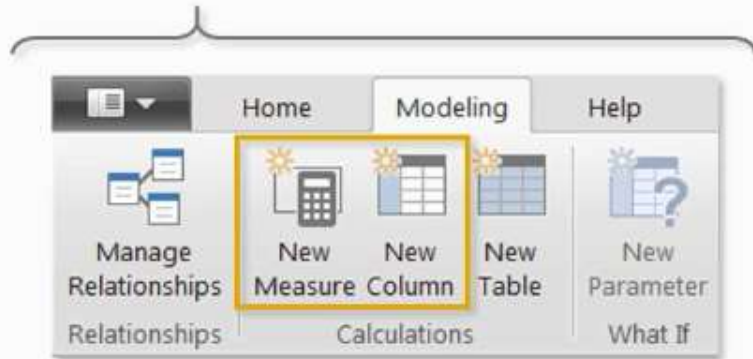
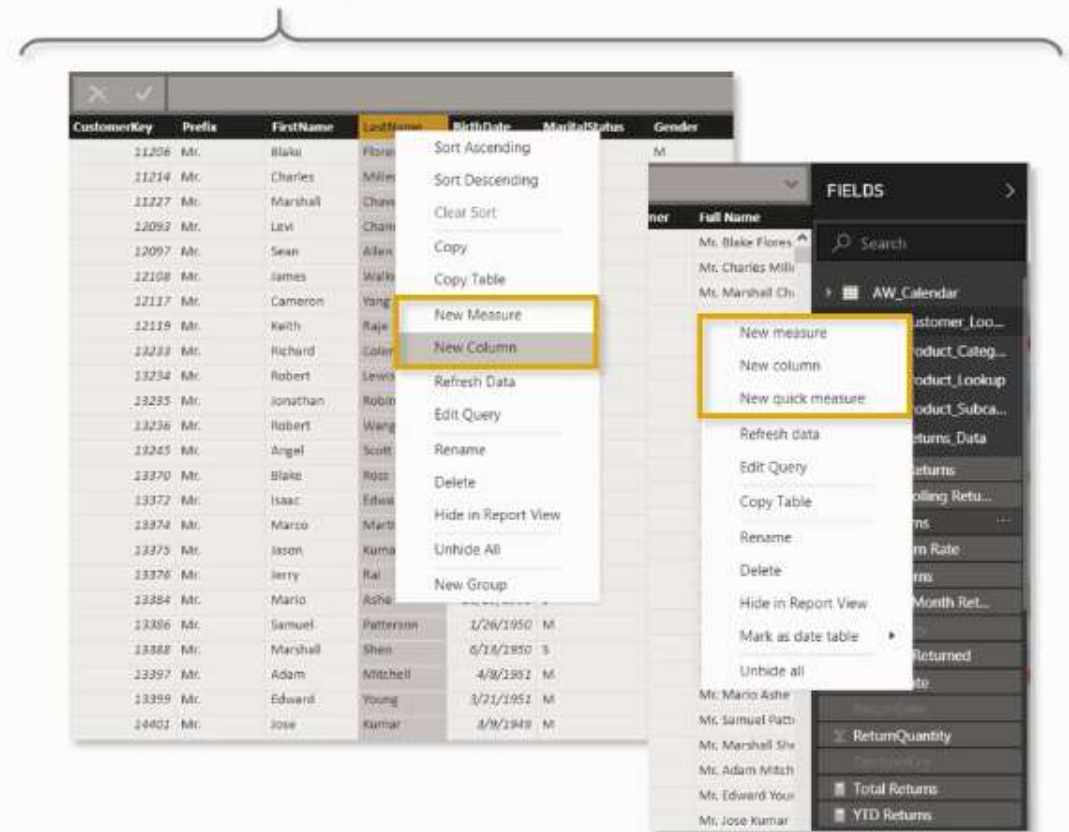


# ADDING COLUMNS & MEASURES

**Option 1:** Select “**New Measure**” or “**New Column**” from the **Modeling** tab



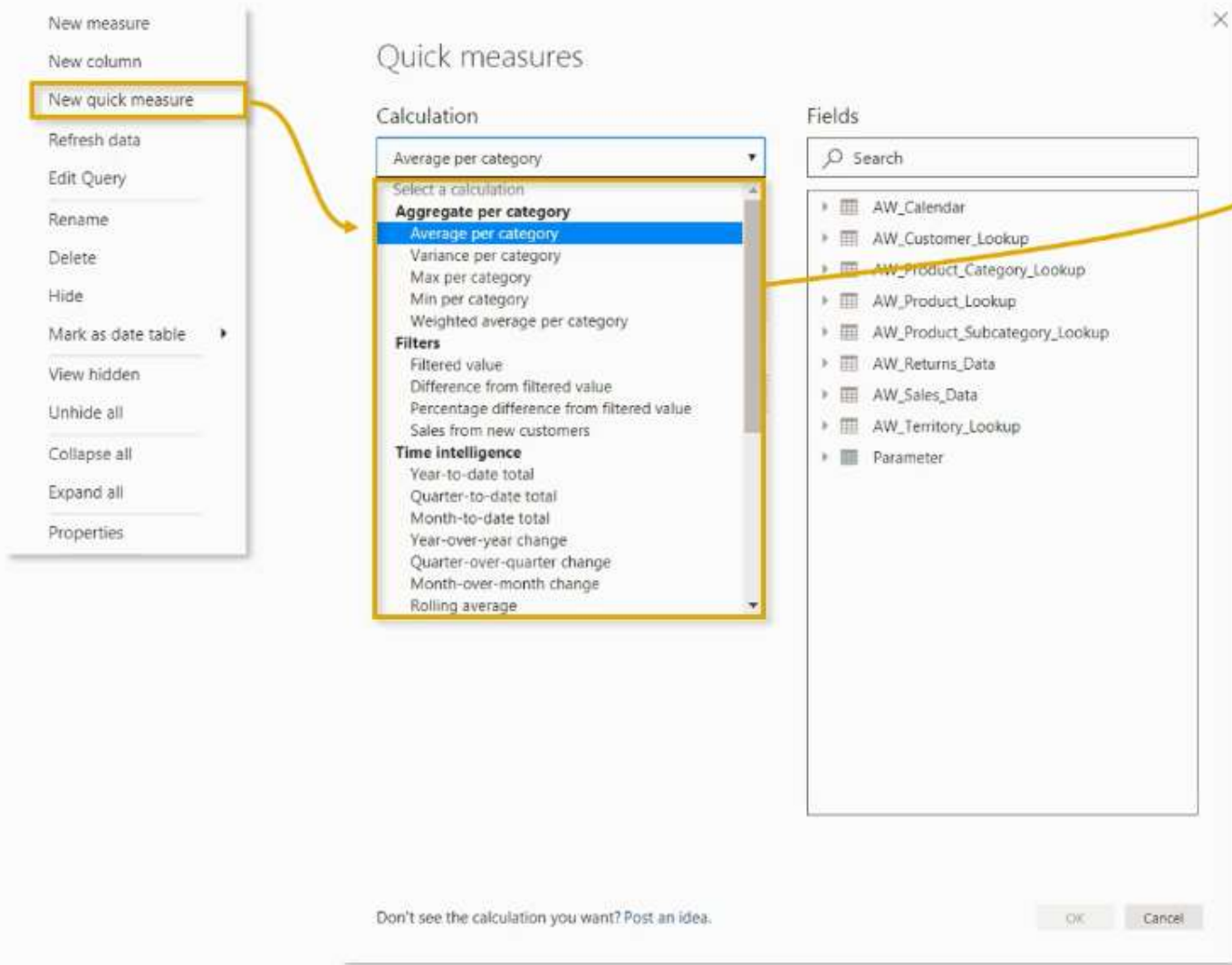
**Option 2:** Right-click within the **table** (in the **Data** view) or the **Field List** (in either the **Data** or **Report** view)



When you insert Columns or Measures using the **Modeling** tab (Option 1), they are assigned to whichever table is *currently selected*, or the *first table in the field list* by default

- Measures can be reassigned to new “Home” tables (under the “**Properties**” options in the **Modeling** tab), but the Option 2 allows you to be more deliberate about placing them
  - Note:** Assigning measures to specific tables doesn’t have any impact on functionality – it’s just a way to keep them organized

# QUICK MEASURES



**Quick Measures** are pre-built formula templates that allow you to drag and drop fields, rather than write DAX from scratch

While these tools can be helpful for defining more complex measures (*like weighted averages or time intelligence formulas*), they encourage laziness and don't help you understand the fundamentals of DAX



## PRO TIP:

Just say **"NO"** to quick measures  
(you're better than that)

# Demo

- Go to report view tab
- Select AW\_salse and then write click then Select new measure
- Write the following DAX

```
Quantity Sold = SUM(AW_Sales[OrderQuantity])
```

- Select the new quantity sold field under AW\_Sales and drag it to value.

ProductKey	OrderQuantity
214	2099
215	1940
220	1995
223	4151
226	392
229	408
232	424
235	381
310	169
311	139
312	179
313	168
314	157
320	65
322	39
324	72
Total	84174

# STEP-BY-STEP MEASURE CALCULATION

CategoryName	Total Returns
Accessories	1,115
Bikes	342
Clothing	267

How *exactly* is this measure calculated?

- REMEMBER: This all happens *instantly* behind the scenes, every time the filter context changes

## STEP 1

Filter context is detected & applied

CategoryName	Total Returns
Accessories	1,115
Bikes	342
Clothing	267

Product[CategoryName] = "Accessories"

Product Table	
Accessories	

## STEP 2

Filters flow "downstream" to all related tables

Product Table	
Accessories	

AW_Sales_Data	
Accessories	

AW>Returns_Data	
Accessories	

## STEP 3

Measure formula evaluates against the filtered table

Total Returns =	COUNTROWS(AW_Returns_Data)
-----------------	----------------------------

Count of rows in the AW\_Returns\_Data table, filtered down to only rows where the product category is "Accessories" = 1,115



# DAX SYNTAX

## MEASURE NAME

- **Note:** Measures are always surrounded in brackets (i.e. **[Total Quantity]**) when referenced in formulas, so spaces are OK

Total Quantity: =SUM(Transactions[quantity])

Referenced  
**TABLE NAME**

Referenced  
**COLUMN NAME**

## FUNCTION NAME

- Calculated columns don't always use functions, but measures do:
  - In a **Calculated Column**, =Transactions[quantity] returns the value from the quantity column in each row (since it evaluates one row at a time)
  - In a **Measure**, =Transactions[quantity] will return an **error** since Power BI doesn't know how to translate that as a single value (you need some sort of aggregation)

**Note:** This is a “fully qualified” column, since it's preceded by the table name -- table names with spaces must be surrounded by **single quotes**:

- Without a space: **Transactions[quantity]**
- With a space: **'Transactions Table'[quantity]**



## PRO TIP:

For **column** references, use the fully qualified name (i.e. **Table[Column]**)  
For **measure** references, just use the measure name (i.e. **[Measure]**)

# DAX OPERATORS

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

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

# DAX OPERATORS

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

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

*Pay attention to these!*

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" }



# COMMON FUNCTION CATEGORIES

## MATH & STATS Functions

*Basic **aggregation** functions as well as “**iterators**” evaluated at the row-level*

### **Common Examples:**

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

### **Iterator Functions:**

- SUMX
- AVERAGEX
- MAXX/MINX
- RANKX
- COUNTX

## LOGICAL Functions

*Functions for returning information about values in a given **conditional expression***

### **Common Examples:**

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

## TEXT Functions

*Functions to manipulate **text strings** or **control formats** for dates, times or numbers*

### **Common Examples:**

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

## FILTER Functions

***Lookup** functions based on related tables and **filtering** functions for dynamic calculations*

### **Common Examples:**

- CALCULATE
- FILTER
- ALL
- ALLEXCEPT
- RELATED
- RELATEDTABLE
- DISTINCT
- VALUES
- EARLIER/EARLIEST
- HASONESVALUE
- HASONEFILTER
- ISFILTERED
- USERELATIONSHIP

## DATE & TIME Functions

*Basic **date and time** functions as well as advanced **time intelligence** operations*

### **Common Examples:**

- DATEDIFF
- YEARFRAC
- YEAR/MONTH/DAY
- HOUR/MINUTE/SECOND
- TODAY/NOW
- WEEKDAY/WEEKNUM

### **Time Intelligence Functions:**

- DATESYTD
- DATESQTD
- DATESMTD
- DATEADD
- DATESINPERIOD



# BASIC DATE & TIME FUNCTIONS

**DAY/MONTH/  
YEAR()**

*Returns the day of the month (1-31), month of the year (1-12), or year of a given date*

=**DAY/MONTH/YEAR**(Date)

**HOUR/MINUTE/  
SECOND()**

*Returns the hour (0-23), minute (0-59), or second (0-59) of a given datetime value*

=**HOUR/MINUTE/SECOND**(Datetime)

**TODAY/NOW()**

*Returns the current date or exact time*

=**TODAY/NOW**()

**WEEKDAY/  
WEEKNUM()**

*Returns a weekday number from 1 (Sunday) to 7 (Saturday), or the week # of the year*

=**WEEKDAY/WEEKNUM**(Date, [ReturnType])

**EOMONTH()**

*Returns the date of the last day of the month, +/- a specified number of months*

=**EOMONTH**(StartDate, Months)

**DATEDIFF()**

*Returns the difference between two dates, based on a selected interval*

=**DATEDIFF**(Date1, Date2, Interval)

# Demo

---

- Go to table view → select calendar table
- Create new column
- Write DAX formula

```
Day of week = WEEKDAY(AW_Calendar_Lookup[Date],1)
```

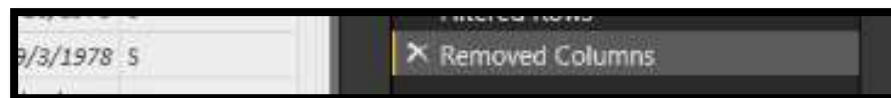
- Go to (Customer\_Lookup) table (Birth Year )column , create same (birth year) column using DAX and remove the original.

```
BirthYear_CC = year(AW_Customer_Lookup[BirthDate])
```

- Also we can get same column values without using DAX functions by

```
BirthYear_CC = AW_Customer_Lookup[BirthDate].[Year]
```

- Delete (BirthYear) column generate by Query Editor and see if this step will be added.



# Demo (Cont) get the age of the customers

---

- Create new column
- Write DAX formula to calculate the (customer age) using

```
Current Age = DATEDIFF(AW_Customer_Lookup[BirthDate], TODAY(), YEAR)
```

# BASIC LOGICAL FUNCTIONS (IF/AND/OR)

**IF()**

*Checks if a given condition is met, and returns one value if the condition is TRUE, and another if the condition is FALSE*

=**IF**(LogicalTest, ResultIfTrue, [ResultIfFalse])

**IFERROR()**

*Evaluates an expression and returns a specified value if the expression returns an error, otherwise returns the expression itself*

=**IFERROR**(Value, ValueIfError)

**AND()**

*Checks whether both arguments are TRUE, and returns TRUE if both arguments are TRUE, otherwise returns FALSE*

=**AND**(Logical1, Logical2)

**OR()**

*Checks whether one of the arguments is TRUE to return TRUE, and returns FALSE if both arguments are FALSE*

=**OR**(Logical1, Logical2)

**Note:** Use the **&&** and **||** operators if you want to include more than two conditions!



# Demo, check wither the customer is parent or not?

---

- Go to (AW\_customer\_lookup) , create new column
- Write DAX formula to check wither a customer is parent or not

```
parent = if(AW_Customer_Lookup[TotalChildren]>0,"Yes","No")
```

# Demo , Identify weekend day from Day of Week column

---

- Go to (AW\_Calender\_lookup)
- Write the following to check wither the day is week end or not

```
weekend = if(or(AW_Calendar_Lookup[Day of week]=6,AW_Calendar_Lookup[Day of week]=7),"weekend","No")
```

# TEXT FUNCTIONS

**LEN()**

*Returns the number of characters in a string*

=**LEN**(Text)

*Note: Use the & operator as a shortcut,  
or to combine more than two strings!*

**CONCATENATE()**

*Joins two text strings into one*

=**CONCATENATE**(Text1, Text2)

**LEFT/MID/  
RIGHT()**

*Returns a number of characters from the  
start/middle/end of a text string*

=**LEFT/RIGHT**(Text, [NumChars])

=**MID**(Text, StartPosition, NumChars)

**UPPER/LOWER/  
PROPER()**

*Converts letters in a string to  
upper/lower/proper case*

=**UPPER/LOWER/PROPER**(Text)

**SUBSTITUTE()**

*Replaces an instance of existing text with  
new text in a string*

=**SUBSTITUTE**(Text, OldText, NewText,  
[InstanceNumber])

**SEARCH()**

*Returns the position where a specified string  
or character is found, reading left to right*

=**SEARCH**(FindText, WithinText,  
[StartPosition], [NotFoundValue])

# Demo

---

- Go to AW\_Csutomer\_lookup
- Create new column
- Write DAX formula to Concatenate (prefix , First name, last name )

```
fullname_CC = AW_Customer_Lookup[Prefix] &  
" " & AW_Customer_Lookup[FirstName] & " " &  
AW_Customer_Lookup[LastName]
```



# Demo

---

- Go to (AW\_Calendar\_lookup) table
- Write DAX formula to get a shortcut for the month name (first 3 char of each month)

```
month name _CC = left(AW_Calendar_Lookup[Month Name],3)
```

- Change the values in (month name) column to upper case

```
month name _CC = upper(left(AW_Calendar_Lookup[Month Name],3))
```

# BASIC MATH & STATS FUNCTIONS

**SUM()**

*Evaluates the sum of a column*

=**SUM**(ColumnName)

**AVERAGE()**

*Returns the average (arithmetic mean) of all the numbers in a column*

=**AVERAGE**(ColumnName)

**MAX()**

*Returns the largest value in a column or between two scalar expressions*

=**MAX**(ColumnName) or =**MAX**(Scalar1, [Scalar2])

**MIN()**

*Returns the smallest value in a column or between two scalar expressions*

=**MIN**(ColumnName) or =**MIN**(Scalar1, [Scalar2])

**DIVIDE()**

*Performs division and returns the alternate result (or blank) if div/0*

=**DIVIDE**(Numerator, Denominator, [AlternateResult])

# Demo : calculate the Avg of product price

- Go the report view
- Go to product table tab and create new measure.
- Write the followings DAX

```
Avg price = AVERAGE(AW_Product_Lookup[ProductPrice])
```

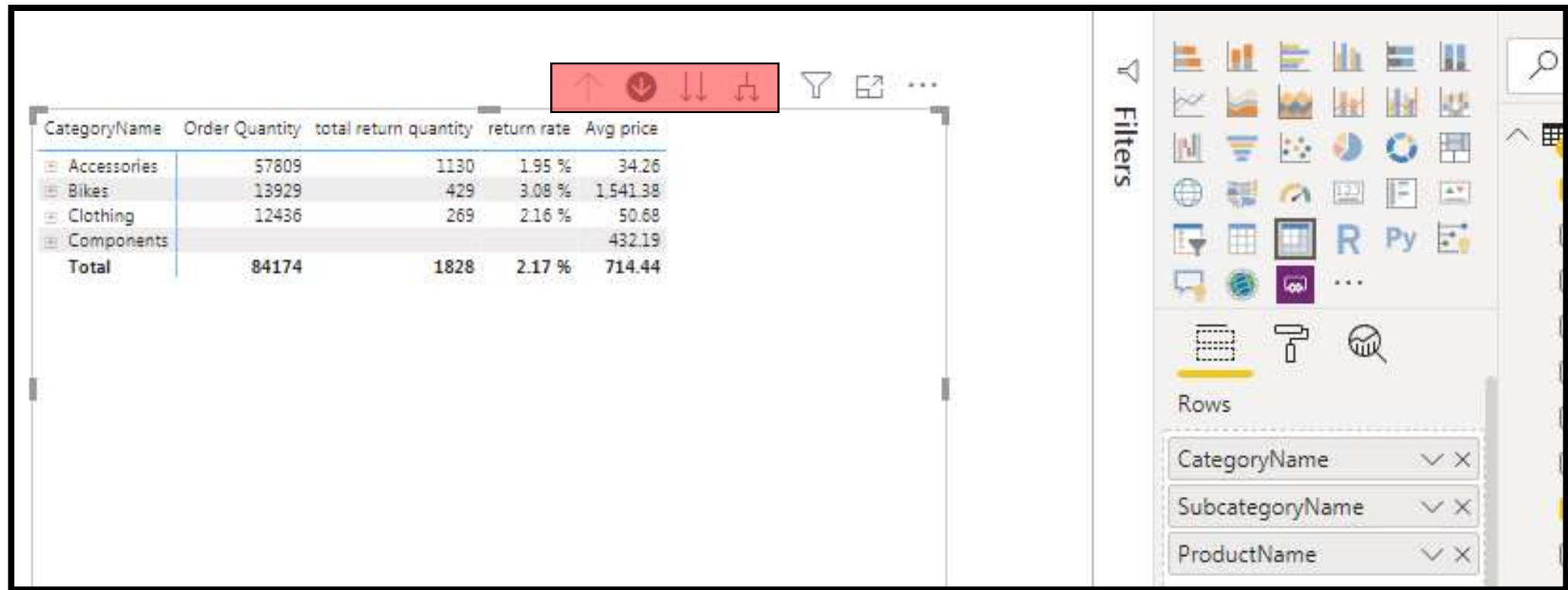
- Drage (Avg price) to the value field from the matrix
- Format the (price) of (Avg price) to currency
- Drag the (subcategory name )to the column

CategoryName	Order Quantity	total return quantity	return rate	Avg price
Accessories	57029	1110	1.91 %	34.28
Skis	12929	409	0.08 %	1,940.38
Clothing	12438	289	2.18 %	30.88
Components	84176	1828	2.17 %	452.13
Total	84176	1828	2.17 %	714.44

CategoryName	Order Quantity	total return quantity	return rate	Avg price
Accessories	57029	1110	1.91 %	34.28
Skis	12929	409	0.08 %	1,940.38
Clothing	12438	289	2.18 %	30.88
Components	84176	1828	2.17 %	452.13
Total	84176	1828	2.17 %	714.44

# Demo (Cont)

- Drag the product name to the (row) field of matrix



The screenshot displays a data visualization tool interface. On the left, a table shows product categories and their associated metrics. Above the table, a red box highlights four icons: an upward arrow, a downward arrow, a double downward arrow, and a refresh icon. On the right, a 'Filters' panel is visible, containing a grid of chart icons and a 'Rows' section. The 'Rows' section lists 'CategoryName', 'SubcategoryName', and 'ProductName', each with a dropdown arrow and a close button. The 'ProductName' field is currently selected.

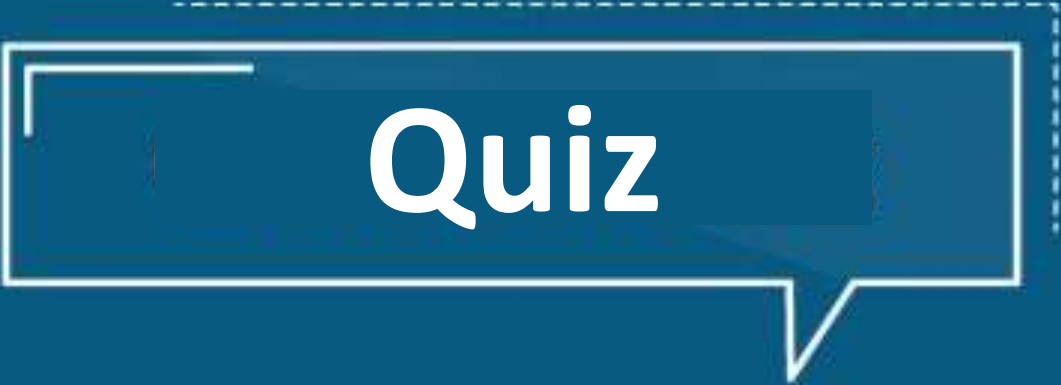
CategoryName	Order Quantity	total return quantity	return rate	Avg price
Accessories	57809	1130	1.95 %	34.26
Bikes	13929	429	3.08 %	1,541.38
Clothing	12436	269	2.16 %	50.68
Components				432.19
<b>Total</b>	<b>84174</b>	<b>1828</b>	<b>2.17 %</b>	<b>714.44</b>

Filters

Rows

- CategoryName
- SubcategoryName
- ProductName





Quiz

# Q1.

---

- **Which two types of fields can DAX be used to create**

☐ Calculated Columns & Calculated Fields

☐ Calculated Columns & Measures

☐ Calculated Fields & Measures

☐ Measures & Calculated Items

---

- **What's the difference between calculated columns and measures**

☐ Calculated columns understand row context

☐ Measures understand filter context

☐ Calculated column values are stored in tables

☐ All of the above

- 
- Which of the following formulas would make the *most* sense as a calculated column?

☐ =SUM(Sales[quantity])

☐ =AVERAGE(Products[RetailPrice])

☐ =DISTINCTCOUNT(Customers[CustomerName])

☐ =Products[RetailPrice] \* 0.75



- 
- **TRUE or FALSE:** The *Grand Total* cell in a Power BI visualization calculates by summing the measure values in the rows above it

**True**

**False**

---

- **When would you need to use the && operator?**

☐ To concatenate two text strings

☐ To create an AND condition between two expressions

☐ To concatenate more than two text strings

☐ To create an AND condition between more than two expressions

# Exercise: Analyzing Data with DAX Calculations in Power BI

---

- Using the Adventure Works report, complete the following:
- **1)** In the **DATA** view, create the following **calculated columns**:
- In the **AW\_Customer\_Lookup** table, add a new column named "**Customer Priority**" that equals "*Priority*" for customers who are under 50 years old and have an annual income of greater than \$100,000, and "*Standard*" otherwise
- In the **AW\_Product\_Lookup** table, add a new column named "**Price Point**", based on the following criteria
  - *If the product price is greater than \$500, **Price Point** = "High"*
  - *If the product price is between \$100 and \$500, **Price Point** = "Mid-Range"*
  - *If the product price is less than or equal to \$100, **Price Point** = "Low"*
- In the **AW\_Calendar\_Lookup** table, add a new column named "**Short Day**" to extract and capitalize the first three letters from the **Day Name** column
- In the **AW\_Product\_Lookup** table, add a column named "**SKU Category**" to extract the first two characters from the **ProductSKU** field
  - **BONUS:** *Modify the **SKU Category** function to return any number of characters up to the first dash (**Hint:** You may need to "search" long and hard for that dash...)*

---

2) In the **REPORT** view, create the following **measures** (*Use a matrix visual to match the "spot check" values provided*)

- Create a measure named "**Product Models**" to calculate the number of unique product model names
  - **Spot check:** *You should see a total of **119** unique product models*
- Create a measure named "**ALL Returns**" to calculate the grand total number of returns, regardless of the filter context
  - **Spot check:** *You should see a total of **1,809** returns*
- Create a measure to calculate "**% of All Returns**"
  - **Spot check:** *You should see a value of **61.64%** for the **Accessories** product category*
- Create a measure named "**Bike Returns**" to calculate total returns for bikes specifically
  - **Spot check:** *You should see a total of **427** bike returns*

- 
- Create a measure named "**Total Cost**", by multiplying order quantities by product costs at the row-level
    - *Spot check: You should see a total cost of **\$14,456,986.32***
  - Once you've calculated **Total Cost**, create a new measure for "**Total Profit**", defined as the total revenue minus the total cost
  - *Spot check: You should see a total profit of **\$10,457,580.86***



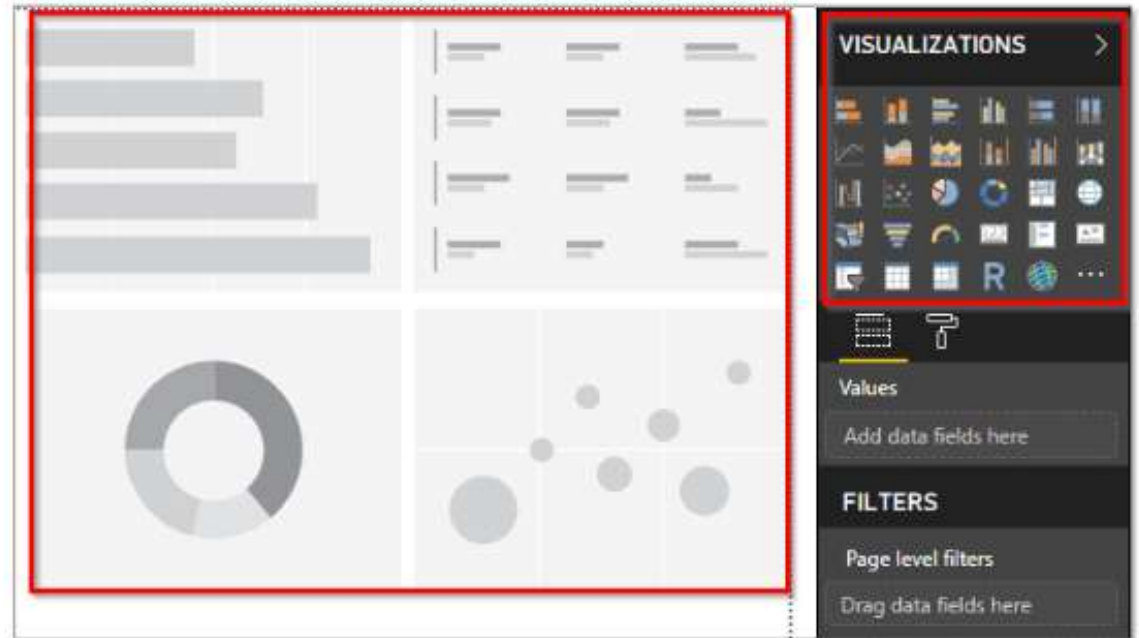
- 
- Total Cost =  $\text{sumx}(\text{'AW-Sales'}, \text{'AW-Sales'}[\text{OrderQuantity}] * \text{related}(\text{AW\_Product\_Lookup}[\text{ProductCost}])))$
  - total profit = [total revenue] - [Total Cost]



# **Section 5: Visualizing Data with Power BI Reports**

# Introduction To Visuals In Power BI

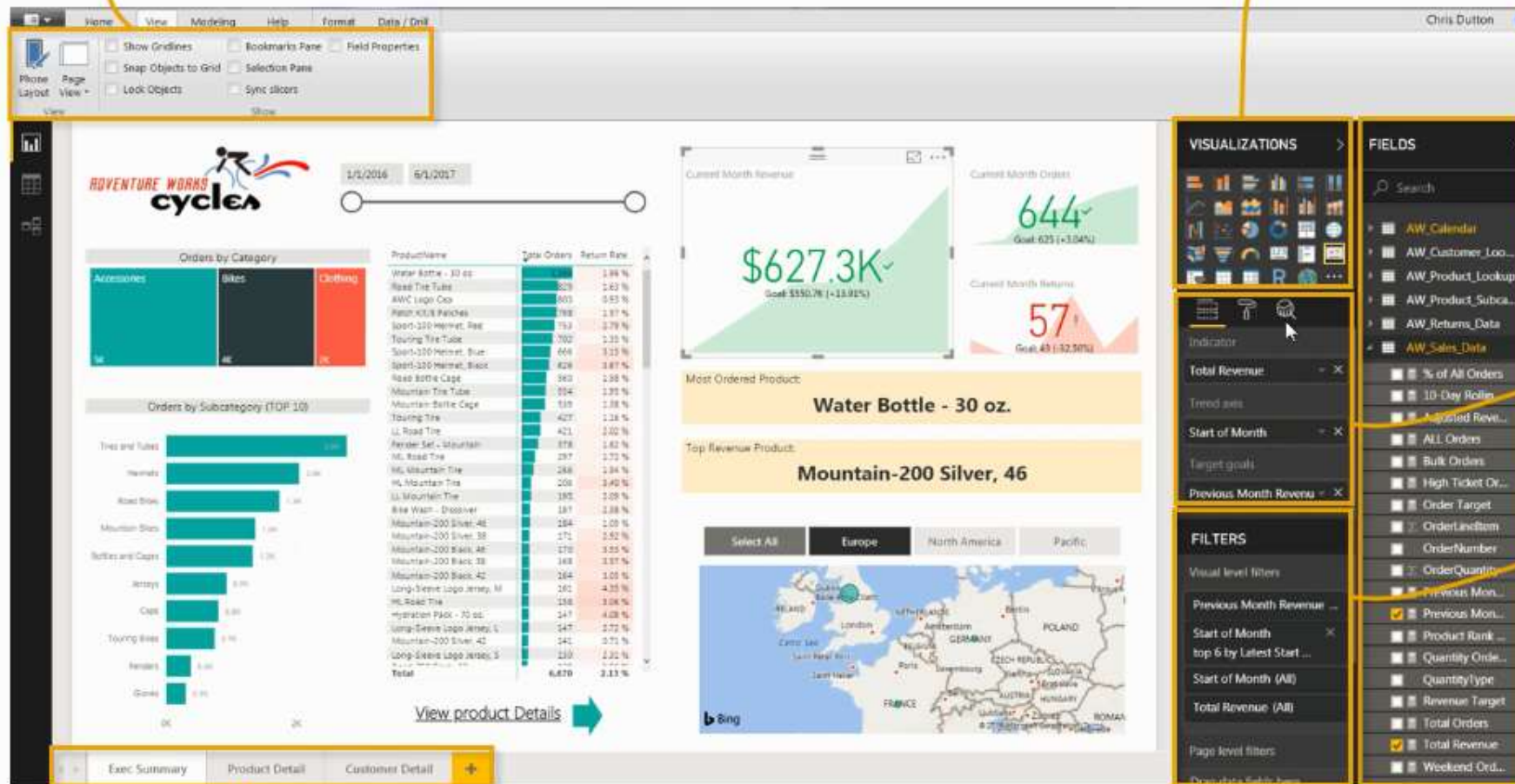
- Visuals are the end-result of any business intelligence and Power BI offers many such visualizations that will help to present your data in an insightful way.
- Visuals in Power BI are the basic building blocks and are available in different categories such as charts, tree maps, gauges and various other page formatting tools.



# THE POWER BI REPORT VIEW

**"View" Options** (Layout, Gridlines, Snap to Grid, Bookmarks/Selection Pane, etc)

**Visualization Options** (Charts, Slicers, Maps, Matrices, etc)



Field List (Tables, columns & measures)

Fields/Format/Analytics Pane (Fields pane shown here)

Filters Pane (Visual-Level, Page-Level, Report-Level, and Drillthrough Filters)

Report Pages (Similar to Excel tabs; each is a blank reporting canvas)

# Reports In Power BI

---

- A Power BI **report** is a multi-perspective view into a dataset, with visualizations that represent different findings and insights from that dataset
- A report can have a single visualization or pages full of visualizations
- Report is highly interactive and highly customizable and the visualizations update as the primary data changes





# How To Use A Visual In Power BI

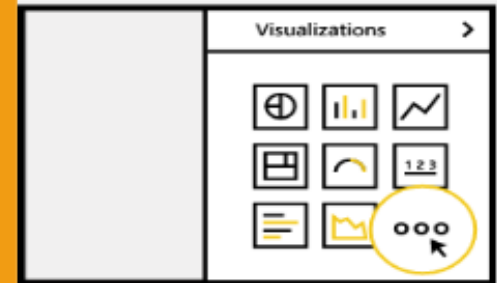
- **Step 1:** Download a visual

Browse Power BI gallery for any number of community created visuals. Click on the visual, download, and read through the terms of use



- **Step 2:** Upload to Power BI

In the report view, look for the upload icon in the visualization pane. Click the icon, follow the instructional popups, and upload your visual file



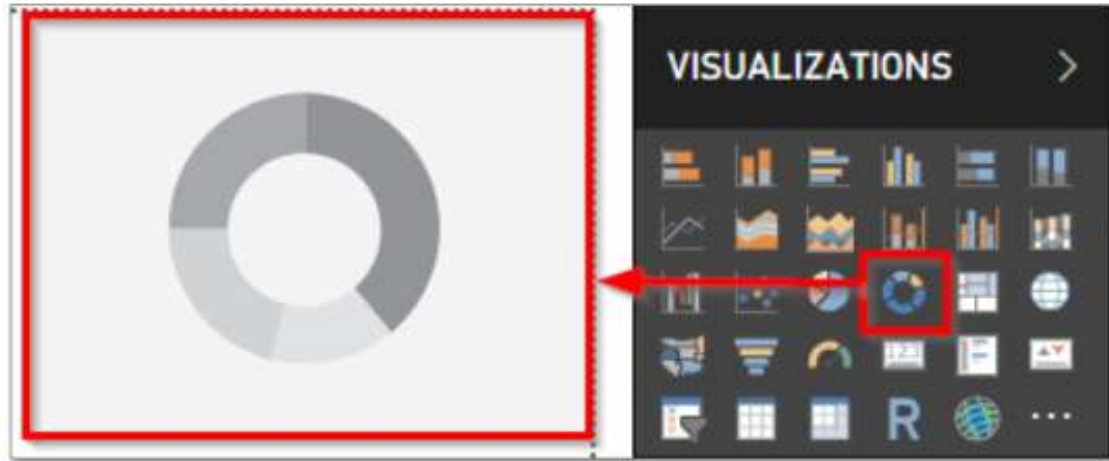
- **Step 3:** Access visual in panel

Upon importing, the visual will be made available in the visualization panel. This visual will work and operate identically to other Power BI visuals



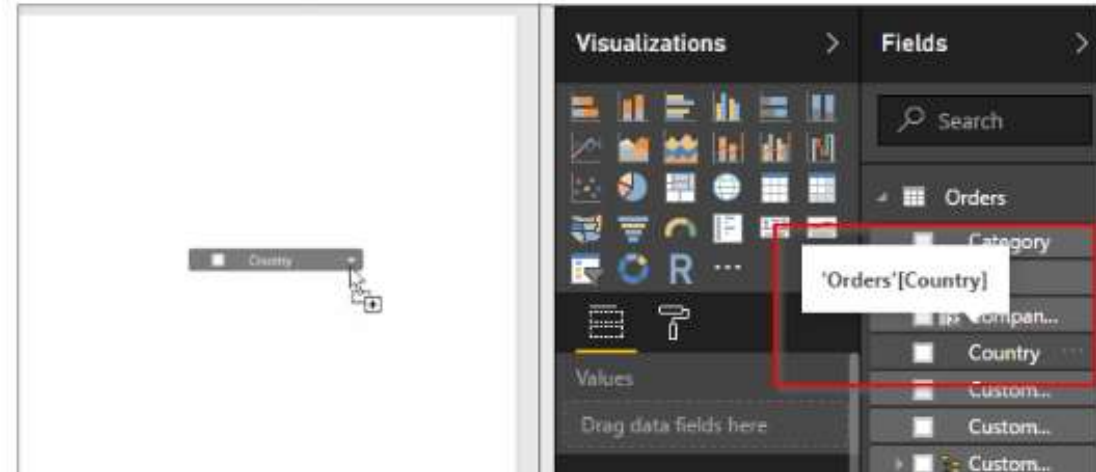
# Create Visualizations In Power BI

- Open your report in *Editing View* and create a visualization. Choose the type of visualization in a Power BI report, to see which one illustrates your data best

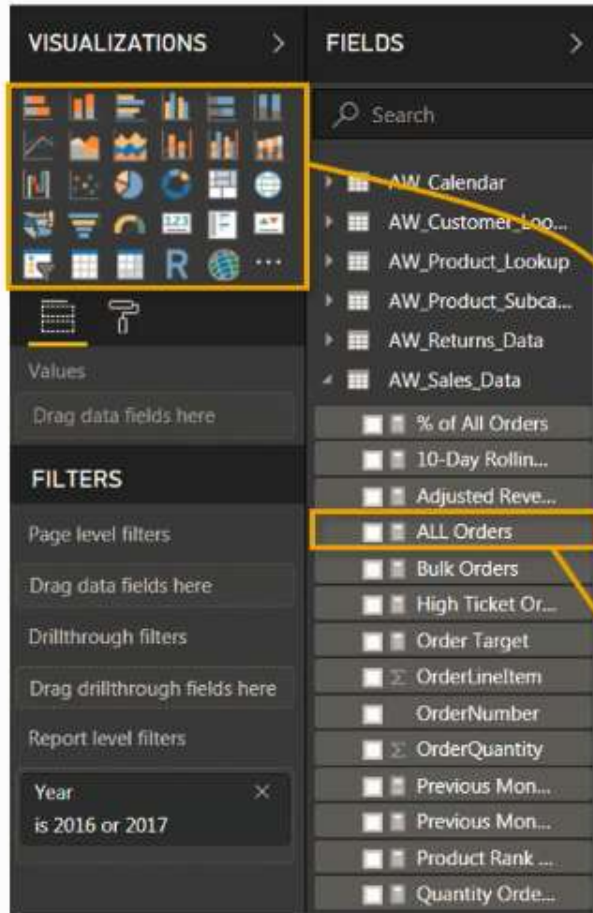


- Click the type of visualization you want to create in the Visualizations pane

- Drag field names from the *Fields* pane, and drop them on the report canvas



# INSERTING OBJECTS & BASIC CHARTS



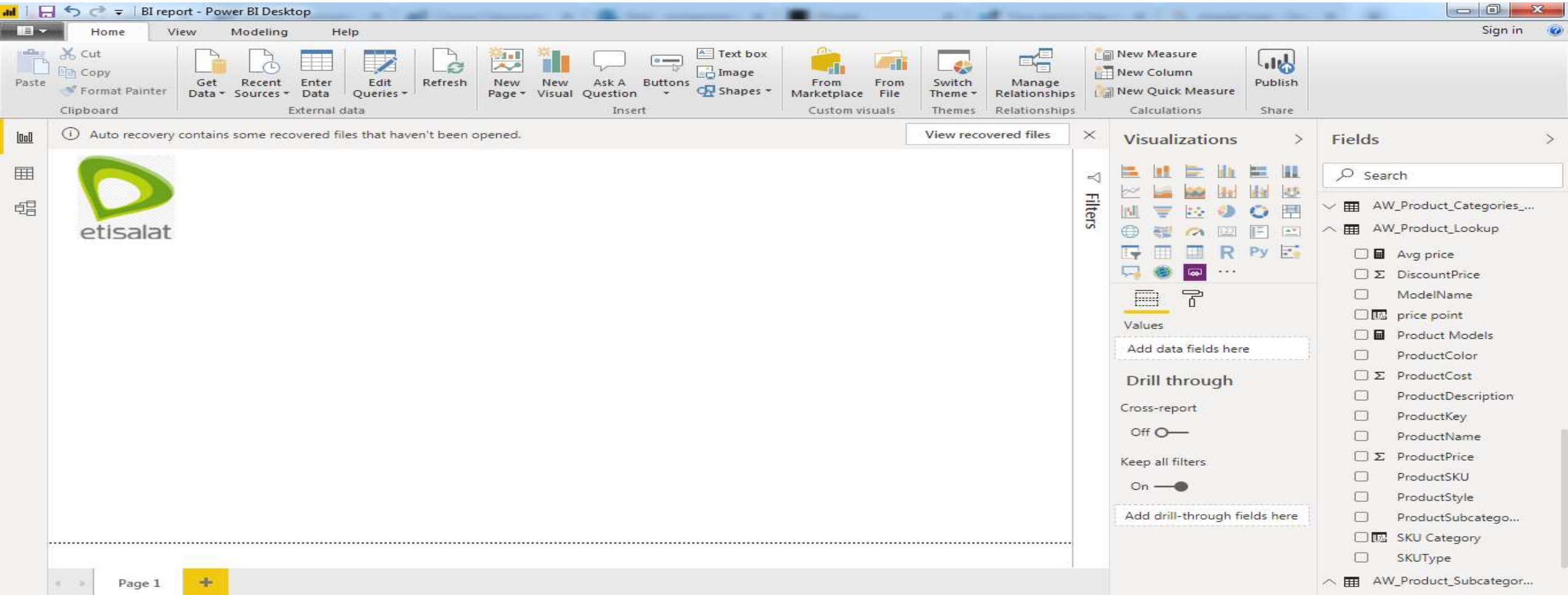
Click on a visualization type or use the “*New Visual*” option in the **Home** tab to insert a blank chart template (usually a column chart by default)

**Note:** You can also add **New Pages**, **Buttons**, **Text Boxes**, **Images** and **Shapes** from this menu



Drag fields or measures directly into the report canvas to automatically generate a new visual

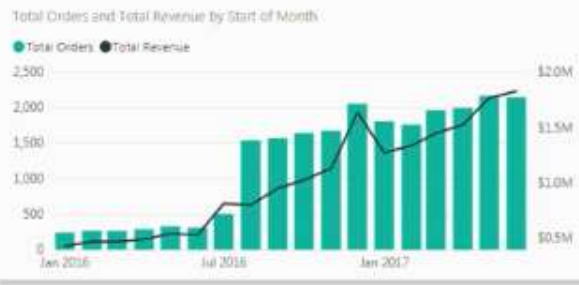
# Demo : Adding Image to the report view (Dashboard)





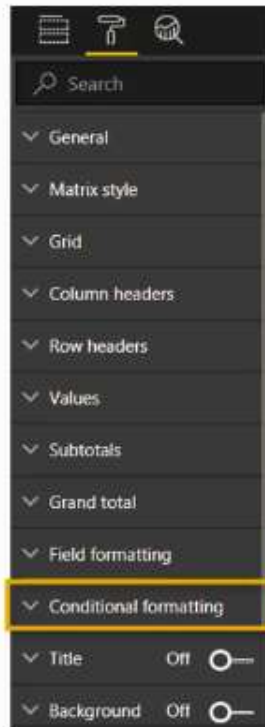
# FORMATTING OPTIONS

Example: Line & Column Chart

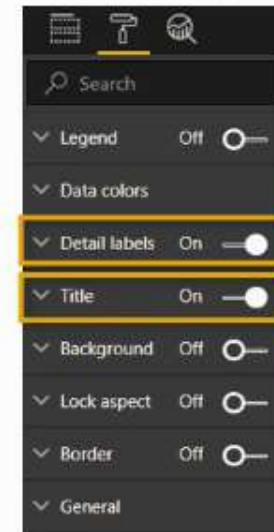


Example: Matrix

ProductName	Total Orders	Return Rate
Water Bottle - 30 oz.	1,164	1.96 %
Road Tire Tube	829	1.63 %
AWC Logo Cap	803	0.93 %
Patch Kit/8 Patches	798	1.57 %
Sport-100 Helmet, Red	753	2.79 %
Touring Tire Tube	702	1.35 %
Sport-100 Helmet, Blue	666	3.15 %
Sport-100 Helmet, Black	626	3.67 %
Road Bottle Cage	560	1.58 %
Mountain Tire Tube	534	1.95 %
Mountain Bottle Cage	539	1.38 %

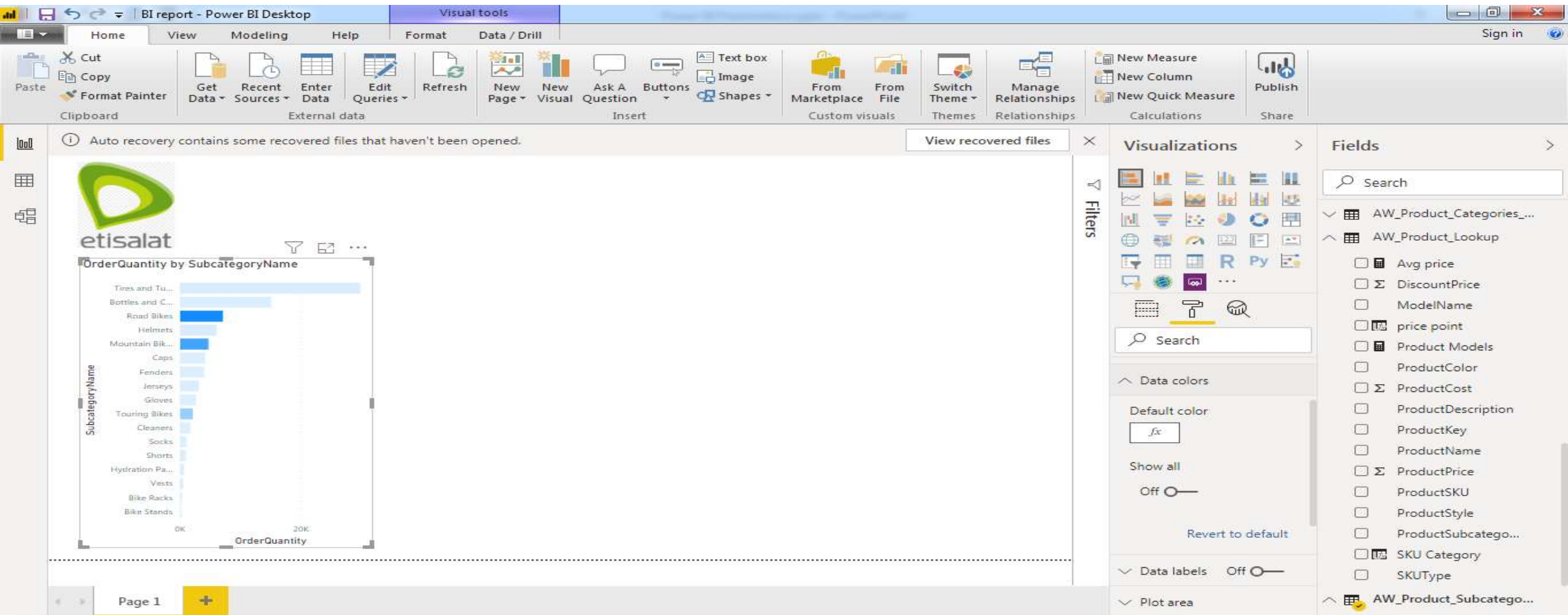


Example: Donut Chart

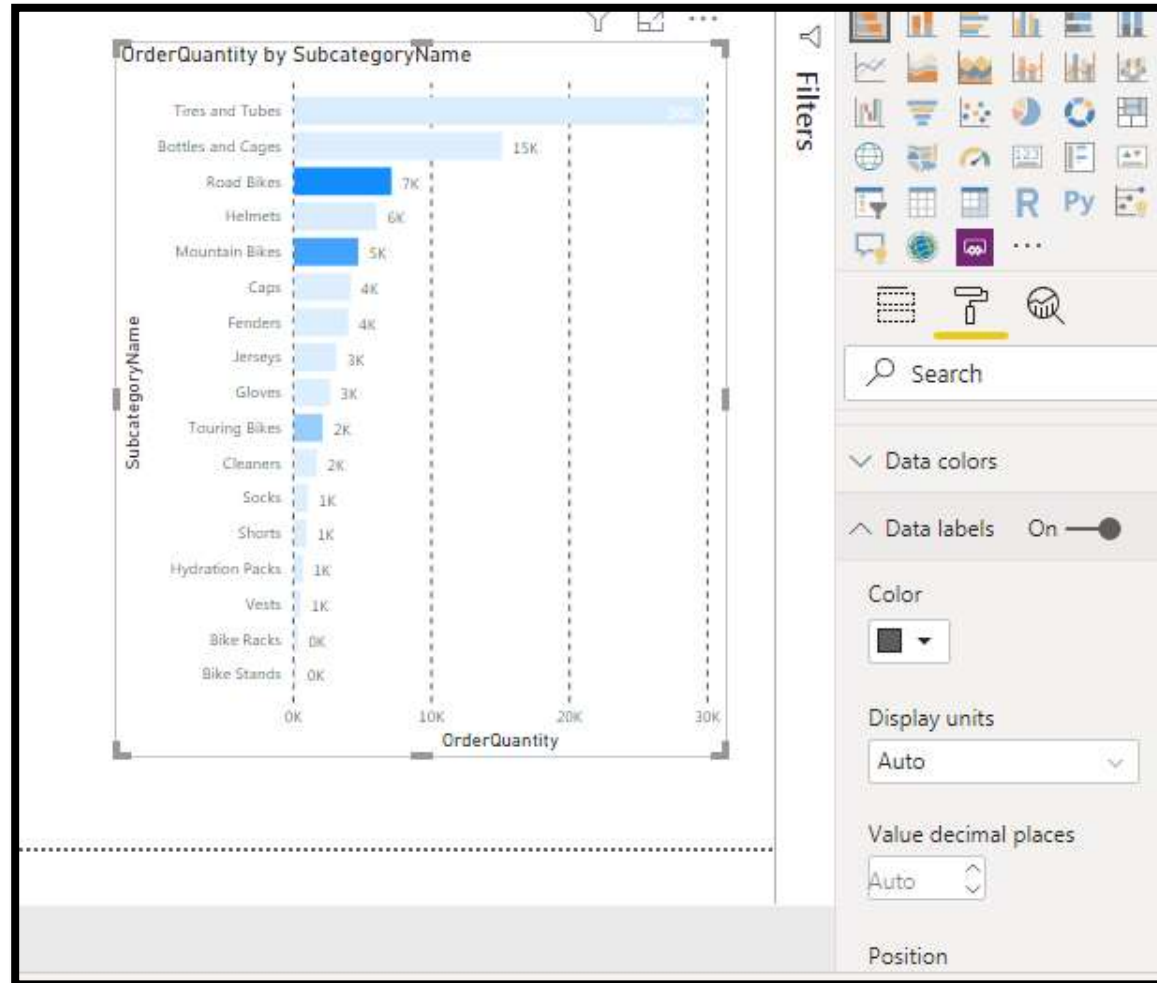




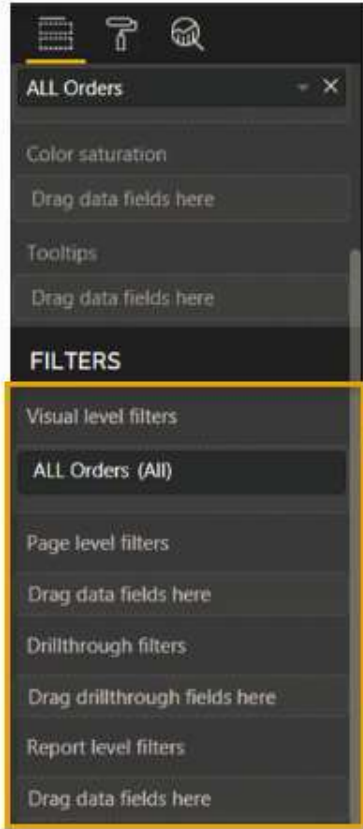
# Demo:



# Demo : exploring format tab



# FILTERING OPTIONS



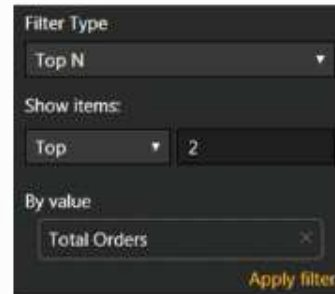
There are **four (x4)** primary filter types in Power BI reports:

1. **Visual Level:** Applies only to the *specific visual* in which it is defined
2. **Page Level:** Applies to *all visuals on the specific page* in which it is defined
3. **Report Level:** Applies to *all visuals* across *all pages* of the report
4. **Drillthrough:** Applies to *specific pages*, and *dynamically changes* based on user paths

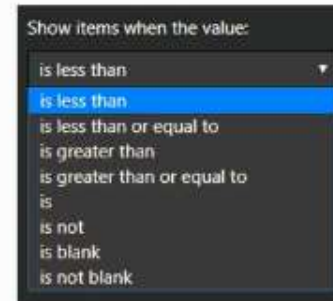
*Filter settings include Basic, Advanced, and Top N options*



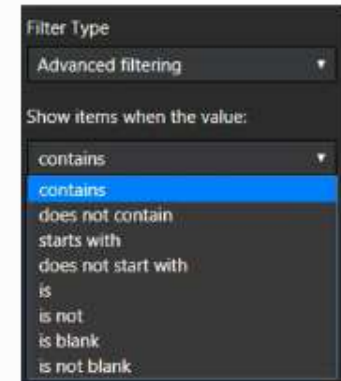
*Basic Options*



*Top N Options*

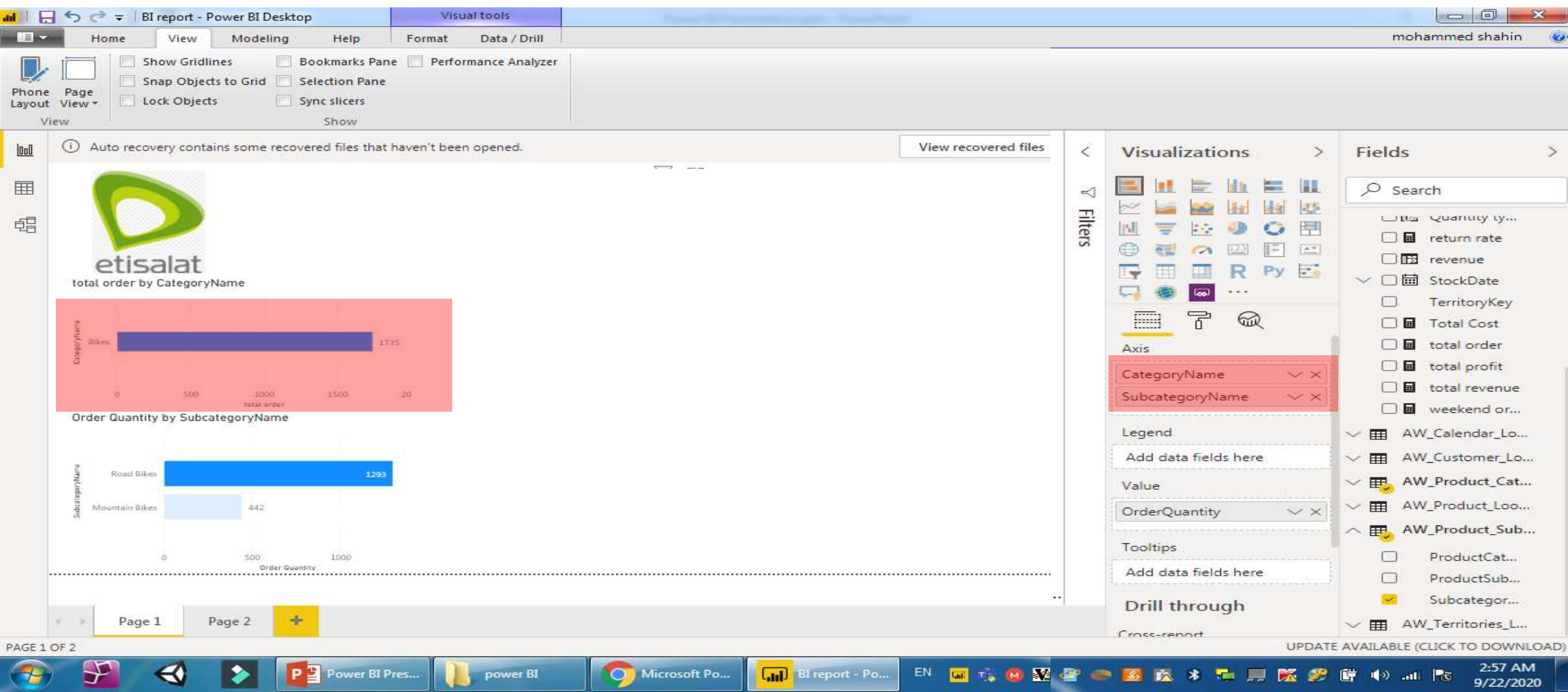


*Advanced (Values)*



*Advanced (Text)*

# Demo : Power BI Report Filtering





# Demo: Exploring Data with Matrix Visuals

BI report - Power BI Desktop

Visual tools

Home View Modeling Help Format Data / Drill

Phone Layout Page View

Show Gridlines Show Snap Objects to Grid Lock Objects Bookmarks Pane Selection Pane Sync slicers Performance Analyzer

Auto recovery contains some recovered files that haven't been opened.

View recovered files

Filters

Visualizations

Fields

Search

etisalat

total order by CategoryName

CategoryName

Order Quantity by SubcategoryName

SubcategoryName

ProductName	total order	return rate
All-Purpose Bike Stand	234	3.42 %
AWC Logo Cap	2062	1.11 %
Bike Wash - Dissolver	850	1.47 %
Classic Vest, L	182	2.20 %
Classic Vest, M	182	3.85 %
Classic Vest, S	157	5.10 %
Fender Set - Mountain	1975	1.36 %
Half-Finger Gloves, L	414	2.14 %
Half-Finger Gloves, M	465	1.74 %
Half-Finger Gloves, S	453	1.69 %
Hitch Rack - 4-Bike	302	2.65 %
HL Mountain Tire	1305	3.75 %
HL Road Tire	795	3.52 %
Hydration Pack - 70 oz.	695	3.60 %
LL Mountain Tire	788	2.50 %
LL Road Tire	957	2.26 %
Long-Sleeve Logo Jersey, L	424	3.54 %
Long-Sleeve Logo Jersey, M	408	3.68 %
Long-Sleeve Logo Jersey, S	392	3.06 %
Long-Sleeve Logo Jersey, XL	381	2.62 %
LL Mountain Tire	1305	3.75 %
Total	25164	2.17 %

Field

Date

Drill through

Cross-report

Off

Keep all filters

On

Add drill-through fields here

AW\_Calendar\_Lookup

AW\_Customer\_Lookup

AW\_Product\_Categories\_L...

Page 1 Page 2

PAGE 1 OF 2

UPDATE AVAILABLE (CLICK TO DOWNLOAD)

9:57 PM 9/20/2020

# Demo

---

- Drag the (Matrix visual) from report view to the workspace.
- Drag (total order) to the (value) field
- Drag (return rate) to the (value) field.
- Drag (product name) to the (rows) field
- Go to format
- Check matrix style
- Check conditional formatting (Data bars ,font color, background)
- Drag the (subcategory name) & (category name) to the row field



# Demo : Filtering with Date Slicers

BI report - Power BI Desktop

Visual tools

Home View Modeling Help Format Data / Drill

Phone Layout Page View

Show Gridlines Show Snap Objects to Grid Lock Objects Show Bookmarks Pane Selection Pane Sync slicers Performance Analyzer

Auto recovery contains some recovered files that haven't been opened.

View recovered files

etisalat  
total order by CategoryName

Order Quantity by SubcategoryName

Product Name total order return rate

Product Name	total order	return rate
Road-250 Red, 58	105	
Road-250 Black, 52	103	4.85 %
Road-250 Red, 48	86	1.16 %
Road-250 Black, 48	82	1.22 %
Road-250 Red, 52	80	2.50 %
Road-250 Black, 58	79	5.06 %
Mountain-200 Black, 46	78	1.28 %
Mountain-200 Black, 42	77	2.60 %
Road-250 Red, 44	69	1.45 %
Road-250 Black, 44	64	4.69 %
Mountain-200 Silver, 38	63	7.94 %
Mountain-200 Silver, 42	62	4.84 %
Mountain-200 Black, 38	50	
Mountain-200 Silver, 46	49	2.04 %
Road-150 Red, 52	46	4.35 %
Road-150 Red, 56	45	4.44 %
Road-150 Red, 48	38	
Road-150 Red, 62	37	
Road-150 Red, 44	36	
Road-550-W Yellow, 42	33	
Total	1735	2.77 %

Visualizations

Fields

Search

AW\_Calendar\_Lookup

Date

Day Name

Day of week

Month Name

month name\_CC

short day

Start of Month

Start of Week

Start of Year

weekend

Year

AW\_Customer\_Lookup

AW\_Product\_Categories\_L...

Drill through

Cross-report

Off

Keep all filters

On

Add drill-through fields here

Page 1 Page 2

PAGE 1 OF 2

UPDATE AVAILABLE (CLICK TO DOWNLOAD)

9:57 PM 9/20/2020

# Demo

---

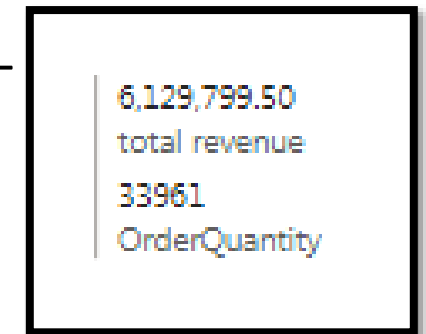
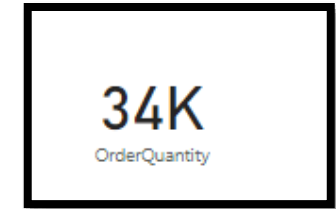
- Drag the slicer visual in the report view to the work space.
- Drag (Country) from territory table to the slicer field.
- Try to select any Country and see how it will work as a filter for all other visuals.
- Drag (date) to the field of slicer instead of (Country).
- Try to filter based on the date
- Go to format tab , check background option.
- Drag the year to the report/page level filter (try basic and advance option)
- Check sync slicer option from view tab



# Demo: Showing Key Metrics with Cards & KPI Visuals

- Drag (card) visual to the work space.
- Drag the (total revenue) to the card field
- Click on the (card) visual and try to change it simply by clicking on (multi-row-card) which will allow you to add another value to show (order Quantity).
- Click on the (KPI) visual to add to the workspace.
- Drag the (total revenue) to the (indicator field) and (start of the month) to the (trend axis).
- Go to format and change the title.

Note: To check the value let's use (matrix) visual and drag (total revenue) and (start of the month) to the matrix to check the values.

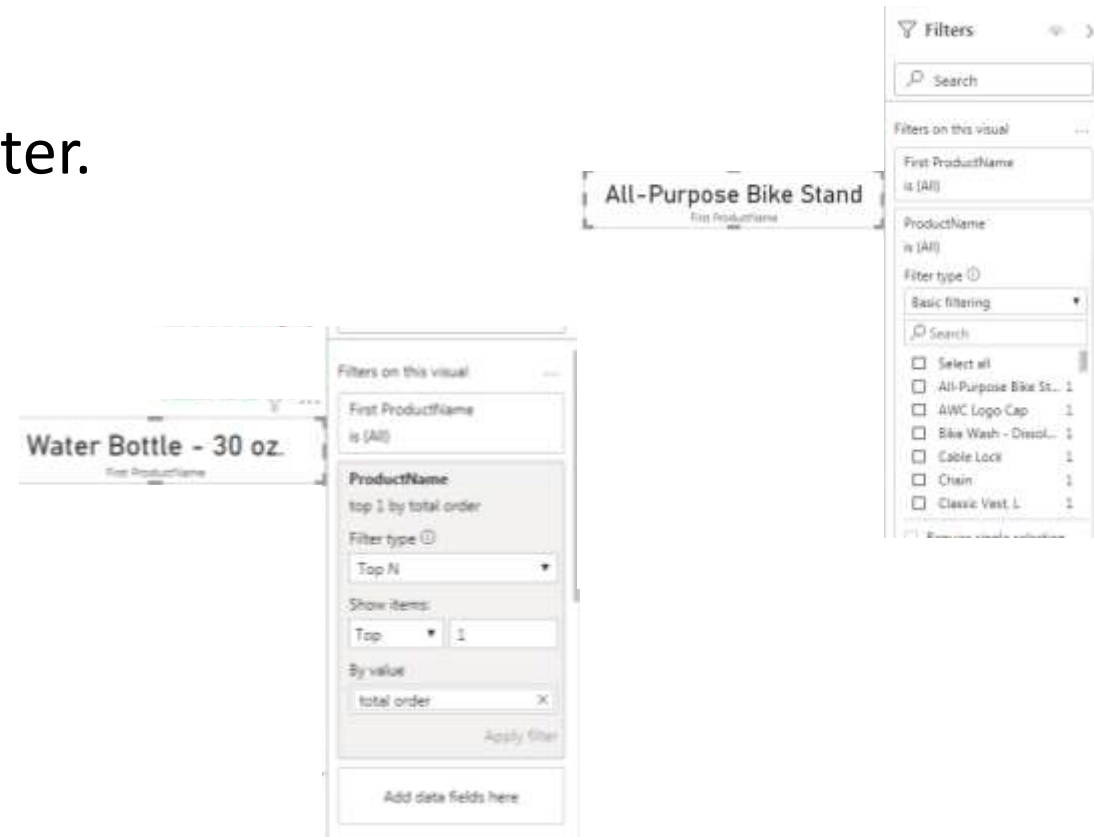
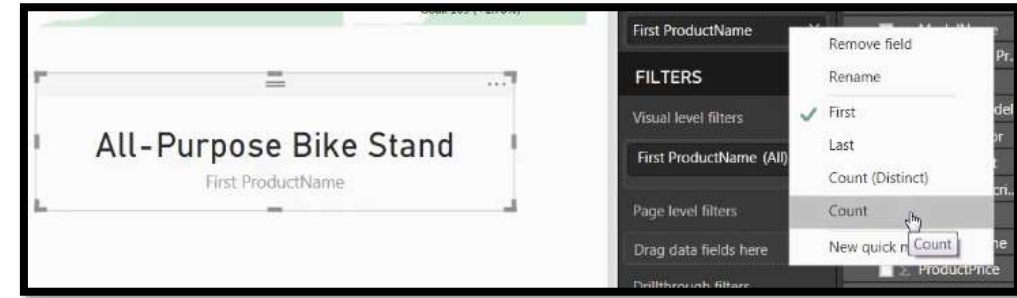


A KPI visual showing '1.83M' in a large, bold, black font. Below it is a line chart showing the trend of total revenue over time. The chart has a light gray background and a dark gray line. The data is as follows:

Start of Month	total revenue
01 March, 2016	471,962.02
01 April, 2016	494,957.58
01 May, 2016	545,535.02
01 June, 2016	533,825.21
01 July, 2016	815,355.86
01 August, 2016	804,191.63
01 September, 2016	952,741.88
01 October, 2016	1,029,819.30
01 November, 2016	1,133,911.38
01 December, 2016	1,635,306.67
01 January, 2017	1,274,377.03
01 February, 2017	1,339,239.38
01 March, 2017	1,448,594.08
01 April, 2017	1,527,811.82
01 May, 2017	1,768,430.63
01 June, 2017	1,826,984.91
Total	24,914,567.18

# Demo: Inserting Text Cards

- Drag (card) visual to the workspace.
- Drag product name to the card.
- Click on the card visual and go to visual filter.
- Drag product name again to the filter.
- Go to show item , insert (1).
- Drag (total orders) to the value field
- Go to format and write a title for the card visual



# Demo: Visualizing Geospatial Data with Maps

- Drag a basic (map) visual.
- Drag (country) to the (location) field
- Drag (order quantity) to the (size) field of the map visual.
- Add (total revenue) & (total profit) to the (tooltip) and you will see that more Data is been added to the map.

Note: you can try also (Filled map) & (ArcGIS).



price

ProductKey

Quantity ty...

return rate

revenue

StockDate

target

TerritoryKey

Total Cost

total order

total profit

total revenue

weekend or...

price

ProductKey

Quantity ty...

return rate

revenue

StockDate

target

TerritoryKey

Total Cost

total order

total profit

total revenue

weekend or...

price

ProductKey

Quantity ty...

return rate

revenue

StockDate

target

TerritoryKey

Total Cost

total order

total profit

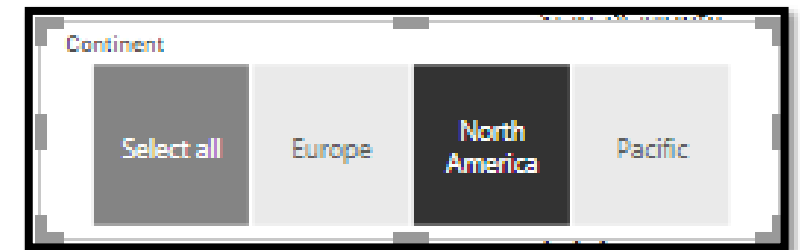
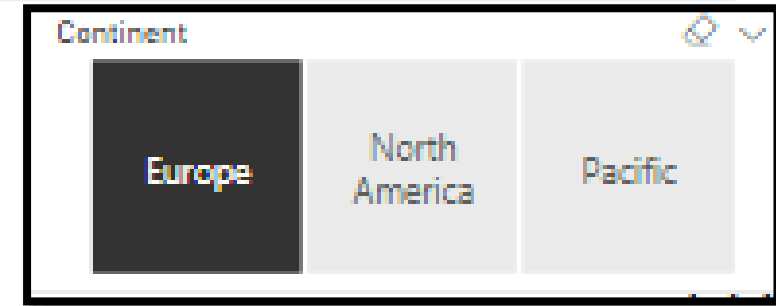
total revenue

weekend or...

# Demo (Cont)

---

- Drag the (Slicer) to the workspace.
- Drag the (Continent) to the (Field) of the Slicer.
- Go to Format → General → Orientation → Horizontal.
- Go to (selection control) → click on (show select all).

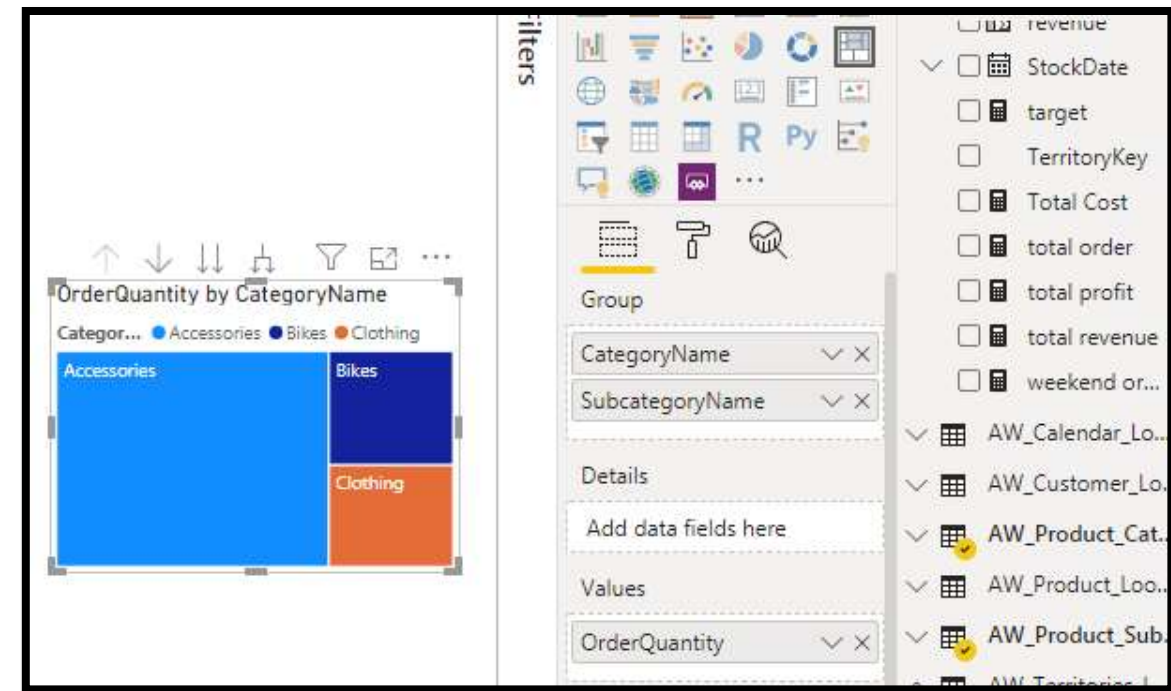
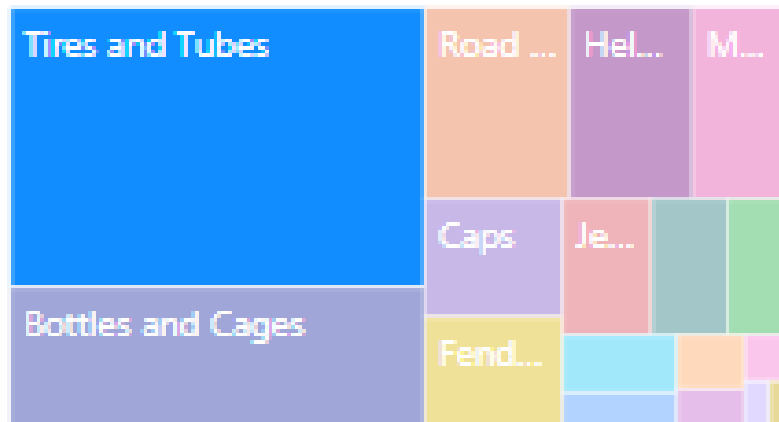


# Demo: Visualizing Data with Treemaps

- Drag tree map

OrderQuantity by SubcategoryName

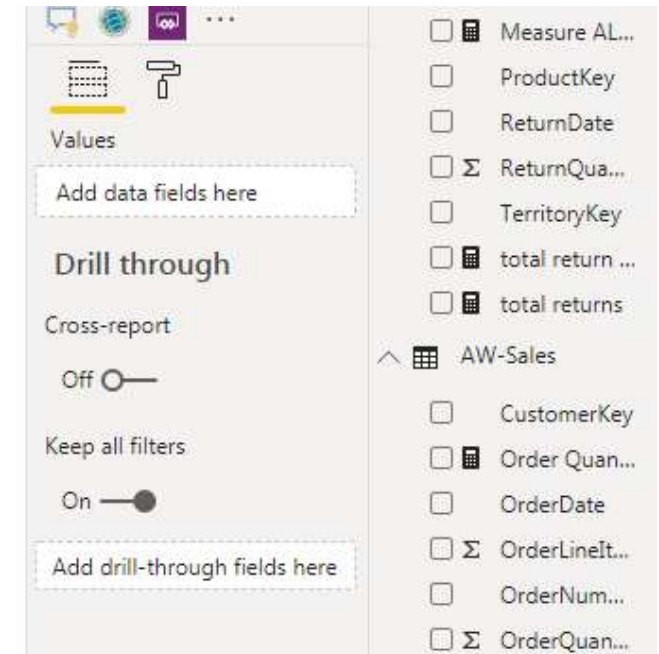
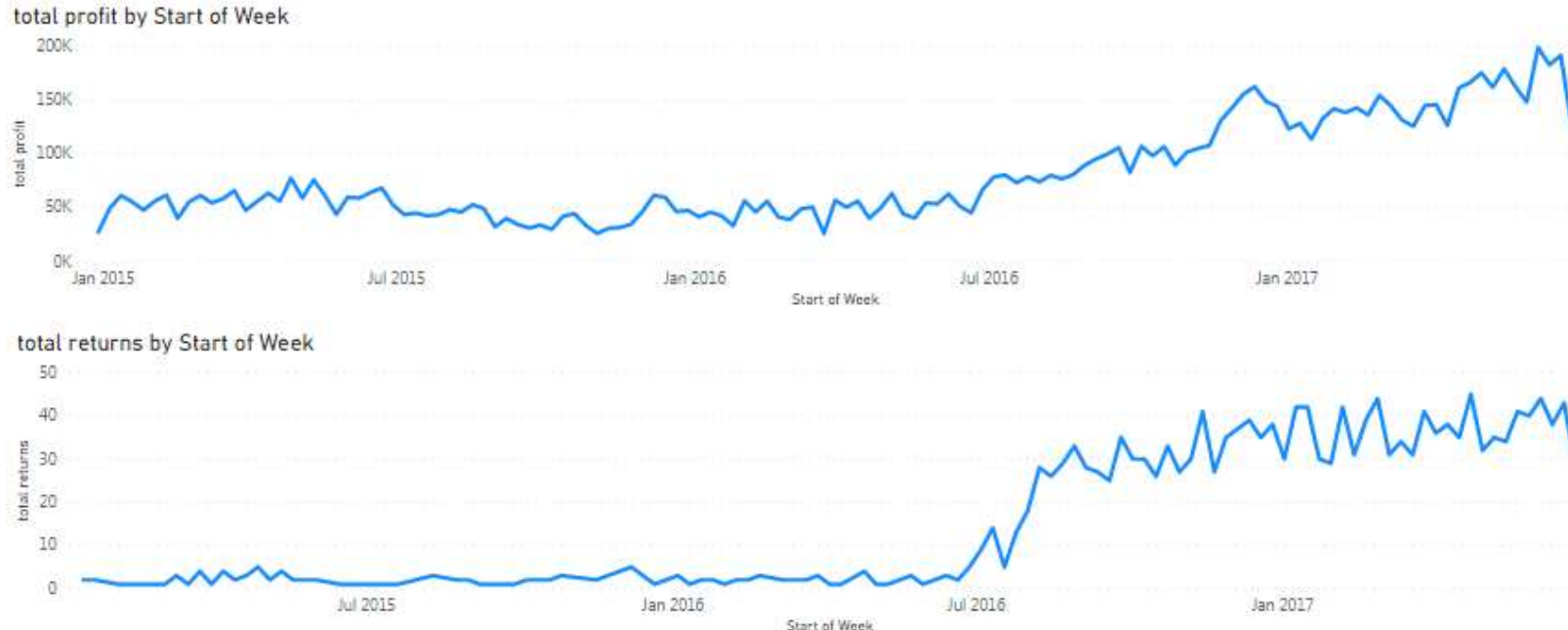
Subcate... ● Tires and Tubes ● Bottles and C...





# Demo: Showing Trends with Line & Area Charts

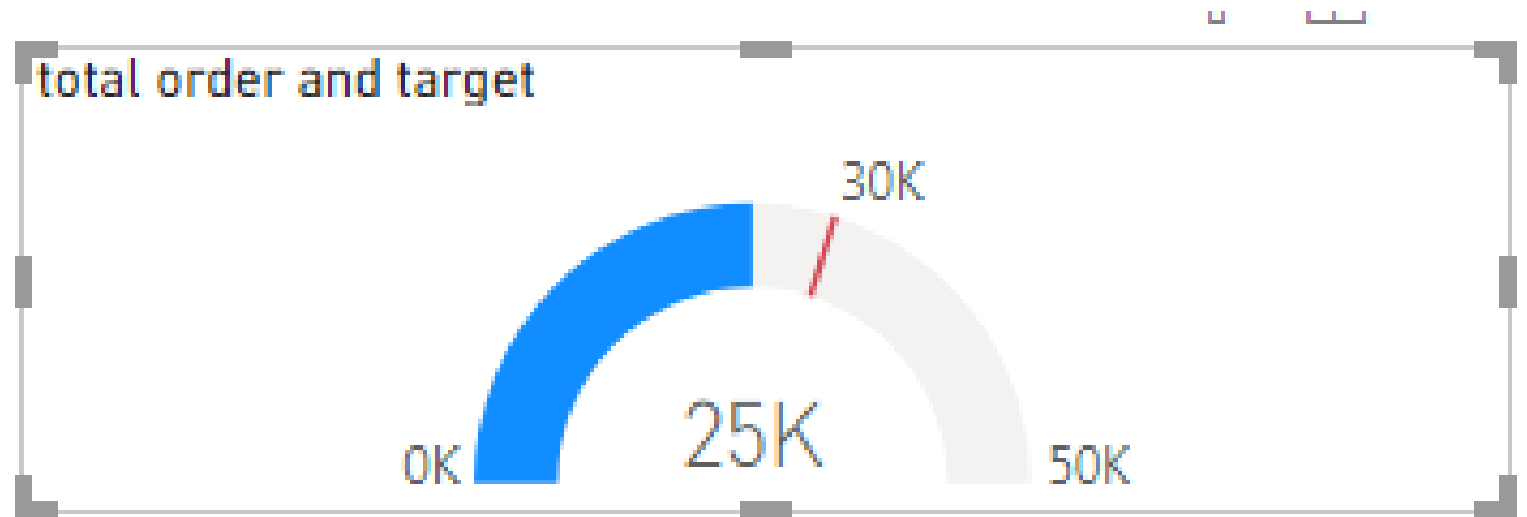
- Drag a line chart
- Drag the total profit to the value field and the start of the week to the Axis field.



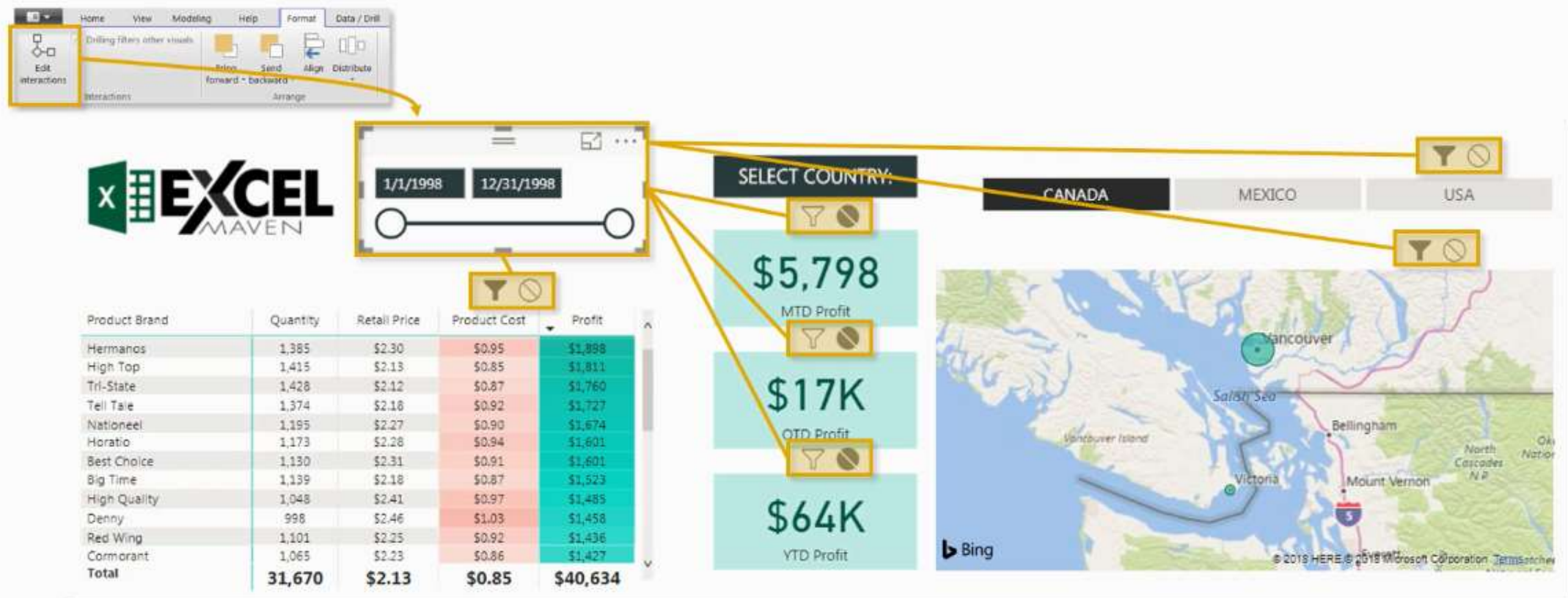
# Demo: Goal Pacing with Gauge Charts

---

- Drag the (Gauge) visual to the work space.
- Drag total order to the value field.
- Drag the (target) to the (target) field.



# EDITING REPORT INTERACTIONS



**Report interactions** allow you to determine how filters applied to *one* visual impact the *others*

- For example, by selecting the Timeline visual and enabling “*Edit interactions*” from the **Format** tab, we can manually determine which visuals should “*react*” when the date range changes
- In this case the **Product matrix**, **Country slicer** and **Map** will filter in response to timeline changes (🔿🕒), but the **MTD**, **QTD**, and **YTD Profit** cards *will not* (🔿🕒)

# EDITING REPORT INTERACTIONS (CONT.)

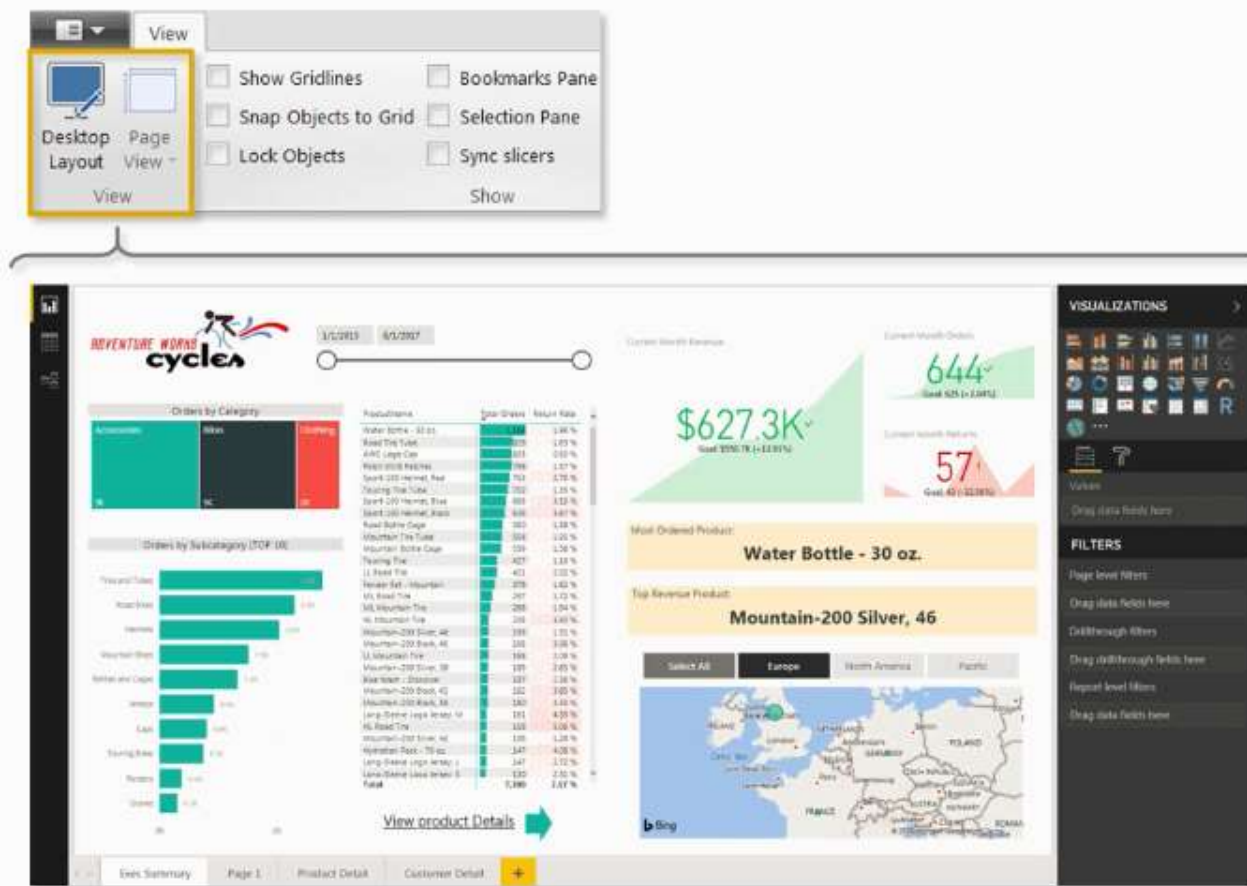


For certain types of visuals, a third option allows you to **"highlight"** sub-segments of the data, rather than simply filtering vs. not filtering

- When the interaction mode is set to **"filter"**, selecting the **"Bikes"** category in the treemap produces a filtered list of subcategories in the chart below
- When the interaction mode is set to **"highlight"**, selecting the **"Bikes"** category in the treemap highlights the relevant subsegment of data in the chart below



# DESKTOP VS. PHONE LAYOUT



Phone Layout view allows you to design on a canvas size optimized for mobile viewing (vs. *desktop*)

- **NOTE:** You can't actually build content within the Phone Layout view; recommend building in **Desktop Layout**, and assembling select visuals for mobile if you plan to share content via the Power BI app

View

☐ Show Gridlines ☐ Bookmarks Pane ☐ Field Properties

☐ Snap Objects to Grid ☐ Selection Pane

☐ Lock Objects ☐ Sync slicers

View Show



## VISUALIZATIONS

REMOVE ALL VISUALIZATIONS



# DATA VISUALIZATION BEST PRACTICES

---



## Strive for clarity & simplicity, above all else

- *Aim to maximize impact and minimize noise; it's all about balancing design and function*



## Don't just build charts and graphs; create a *narrative*

- *Without context, data is meaningless; use filters, bookmarks, and effective visualizations to translate raw data into powerful insights and implications*



## Always ask yourself the three key questions:

1. What **type of data** are you visualizing? (Integer, categorical, time-series, geo-spatial, etc)
2. What are you trying to **communicate**? (Relationships, compositions, trending, etc)
3. Who is the **end user** consuming this information? (Analyst, CEO, client, intern, etc)