

تحليل الأعمال والبيانات باستخدام Power BI

الجزء الأول

Business and Data analysis by using Power BI

Part 1





المحاضرة الأولى

تعرف على عالم تحليل الأعمال وربط وتشكيل البيانات

أجندة المحاضرة Lecture Agenda

• مقدمة عن تحليل البيانات والأعمال Introduction

- مقدمة وأسباب تميزه Power BI
- Why using Power BI

أجندة المحاضرة Lecture Agenda

- التعامل مع جميع أنواع البيانات
Deal With All Types Of Data
- جمع ودمج البيانات
Data Appending and Merging

لماذا Power BI هو الأفضل للاستخدام؟ Why Power BI?



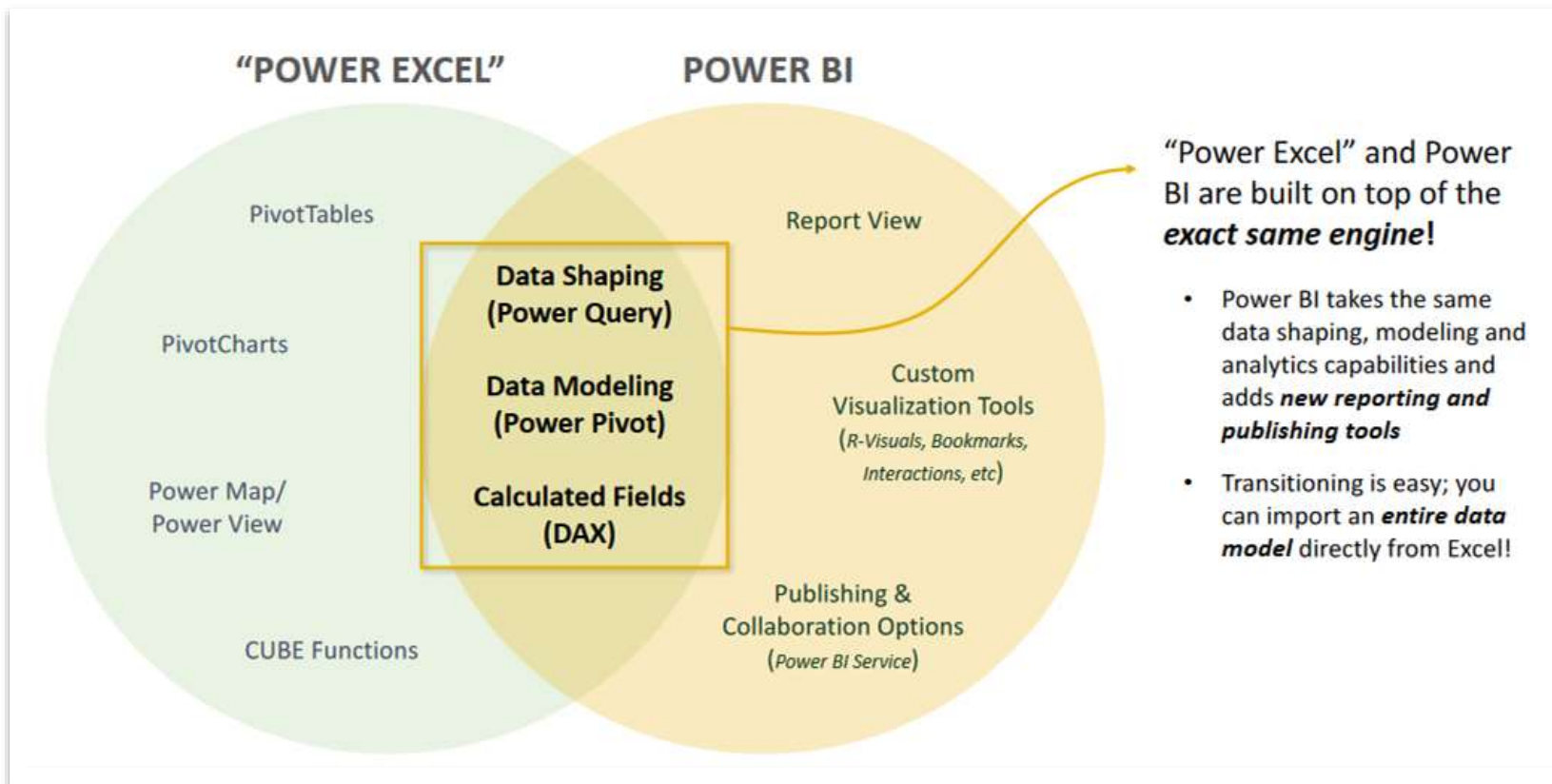
Power BI is a standalone Microsoft business intelligence product, which includes both desktop and web-based applications for loading, modeling, and visualizing data

الرحلة الرئيسية للـ Power BI

Power BI Main Journey

- Connect and transform the raw data
- Build a relational data model
- Create new calculated columns and DAX measures
- Design an interactive report to analyze and visualize the data

Power Excel Vs. Power BI



واجهة المستخدم لبرنامج Power BI

Power BI interface

Three Core Views:

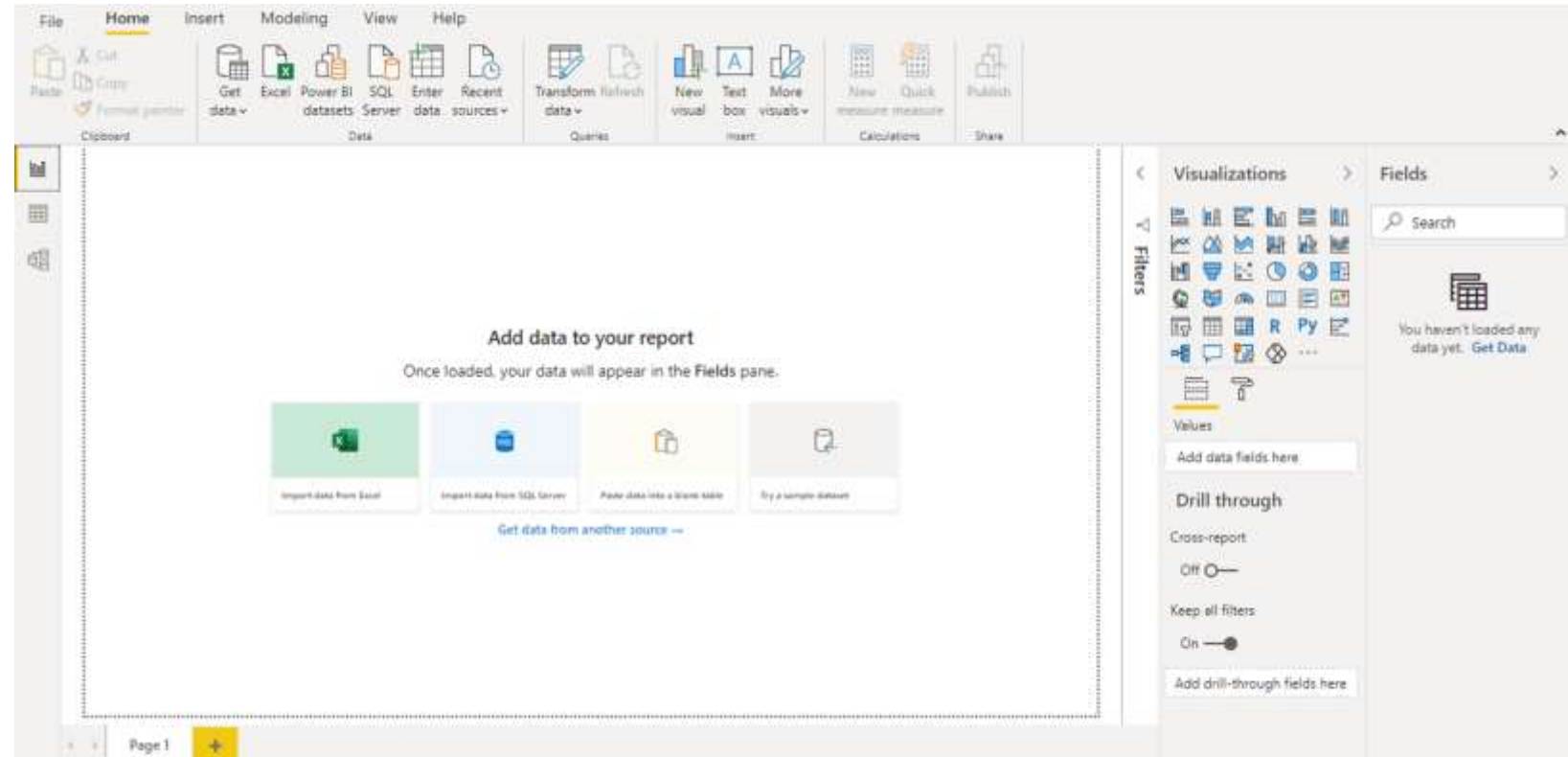
Report



Data

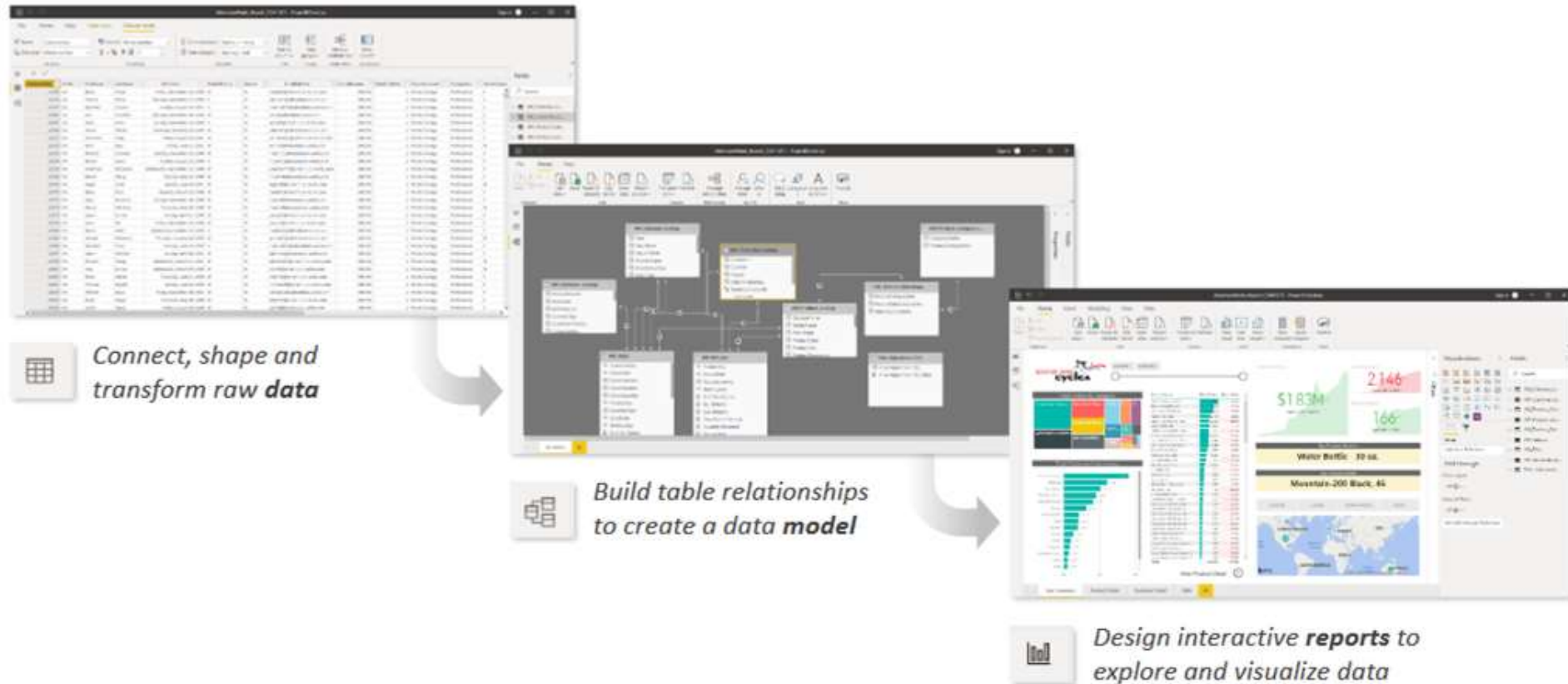


Relationships
(aka Model)



واجهة المستخدم لبرنامج Power BI

Power BI interface





التعامل مع جميع أنواع البيانات

Deal With All Types Of Data

Using Index Column

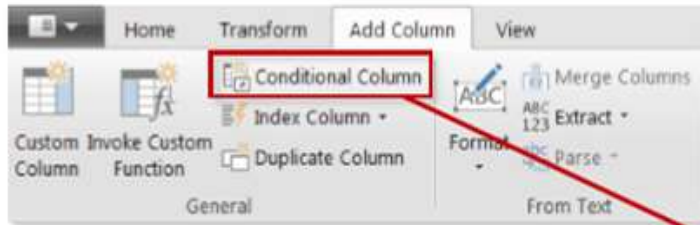


Index Columns contain a list of sequential values that can be used to identify each unique row in a table (*typically starting from 0 or 1*)

These columns are often used to create **unique IDs** that can be used to form relationships between tables (*more on that later!*)

	¹² ₃ Index	transaction_date	stock_date	¹² ₃ product_id	¹² ₃ customer_id	¹² ₃ promotion_id	¹² ₃ st
1	0	1/1/1997	12/28/1996	761	6613	0	
2	1	1/1/1997	12/30/1996	1435	8830	0	
3	2	1/1/1997	12/29/1996	1175	8830	0	
4	3	1/1/1997	12/30/1996	1152	8830	0	
5	4	1/1/1997	12/31/1996	1245	5005	0	
6	5	1/1/1997	12/27/1996	209	5005	0	
7	6	1/1/1997	12/28/1996	1345	5005	0	
8	7	1/1/1997	12/28/1996	1468	5005	0	
9	8	1/1/1997	12/26/1996	84	7962	0	
10	9	1/1/1997	12/30/1996	966	7962	0	
11	10	1/1/1997	12/27/1996	1022	7962	0	
12	11	1/1/1997	12/29/1996	440	7962	0	
13	12	1/4/1997	12/28/1996	151	2274	1054	
14	13	1/4/1997	12/28/1996	1287	8648	1054	
15	14	1/4/1997	12/30/1996	1264	8648	1054	
16	15	1/4/1997	12/31/1996	188	8648	1054	
17	16	1/4/1997	1/1/1997	1526	8648	1054	
18	17	1/4/1997	12/29/1996	518	8762	1054	
19	18	1/5/1997	12/31/1996	963	4018	0	
20	19	1/5/1997	12/29/1996	154	1418	0	
21							

Adding Conditional Column



Conditional Columns allow you to define new fields based on logical rules and conditions (*IF/THEN statements*)

In this case we're creating a new conditional column called "**Order Size**", which depends on the values in the "**quantity**" column, as follows:

- If quantity >5, Order Size = "**Large**"
- If quantity is **from 2-5**, Order Size = "**Medium**"
- If quantity =1, Order Size = "**Small**"
- Otherwise Order Size = "**Other**"

Add Conditional Column

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

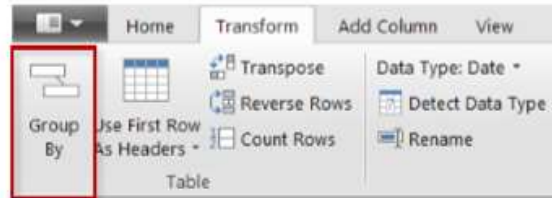
New column name
Order Size

	Column Name	Operator	Value (1)		Output (1)
If	quantity	is greater than	ABC 123 5	Then	ABC 123 Large
Else If	quantity	is greater than or...	ABC 123 2	Then	ABC 123 Medium
Else If	quantity	equals	ABC 123 1	Then	ABC 123 Small ...

Add Rule

Otherwise (1)
ABC 123 Other

Grouping and Advanced Grouping



Group By allows you to aggregate your data at a different level
(i.e. transform daily data into monthly, roll up transaction-level data by store, etc.)

transaction_date	stock_date	product_id	customer_id	store_id	quantity
1	8/11/2007	4	3443	2	4
2	8/11/2007	5	438	15	4
3	8/20/2007	6/15/2007	10140	17	3
4	1/1/2007	10/26/2006	4108	7	4
5	1/1/2007	1/1/2007	2768	2	2
6	11/26/2007	11/11/2007	2270	11	3
7	5/1/2007	4/26/2007	2322	2	3
8	8/16/2007	8/16/2007	8632	24	2
9	8/11/2007	8/11/2007	4608	17	3
10	11/11/2007	11/6/2007	8081	8	4
11	11/21/2007	11/11/2007	4767	11	4
12	8/16/2007	8/11/2007	4248	24	4
13	8/1/2007	8/1/2007	1363	24	3
14	11/26/2007	11/11/2007	117	21	4
15	8/11/2007	8/11/2007	3807	8	4
16	4/1/2007	4/2/2007	818	7	4
17	1/11/2007	1/6/2007	9188	19	3
18	11/11/2007	11/22/2007	4010	8	3
19	1/11/2007	1/16/2007	433	18	3
20	5/11/2007	5/9/2007	8189	23	4
21	10/6/2007	10/1/2007	3528	17	3
22	8/16/2007	8/16/2007	3829	19	3
23	4/16/2007	4/16/2007	4461	11	3

Group By

Specify the columns to group by.

Group by

product_id

Add grouping

New column name

Total_Quantity

Add aggregation

Operation

Sum

Column

quantity

OK Cancel

product_id	Total_Quantity
1	4
2	11
3	12
4	14
5	18
6	23
7	48
8	50
9	56
10	59
11	61
12	75
13	89
14	90
15	112
16	115
17	119
18	120

Merge Quires

Product ID	Sales
1	122322
2	221234
3	135335

Product ID	Product name
1	Book
2	Bike
3	Chair

Product ID	Product name	Sales
1	Book	122322
2	Bike	221234
3	Chair	135335

Append Quires

Product ID	Sales	Return
1	232344	233

+

Product ID	Sales	Return
2	43344	21

+

Product ID	Sales	Return
3	53433	433

+

Product ID	Sales	Return
4	5432	133

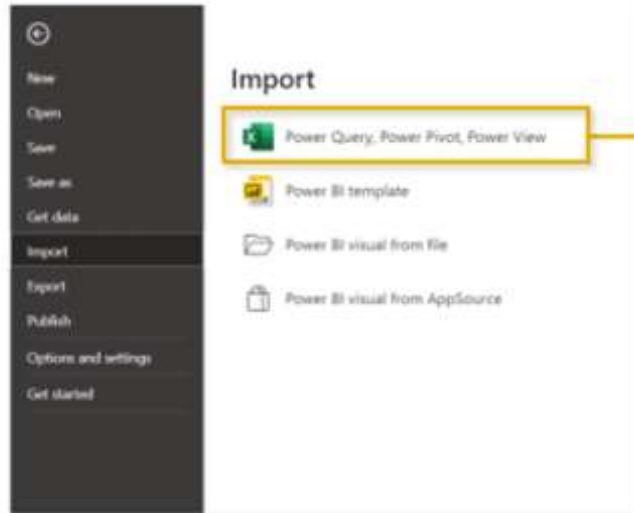
=

Product ID	Sales	Return
1	232344	233
2	43344	21
3	53433	433
4	5432	133

Tips : Appending add rows to the table
Vs. Merge add column to the table
Taller Vs Wider

إضافة ملفات الاكسيل الجاهزة مباشرة لـ Power BI

Import Excel Model



Already have a fully-built model in Excel?

Import models built in Excel directly into Power BI Desktop using ***Import > Power Query, Power Pivot, Power View****

Imported models retain the following:

- Data source connections and queries
- Query editing procedures and applied steps
- Table relationships, hierarchies, field settings, etc.
- All calculated columns and DAX measures



المحاضرة الثانية

الـ Data Modeling وإنشاء العلاقات بين البيانات

أجندة المحاضرة Lecture Agenda

- مفهوم العلاقات بين الجداول
Relationships concept
- أنواع الـ Relationships المختلفة
Relationships Types

أجندة المحاضرة Lecture Agenda

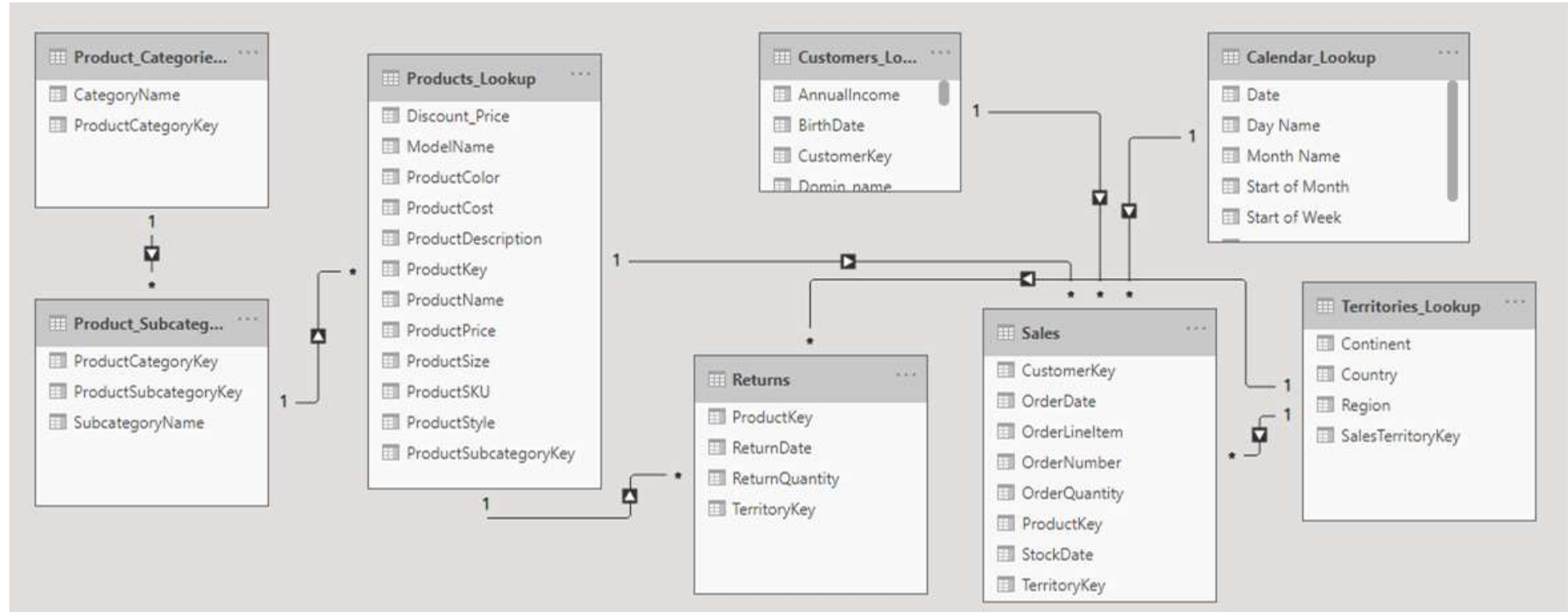
- مفهوم تدفق البيانات والفلاتر

Filter Flow Concept

- الاستفادة من العلاقات للوصول إلى معلومات أعمق وأكثر دقة

Take advantage of Related Data

Normalization Concept



Data Table Vs. Lookup Table

date	product_id	quantity
1/1/1997	869	5
1/1/1997	1472	3
1/1/1997	76	4
1/1/1997	320	3
1/1/1997	4	4
1/1/1997	952	4
1/1/1997	1222	4
1/1/1997	517	4
1/1/1997	1359	4
1/1/1997	357	4
1/1/1997	1426	5
1/1/1997	190	4
1/1/1997	367	4
1/1/1997	250	5
1/1/1997	600	4
1/1/1997	702	5

date	day_of_month	month	year	weekday	week_of_year	week_ending	month_name	quarter
1/1/1997	1	1	1997	Wednesday	1	1/5/1997	January	Q1
1/2/1997	2	1	1997	Thursday	1	1/5/1997	January	Q1
1/3/1997	3	1	1997	Friday	1	1/5/1997	January	Q1
1/4/1997	4	1	1997	Saturday	1	1/5/1997	January	Q1
1/5/1997	5	1	1997	Sunday	2	1/5/1997	January	Q1
1/6/1997	6	1	1997	Monday	2	1/12/1997	January	Q1

This Calendar Lookup table provides additional attributes about each date (month, year, weekday, quarter, etc.)

product_id	product_brand	product_name	product_sku	product_retail_price	product_cost	product_weight
1	Washington	Washington Berry Juice	90748583674	2.85	0.94	8.39
2	Washington	Washington Mango Drink	96516502499	0.74	0.26	7.42
3	Washington	Washington Strawberry Drink	58427771925	0.83	0.4	13.1
4	Washington	Washington Cream Soda	64412155747	3.64	1.64	10.6
5	Washington	Washington Diet Soda	85561191439	2.19	0.77	6.66
6	Washington	Washington Cola	29804642796	1.15	0.37	15.8
7	Washington	Washington Diet Cola	20191444754	2.61	0.91	18
8	Washington	Washington Orange Juice	89770532250	2.59	0.8	8.97

This Product Lookup table provides additional attributes about each product (brand, product name, sku, price, etc.)

Primary Vs. Foreign Keys

Primary key

Primary key

	customer_id	customer_acct_num	first_name
1	213	89905924797	David
2	227	90065194368	Michael
3	246	90350758433	Rita
4	255	90442248582	Jacky
5	258	90463679432	John
6	273	90642099976	Jennifer
7	289	90872421300	Bertha
8	294	90917686200	Michelle
9	311	91123164858	Janice
10	313	91158838272	Ole
11	331	91323767304	Lanna
12	390	92005987200	Paul
13	394	92050951719	Jared

	product_id	Product_Name	Therapy_area
1	1	Rani	Gastrointestinal...
2	2	Depovit B12	Vitamins & Sup...
3	3	Gratisovir	Respiratory
4	4	Dexamethasone	Hormones and ...
5	5	Oplex	Hormones and ...
6	6	Cefaxone	Anti-infectives
7	7	Spasmodigestin	Gastrointestinal...
8	8	Ketolac	Gynaecology

	transaction_date	stock_date	product_id	customer_id	store_id	quantity
1	10/23/1998 12:00:0...	10/20/1998 1...	14	1814	19	3
2	10/23/1998 12:00:0...	10/19/1998 1...	46	5326	19	3
3	10/23/1998 12:00:0...	10/22/1998 1...	67	8236	19	3
4	10/23/1998 12:00:0...	10/20/1998 1...	73	10121	19	3
5	10/23/1998 12:00:0...	10/17/1998 1...	77	8236	19	3
6	10/23/1998 12:00:0...	10/17/1998 1...	84	10121	19	3
7	10/23/1998 12:00:0...	10/21/1998 1...	86	6770	19	3

Foreign keys

Merge Vs. Relationships

Original Fact Table fields			Attributes from Calendar Lookup table						Attributes from Product Lookup table			
date	product_id	quantity	day_of_month	month	year	weekday	month_name	quarter	product_brand	product_name	product_sku	product_weight
1/1/1997	869	5	1	1	1997	Wednesday	January	Q1	Nationeel	Nationeel Grape Fruit Roll	52382137179	17
1/7/1997	869	2	7	1	1997	Tuesday	January	Q1	Nationeel	Nationeel Grape Fruit Roll	52382137179	17
1/3/1997	1	4	3	1	1997	Friday	January	Q1	Washington	Washington Berry Juice	90748583674	8.39
1/1/1997	1472	3	1	1	1997	Wednesday	January	Q1	Fort West	Fort West Fudge Cookies	37276054024	8.28
1/6/1997	1472	2	6	1	1997	Monday	January	Q1	Fort West	Fort West Fudge Cookies	37276054024	8.28
1/5/1997	2	4	5	1	1997	Sunday	January	Q1	Washington	Washington Mango Drink	96516502499	7.42
1/1/1997	76	4	1	1	1997	Wednesday	January	Q1	Red Spade	Red Spade Sliced Chicken	62054644227	18.1
1/1/1997	76	2	1	1	1997	Wednesday	January	Q1	Red Spade	Red Spade Sliced Chicken	62054644227	18.1
1/5/1997	3	2	5	1	1997	Sunday	January	Q1	Washington	Washington Strawberry Drink	58427771925	13.1
1/7/1997	3	2	7	1	1997	Tuesday	January	Q1	Washington	Washington Strawberry Drink	58427771925	13.1
1/1/1997	320	3	1	1	1997	Wednesday	January	Q1	Excellent	Excellent Cranberry Juice	36570182442	16.4

Sure, but **it's extremely inefficient.**

- Merging data in this way creates **redundant data** and utilizes **significantly more memory and processing power** than creating relationships between multiple small tables

Relationship types

Emp_id	Emp_name
1	Samy
2	Hany
3	Hosam
4	Ahmed

One to One Relationship

1 ←→ 1

Emp_id	Salary
1	5500
2	4500
3	3455
4	7500

Emp_id	Emp_name	salary
1	Samy	5500
2	Hany	4500
3	Hosam	3455
4	Ahmed	7500

One to Many Relationship

Primary key

	customer_id	customer_acct_num	first_name
1	213	89905924797	David
2	227	90065194368	Michael
3	246	90350758433	Rita
4	255	90442248582	Jacky
5	258	90463679432	John
6	273	90642099976	Jennifer
7	289	90872421300	Bertha
8	294	90917686200	Michelle
9	311	91122164892	Janice
10	313	91158838272	Ole
11	331	91323767304	Lana
12	390	92005987200	Paul
13	394	92050951719	Jared

Primary key

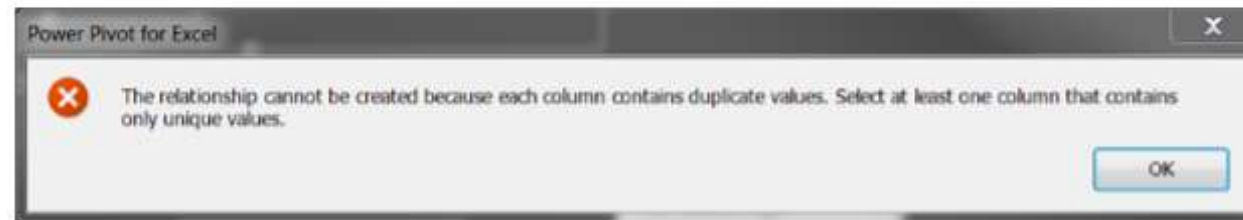
	product_id	Product_Name	Therapy_area
1	1	Rani	Gastrointestinal...
2	2	Depovit B12	Vitamins & Sup...
3	3	Gratisovir	Respiratory
4	4	Dexamethasone	Hormones and ...
5	5	Oplex	Hormones and ...
6	6	Cefaxone	Anti-infectives
7	7	Spasmodigestin	Gastrointestinal...
8	8	Ketolac	Gynaecology

Foreign keys

	transaction_date	stock_date	product_id	customer_id	store_id	quantity
1	10/23/1998 12:00:0...	10/20/1998 1...	14	1814	19	3
2	10/23/1998 12:00:0...	10/19/1998 1...	46	5326	19	3
3	10/23/1998 12:00:0...	10/22/1998 1...	67	8236	19	3
4	10/23/1998 12:00:0...	10/20/1998 1...	73	10121	19	3
5	10/23/1998 12:00:0...	10/17/1998 1...	77	8236	19	3
6	10/23/1998 12:00:0...	10/17/1998 1...	84	10121	19	3
7	10/23/1998 12:00:0...	10/21/1998 1...	86	6770	19	3

Many to many relationship

product_id	product_name	product_sku	date	product_id	transactions
4	Washington Cream Soda	64412155747	1/1/2017	4	12
4	Washington Diet Cream Soda	81727382373	1/2/2017	4	9
5	Washington Diet Soda	85561191439	1/3/2017	4	11
7	Washington Diet Cola	20191444754	1/1/2017	5	16
8	Washington Orange Juice	89770532250	1/2/2017	5	19
			1/1/2017	7	11





مقدمة عن الـ DAX

Data analysis Expressions

هي لغة نستخدمها لنستطيع عمل عمليات حسابية أكثر تعقيداً ومتقدمة لكي توصلنا لتفاصيل أعلى من الـ بيزنس الخاص بنا.

وتنقسم إلى :

- Calculated Column
- Measures

Calculated Columns

• الأعمدة الحسابية تقوم بإضافة قيم جديدة لكل صف من صفوف الجدول كناتج لعملية حسابية معينة

- Calculated columns generate values for each row, which are visible within tables in the Data view

• الأعمدة الحسابية تعمل على مستوى الصف، وليس التجميعية الكاملة برقم فردي

- Calculated columns understand row context; they're great for defining properties based on information in each row, but generally useless for aggregation (SUM, COUNT, etc)

Calculated Columns

• قم باستخدام الأعمدة الحسابية عندما ترغب في إضافة معلومات ثابتة بشكل دائم مع محتوى الجدول

- Use calculated columns when you want to “stamp” static, fixed values to each row in a table (or use the Query Editor!)

• لا تستخدم الأعمدة الحسابية عندما ترغب في عمل عمليات حسابية، أو جمع متوسط للقيم أو أي قيم تريد تحليلها رقمياً كقيمة في المستقبل

- DO NOT use calculated columns for aggregation formulas, or to calculate fields for the “Values” area of a visualization (use measures instead)

Measures

- تأثير القياسات Measures يمتد على جميع جداول الأعمدة مثل الأعمدة الحسابية على عكس الأعمدة الحسابية فإن ال Measures لا تظهر داخل جدول البيانات، فالمكان الوحيد التي يمكن استخدامها به هو ال Values داخل ال Pivot Table
- Like calculated columns, measures reference entire tables or columns
- على نقيض الأعمدة الحسابية -تلك القيم غير المرئية في الجداول يمكن فقط رؤيتها عن طريق صورة مثل رسم بياني
- Unlike calculated columns, measure values aren't visible within tables; they can only be "seen" within a visualization like a chart or matrix

Measures

• ال Measure تقوم بتغيير القيم الخاصة بها بناء على محتوى الفلتر المستخدم مع البيانات، وهذا يعني أن ال Measure تعيد العمليات الحسابية بمجرد تغير الفلتر لإعطاء نتائج مختلفة مطابقة ومناسبة للفلتر الجديد

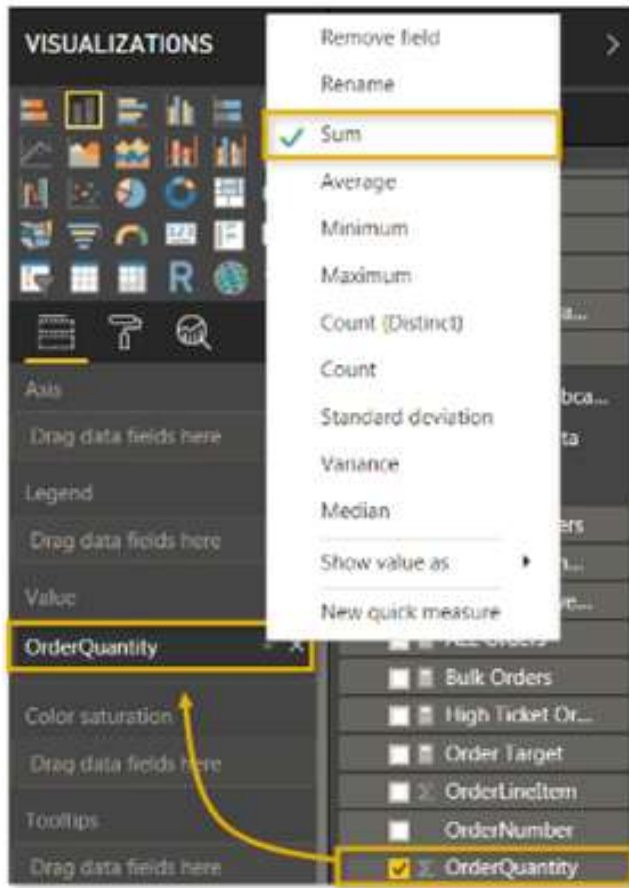
- Measures are evaluated based on filter context, which means they recalculate when the fields or filters around them change (like when new row or column labels are pulled into a matrix or when new filters are applied to a report)

Measures

• قم باستخدام ال Measures وليس الأعمدة الحسابية عندما تكون المعلومة غير ملائمة للسكن داخل صفوف الجدول، أو بمعنى آخر عندما ترغب في جمع أو إعطاء متوسط، وغيرها من العمليات الجمعية الشاملة

- Use measures (vs. calculated columns) when a single row can't give you the answer (in other words, when you need to aggregate)

Implicit Vs Explicit Measures



- هو إضافة حقل حسابي إلى منطقة القيم ليتم حسابه
- Implicit measures are created when you drag raw numerical fields (like “Order Quantity”) into the values pane of a visual and manually select the aggregation mode (Sum, Average, Min/Max, etc)
- هو إنشاء عملية حسابية متغيرة بتغير المعطيات
- Explicit measures are created by actually entering DAX functions (or adding “quick measures”) to define calculated columns or measures

Understanding Filter Context

ProductName	Total Orders	Return Rate
Water Bottle - 30 oz.	1164	1.96 %
Road Tire Tube	929	1.63 %
AWC Logo Cap	903	0.93 %
Patch Kit/8 Patches	798	1.57 %
Sport-100 Helmet, Red	755	2.79 %
Touring Tire Tube	702	1.35 %
Sport-100 Helmet, Blue	668	3.15 %
Sport-100 Helmet, Black	626	3.67 %
Road Bottle Cage	560	1.58 %
Mountain Tire Tube	554	1.95 %
Mountain Bottle Cage	539	1.38 %
Touring Tire	427	1.16 %
LL Road Tire	421	2.02 %
Fender Set - Mountain	378	1.82 %
ML Road Tire	297	1.72 %
ML Mountain Tire	266	1.94 %
HL Mountain Tire	206	3.40 %
Mountain-200 Silver, 46	199	1.11 %
Mountain-200 Black, 46	195	3.06 %
LL Mountain Tire	195	2.09 %
Mountain-200 Silver, 38	189	2.65 %
Bike Wash - Dissolver	187	2.38 %
Mountain-200 Black, 42	182	3.85 %
Mountain-200 Black, 38	180	3.33 %
Long-Sleeve Logo Jersey, M	161	4.35 %
HL Road Tire	158	5.06 %
Mountain-200 Silver, 42	155	1.28 %
Hydration Pack - 30 oz.	147	4.08 %
Long-Sleeve Logo Jersey, L	147	2.72 %
Long-Sleeve Logo Jersey, S	130	2.31 %
Total	7,380	2.17 %

Product name =
Touring Tire tube

Remove all Filters
and just Give me
the Total Orders

Remember that measures are evaluated based on filter context, which means that they recalculate whenever the fields or filters around them change

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

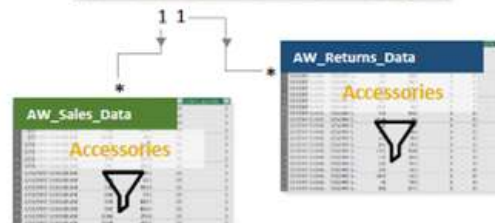
Product[CategoryName] = "Accessories"

Product Table	
Accessories	

STEP 2

Filters flow "downstream" to all related tables

Product Table	
Accessories	



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 Operations

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" }

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)

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!

Text Functions

LEN()	Returns the number of characters in a string	= LEN (Text)	<i>Note: Use the & operator as a shortcut, or to combine more than two strings!</i>
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)	

ما تم شرحه بالكورس

- مقدمة عن تحليل البيانات والأعمال Introduction

- مقدمة وأسباب تميزه Power BI
- Why using Power BI

- التعامل مع جميع أنواع البيانات Deal With All Types Of Data

ما تم شرحه بالكورس

- جمع ودمج البيانات

Data Appending and Merging

- مفهوم العلاقات بين الجداول

Relationships concept

- أنواع الـ Relationships المختلفة

Relationships Types

ما تم شرحه بالكورس

- مفهوم تدفق البيانات والفلاتر

Filter Flow Concept

- الاستفادة من العلاقات للوصول إلى معلومات أعمق وأكثر دقة

Take advantage of Related Data

- مقدمة عن الـ DAX

Introduction to Data analysis Expressions

شكراً لكم