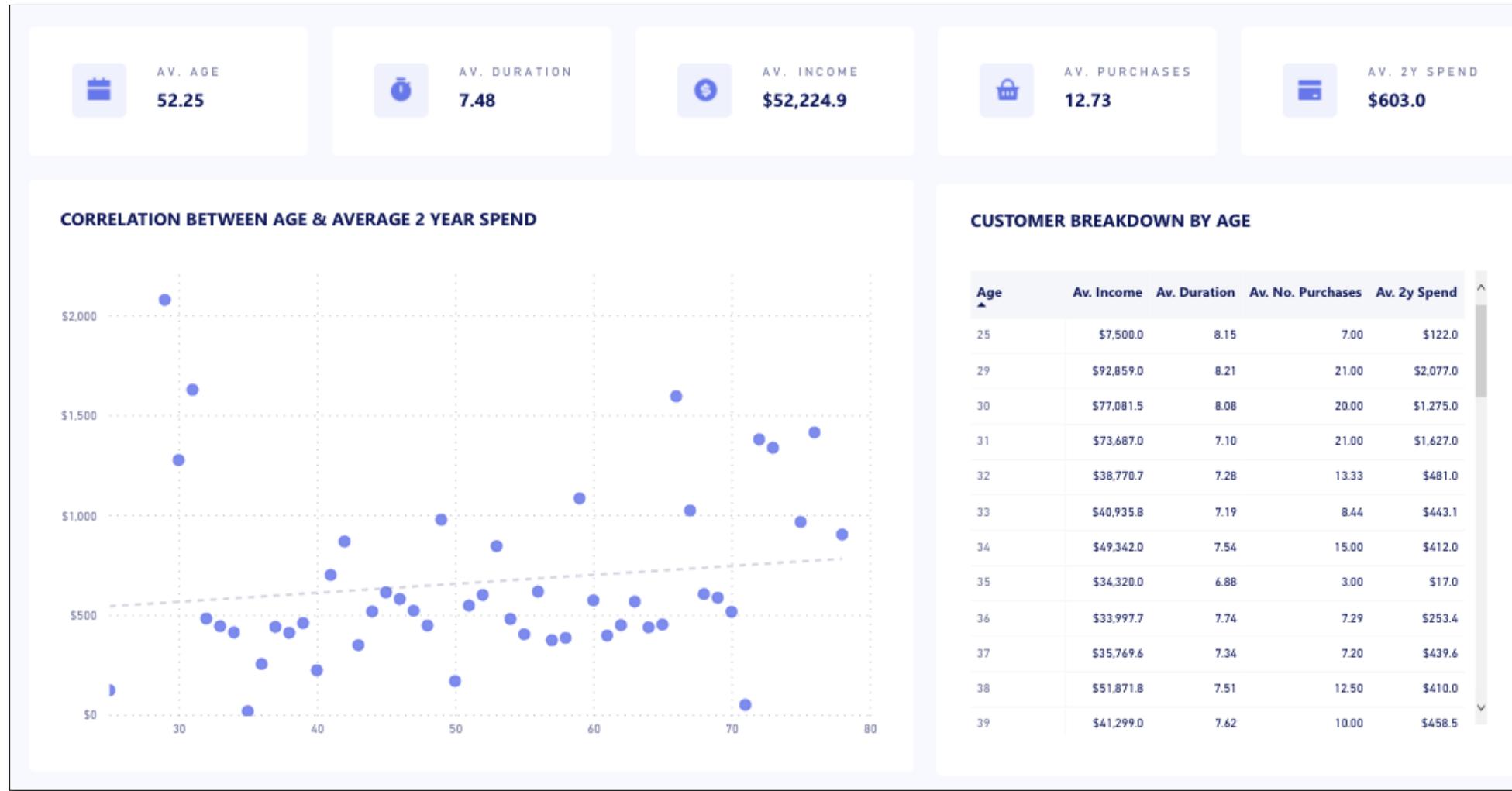
Whiteboard Mockup



Wireframe



Final Product



Page Navigation

Company Analysis Genre Analysis Platform Analysis

Global Sales by Year and Company

Company: Microsoft (teal), Nintendo (dark grey), PC (black), Sony (red)

Global Sales

Year

Year	Game	Company	Critic Score	Rating
2002	Spider-Man: The Movie	Nintendo	78	E
2002	SpongeBob SquarePants: Revenge of the Flying Dutchman	Nintendo	71	E
2002	Spyro 2: Season of Flame	Nintendo	76	E
2002	Star Fox Adventures	Nintendo	82	T
2002	Super Mario Sunshine	Nintendo	92	E
2002	Super Monkey Ball 2	Nintendo	87	E
2002	The Legend of Zelda: A Link to the Past	Nintendo	95	E
2002	The Legend of Zelda: The Wind Waker	Nintendo	96	E
2002	Age of Mythology	PC	89	T

Microsoft

Sony

PC

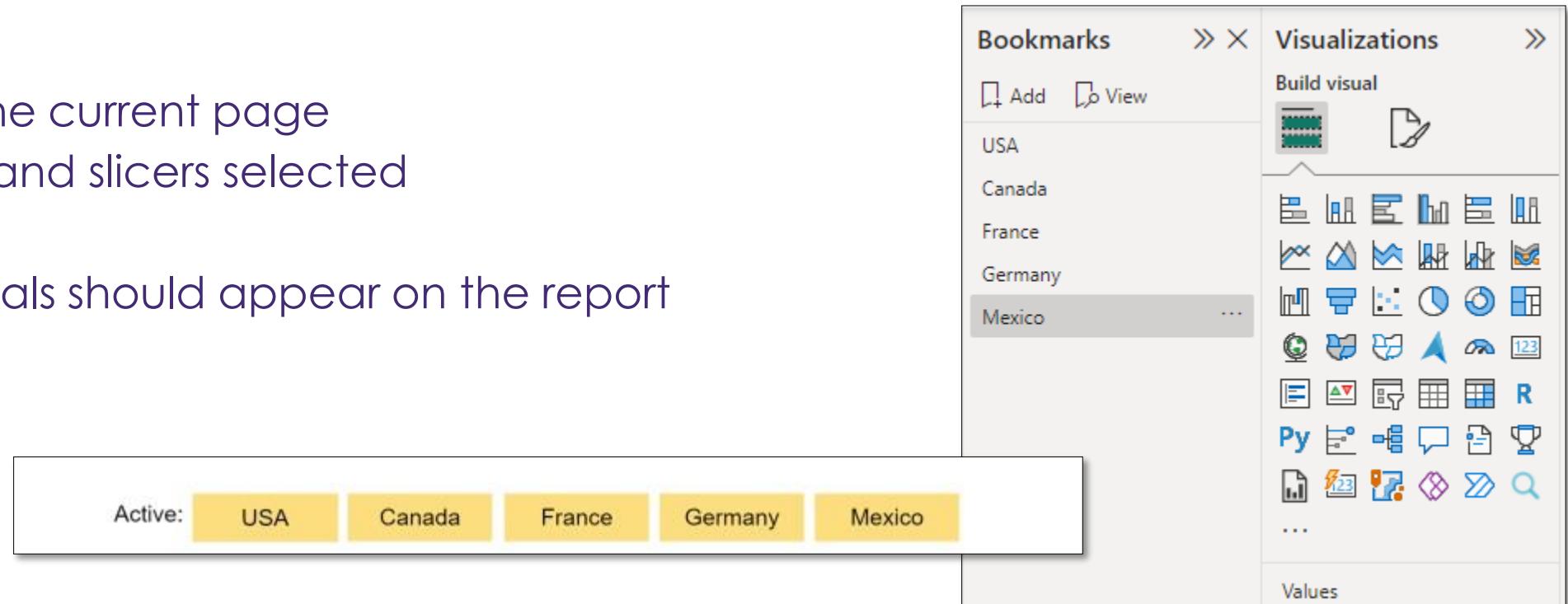
Nintendo

Details

Bookmark

Bookmarks let you save interesting states as part of your report. Once you have a list of bookmarks, you can use these in several ways including organizing and transitioning visuals:

- Includes the current page
- Any filters and slicers selected
- Sort Order
- Which visuals should appear on the report



Report Drill-through

Total Actual Award by Job Type



Grant Line Details

	Program	Company	Award Date	Funding ID	Status	Actual Job Count	Required Job Counts
1	JDIG	ABB Inc.	Thursday, September 09, 2010	273	Terminated, Funds Disbursed	156	130
2	JDIG	American Roller Bearing Company of North Carolina	Thursday, December 08, 2011	409	Terminated, Funds Disbursed	0	208
3	JDIG	AptarGroup, Inc.	Monday, May 16, 2011	355	Active	106	135
4	JDIG	ASCO Power Technologies, L.P.	Wednesday, February 18, 2009	142	Terminated, Funds Disbursed	198	295
5	JDIG	Ashley Furniture Industries, Inc. I	Friday, April 20, 2012	437	Active	1229	468
6	JDIG	Brunswick Corporation (Hatteras Yachts Division)	Tuesday, July 20, 2010	257	Terminated, Funds Disbursed	221	315
7	JDIG	Caterpillar Inc. (Bee)	Wednesday, February 01, 2012	422	Terminated, Funds Disbursed	111	169

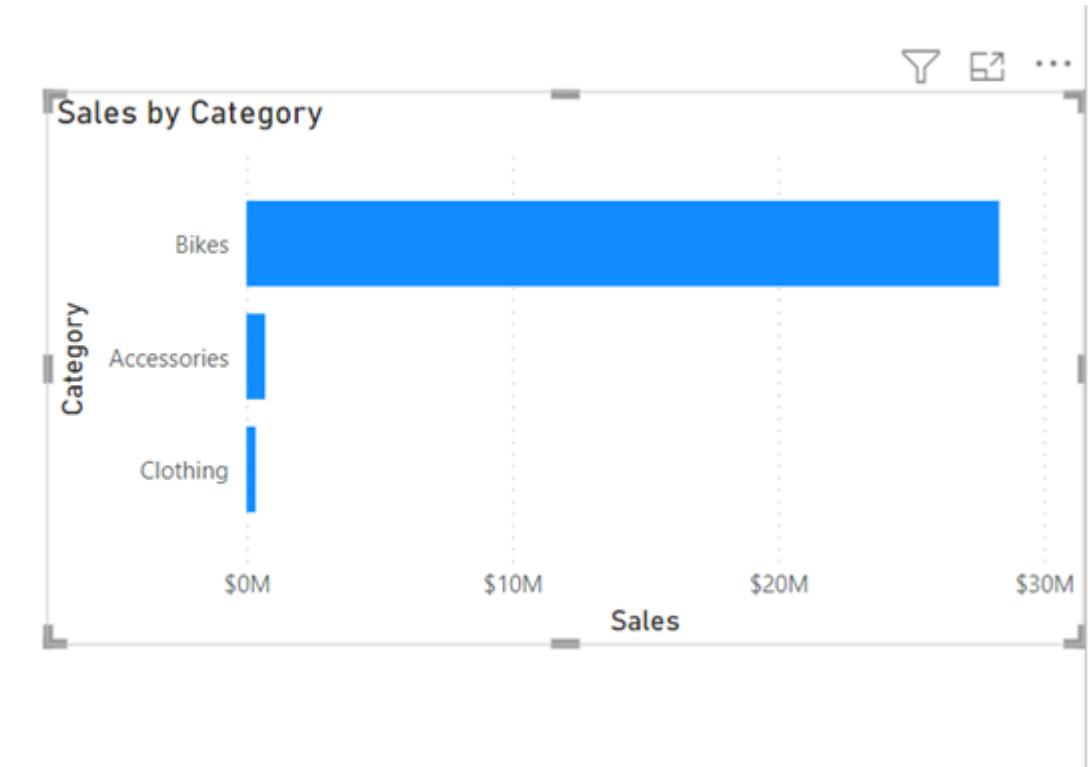
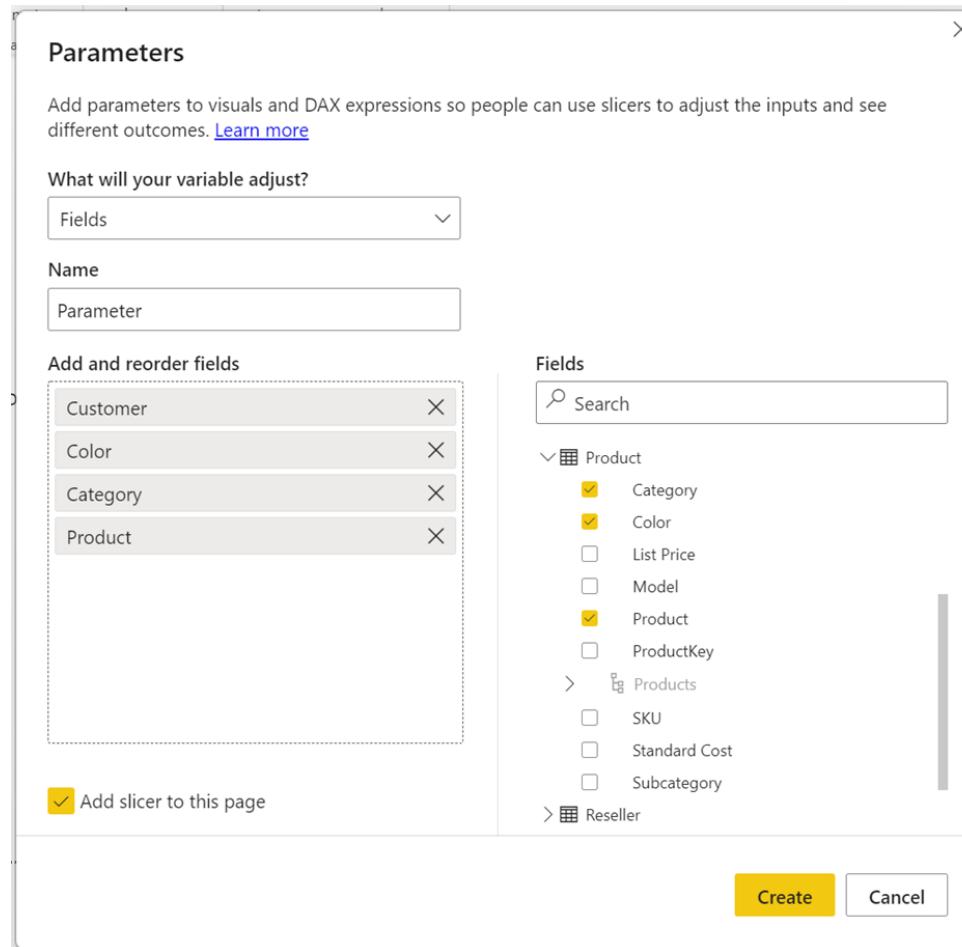
Custom Tooltip



Custom tooltips in Power BI let you create personalized and interactive data point descriptions that appear when users hover over visuals, enhancing the user experience with tailored information and insights.



Field Parameters



Performance Analyzer

Built-in performance analysis

- Each visual has three components
 - How long did the DAX query take?
 - How long did my visual take to render?
 - How long was everything else?

The screenshot shows the Power BI Performance Analyzer interface. At the top, there are buttons for 'Start recording' (disabled), 'Refresh visuals', and 'Stop'. Below this is a table of recorded queries with columns for 'Name' and 'Duration (ms)'. The table shows the following data:

Name	Duration (ms)
Recording started /9/27/2023 5:21:00 PM	-
GLOBAL	237
Data Load	638
Power Query Editor	79
DirectQuery	18
R scripting	541
Python scripting	637
Security	580
Privacy	670
Updates	715
Usage Data	
Diagnostics	
Preview features	
Auto recovery	
Report settings	
CURRENT FILE	
Data Load	
Regional Settings	
Privacy	
Auto recovery	
DirectQuery	
Query reduction	
Report settings	

On the left, there is a sidebar titled 'Options' with sections for 'GLOBAL' and 'CURRENT FILE'. Under 'GLOBAL', there are checkboxes for 'Reduce number of queries sent by' (unchecked) and 'Disabling cross highlighting/filtering by default' (unchecked). Under 'CURRENT FILE', there are checkboxes for 'Show an Apply button and only send queries once for' (unchecked) and 'Slicer selections' (unchecked) and 'Filter selections' (unchecked). At the bottom right are 'OK' and 'Cancel' buttons.

Use Query Reduction Features

Scenario

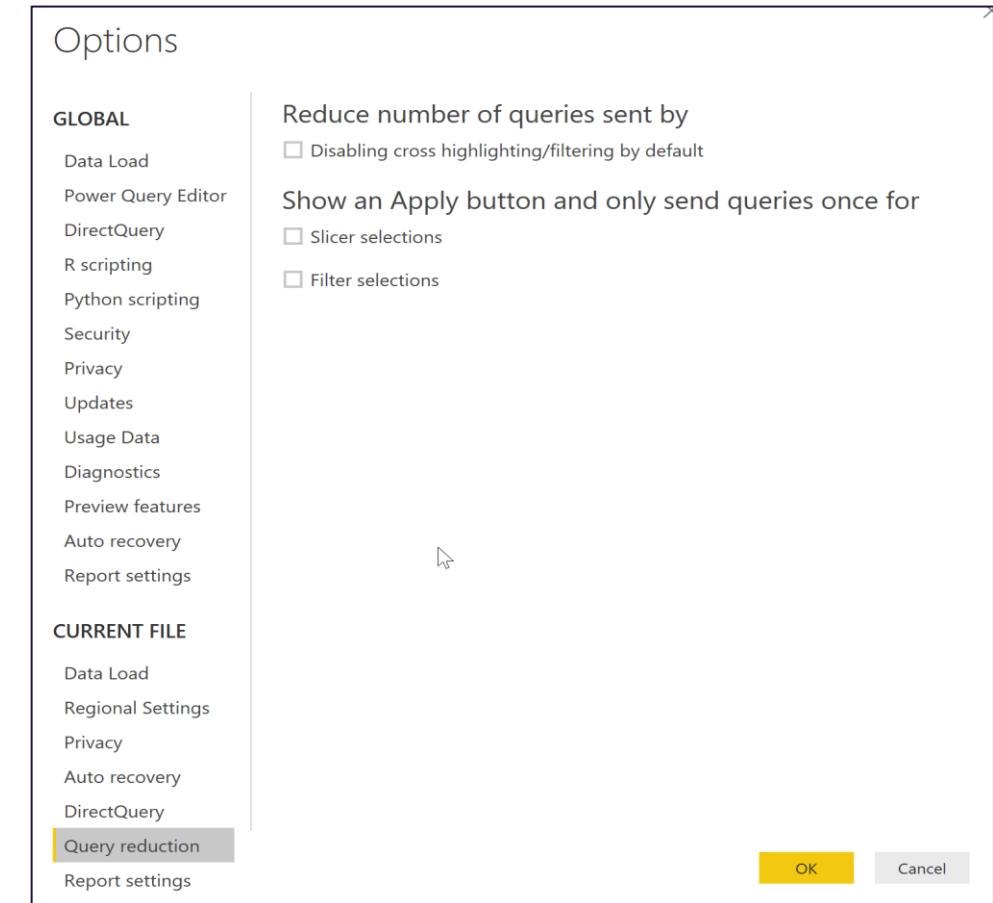
- Previously discussed high number of visuals/slicers

Why is it desired?

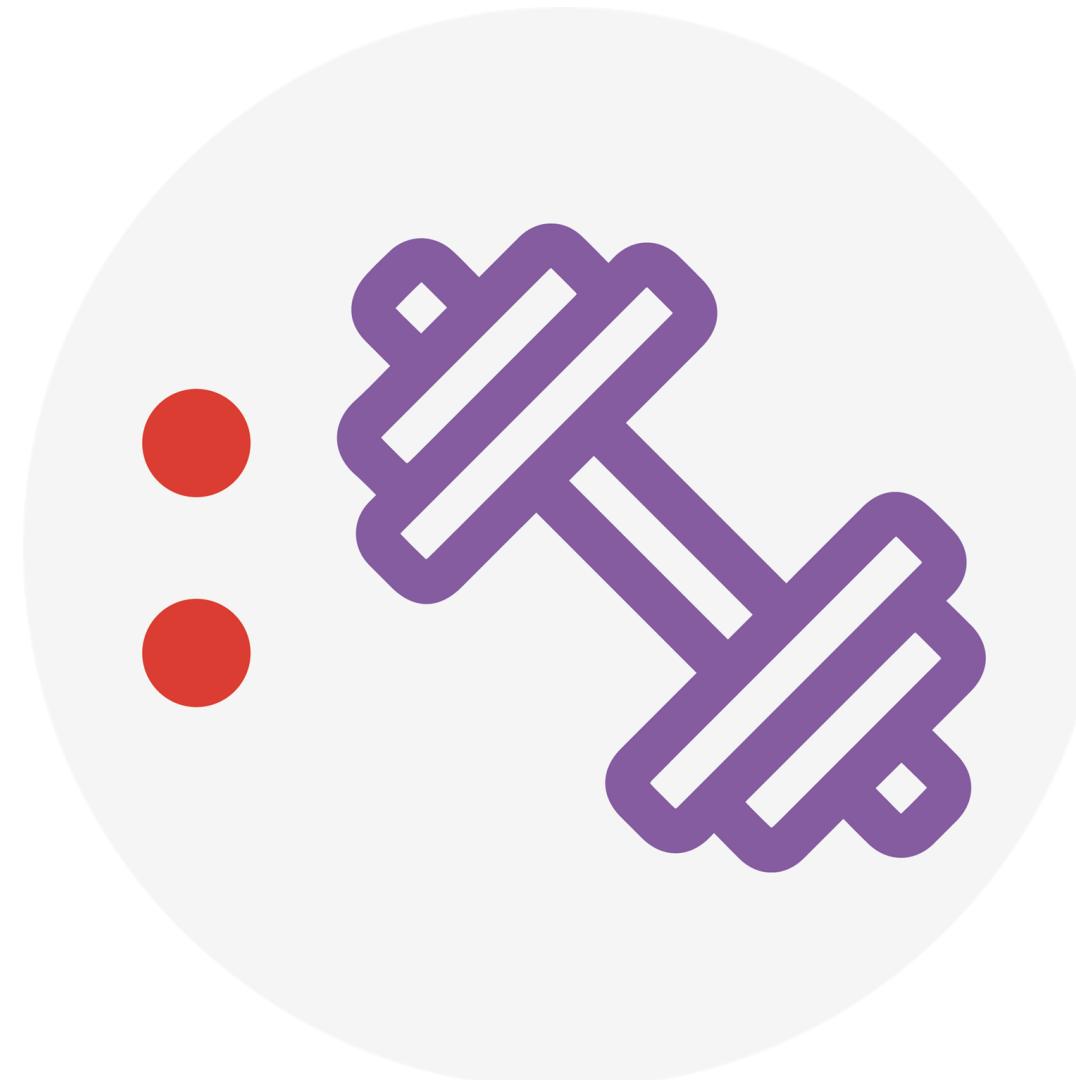
- Lower memory load, less data to fetch and process

Proposed solutions

- consider no cross-highlighting, and
- adding Apply buttons to slicers and filters



Live Demo



KC7



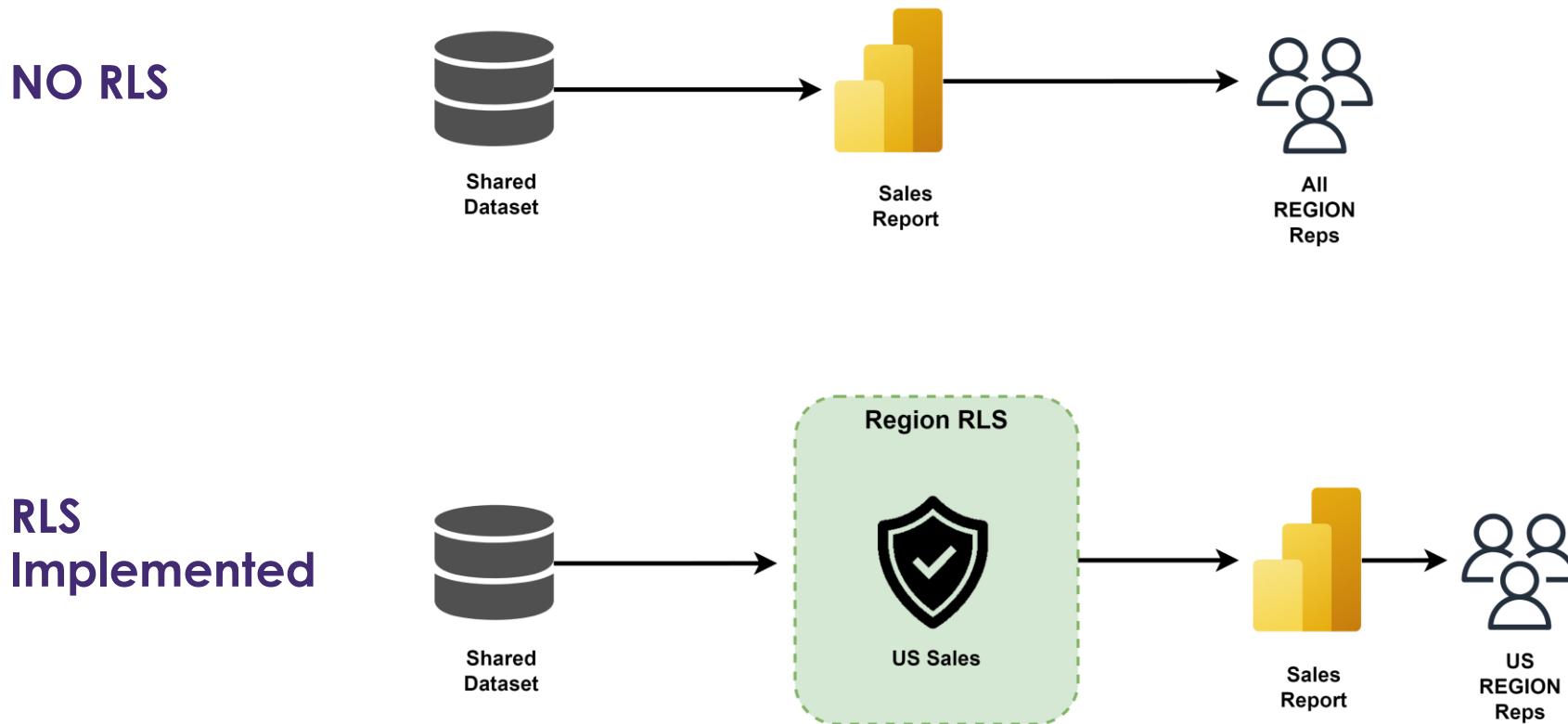
Lesson 8

Row-Level Security

After completing this module, students will be able to:

- Define RLS
- Implement Static RLS
- Implement Dynamic RLS

What is Row-Level Security



Row-Level Security (RLS) in Power BI is a security feature that allows you to control access to data at a granular level. With RLS, you can define rules that restrict which rows of data users can see in a dataset or report based on their roles.

Static Row-Level Security

Manage roles

Roles

Automotive

Clothing

Game

Sports

Create

Delete

Tables

Employees

Product

Table filter DAX expression

[department] = "Game"

Filter the data that this role can see by entering a DAX filter expression that returns a True/False value. For example: [Entity ID] = "Value"

Save

Cancel



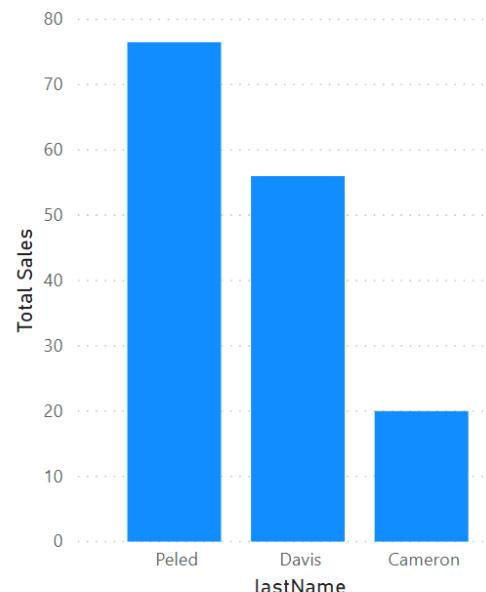
152.47

Total Sales

Top Products Sales

product	department	Total Sales
Spider, spider	Game	44.50
Invest in it All	Game	31.99
Santo Domingo	Game	31.00
Settlers of Air	Game	24.99
Lords of Avalon	Game	19.99
Total		152.47

Top Employees



Setting Up RLS in Power BI Service

The screenshot shows the Power BI Intermediate Training workspace. In the center, there is a list of items. One item, "Lab 7 -RLS", is highlighted with a red box. To its right is a three-dot menu icon, also highlighted with a red box. A context menu has appeared, listing options like "Analyze in Excel", "Create report", etc., with "Security" at the bottom, also highlighted with a red box.

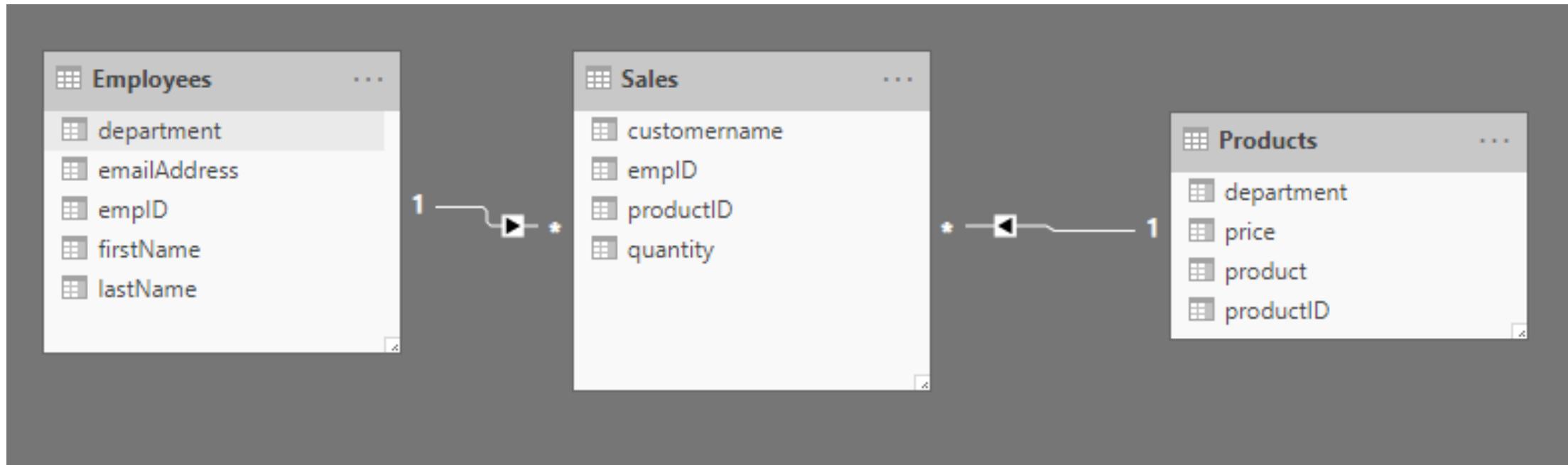
From your Power BI service workspace you will need to select the dataset security settings

The screenshot shows the Row-Level Security setup page. It lists two regions: "Canada Region (0)" and "US Region (0)". A red arrow points from the "US Region (0)" area to the "Members (0)" section. The "Members (0)" section displays a list of users: "Shannon Lynch" and "Tripat Gill", each with a delete icon. Below the list is a text input field "Enter email addresses" and a green "Add" button.

Add the users that should be assigned to the applicable RLS role

RLS Security

Secure reports and workspace by sharing them to Active Directory users and groups.



Dynamic RLS

Manage roles

Roles

EmployeeEmailAddress	...
Create	Delete

Tables

Employees	...
Products	...
Sales	...

Table filter DAX expression

```
[emailAddress] = userprincipalname()
```

Row-Level Security

Eastern US (0)

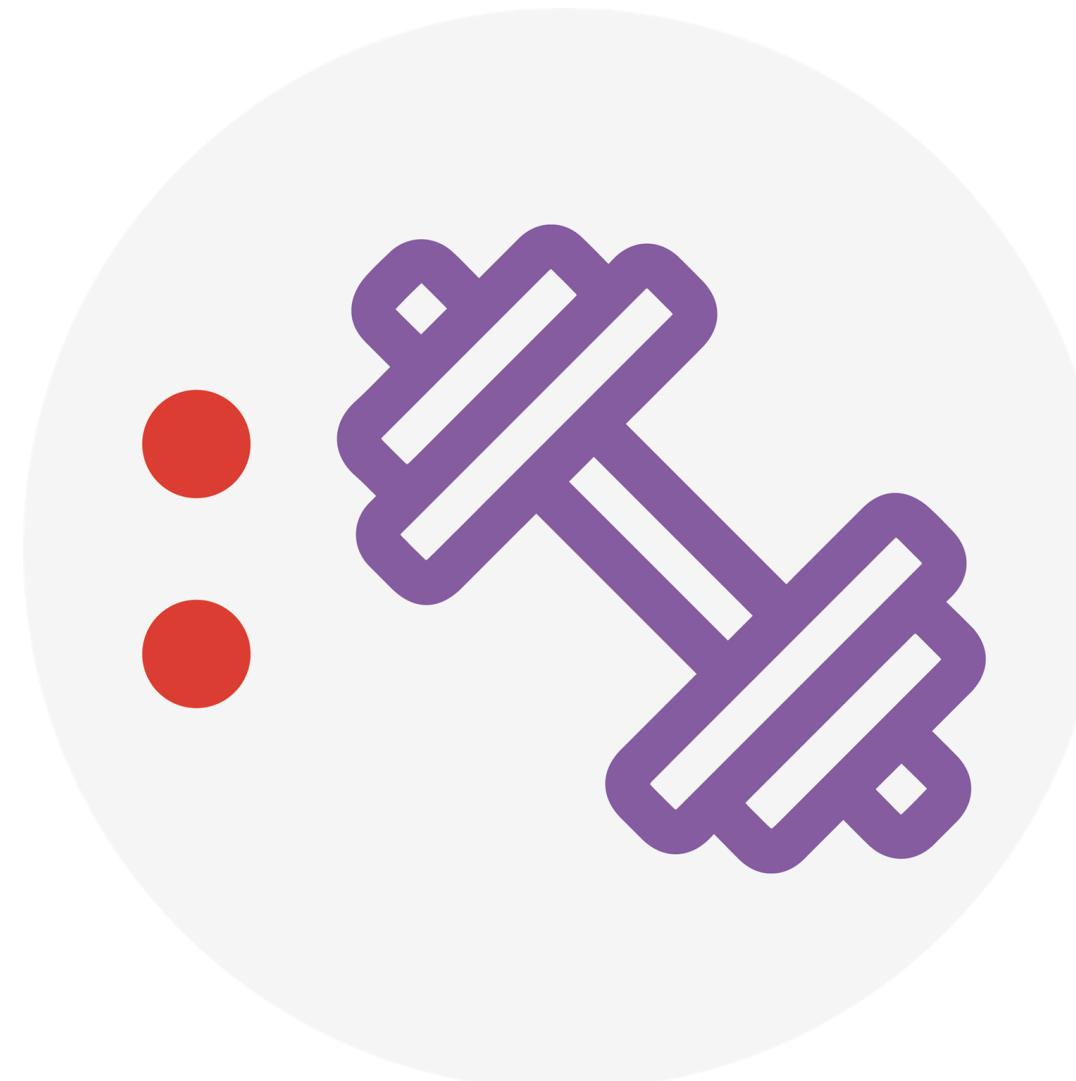
Members (0)

People or groups who belong to this role

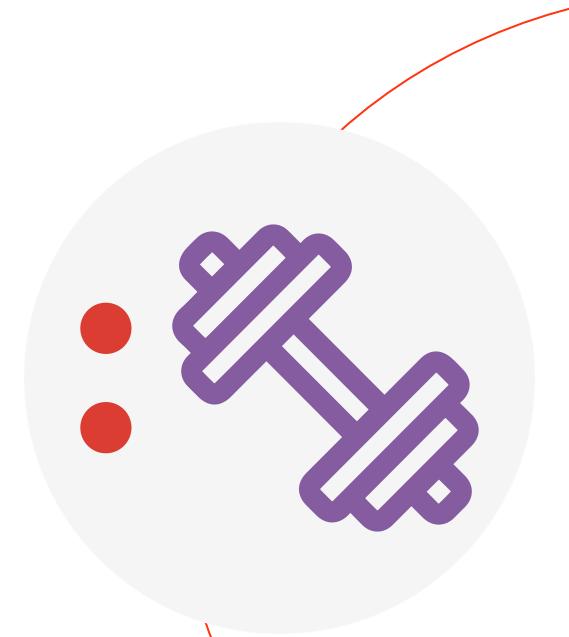
Enter email addresses

Add

Live Demo



Exercise 5



KC8

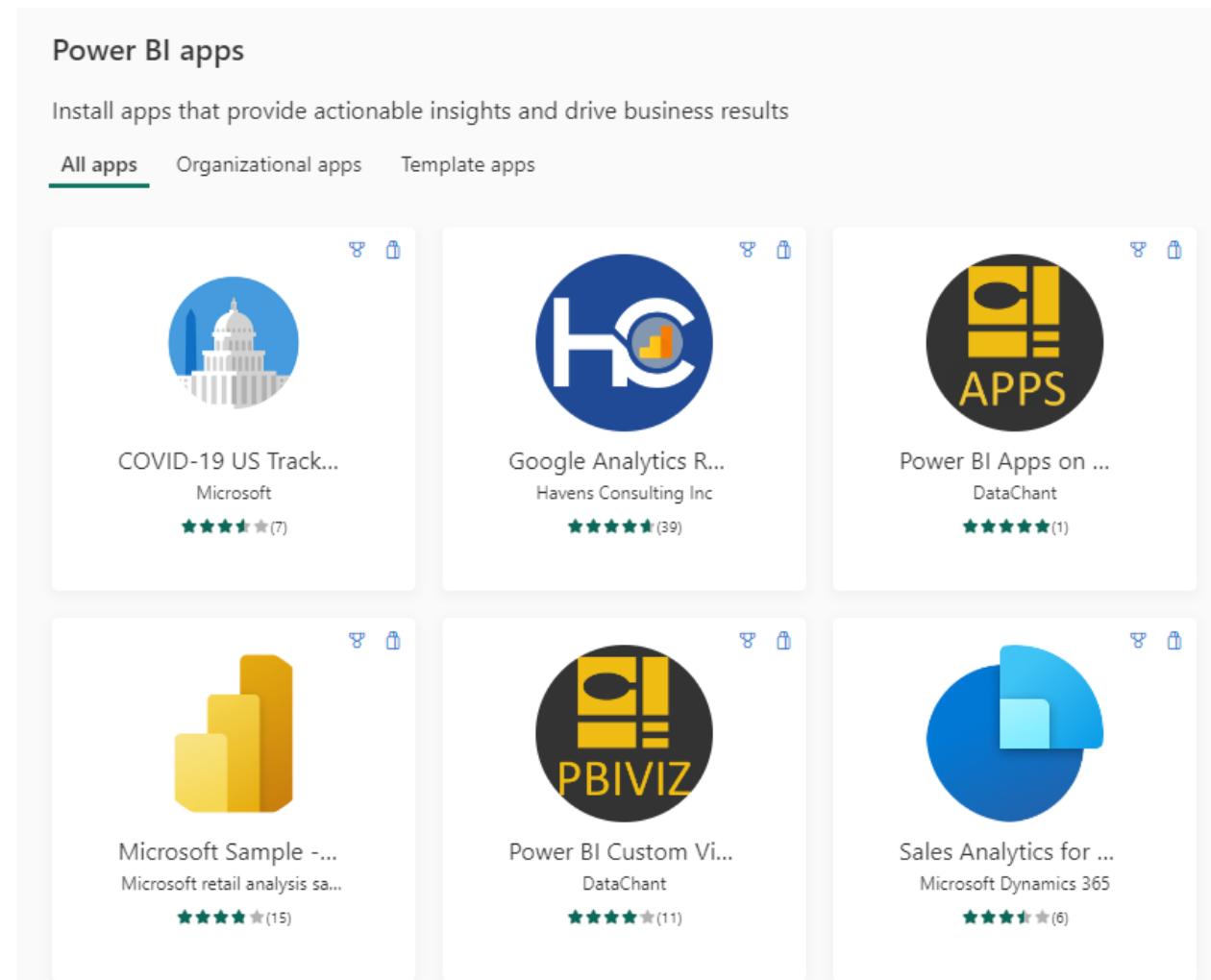


Power BI App Deployment

Publishing your report App

Within Power BI, you have the capability to craft formalized content packages and subsequently share them widely in the form of apps.

These apps are developed within workspaces, fostering collaborative work on Power BI content alongside your peers. After completion, you can then release the finalized app to a broad audience within your organization.

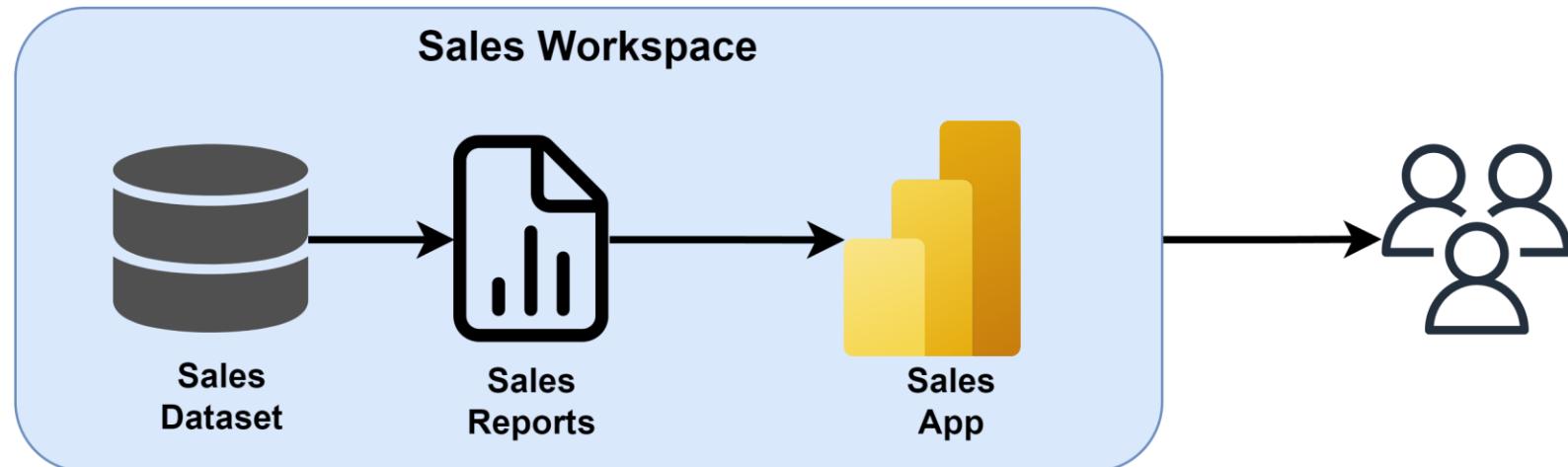
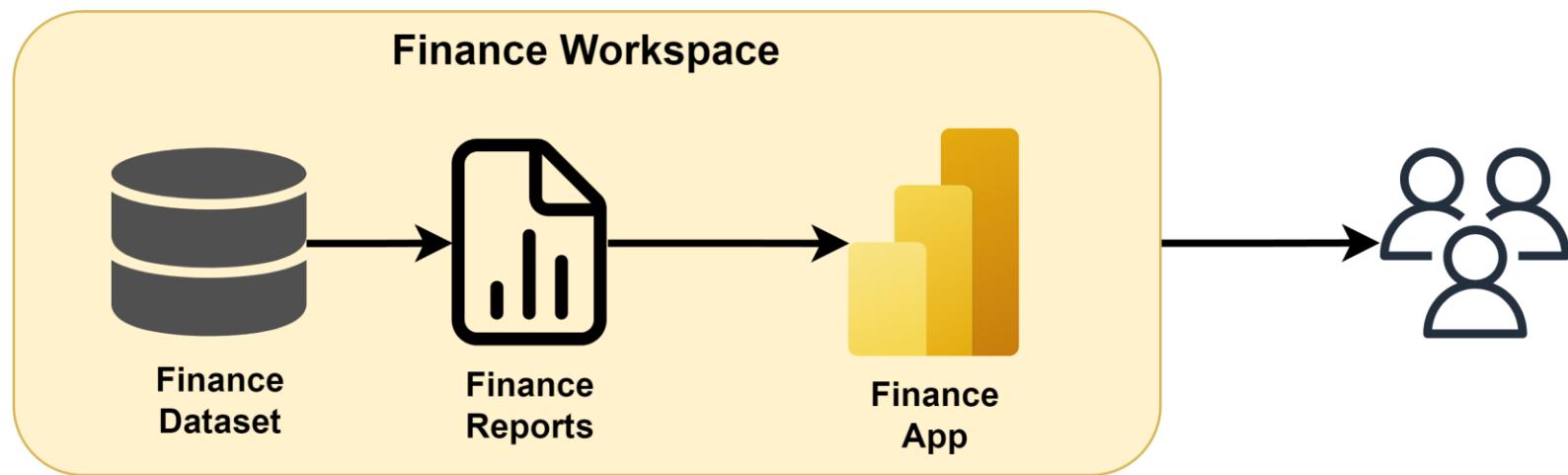


App Benefits of Report Distribution

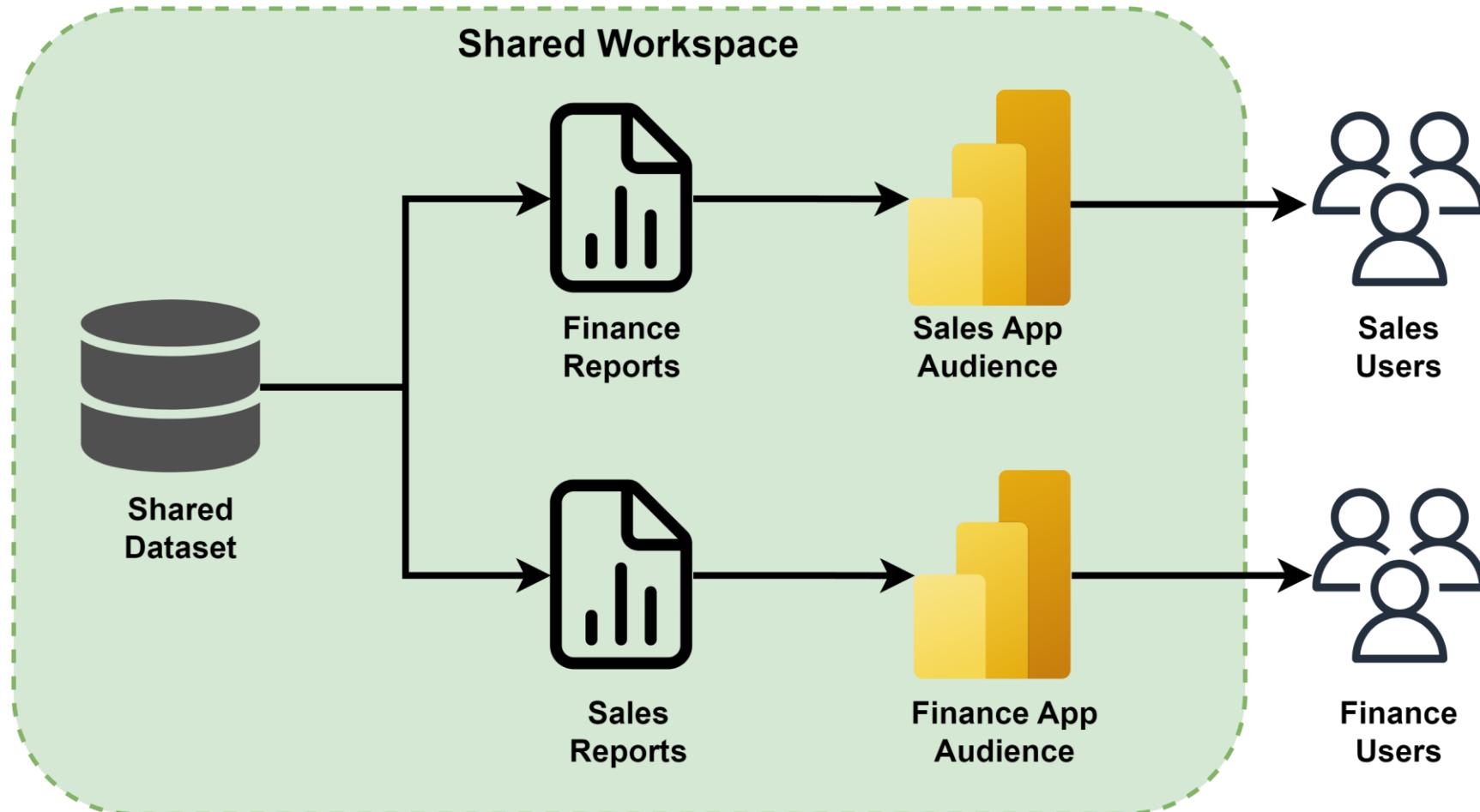
- ✓ Streamlined Access
- ✓ Enhanced Security
- ✓ Consistency
- ✓ Simplified Sharing
- ✓ Usage Metrics
- ✓ Mobile Access
- ✓ Automatic Updates
- ✓ User-Friendly Experience
- ✓ Scalability
- ✓ Version Control



Strategy #1: Separate Workspaces for Each User Group



Strategy #2: Shared Workspace for Multiple User Groups



Create and publish your App

Power BI Power BI Intermediate Training 1 ? ? ? ?

Power BI Intermediate Training

+ New + Upload + Create app Manage access Workspace settings Filter by keyword Filter Sort

Name	Type	Owner	Refreshed	Next refresh	Endorsement	Sensitivity	Included in app
Lab 6 -Advanced Visualization Completed	Report	Power BI Intermediate...	9/17/23, 10:56:37 PM	—	① Setup	② Content*	③ Audience*
Lab 6 -Advanced Visualization Completed	Dataset	Power BI Intermediate...	9/17/23, 10:56:37 PM	N/A	Build your app		

Setup **Content*** **Audience***

App name *
Power Cycle Report

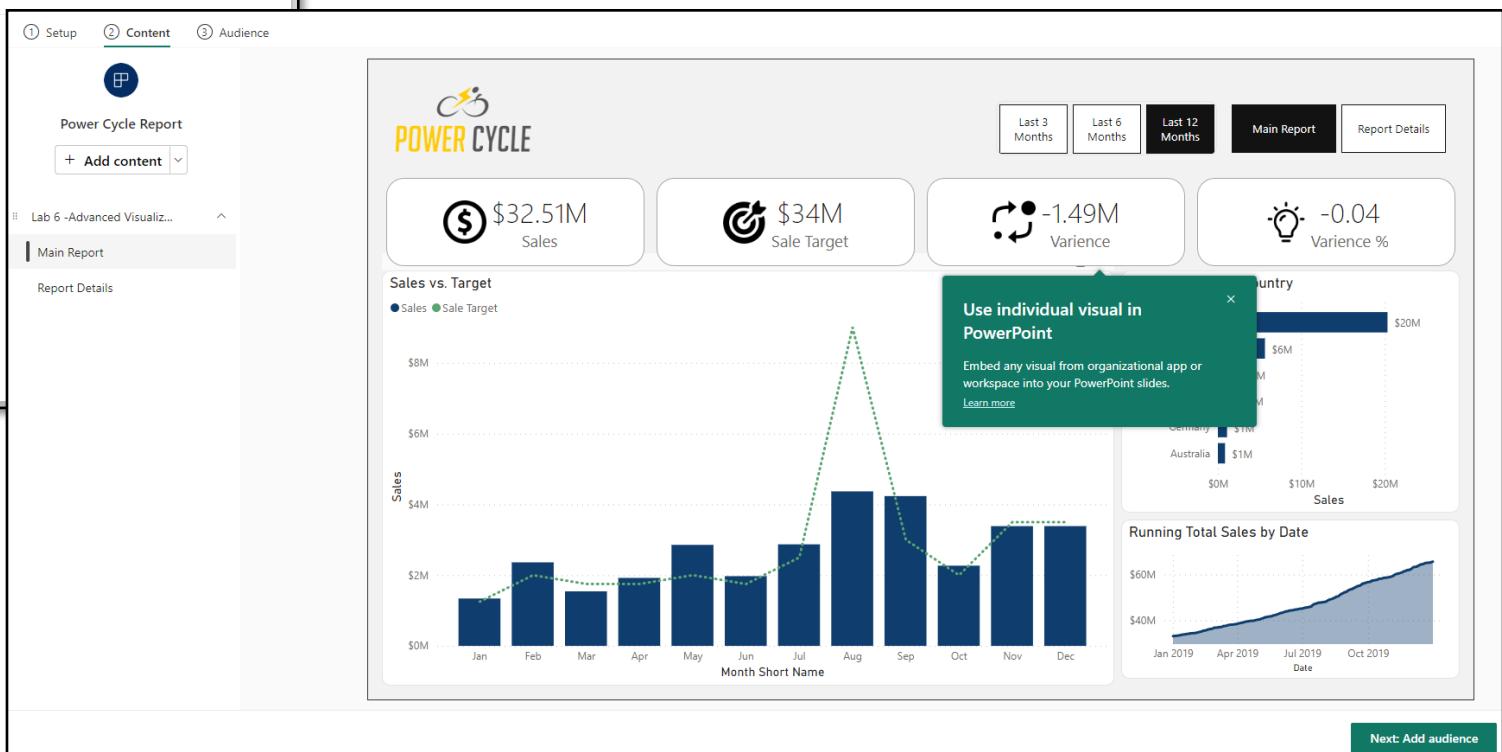
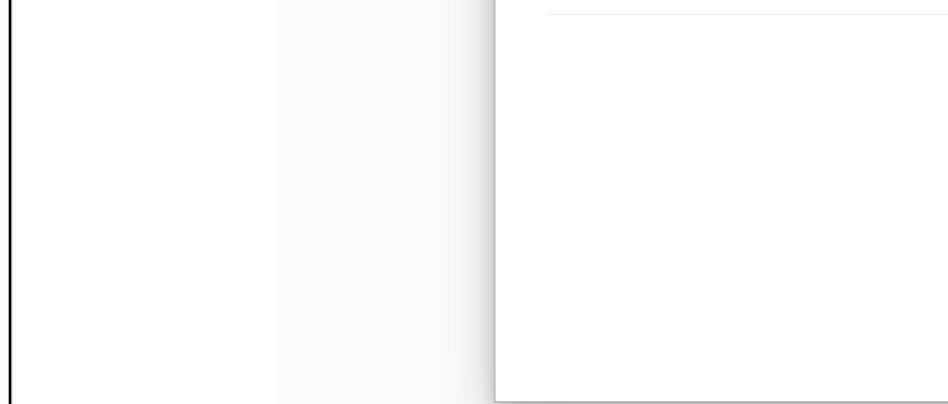
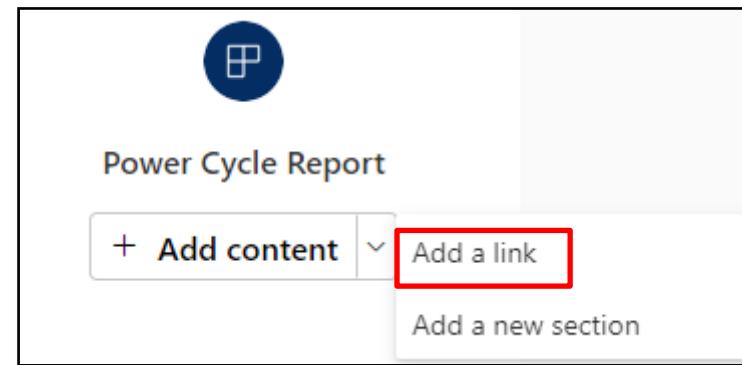
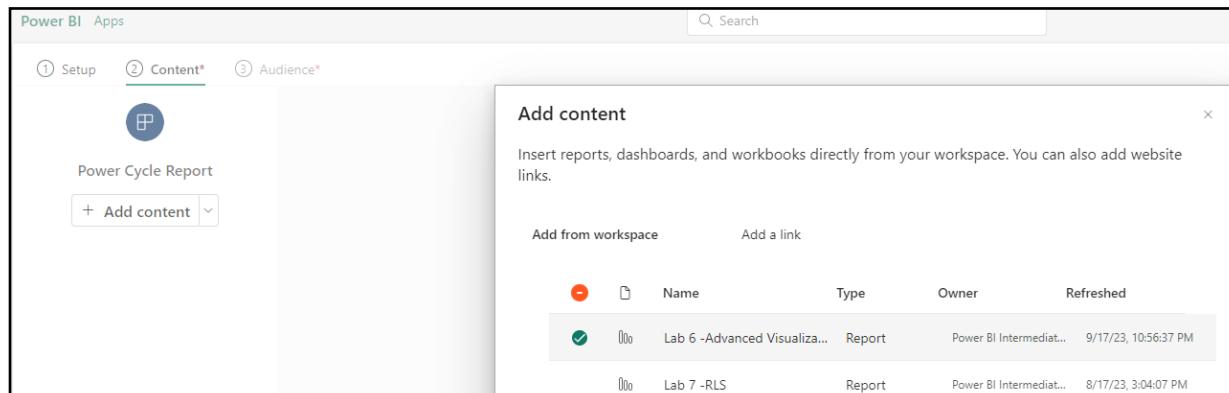
Description *
Power Cycle operational overview
168 characters left

App logo
 Upload Delete

App theme color


Contact information
 Show app publisher
 Show items contacts from the workspace
 Show specific individuals or groups

Report Content



Course Name Here

DATA SOCIETY © 2020 110

Managing App Audiences

① Setup ② Content ③ Audience

X

Audience

Manage your audiences and their permissions. Select what content each audience can see by toggling the eye icon.

Power Cycle Rep...

New Audience

+ New Audience



Add up to 10 audience views



Power Cycle Report

Lab 6 -Advanced Visu...

Main Report

Report Details

Lab 7 -RLS



Hide reports that should not be accessible to this audience

POWER CYCLE

\$32.51M
Sales

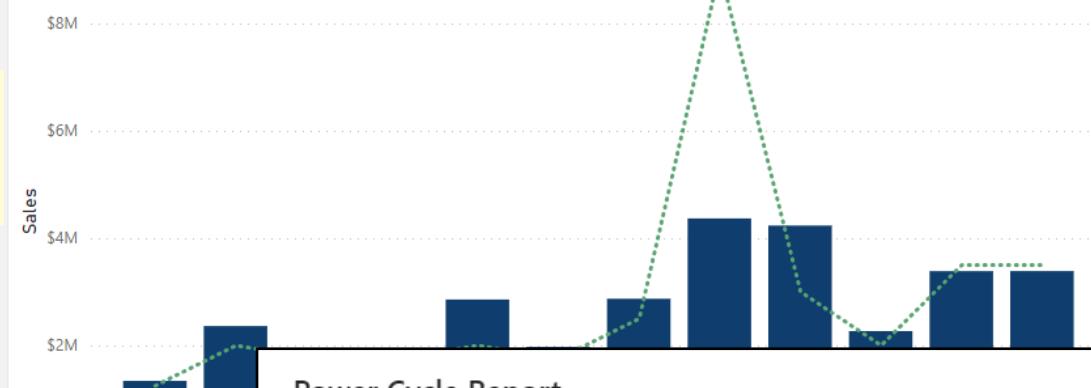
\$34M
Sale Target

-1.49M
Varience

-0.04
Varience %

Sales vs. Target

● Sales ● Sale Target



Power Cycle Report

When you publish an app that has large distribution, it might take a little while to process. Typically, the content will be available within 5-10 minutes, but it can take up to one day.

Publish

Cancel

Last 3 Months

Last 6 Months

Last 12 Months

Main Report

Report Detail

Edit Audience

Power Cycle Report

Grant access to

Entire organization [Learn more](#)

Specific users or groups

SL Shannon Lynch X

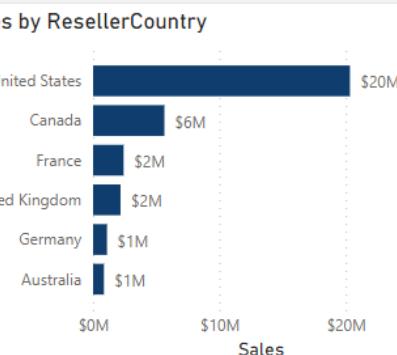
Enter a name or email address

> Advanced

Workspace users ⓘ



Determine who needs to have access to this view



Publish app

Cancel

Link to App

Successfully published

Power Cycle Report

Give people the link below, or direct them to Apps > Get apps in the Power BI service.

<https://app.powerbi.com/Redirect?action=OpenApp&appId=ab92294c-fee3-4bfe-98f9-799429350f55&ctid=451>

Copy

Go to app Close

Power Cycle Report

Main Report Report Details

Last 3 Months Last 6 Months Last 12 Months Main Report Report Details

POWER CYCLE

Sales \$32.51M **Sale Target** \$34M **Variance** -1.49M **Variance %** -0.04

Sales vs. Target

● Sales ● Sale Target

Sales

\$8M
\$6M
\$4M
\$2M
\$0M

Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

Month Short Name

Sales by ResellerCountry

ResellerCountry	Sales
United States	\$20M
Canada	\$6M
France	\$2M
United Kingdom	\$2M
Germany	\$1M
Australia	\$1M

Running Total Sales by Date

\$60M
\$40M

Jan 2019 Apr 2019 Jul 2019 Oct 2019

Date

Go back

Managing App users

The screenshot shows the Data Hub interface. On the left, there's a sidebar with icons for 'data hub' (selected), 'Apps' (highlighted with a red box), 'Metrics', and 'Monitoring'. In the center, a card for the 'Power Cycle Report' app is displayed, featuring a thumbnail, the app name, a star icon, three dots for more options, the creator 'Shannon Lynch', the creation date '9/17/23, 11:18:52 PM', and 'Org app' status. A context menu is open from the three-dot icon, listing 'Edit', 'Delete', and 'Manage permissions', with 'Manage permissions' also highlighted with a red box.

The screenshot shows the 'Power Cycle Report' app settings page. At the top, there's a header with a blue circular icon containing a white 'P', the app name, and a 'Edit' button. Below the header, there are two buttons: '+ Add user' and 'Manage audiences' (both highlighted with a red box). Underneath these buttons, there are three tabs: 'Direct access', 'Pending requests', and 'Pending invitations' (all highlighted with a red box). The main content area has four sections: 'People and groups with access' (with a 'Workspace users' button), 'Email Address' (with a dropdown set to 'All'), 'Audiences' (with a dropdown set to 'All'), and 'Options' (with a 'More options' button). The entire content area is enclosed in a red box.

App Endorsement

The screenshot shows the 'Power BI Intermediate Training' app settings page. At the top, there's a navigation bar with 'New', 'Upload', 'Update app', 'Manage access', 'Workspace settings', and a three-dot menu. The 'Endorse this app' option from the menu is highlighted with a red box. Below the navigation bar is a table with columns: Name, Type, Owner, and tabs for General, Dashboards, Datasets, Workbooks, Dataflows, and App (which is selected). A sidebar on the left shows 'Power Cycle Report'. The main content area has a heading 'Settings for Power Cycle Report' and a section 'Endorsement' with three options: 'None' (radio button), 'Promoted' (radio button, selected), and 'Certified' (radio button). The 'Promoted' option is also highlighted with a red box. There's a checked checkbox for 'Feature on Home'. At the bottom, there's a section titled 'Apps' with a table showing one item: 'Power Cycle Report' by 'Shannon Lynch' published on '9/17/23, 11:18:52 PM' as an 'Org app'. The 'Endorsement' column for this item is also highlighted with a red box and shows a blue 'Promoted' button.

Power BI Intermediate Training

+ New ▾ Upload ▾ Update app Manage access Workspace settings ...

Endorse this app

Feature this app on Home

Name	Type	Owner
General		
Dashboards		
Datasets		
Workbooks		
Dataflows		
App		

Power Cycle Report

Settings for Power Cycle Report

Endorsement

Help coworkers find your quality content by endorsing this app. [Learn more](#)

None
The app will appear in search results but isn't endorsed.

Promoted
When you're ready to distribute the app to your coworkers, promote it to let them know.

Certified
Certify your app to show coworkers that it's been reviewed and meets your org's certification criteria. [How do I get my app certified?](#)

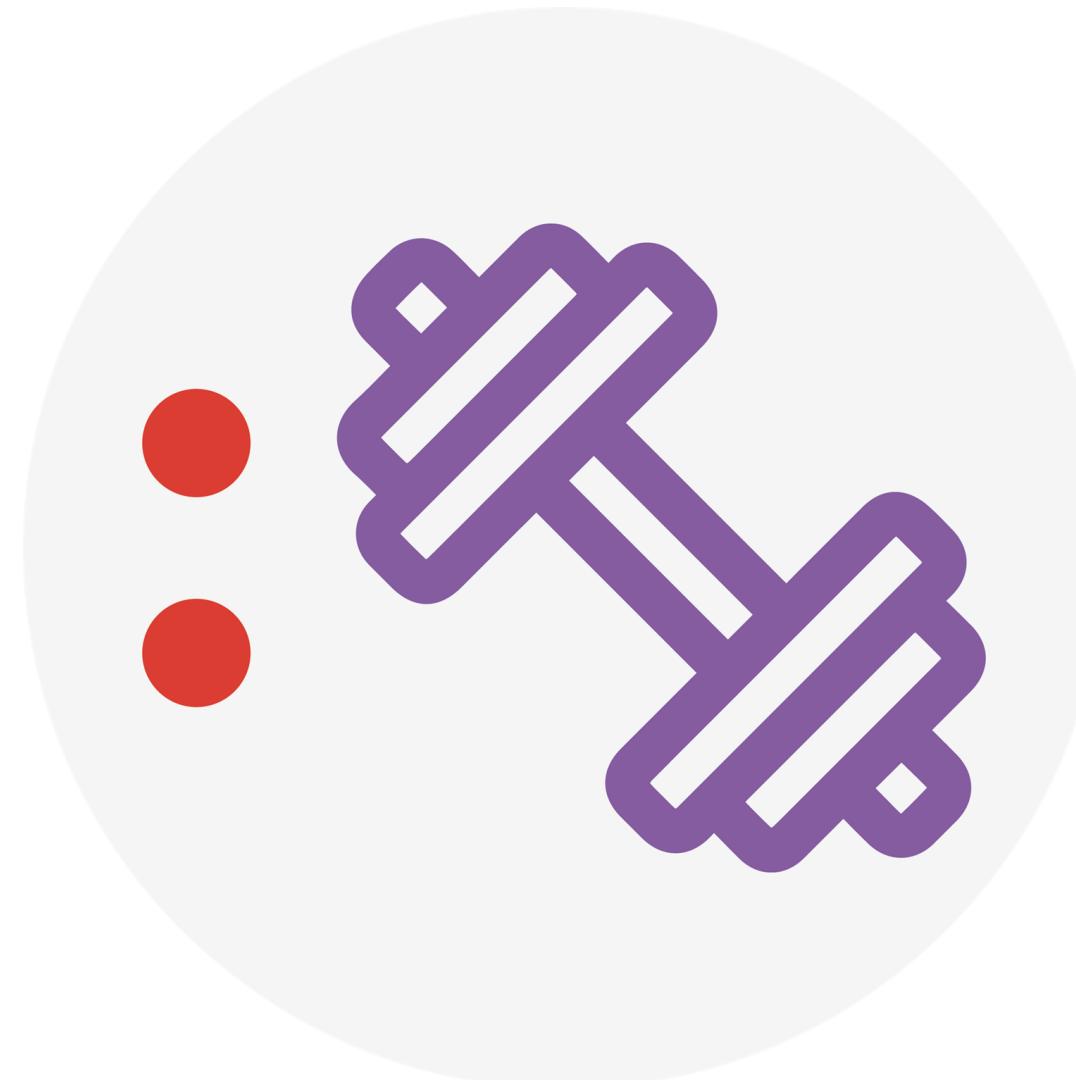
Feature on Home

Apps

Apps are collections of dashboards and reports in one easy-to-find place.

Name	Publisher	Published	App type	Version	Endorsement ↑
Power Cycle Report	Shannon Lynch	9/17/23, 11:18:52 PM	Org app	—	Promoted

Live Demo



KC9



Power BI Support Resources

[Community.PowerBI.com](#) – Community Forum

[Report Theme Gallery](#) – A showcase for stunning report themes

[Data Stories Gallery](#) – Get inspired with Data Stories by other Power BI users

[R-Visuals Gallery](#) – Get inspired by others use of R for analyzing their data

[Store.office.com](#) – Custom PBI visuals and R visuals you can download and use in your story

[Power BI Blog](#) - weekly updates

[User Voice for Power BI](#) – Vote on (or submit) your favorite new ideas for Power BI

[Issues.PowerBI.Com](#) – log issues with the community

[Whitepaper](#) - Creating an Enterprise Class Dashboard *Solution with Power BI*

[Guided Learning](#) Self Service Power BI training

[DAX Formula Language](#) – syntax for DAX

[DAX Patterns](#) – Great website to learn new patterns for the DAX Language

[Power Query Formula Language](#) – syntax for the “Query” language

[Paletton.com](#) – a color scheme generator

<https://unicode-table.com/en/> – Unicode Character Table

[Theme Generator](#)

[Contrast Analyzer](#): a tool that creates a “lens” to show how people with different visual disabilities might see your reports

[Charticulator](#): a tool that helps to build custom visuals



Thank you!

hello@datasociety.com

1100 15th St. NW, Floor 4
Washington, D.C. 20005

(202)600-9635

datasociety.com