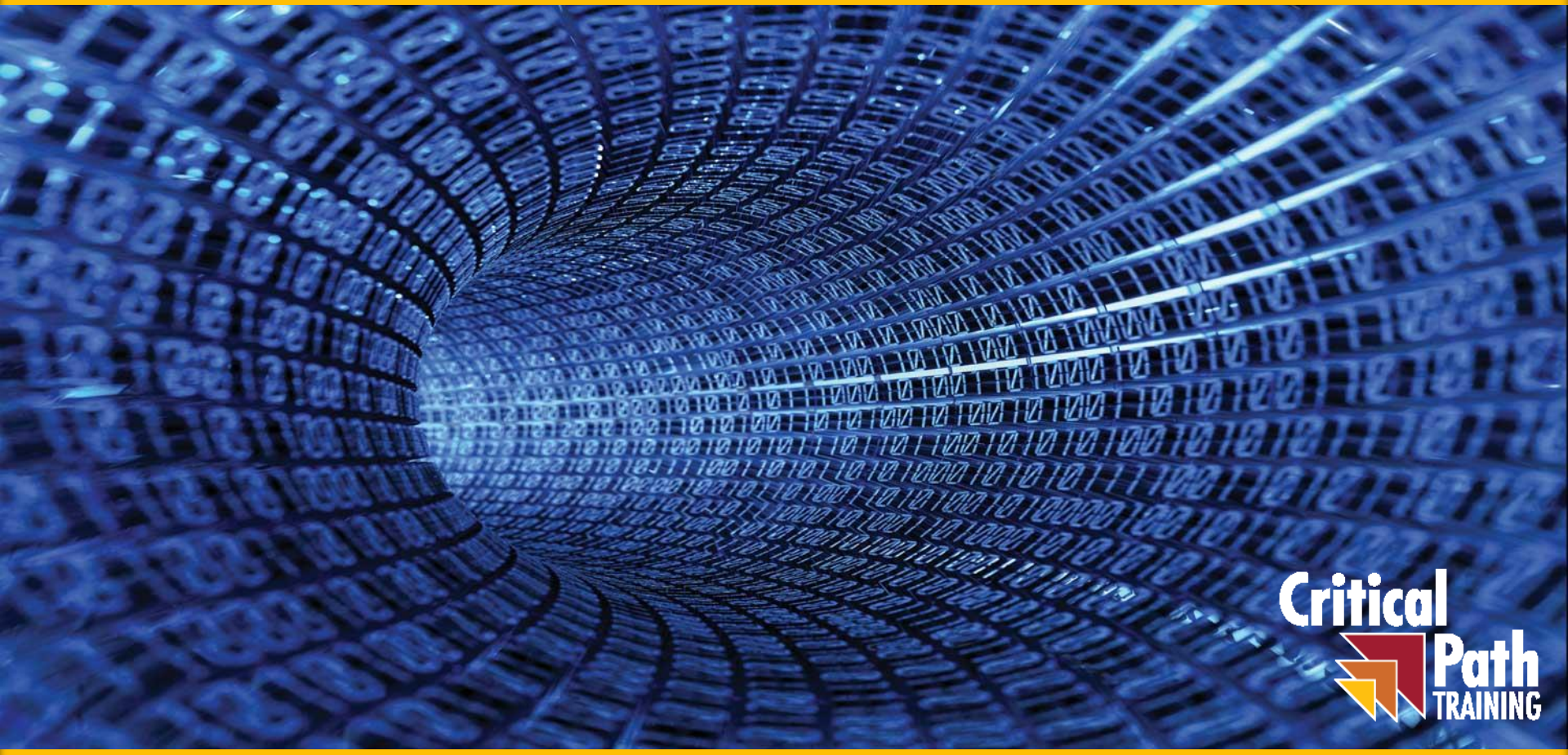
Designing Queries with Power BI Desktop



Agenda

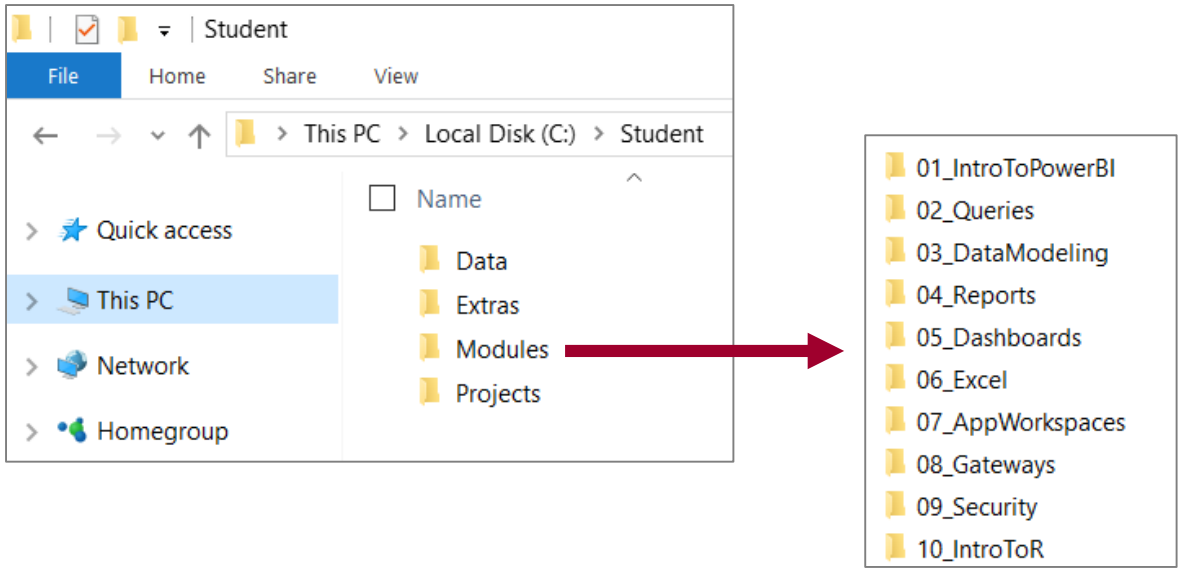
- Class Introduction
- Query Design Fundamentals
- SharePoint Online
- Query Folding
- Designing Data Model using a Star Schema
- Advanced Query Design

Student Background

- What is your name?
- What are you doing with Power BI?
- Which products/services have you used
  - Excel
  - Access
  - SQL Server, SSRS, SSAS
  - SharePoint and Office 365
  - Tableau
  - Dynamics 365
  - Salesforce
  - Others

Student Files for This Course

- Copy the **Student** folder from the master zip archive
  - Create a new local folder at **C:\Student**
- Each module has folder inside **Student\Modules** folder
  - Slides and lab writeup available through student manual (not in GitHub repository)

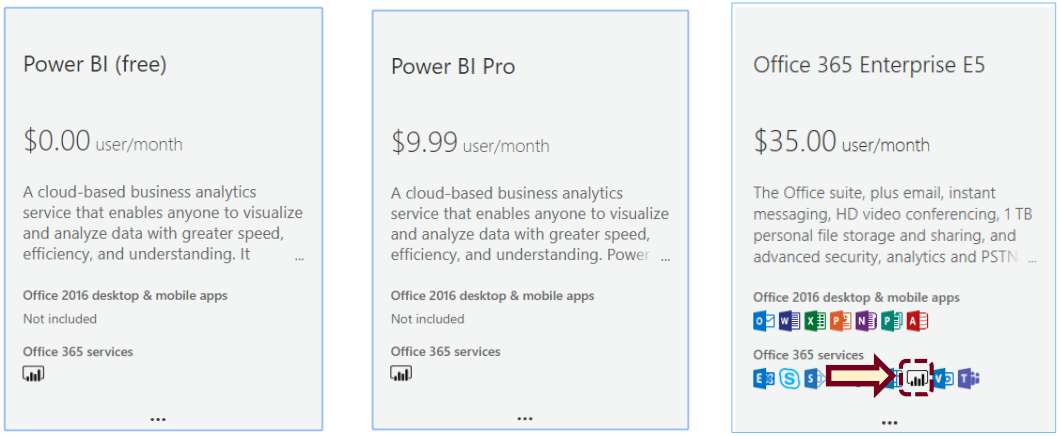


What is Power BI?

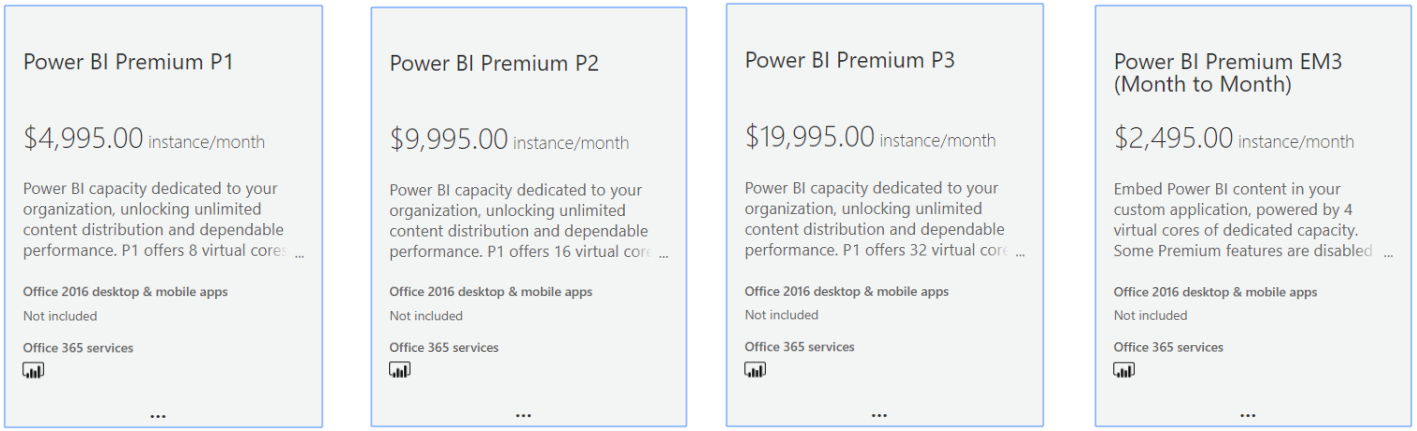
- What is Power BI?
  - A cloud-based analytics service for licensed subscribers
  - Environment which supports and promotes self-service BI
  - Powerful builder tools for importing, modeling and visualizing data
  - Enterprise-grade platform for deploying reports and dashboards
  - On-premises server product supporting subset of cloud features

Power BI Licensing

- User-based licensing

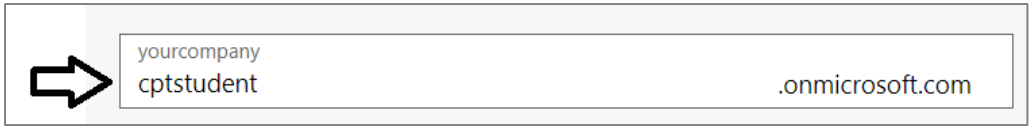


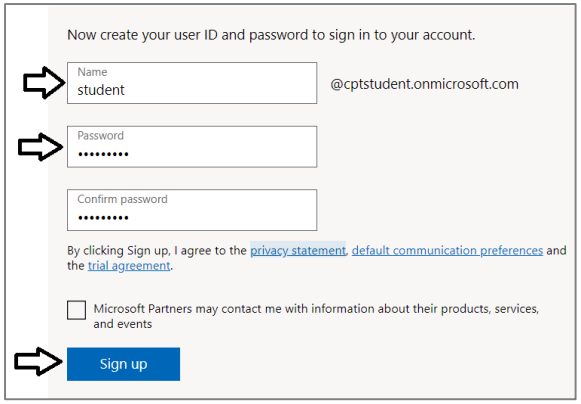
- Capacity-based Licensing

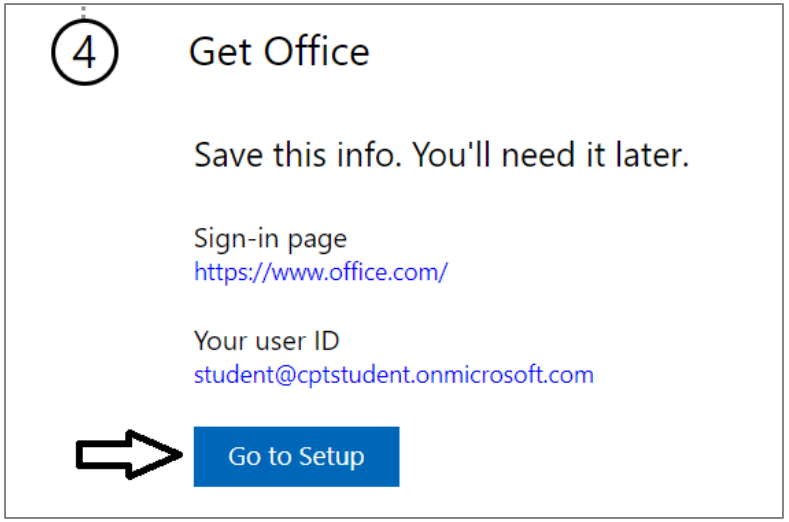


Creating a Power BI Lab Environment

- Sign up for an Office 365 E5 trial account
  - Sign up process creates new Azure AD tenant
  - Tenant created with user account which is Global tenant admin
  - Tenant gets 30-day trial subscription for 25 Office 365 E5 licenses

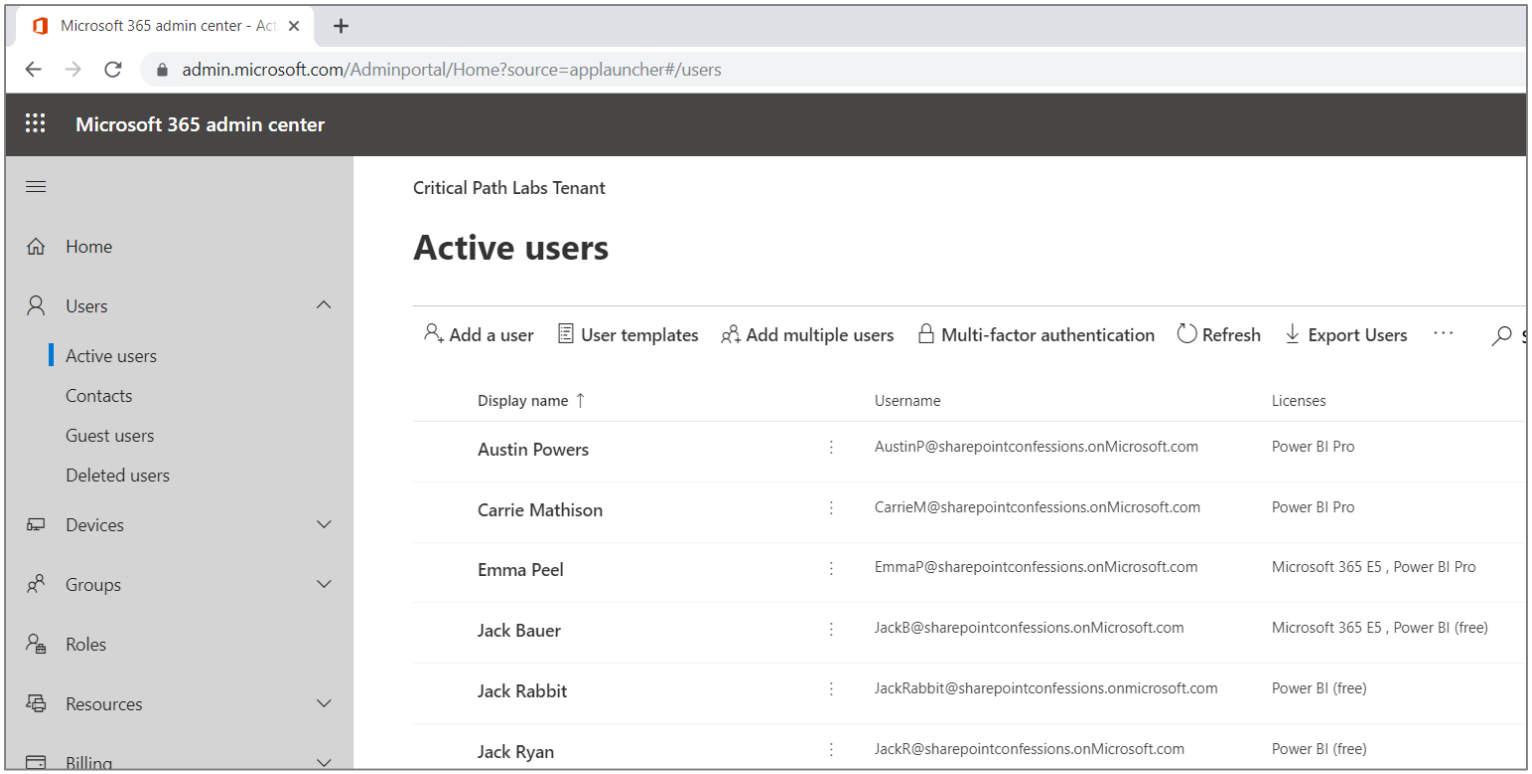






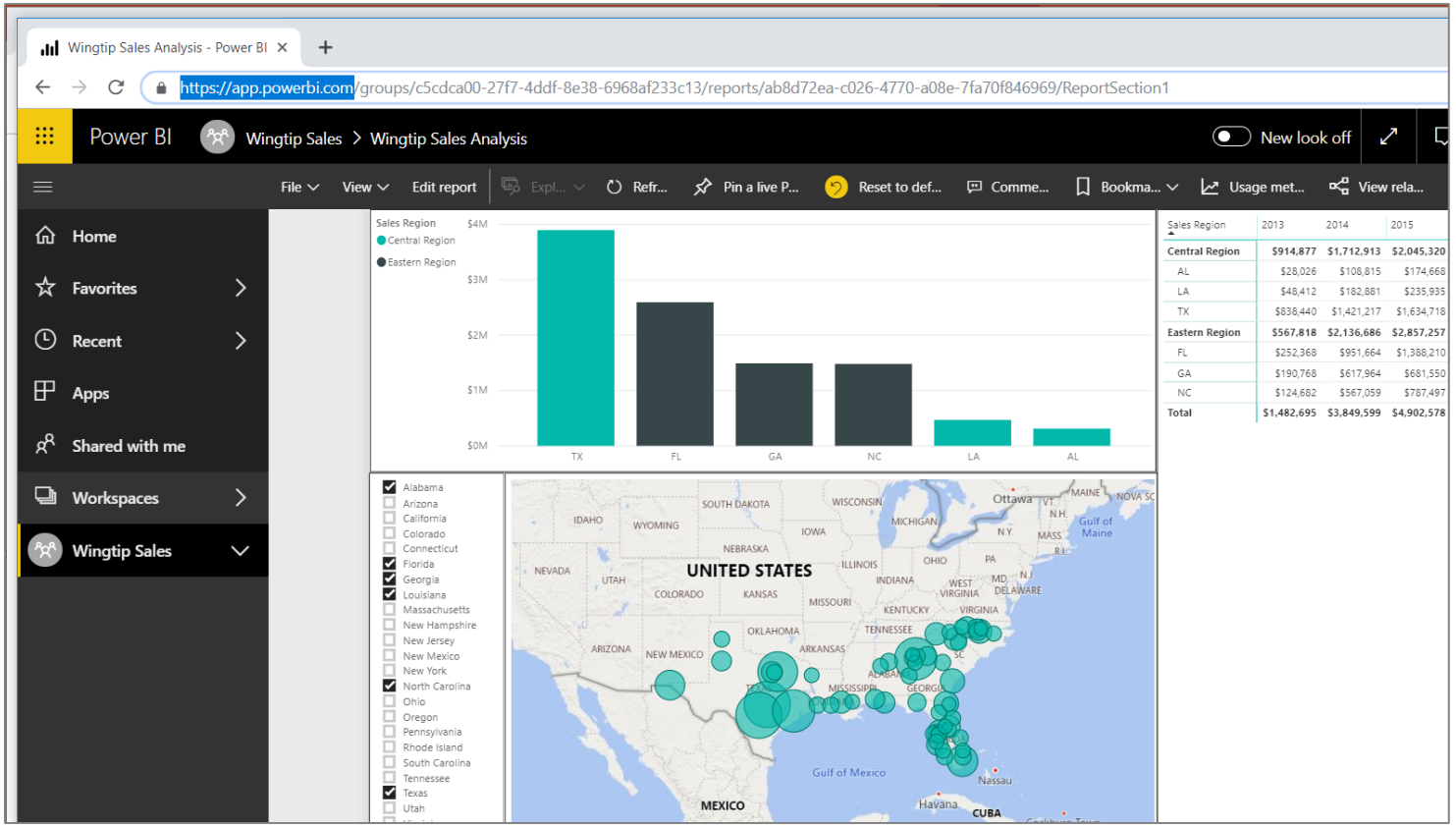
Microsoft 365 admin center

- Microsoft 365 admin center used for tenant administration
  - You will have full control because you are a Global tenant admin
  - You can create user accounts and groups
  - You can create subscriptions and assign licenses

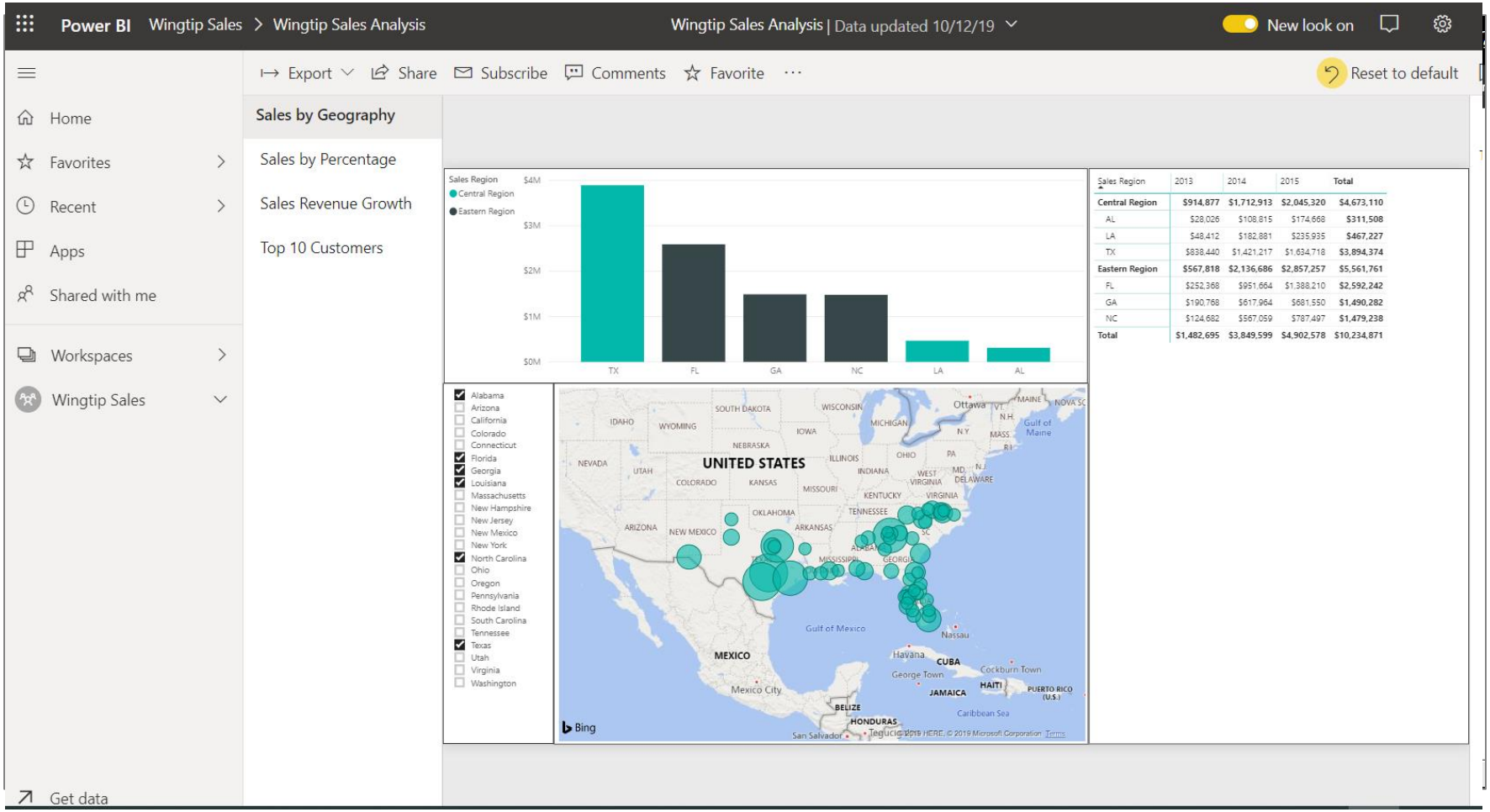


The Power BI Service

- The Power BI Service
  - Provides cloud-based foundation for Power BI platform
  - Provides browser-based portal at https://app.powerbi.com

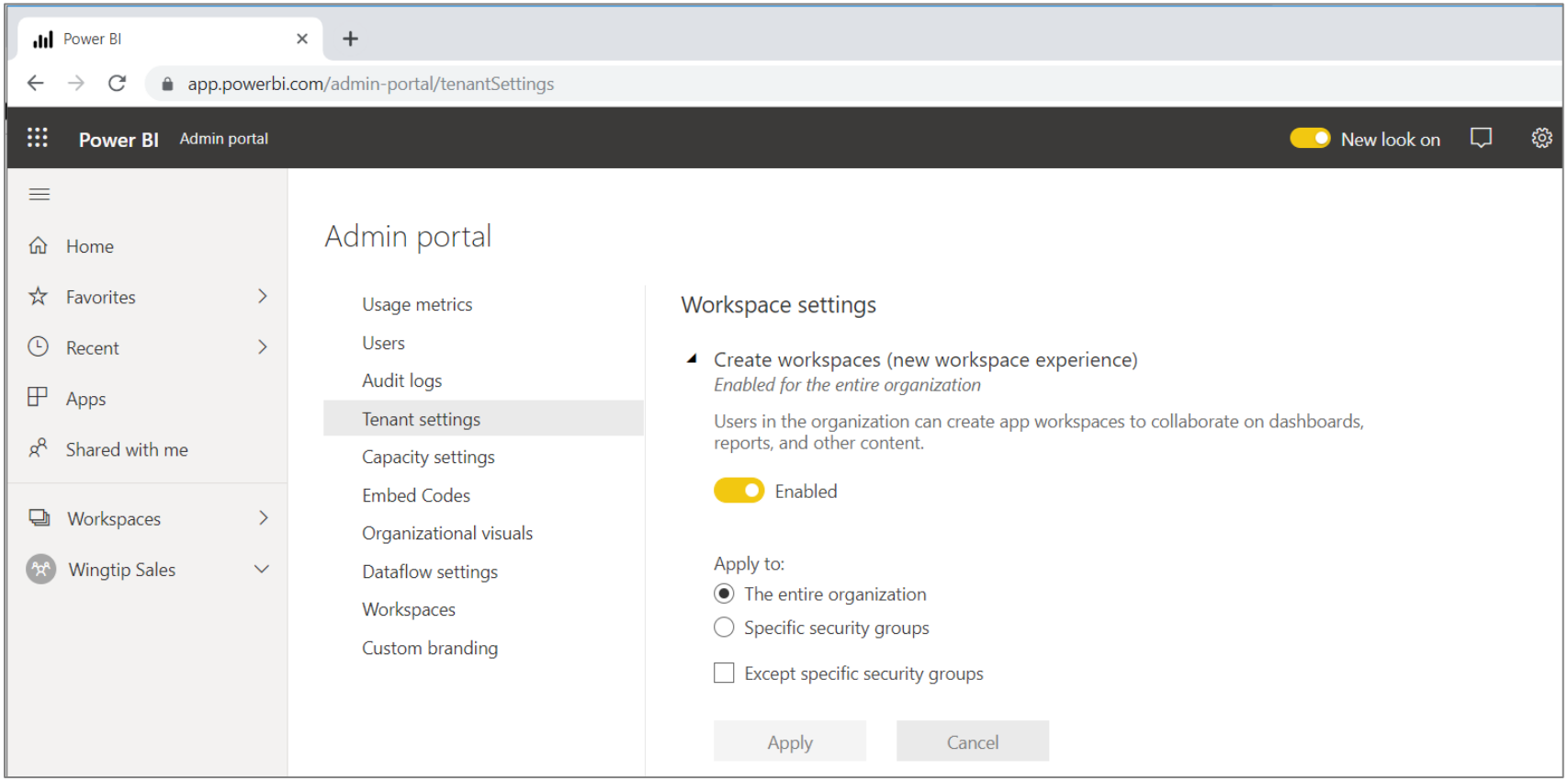


Light Grey is the New Black



The Power BI Admin Portal

- Admin portal used to configure tenant-level settings
  - Accessible at https://app.powerbi.com/admin-portal
  - Accessible by global tenant admins and Power BI Service admins



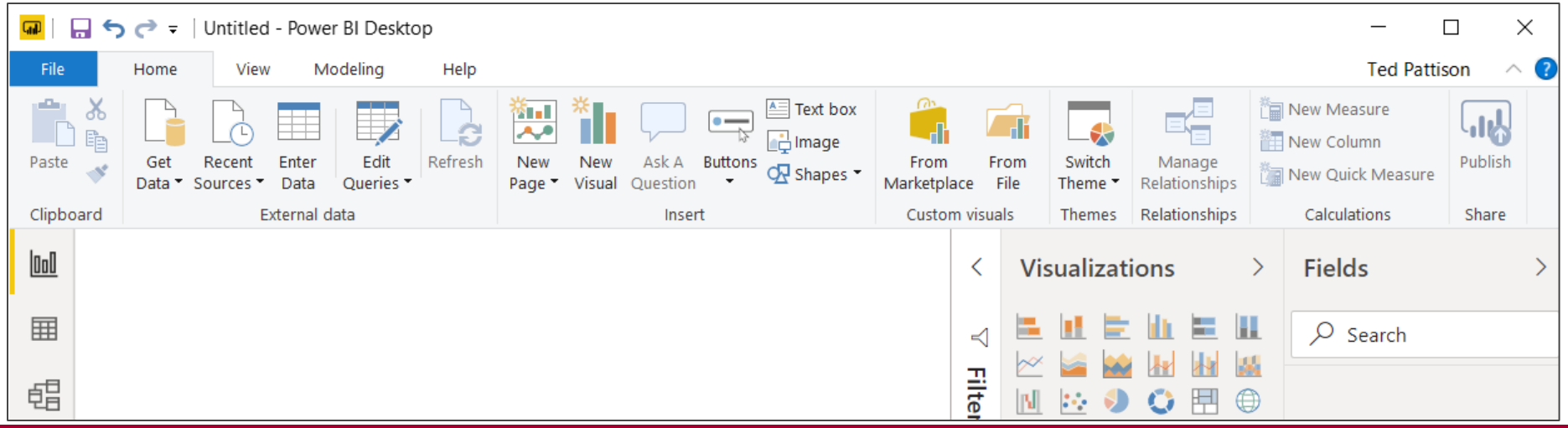
Central Power BI Concepts

- Workspace
  - Secured container for Power BI resources
  - Created as personal workspaces and app workspaces
- Dashboard
  - Consolidated high-level view into reports and datasets
  - Provides great experience on mobile device (*e.g. iPhone, Android, tablet, etc.*)
- Report
  - Collection of one or more pages with tables & visualizations
  - Provides consumer with interactive control through filtering and bookmarks
- Dataset
  - In-memory data model containing one or more tables
  - Used to supply the underlying data to reports and dashboards
- Dataflows
  - Persistent data store used for more complex ETL requirements
  - Not required in most Power BI scenarios


DEMO
Creating Azure AD User Accounts in an Office 365 Tenant

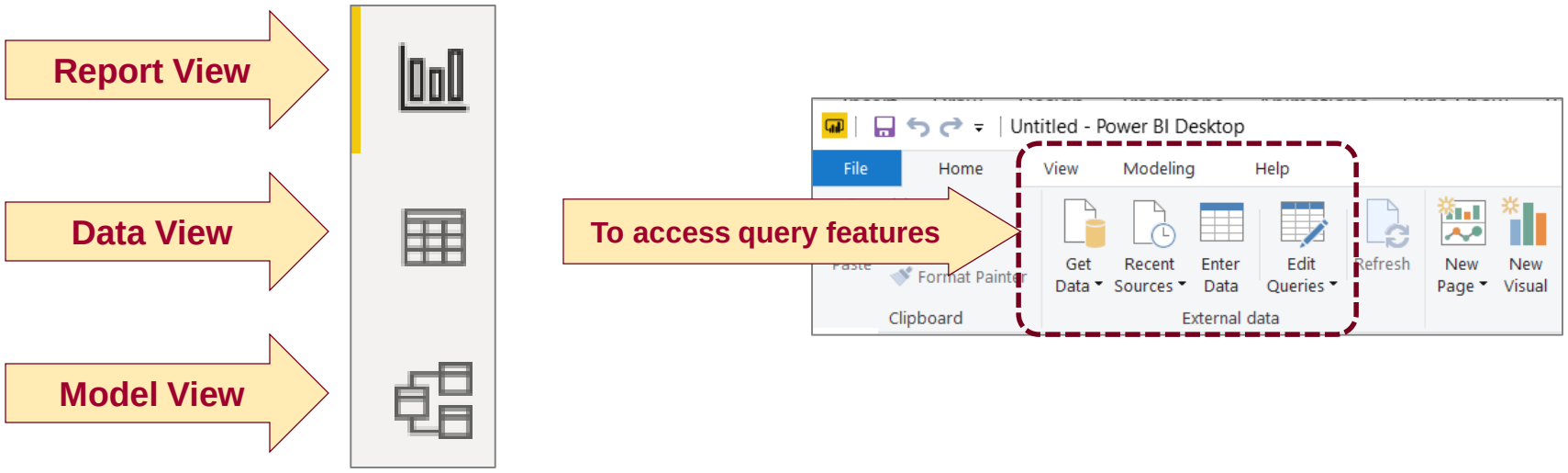
Working with Power BI Desktop

- Power BI Desktop is a Windows application
  - Work is saved and published in terms of PBIX projects
  - You can work on multiple projects at once
  - Each project runs in its own Power BI Desktop instance
  - Power BI Desktop can freeze up or act buggy
  - Quit & restart Power BI Desktop if it acts strangely



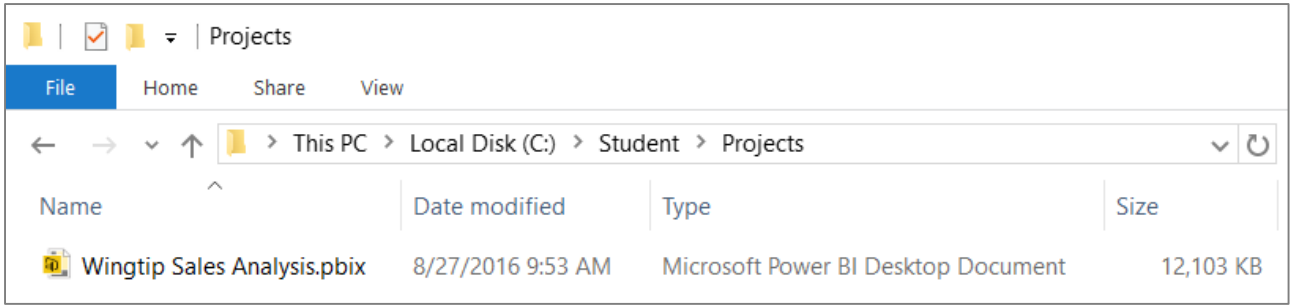
Getting Around in Power BI Desktop

- What do you need to learn to use Power BI Desktop?
  - Query features for importing data
  - Designing data model & writing DAX expressions
  - Designing reports with Power BI Desktop report designer
- Navigating between view modes



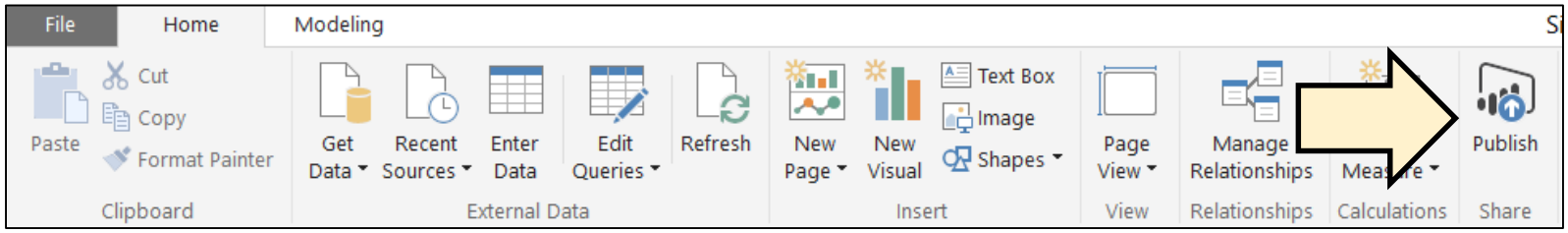
Projects and PBIX Files

- Power BI Desktop projects saved using PBIX files
  - PBIX file contains data source definitions
  - PBIX file contains query definitions
  - PBIX file contains data imported from queries
  - PBIX file contains exactly one data model definition
  - PBIX file contains exactly one report
  - PBIX file never contains data source credentials

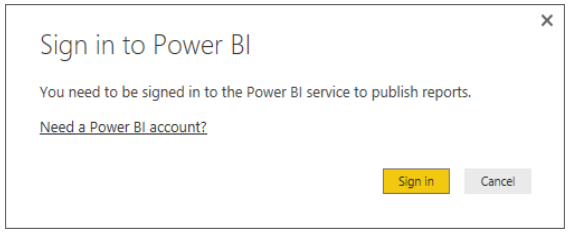


Publishing a Power BI Desktop Project

- Power BI Desktop provides **Publish** command
  - Used to publish project to Power BI service



  - Requires logging into your Office 365 account



  - Published articles added to target workspace

Agenda

- ✓ Class Introduction
- ➢ Query Design Fundamentals
- SharePoint Online
- Query Folding
- Designing Data Model using a Star Schema
- Advanced Query Design

Data Discovery

- Data can live in a variety of sources
  - Files (e.g. CSV file, Excel workbook)
  - OLTP Databases
  - OLAP Databases
  - SharePoint Lists and Document Libraries
  - Azure-based services
  - Online services & SaaS applications
- You Must Determine Measurable Objectives
  - Financial (revenue, expenses, profit margin, etc.)
  - Business processes efficiency
  - Customer Satisfaction Levels

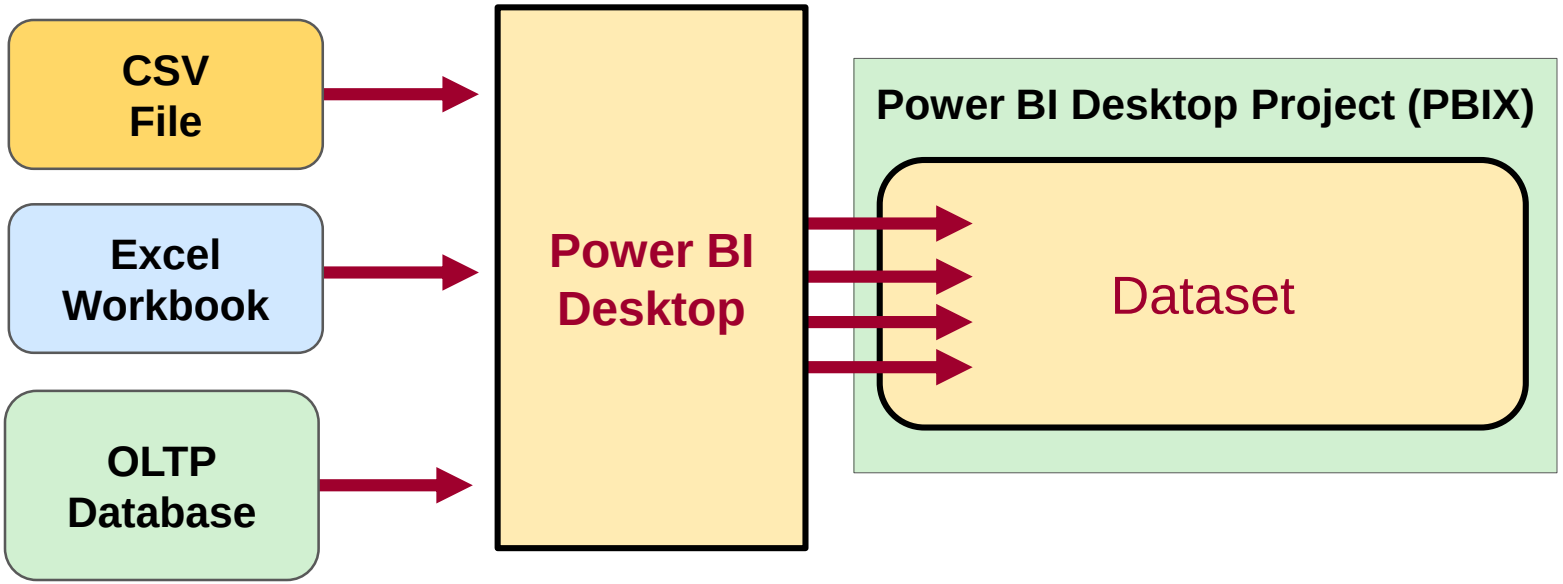
Defining Grain Statements

- Grain statements should be defined in initial design phase
  - Grain statements helps determine requirements for BI queries
  - Grain statements can be created & understood by business users

- Example grain statements for BI project at Wingtip Toys
  - What was the total sales revenue over the last 4 years?
  - What was the sales revenue by year, quarter and month?
  - What was the sales revenue by region, state, city and zip code?
  - What was the sales revenue by category, subcategory and product?
  - What was the growth in sales revenue from month to month in 2013?
  - What was profit margin for each product by year, quarter and month?
  - Have their been any products with significantly decreasing profit margin?

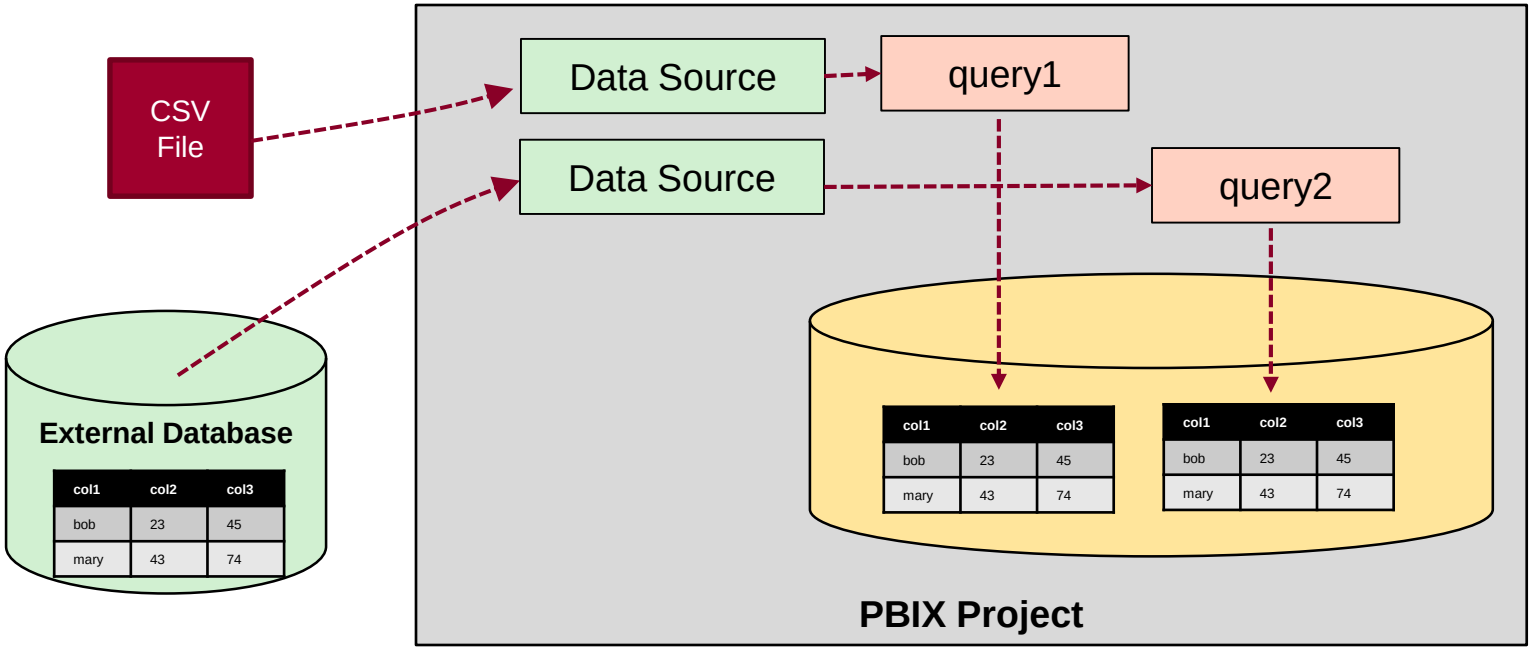
Power BI Desktop is an ETL Tool

- 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



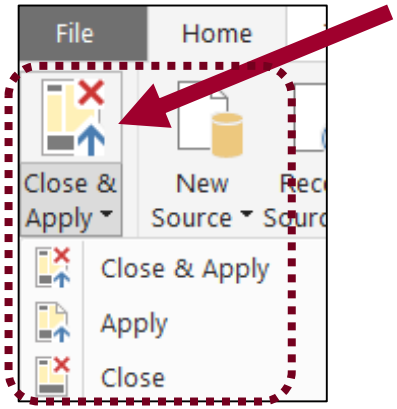
Understanding Query Input and Output

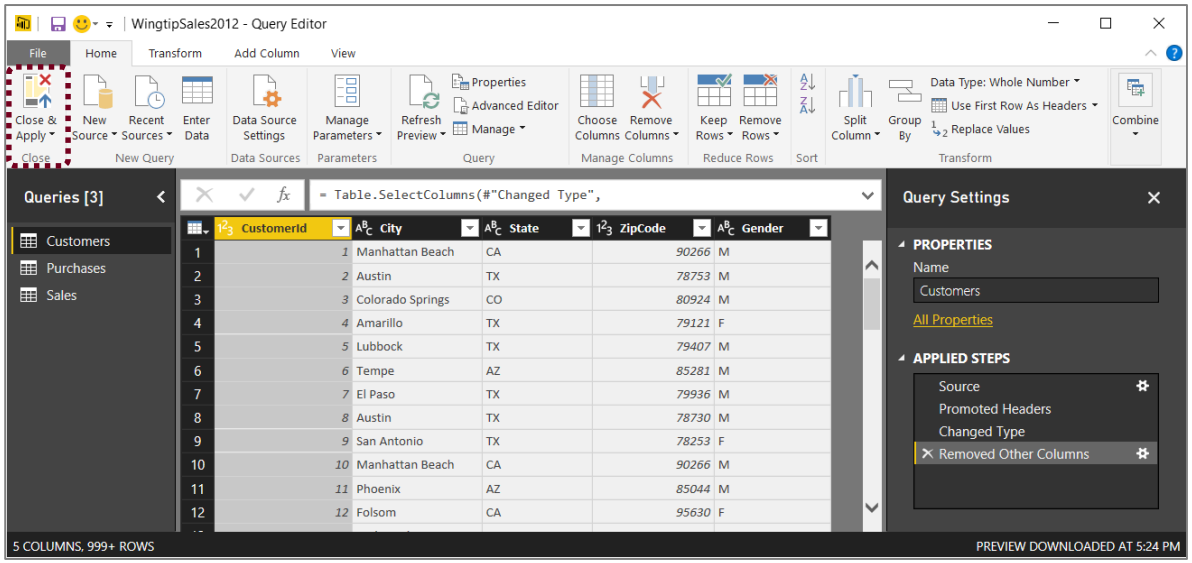
- PBIX project is container for data sources and queries
  - Queries created and saved within scope of Power BI project
  - Queries can pull data from local files
  - Queries can pull data from external content sources
  - Queries main purpose is to load imported data into data model



Query Editor Window

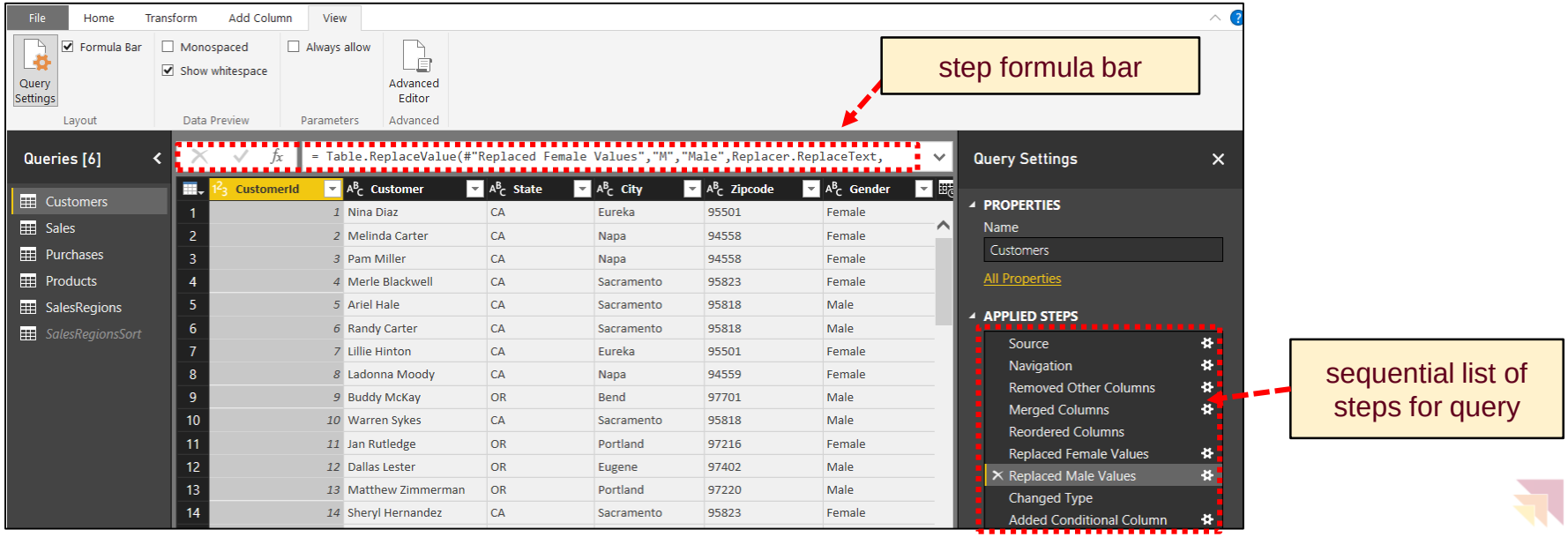
- Power BI Desktop provides separate Query Editor window
  - Provides powerful features for designing queries
  - Displays list of all queries in project on the left
  - Displays **Properties** and **Applied Steps** for selected query on right
  - Preview of table generated by query output shown in the middle
  - Query can be executed using **Apply** or **Close & Apply** command





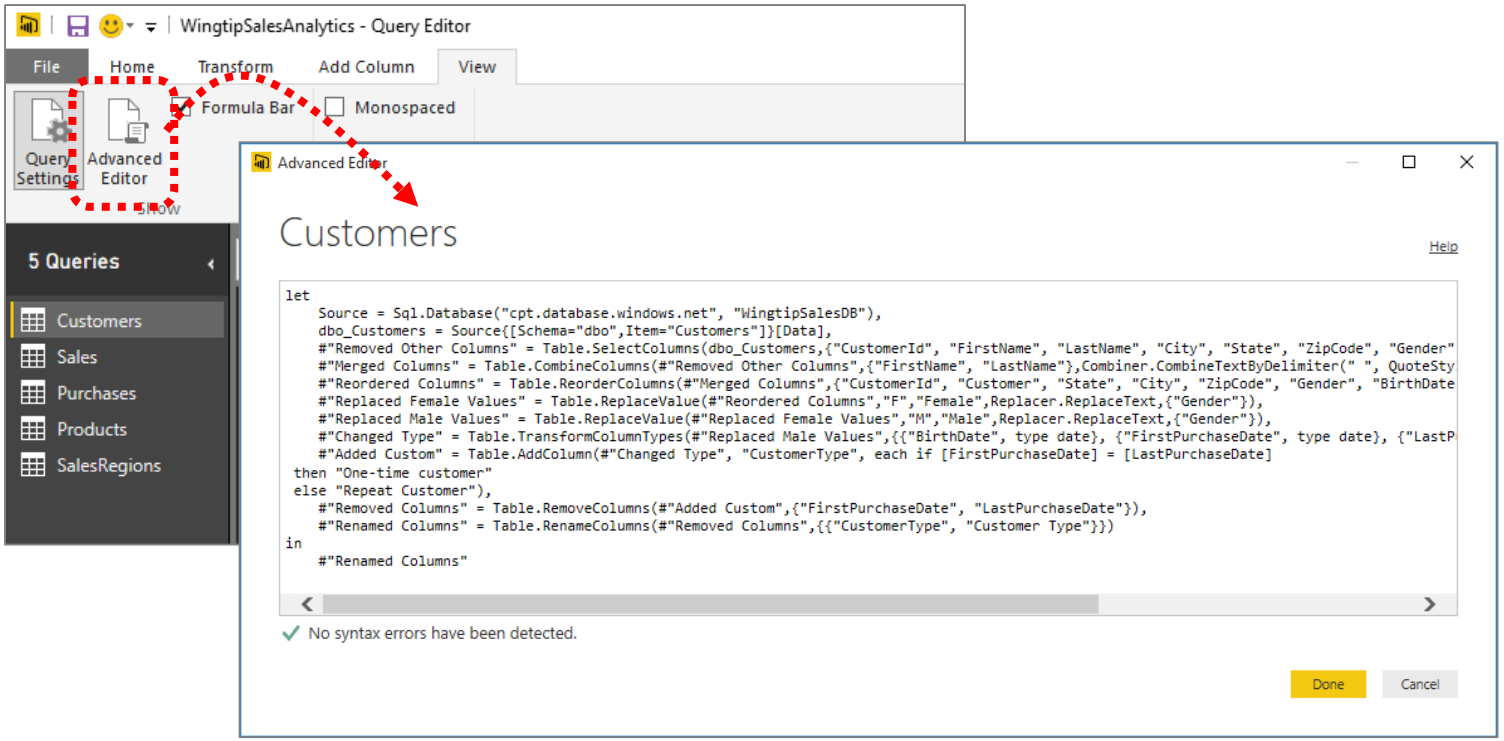
Query Steps

- 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



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

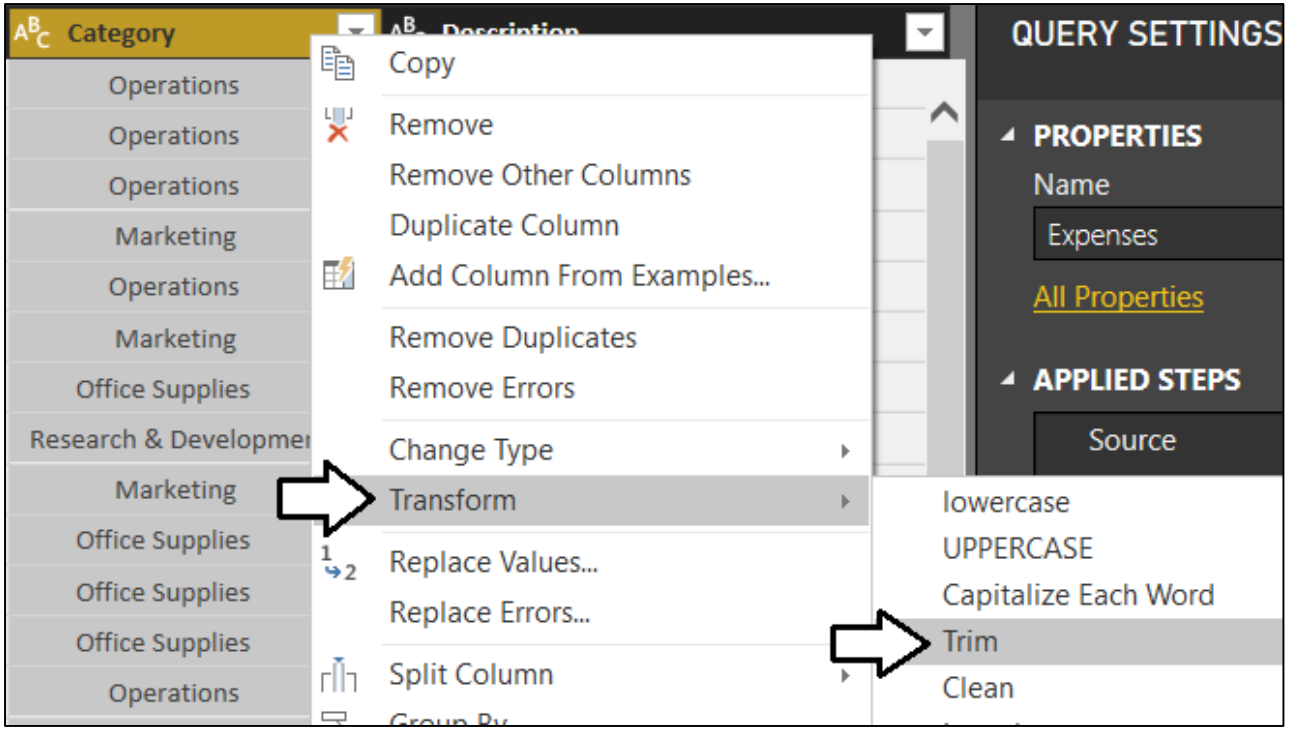


Examples of Basic Power BI Desktop Steps

- Rename column
- Convert column type
- Trim and clean column values
- Replace column values
- Format column values
- Expanding related column
- Merging columns
- Splitting columns

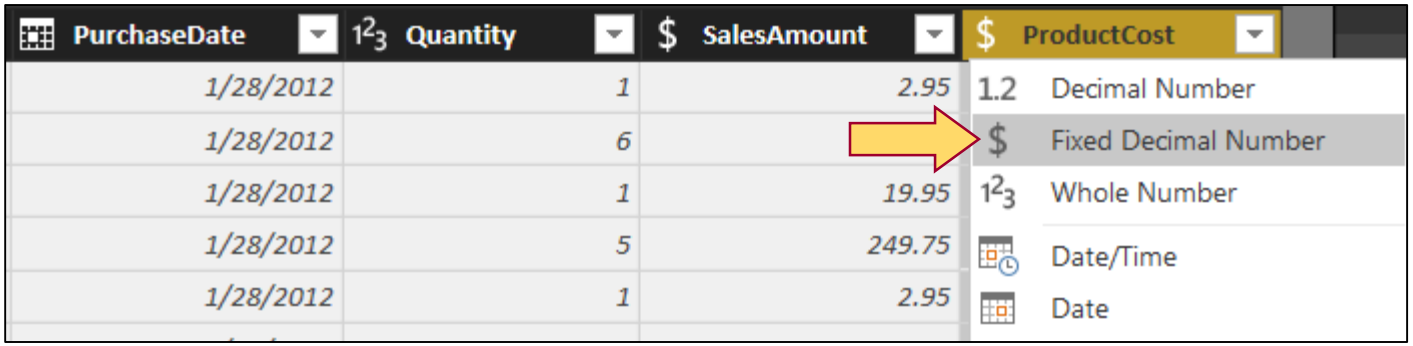
Cleaning Data

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



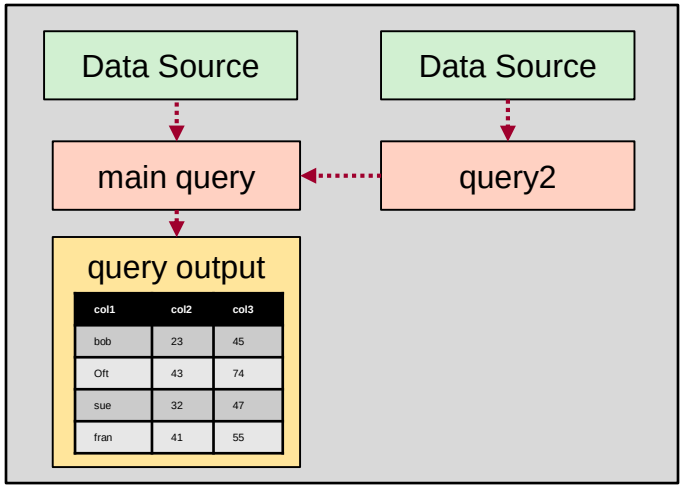
Converting Column Types

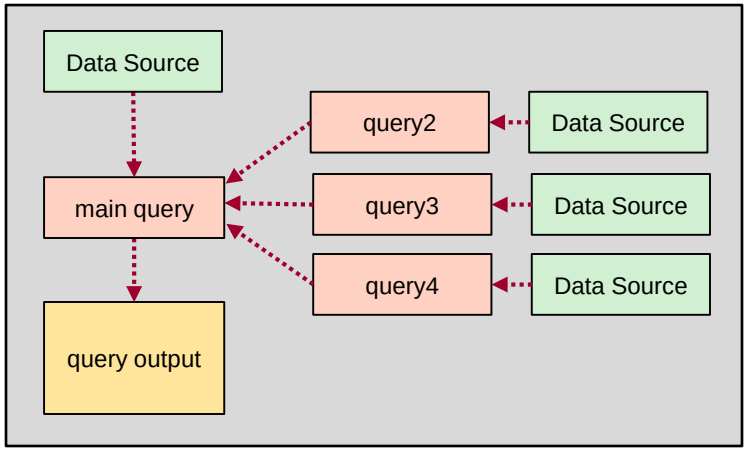
- Transform data to make it more reliable
  - Convert date-time column to date column
- Transform data to make it more efficient
  - Convert decimal to fixed decimal number for currency



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 be can created with connection-only load setting

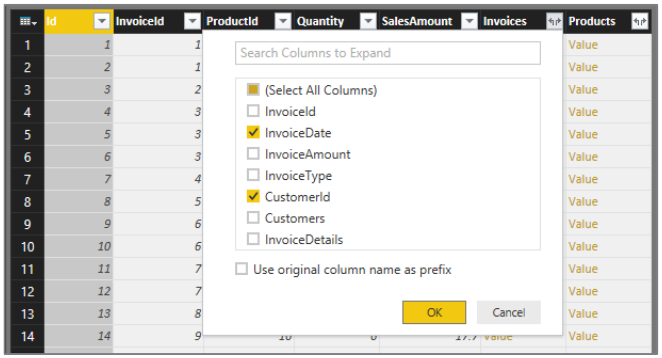




Expanding Related Columns

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

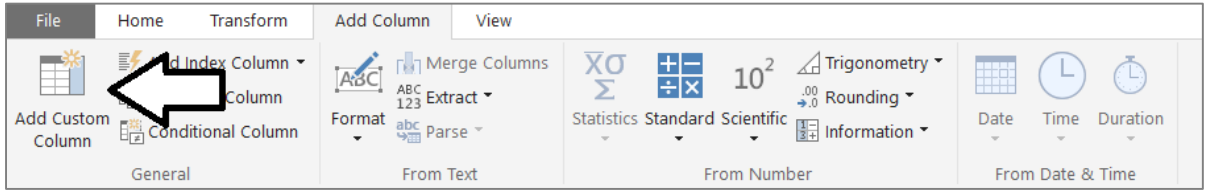
SalesAmount	Invoices	
119.8	Value	Value
29.95	Value	Value
59.9	Value	Value
399.6	Value	Value
29.9	Value	Value
59.8	Value	Value

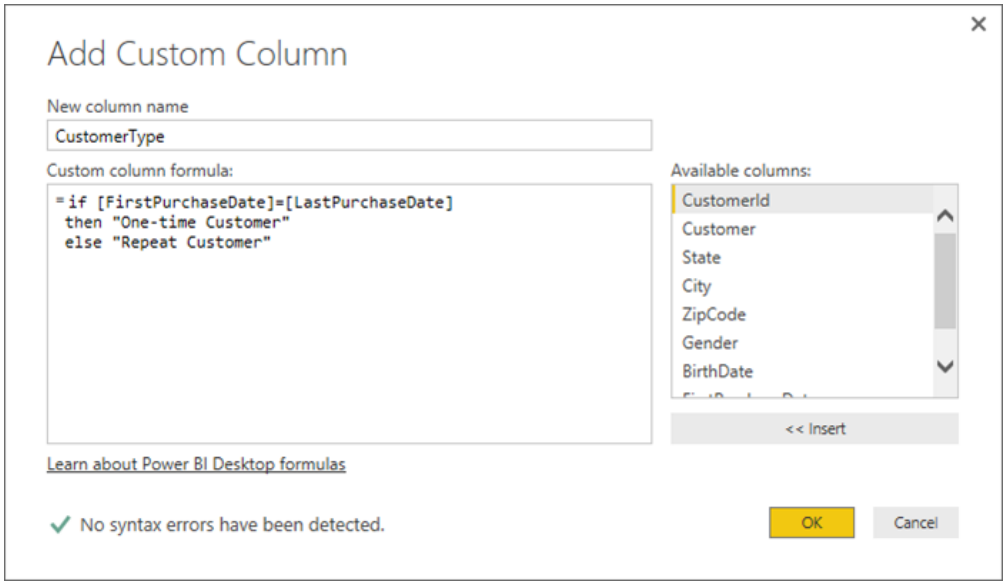


	Id	InvoiceId	ProductId	Quantity	SalesAmount	InvoiceDate	CustomerId	Products
1	1	1	22	4	119.8	1/28/2012 12:00:00 AM	1	Value
2	2	1	22	1	29.95	1/28/2012 12:00:00 AM	1	Value
3	3	2	22	2	59.9	1/28/2012 12:00:00 AM	2	Value
4	4	3	17	8	399.6	1/28/2012 12:00:00 AM	3	Value
5	5	3	18	2	29.9	1/28/2012 12:00:00 AM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

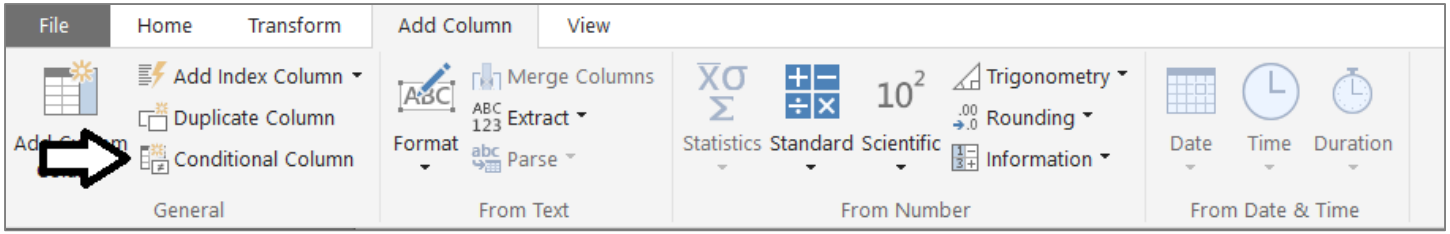


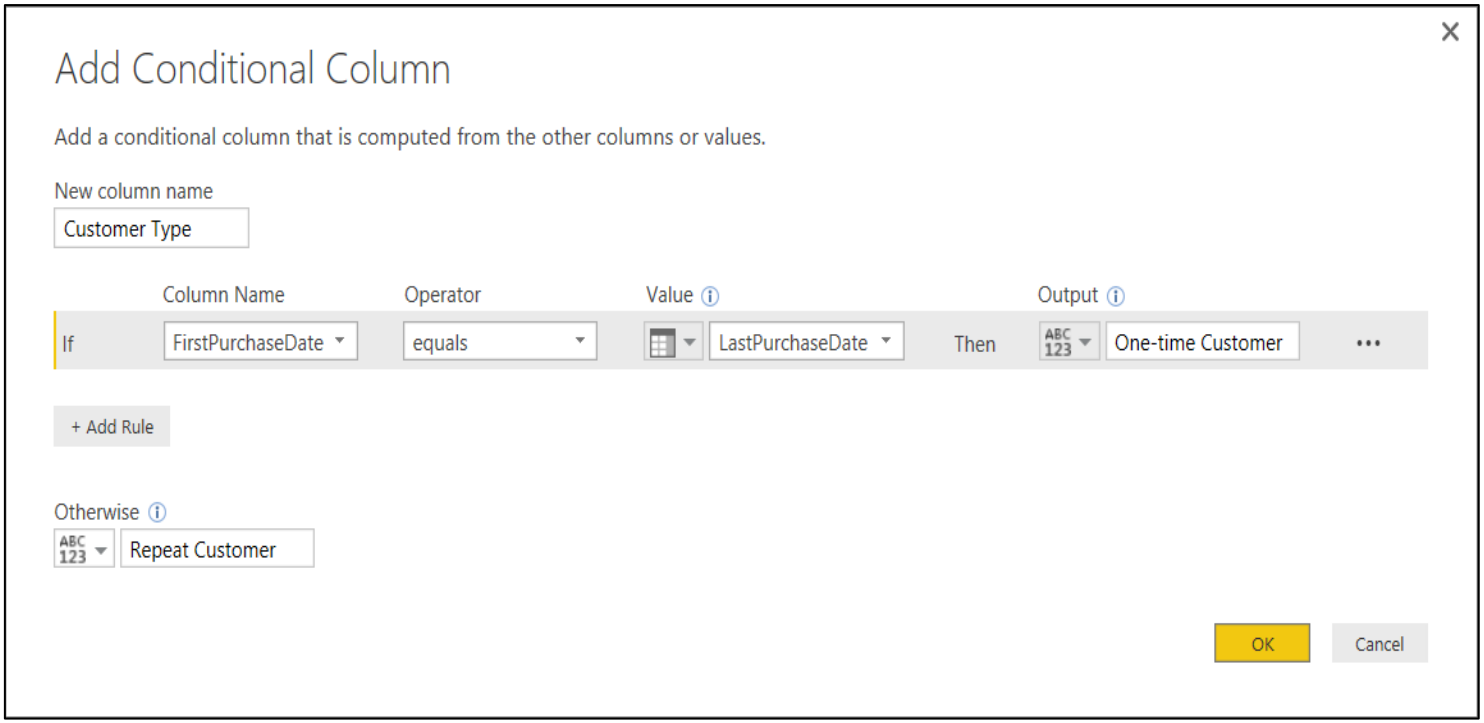


FirstPurchaseDate	LastPurchaseDate	CustomerType
1/28/2012	1/28/2012	One-time Customer
1/28/2012	1/28/2012	One-time Customer
1/28/2012	1/28/2012	One-time Customer
1/28/2012	1/28/2012	One-time Customer
1/28/2012	1/28/2012	One-time Customer
1/28/2012	1/28/2012	One-time Customer
1/29/2012	11/22/2015	Repeat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





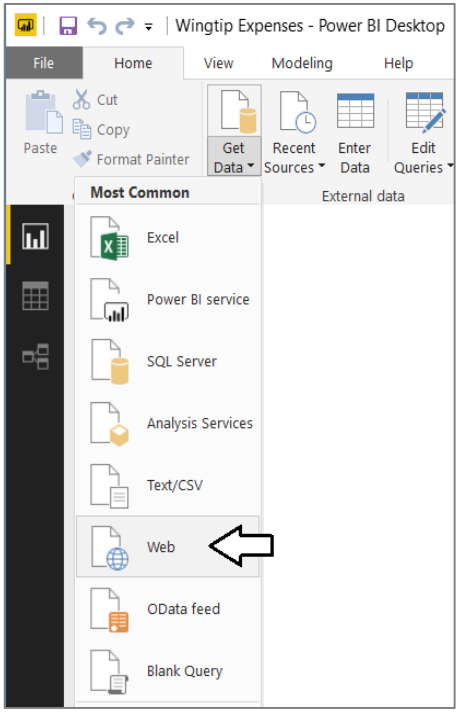

DEMO
Using Queries to Transform Data During the Load Process

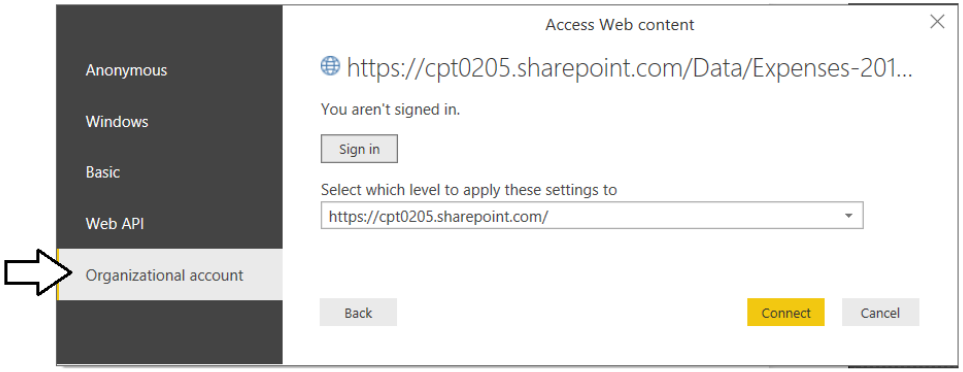
Agenda

- ✓ Class Introduction
- ✓ Query Design Fundamentals
- ➢ SharePoint Online
- Query Folding
- Designing Data Model using a Star Schema
- Advanced Query Design

Importing Files using the Web Datasource

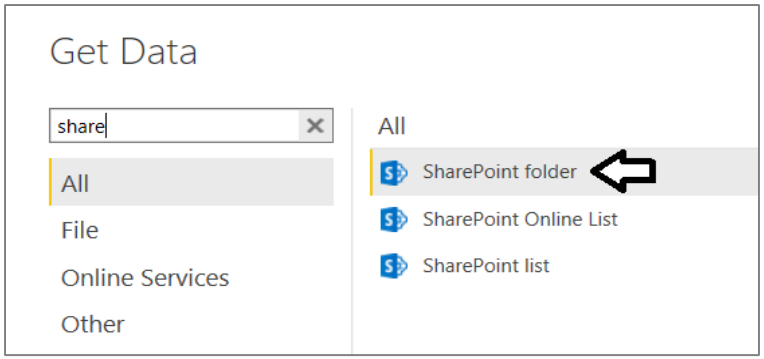
- Files in SharePoint document library exposed via HTTPS
  - Use **Web** datasource to import files in SharePoint Online
  - Use the absolute path to file in document library
  - Authenticate using **Organizational account**



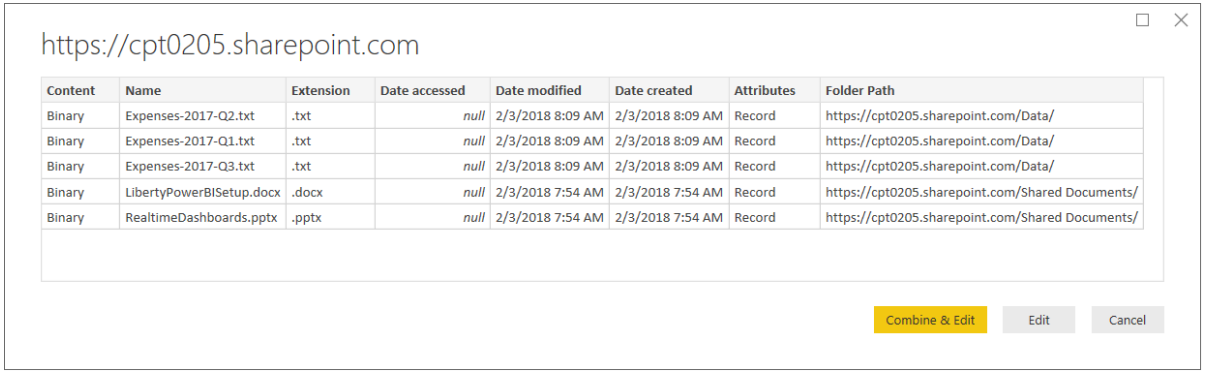


Importing using the SharePoint Folder

- Select the **SharePoint folder** datasource



- Query returns a row for each file in the site



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/


DEMO
Importing Content from Files in SharePoint Online

Agenda

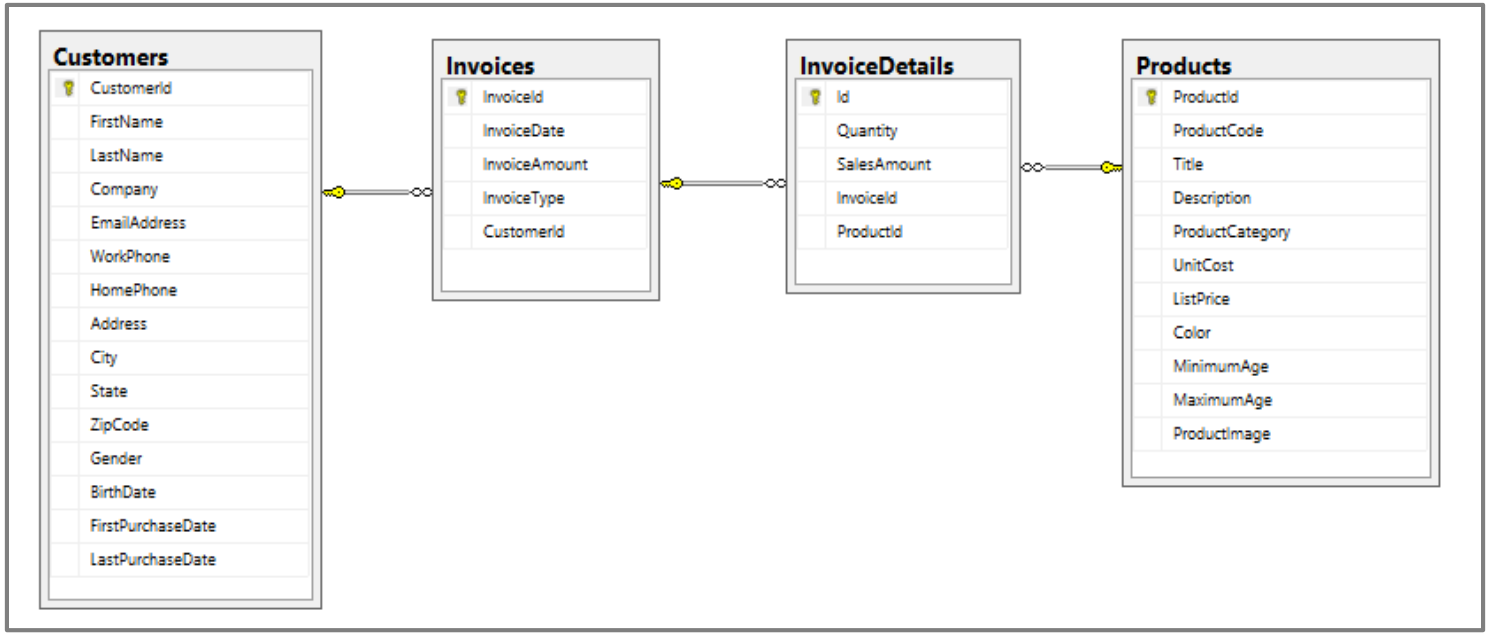
- ✓ Class Introduction
- ✓ Query Design Fundamentals
- ✓ SharePoint Online
- ➢ Query Folding
- Designing Data Model using a Star Schema
- Advanced Query Design

Agenda

- ✓ Class Introduction
- ✓ Query Design Fundamentals
- ✓ SharePoint Online
- ✓ Query Folding
- ➢ Designing Data Model using a Star Schema
- Advanced Query Design

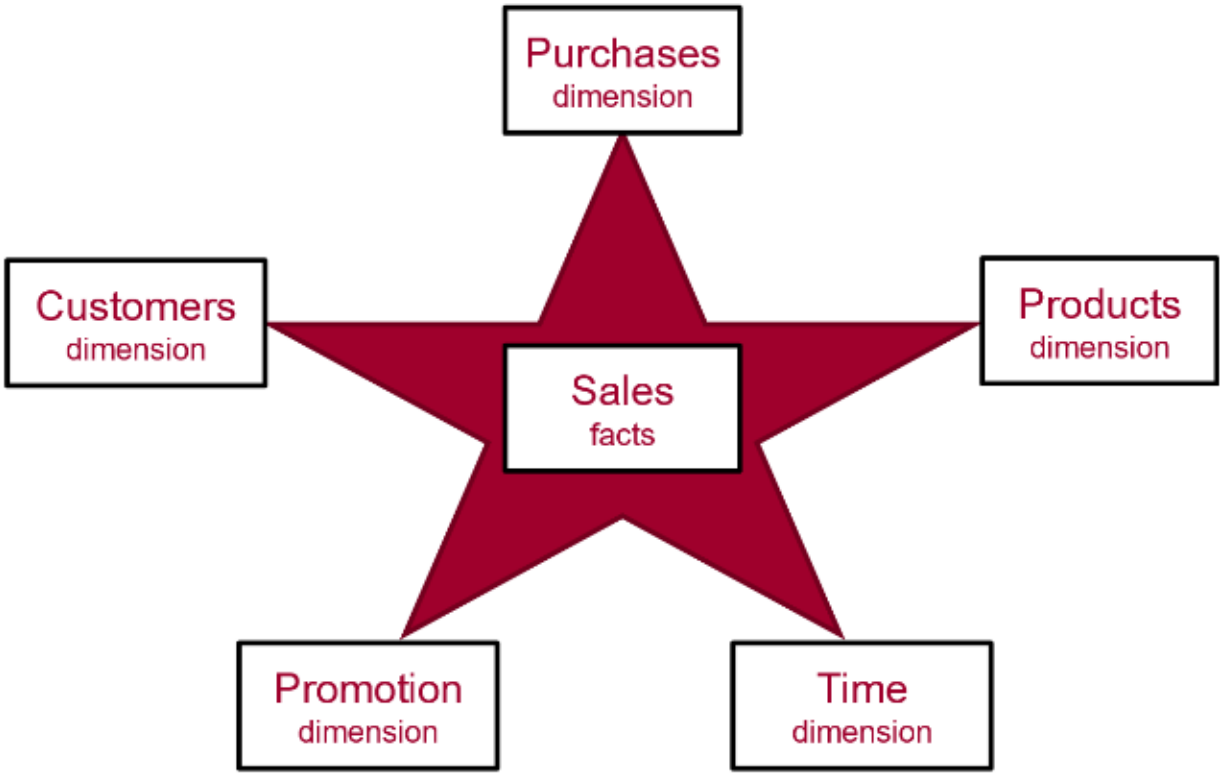
Sample OLTP Database: WingtipSalesDB

- Online Transaction Processing (OLTP) System
  - Used for real-time data access and transaction-based data entry
  - Optimized for faster transactions (e.g. inserts, updates & deletes)
  - Tables normalized to reduce/eliminate redundancies
  - Table schemas can be hard for business users to understand



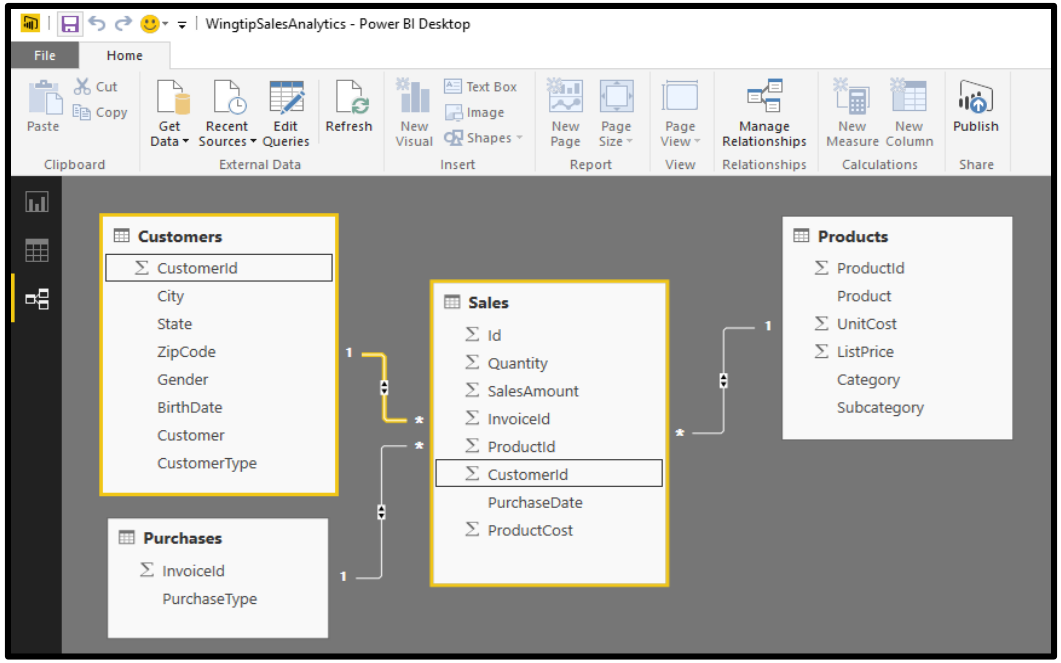
Data Modeling using a Star Schema

- 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



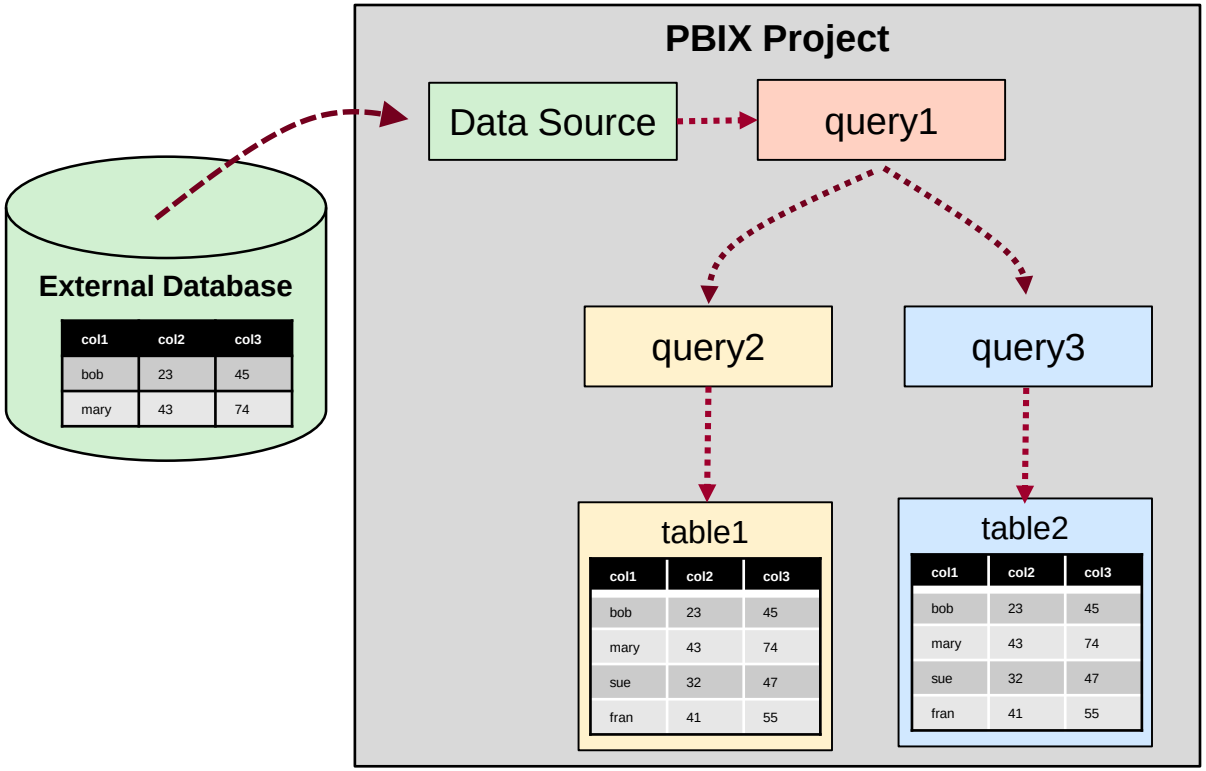

DEMO
Exploring the Wingtip Sales Analysis Demo Project

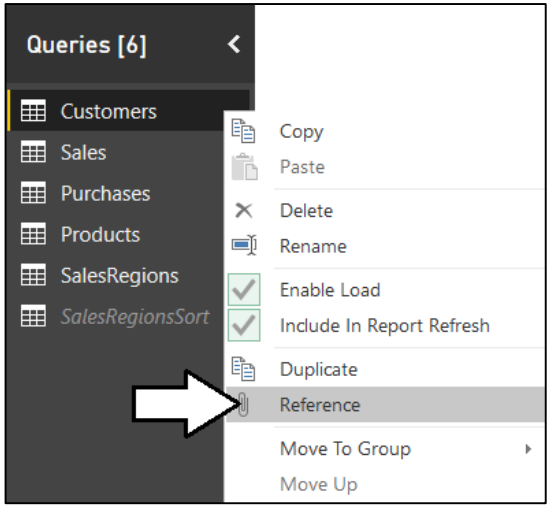
Agenda

- ✓ Class Introduction
- ✓ Query Design Fundamentals
- ✓ SharePoint Online
- ✓ Query Folding
- ✓ Designing Data Model using a Star Schema
- ➢ Advanced Query Design

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





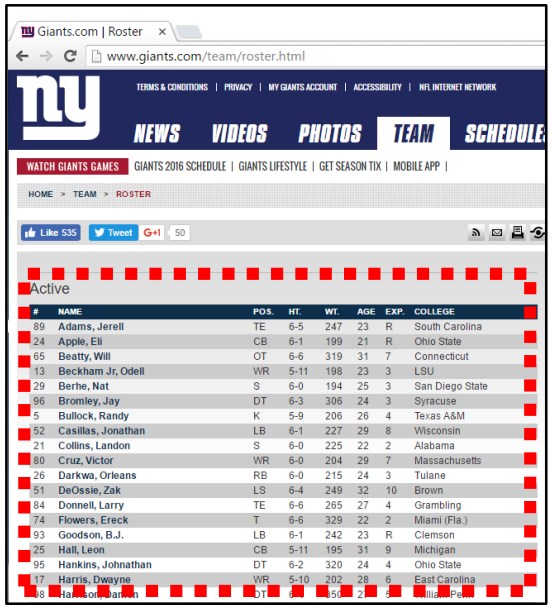
Agenda

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


DEMO
Creating a Function Query

Working with Web Data Sources

- Many public websites publish data using HTML tables
  - Power BI desktop can scrape data from tables in HTML pages



Query Input

	#	Name	Pos.	Ht.	Wt.	Age	Exp.	College
1	89	Adams, Jerell	TE	6-5	247	23	R	South Carolina
2	24	Apple, Eli	CB	6-1	199	21	R	Ohio State
3	65	Beatty, Will	OT	6-6	319	31	7	Connecticut
4	13	Beckham Jr, Odell	WR	5-11	198	23	3	LSU
5	29	Berhe, Nat	S	6-0	194	25	3	San Diego State
6	96	Bromley, Jay	DT	6-3	306	24	3	Syracuse

Query Output

	Number	Last Name	First Name	Weight	Height	Age	Experience	Position	Category	Side	College
1	89	Adams	Jerell	247	77	23	0	Tight End	Backs and Receivers	Offense	South Carolina
2	84	Donnell	Larry	265	78	27	4	Tight End	Backs and Receivers	Offense	Grambling
3	45	Tye	Will	262	74	24	1	Tight End	Backs and Receivers	Offense	Stony Brook
4	24	Apple	Eli	199	73	21	0	Cornerback	Defensive Backs	Defense	Ohio State
5	25	Hall	Leon	195	71	31	9	Cornerback	Defensive Backs	Defense	Michigan
6	20	Jenkins	Janoris	198	70	27	5	Cornerback	Defensive Backs	Defense	North Alabama
7	41	Rodgers-Cromartie	Dominique	205	74	30	8	Cornerback	Defensive Backs	Defense	Tennessee State
8	65	Beatty	Will	319	78	31	7	Offensive Tackle	Offensive Line	Offense	Connecticut
9	13	Beckham Jr	Odell	198	71	23	3	Wide Receiver	Backs and Receivers	Offense	LSU
10	80	Cruz	Victor	204	72	29	7	Wide Receiver	Backs and Receivers	Offense	Massachusetts

Summary

- ✓ Class Introduction
- ✓ Query Design Fundamentals
- ✓ SharePoint Online
- ✓ Query Folding
- ✓ Designing Data Model using a Star Schema
- ✓ Advanced Query Design