

Designing Queries to Extract and Transform Data



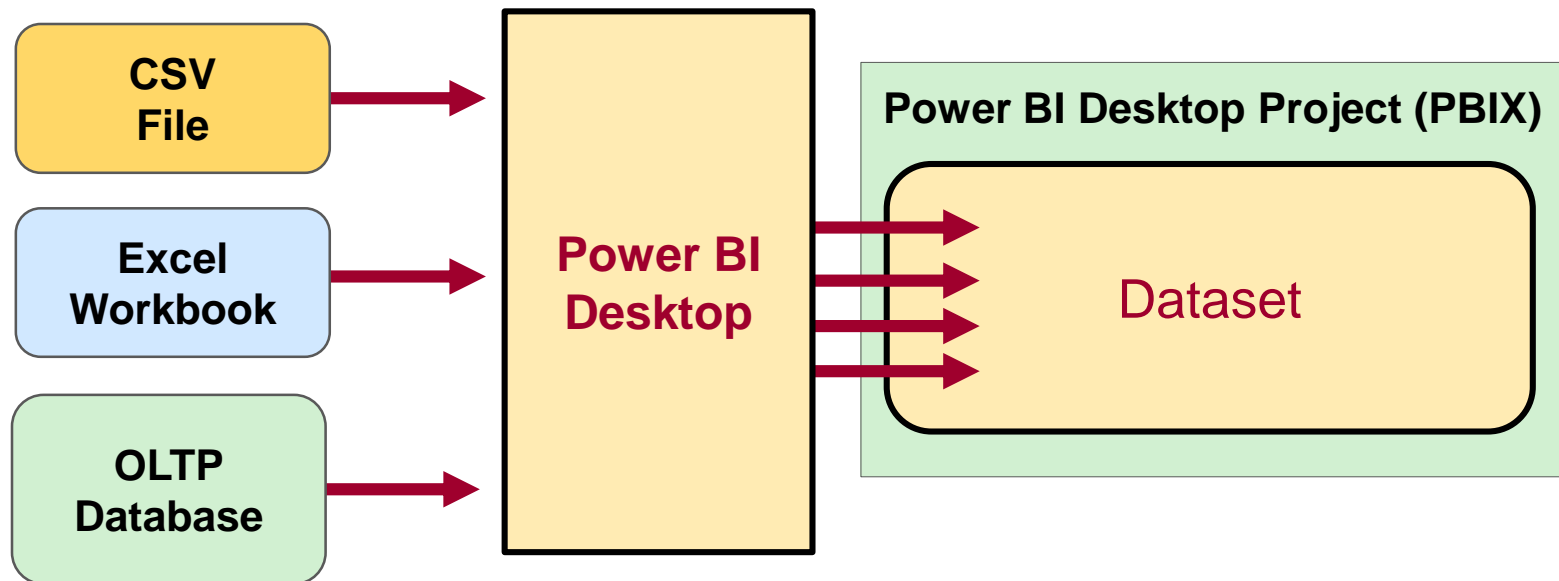
Agenda

- Query Design Fundamentals
 - Designing Data Model using a Star Schema
 - Working with the Query Editor Window
 - Importing Content From SharePoint Online
 - Understanding Parameters and Template Files
 - Designing with Function Queries



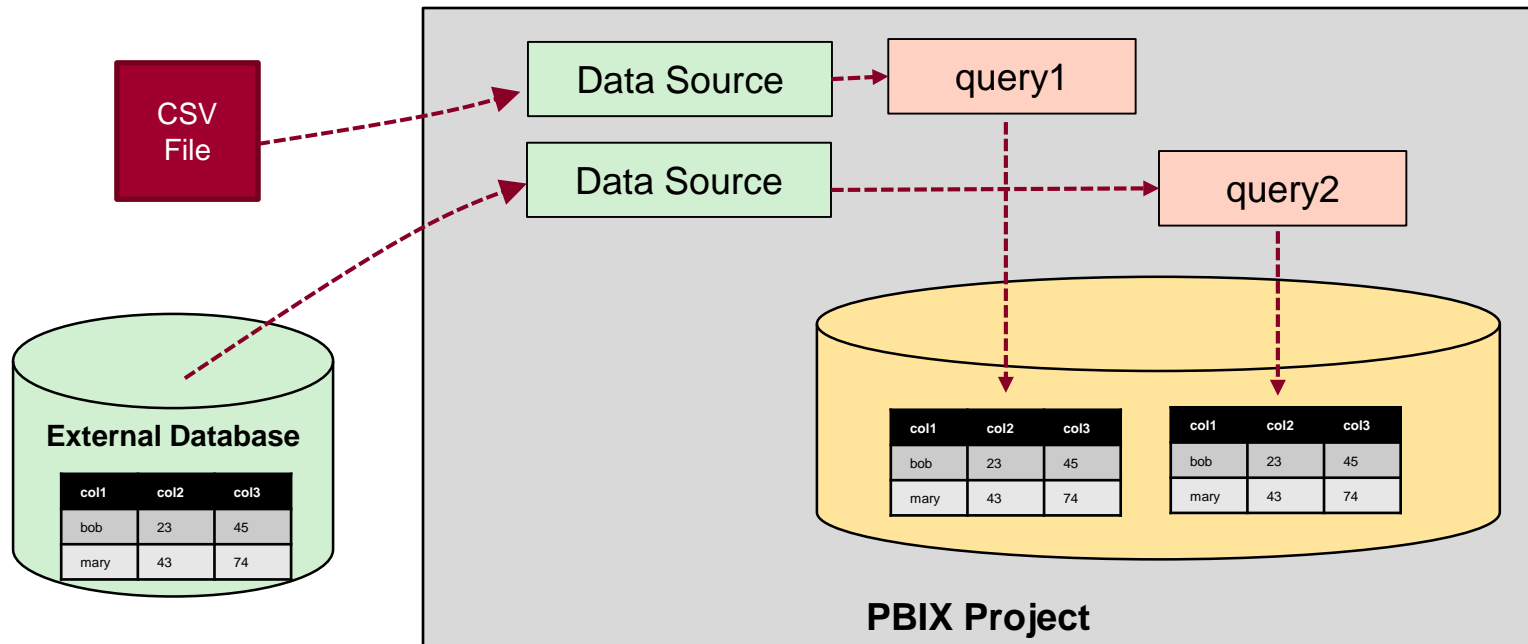
Power BI Desktop is an ETL Tool

- ETL process is essential part of any BI Project
 - **Extract** the data from wherever it lives
 - **Transform** the shape of the data for better analysis
 - **Load** the data into dataset for analysis and reporting



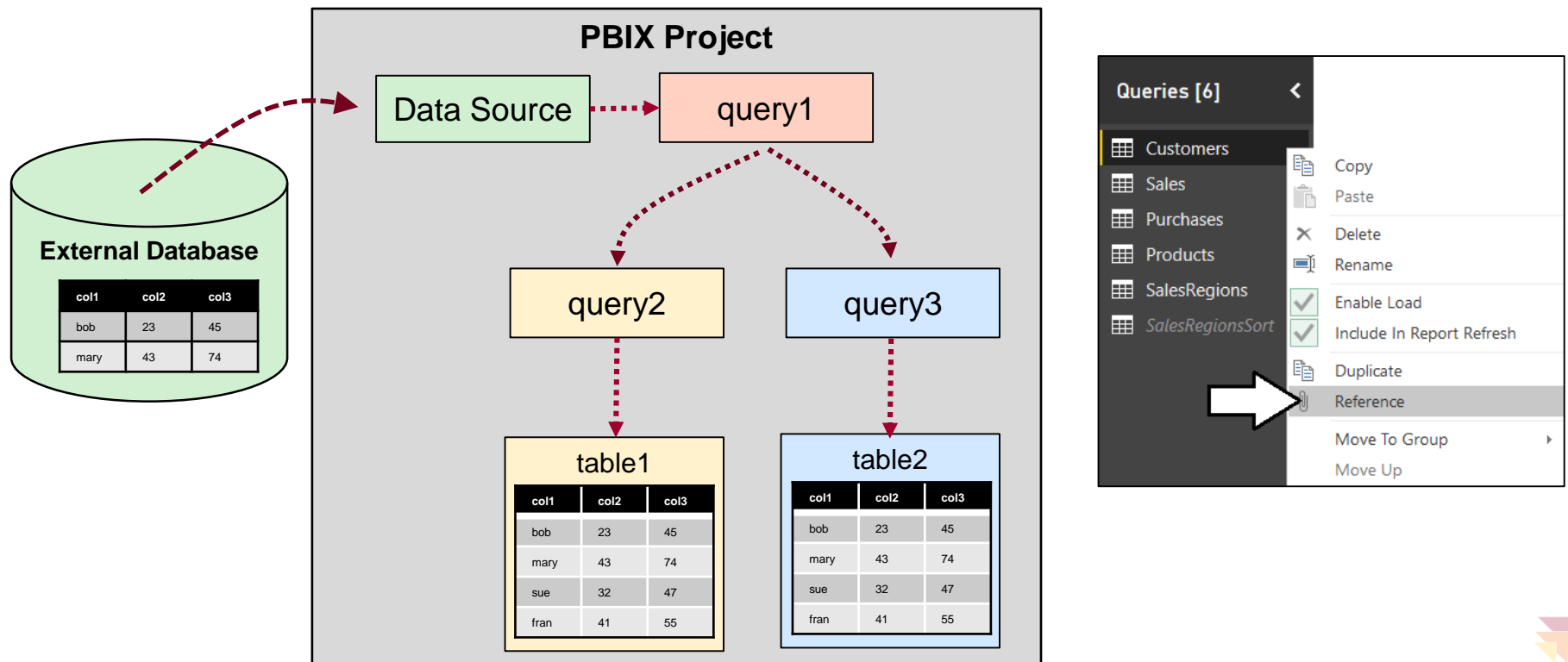
Understanding Query Input and Output

- PBIX project is container for data sources and queries
 - Queries created and saved within scope of Power BI project
 - Queries can pull data from local files
 - Queries can pull data from external content sources
 - Queries main purpose is to load imported data into data model



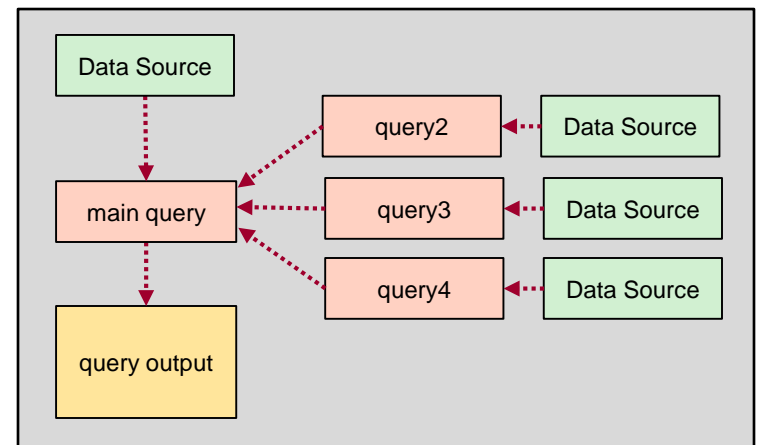
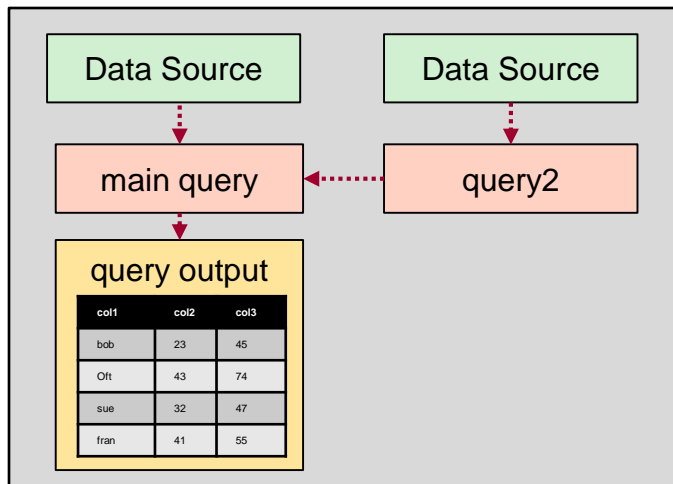
Query Composition

- Query can serve as source for other queries
 - Allows for creation of reusable base queries & query composition
 - Complexity can be hidden in base queries
 - **Reference** command creates new query based on another query



Combining Queries

- Query can be merged or appended with another query
 - Merge operation allows you combine columns from two tables
 - Append operation allows you to combine rows from two tables
- Two queries are combined into single output for loading
 - Load settings of main query determines where output is loaded
 - Secondary query acts as source for main query
 - Secondary query can be created with connection-only load setting



Query Steps

- A query is created as a sequence of steps
 - Each step is a parameterized operation on the data
 - Each step has formula which can be viewed/edited in formula bar
 - Query starts with Source step to extract data from a data source
 - Additional steps added to perform transform operations on data
 - You can replay query operations one by one by clicking on steps

The screenshot displays the Power BI Query Editor interface. At the top, the ribbon includes 'File', 'Home', 'Transform', 'Add Column', and 'View'. Below the ribbon, the 'Formula Bar' is active, showing the formula: `= Table.ReplaceValue("#Replaced Female Values","M","Male",Replacer.ReplaceText,`. A red dashed box highlights the formula bar, with a yellow callout box labeled 'step formula bar' pointing to it. On the left, the 'Queries [6]' pane lists 'Customers', 'Sales', 'Purchases', 'Products', 'SalesRegions', and 'SalesRegionsSort'. The main area shows a data table with columns: CustomerId, Customer, State, City, Zipcode, and Gender. The table contains 14 rows of data. On the right, the 'Query Settings' pane is open, showing the 'Properties' section with 'Name' set to 'Customers'. Below it, the 'Applied Steps' section is highlighted with a red dashed box and a yellow callout box labeled 'sequential list of steps for query'. The 'Applied Steps' list includes: Source, Navigation, Removed Other Columns, Merged Columns, Reordered Columns, Replaced Female Values, Replaced Male Values (which is selected with a mouse cursor), Changed Type, and Added Conditional Column.

CustomerId	Customer	State	City	Zipcode	Gender
1	Nina Diaz	CA	Eureka	95501	Female
2	Melinda Carter	CA	Napa	94558	Female
3	Pam Miller	CA	Napa	94558	Female
4	Merle Blackwell	CA	Sacramento	95823	Female
5	Ariel Hale	CA	Sacramento	95818	Male
6	Randy Carter	CA	Sacramento	95818	Male
7	Lillie Hinton	CA	Eureka	95501	Female
8	Ladonna Moody	CA	Napa	94559	Female
9	Buddy McKay	OR	Bend	97701	Male
10	Warren Sykes	CA	Sacramento	95818	Male
11	Jan Rutledge	OR	Portland	97216	Female
12	Dallas Lester	OR	Eugene	97402	Male
13	Matthew Zimmerman	OR	Portland	97220	Male
14	Sheryl Hernandez	CA	Sacramento	95823	Female

Advanced Editor

- Power BI Desktop based on "M" functional language
 - Query in Power BI Desktop saved as set of M statements in code
 - Query Editor generates code in M behind the scenes
 - Advanced users can view & modify query code in Advanced Editor



Agenda

- ✓ Query Design Fundamentals
- Designing Data Model using a Star Schema
 - Working with the Query Editor Window
 - Importing Content From SharePoint Online
 - Understanding Parameters and Template Files
 - Designing with Function Queries



Data Modeling using a Star Schema

- OLAP Modeling often based on Star Schema
 - Tables defined as fact tables or dimension tables
 - Fact tables related to dimension table using 1-to-many relationships



Designing Queries to Build a Star Schema

- Converts OLTP Data Model to OLAP Data Model
 - Sales table is modeled as a OLAP Fact Table
 - Other tables are modeled as OLAP Dimension tables
 - Requires pulling CustomerId column into Sales table
 - All dimension tables should be directly related to fact table





DEMO

Exploring the Wingtip Sales Analysis Demo Project

Agenda

- ✓ Query Design Fundamentals
- ✓ Designing Data Model using a Star Schema
- Working with the Query Editor Window
 - Importing Content From SharePoint Online
 - Understanding Parameters and Template Files
 - Designing with Function Queries



Query Editor Ribbon Tabs

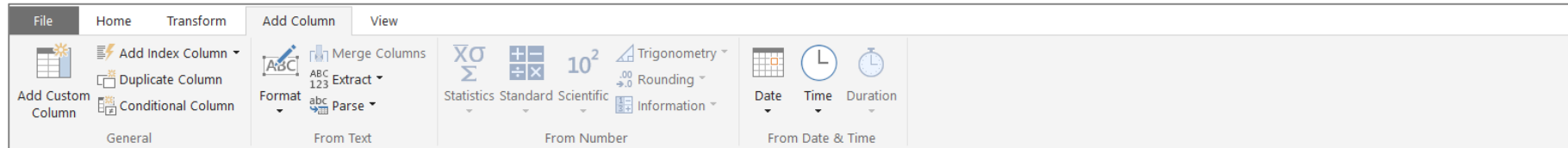
Home tab



Transform tab



Add Column tab



View tab



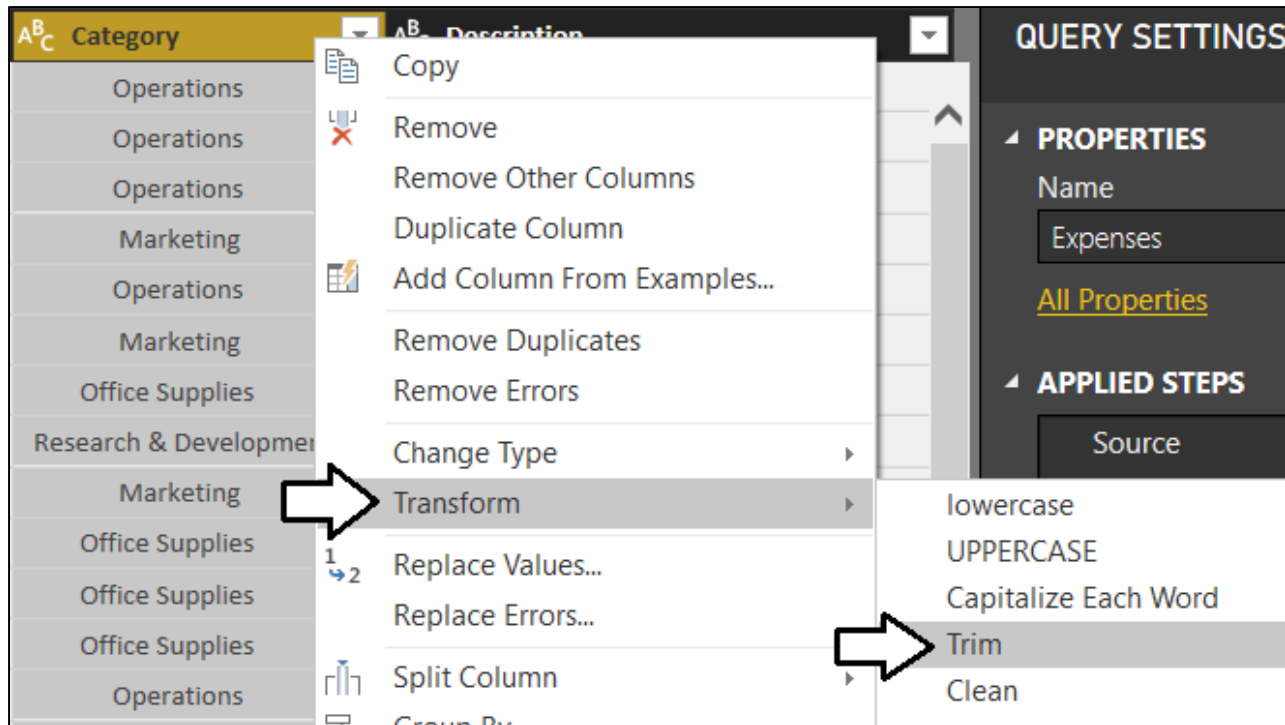
Examples of Basic Power BI Desktop Steps

- Rename column
- Convert column type
- Format column values
- Reorder columns
- Replace column values
- Expanding related column
- Merging columns
- Splitting columns











Cleaning Data

- Special steps available to clean up string-based data
 - **Transform > Trim** removes whitespace
 - **Transform > Clean** removed non-printable characters



Converting Column Types

- Transform data to make it more reliable
 - Convert date-time column to date column
- Transform data to make it more efficient
 - Convert decimal to fixed decimal number for currency

 PurchaseDate		10^3 Quantity		\$ SalesAmount		\$ ProductCost	
1/28/2012		1		2.95		1.2	Decimal Number
1/28/2012		6				\$	Fixed Decimal Number
1/28/2012		1		19.95		10^3	Whole Number
1/28/2012		5		249.75			Date/Time
1/28/2012		1		2.95			Date



Expanding Related Columns

- Used to pull data from related tables
 - Saves you from performing SQL joins or VLOOKUP

SalesAmount	Invoices	
119.8	Value	Value
29.95	Value	Value
59.9	Value	Value
399.6	Value	Value
29.9	Value	Value
59.8	Value	Value

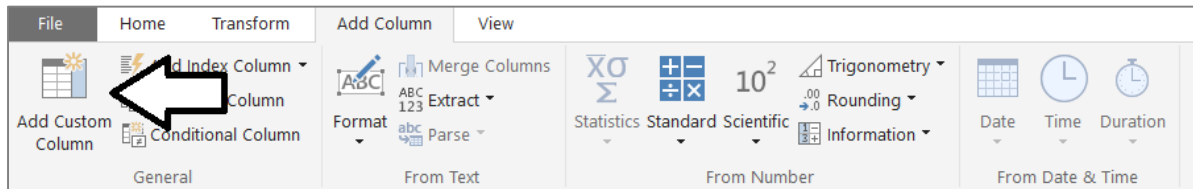
Id	InvoiceId	ProductId	Quantity	SalesAmount	Invoices	Products
1	1	1				Value
2	2	1				Value
3	3	2				Value
4	4	3				Value
5	5	3				Value
6	6	3				Value
7	7	4				Value
8	8	5				Value
9	9	6				Value
10	10	6				Value
11	11	7				Value
12	12	7				Value
13	13	8				Value
14	14	9				Value

Id	InvoiceId	ProductId	Quantity	SalesAmount	InvoiceDate	CustomerId	Products
1	1	1	22	4	119.8	1/28/2012 12:00:00 AM	1 Value
2	2	1	22	1	29.95	1/28/2012 12:00:00 AM	1 Value
3	3	2	22	2	59.9	1/28/2012 12:00:00 AM	2 Value
4	4	3	17	8	399.6	1/28/2012 12:00:00 AM	3 Value
5	5	3	18	2	29.9	1/28/2012 12:00:00 AM	3 Value
6	6	3	18	4	59.8	1/28/2012 12:00:00 AM	3 Value
7	7	4	16	1	2.95	1/28/2012 12:00:00 AM	4 Value



Adding a Custom Column

- Custom column provide custom logic
 - Logic must be written in M programming language



Add Custom Column

New column name:

Custom column formula:

```
= if [FirstPurchaseDate]=[LastPurchaseDate]  
then "One-time Customer"  
else "Repeat Customer"
```

Available columns:
CustomerId
Customer
State
City
ZipCode
Gender
BirthDate
FirstPurchaseDate
LastPurchaseDate
CustomerType

<< Insert

[Learn about Power BI Desktop formulas](#)

✓ No syntax errors have been detected.

OK Cancel

FirstPurchaseDate	LastPurchaseDate	CustomerType
1/28/2012	1/28/2012	One-time Customer
1/28/2012	1/28/2012	One-time Customer
1/28/2012	1/28/2012	One-time Customer
1/28/2012	1/28/2012	One-time Customer
1/28/2012	1/28/2012	One-time Customer
1/28/2012	1/28/2012	One-time Customer
1/29/2012	11/22/2015	Repeat Customer
1/29/2012	10/2/2015	Repeat Customer
1/29/2012	1/29/2012	One-time Customer
1/29/2012	5/6/2015	Repeat Customer
1/29/2012	1/29/2012	One-time Customer



Adding a Conditional Column

- Abstracts away need to write M code



Add Conditional Column

Add a conditional column that is computed from the other columns or values.

New column name

	Column Name	Operator	Value		Output
If	<input type="text" value="FirstPurchaseDate"/>	<input type="text" value="equals"/>	<input type="text" value="LastPurchaseDate"/>	Then	<input type="text" value="One-time Customer"/>

+ Add Rule

Otherwise

<input type="text" value="Repeat Customer"/>
--



DEMO

Using Queries to Transform Data During the Load Process

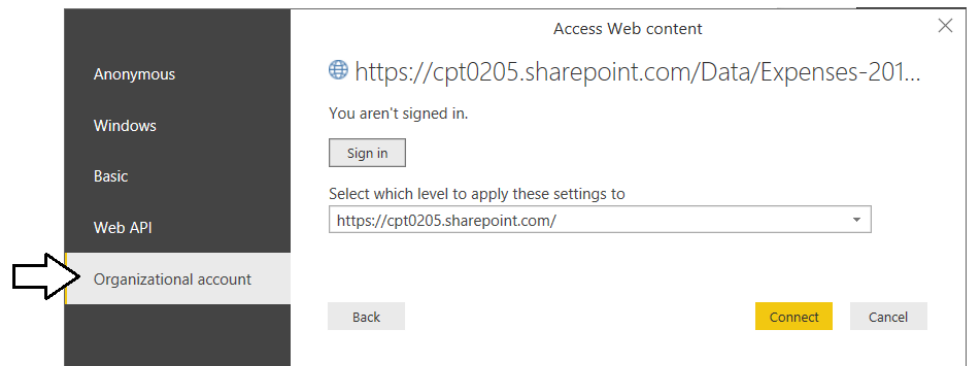
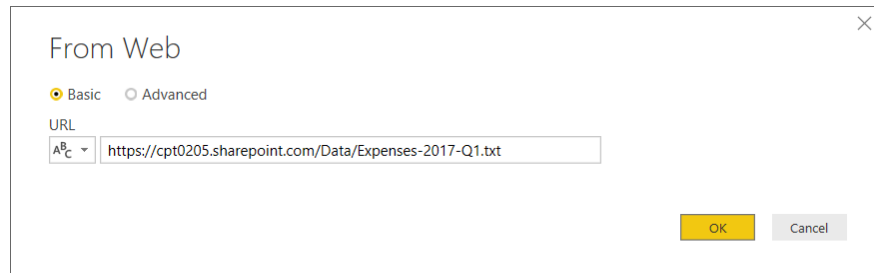
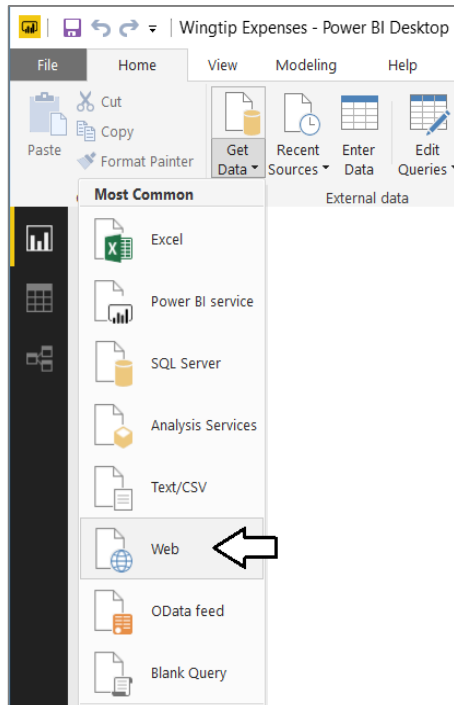
Agenda

- ✓ Query Design Fundamentals
- ✓ Designing Data Model using a Star Schema
- ✓ Working with the Query Editor Window
- Importing Content From SharePoint Online
 - Understanding Parameters and Template Files
 - Designing with Function Queries



Importing Files using the Web Datasource

- Files in SharePoint document library exposed via HTTPS
 - Use **Web** datasource to import files in SharePoint Online
 - Use the absolute path to file in document library
 - Authenticate using **Organizational account**



Importing using the SharePoint Folder

- Select the **SharePoint folder** datasource



- Query returns a row for each file in the site

The query results window displays the URL <https://cpt0205.sharepoint.com> and a table of files. The table has columns for Content, Name, Extension, Date accessed, Date modified, Date created, Attributes, and Folder Path. The data shows five files, all created on 2/3/2018 at 8:09 AM, with various extensions (.txt, .docx, .pptx).

Content	Name	Extension	Date accessed	Date modified	Date created	Attributes	Folder Path
Binary	Expenses-2017-Q2.txt	.txt	null	2/3/2018 8:09 AM	2/3/2018 8:09 AM	Record	https://cpt0205.sharepoint.com/Data/
Binary	Expenses-2017-Q1.txt	.txt	null	2/3/2018 8:09 AM	2/3/2018 8:09 AM	Record	https://cpt0205.sharepoint.com/Data/
Binary	Expenses-2017-Q3.txt	.txt	null	2/3/2018 8:09 AM	2/3/2018 8:09 AM	Record	https://cpt0205.sharepoint.com/Data/
Binary	LibertyPowerBISetup.docx	.docx	null	2/3/2018 7:54 AM	2/3/2018 7:54 AM	Record	https://cpt0205.sharepoint.com/Shared Documents/
Binary	RealtimeDashboards.pptx	.pptx	null	2/3/2018 7:54 AM	2/3/2018 7:54 AM	Record	https://cpt0205.sharepoint.com/Shared Documents/

Buttons at the bottom: Combine & Edit, Edit, Cancel.





DEMO

Importing Content from Files in SharePoint Online

Agenda

- ✓ Query Design Fundamentals
- ✓ Designing Data Model using a Star Schema
- ✓ Working with the Query Editor Window
- ✓ Importing Content From SharePoint Online
- Understanding Parameters and Template Files
- Designing with Function Queries



Query Parameters

- What is a Query Parameter?
 - Configurable setting with project scope
 - Strongly-typed value to which you can apply restrictions
 - Can be referenced from a query
 - Can be referenced from DAX code in data model
- Where are Parameters commonly used
 - To parameterize data source connection details
 - To filter rows when importing data



Creating Query Parameters

- Parameters can be created using **Manager Parameters** menu



- Parameter properties

- Name
- Description
- Required
- Allowed Values
- Default Value
- Current Value

Parameters

New

Customer State

Name

Customer State

Description

This parameter is used in the Customers query to filter the customer rows which are loaded into the dataset for the Power BI Desktop project.

☒ Required

Type

Text

Allowed Values

List of values

1	CA
2	OR
3	WA
4	AZ
5	TX
*	

Default Value

CA

Current Value

CA

OK Cancel

Referencing Parameters in a Query

- Parameters can be referenced inside query
 - Next query execution uses current parameter value

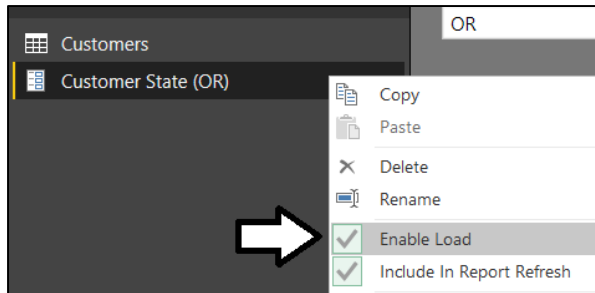
The screenshot shows a 'Filter Rows' dialog box with a close button (X) in the top right corner. It has two tabs: 'Basic' (unselected) and 'Advanced' (selected). Below the tabs, it says 'Show rows where:'. There are two columns: 'And/Or' and 'Column'. The 'And/Or' column has a dropdown menu with 'And' selected. The 'Column' column has a dropdown menu with 'State' selected. The 'Operator' column has a dropdown menu with 'equals' selected. The 'Value' column has a dropdown menu with 'Customer State' selected. There is a '...' button to the right of the 'Value' dropdown. Below the filter rules, there is an 'Add Clause' button. At the bottom right, there are 'OK' and 'Cancel' buttons.

And/Or	Column	Operator	Value
	State	equals	Customer State
And	State		ABC

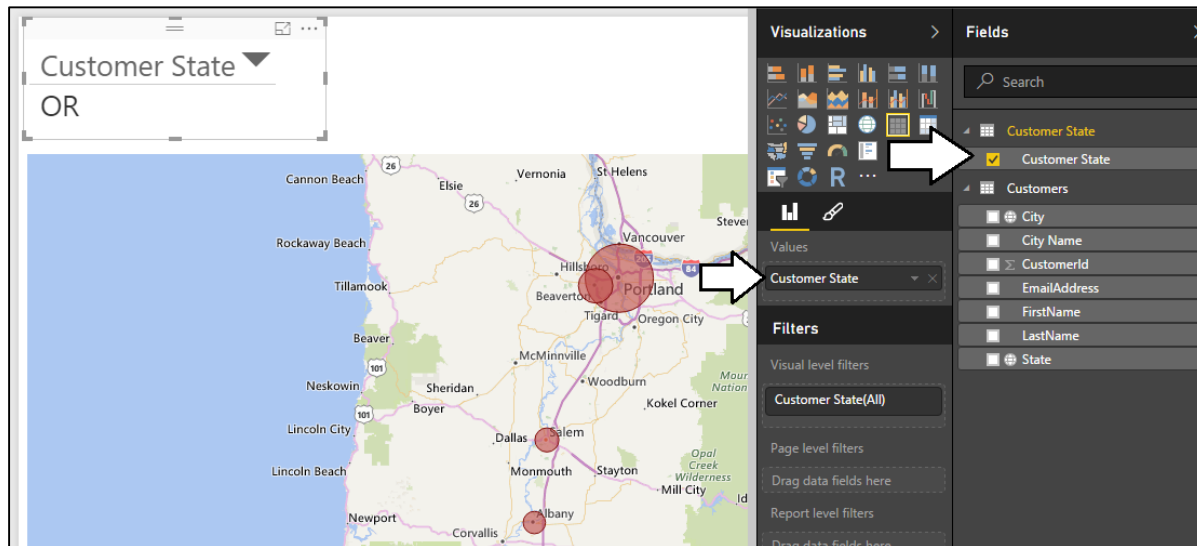


Making Parameters Available to Data Model

- Configure parameter's Enable Load setting

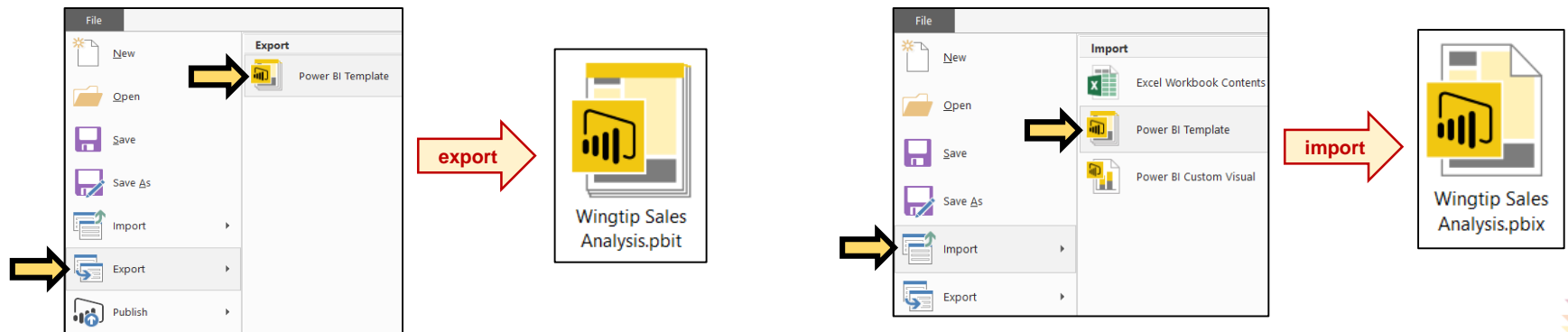


- Parameter becomes visible within fields list in report view



Power BI Project Template Files

- PBIX project can be exported to project template file
 - Template file created with PBIT file extension
 - Generated template files contains everything except for the data
 - PBIT template file can be imported to create new PBIX projects
 - Template files are powerful when used together with parameters
- How are template files used?
 - Export PBIX project to create a PBIT template file
 - Import the PBIT template file to create a new PBIX project



Agenda

- ✓ Query Design Fundamentals
- ✓ Designing Data Model using a Star Schema
- ✓ Working with the Query Editor Window
- ✓ Importing Content From SharePoint Online
- ✓ Understanding Parameters and Template Files
- Designing with Function Queries



Understanding Function Queries

- Query can be converted into reusable function
 - Requires editing query M code in Advanced Editor
 - Function query can be defined to accept parameters

```
GetExpensesFromFile

(FilePath as text) =>

let
    Source = Csv.Document(Web.Contents(FilePath))
    #"Changed Type" = Table.TransformColumnTypes
```

- Function query can't be edited with visual designer





DEMO

Creating a Function Query

Summary

- ✓ Query Design Fundamentals
- ✓ Designing Data Model using a Star Schema
- ✓ Working with the Query Editor Window
- ✓ Importing Content From SharePoint Online
- ✓ Understanding Parameters and Template Files
- ✓ Designing with Function Queries

