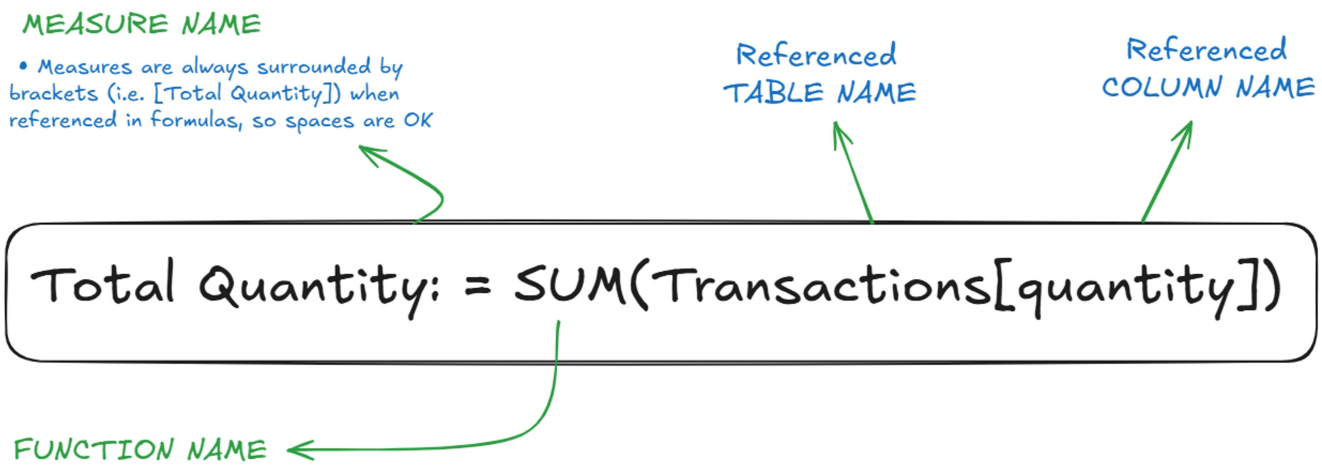
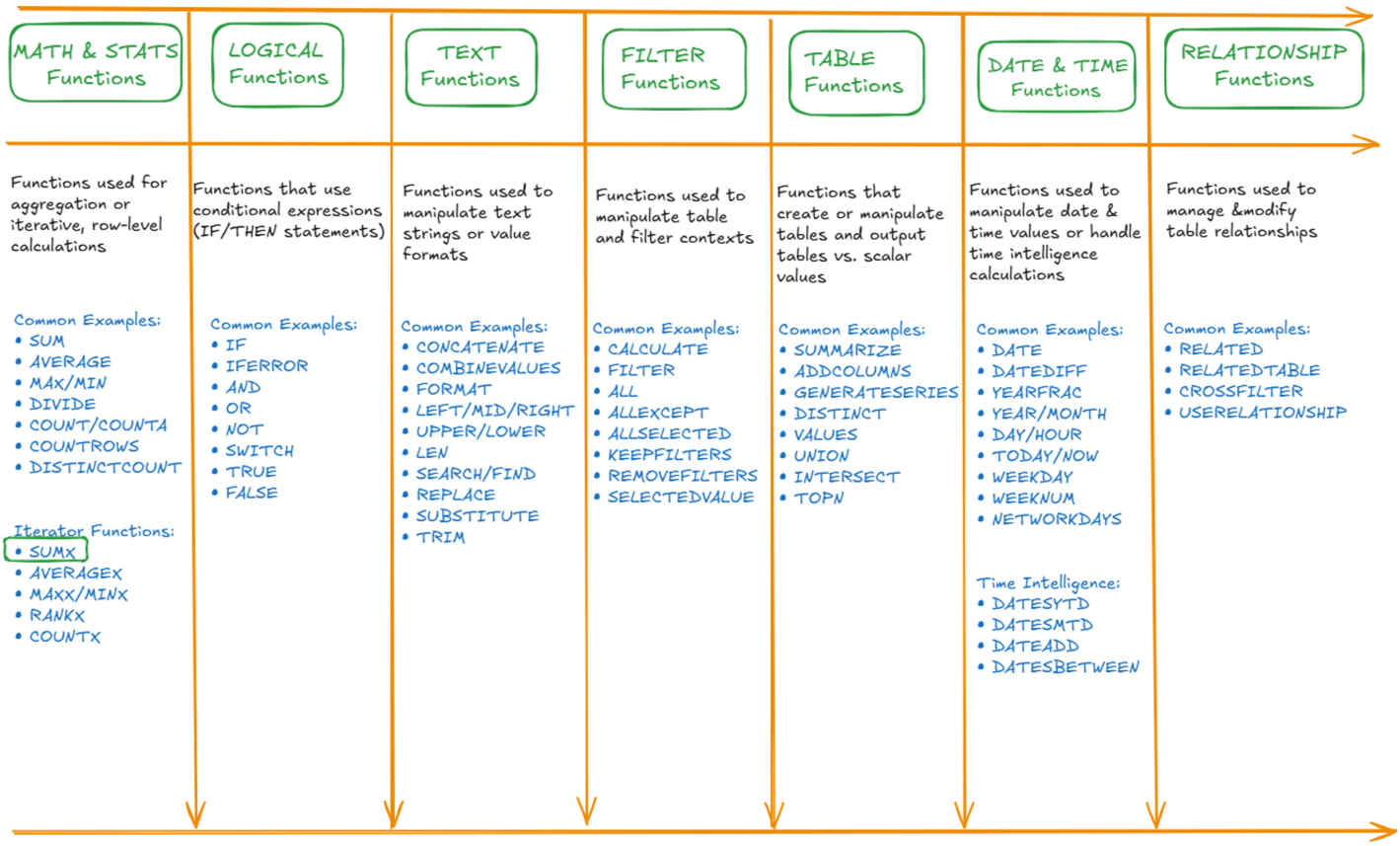
CN- Lec8- DAX Functions - p2

DAX Functions - p2

DAX SYNTAX



COMMON FUNCTION CATEGORIES



setA = {1,2,3,4,5} SetB = {3,4,5,6,7,8}
Union = {1,2,3,4,5,6,7,8} , Intersect = {3,4,5}

Department	Salary
HR	20K
Tech	90K
Sales	35K
HR	15K
Tech	95K
Sales	45K
Operation	25K
Marketing	50K
IT	80K
Sales	25K
IT	100K

Filter

Department	Salary	ALL	% of Cont
HR	35K	580	35/580
Tech	185K	580	
Sales	105K	580	
Operation	25K	580	
Marketing	50K	580	
IT	180K	580	

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(ColumnNameOrScalar1 , [Scalar2])
MIN	Returns the smallest value in a column or between two scalar expressions	=MIN(ColumnNameOrScalar1 , [Scalar2])
DIVIDE	Performs division and returns the alternate result (or blank) if DIV/0	=DIVIDE(Numerator, Denominator, [AlternateResult])

COUNTING FUNCTIONS

COUNT	Counts the number of non-empty cells in a column(excluding Boolean values)	=COUNT(ColumnName)
COUNTA	Counts the number of non-empty cells in a column (including Boolean values)	=COUNTA(ColumnName)
DISTINCT COUNT	Counts the number of distinct values in a column	=DISTINCTCOUNT(Column Name)
COUNTROWS	Counts the number of rows in the specified table, or a table defined by an expression	=COUNTROWS([Table])

BASIC LOGICAL FUNCTIONS

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 it returns an error, otherwise returns the expression itself	=IFERROR(Value, ValueIfError)
SWITCH	Evaluates an expression against a list of values and returns one of multiple possible expressions	=SWITCH(Expression, Value1, Result1, ..., [Else])
AND	Checks whether both arguments are TRUE to return TRUE, otherwise returns FALSE	=AND(Logical1, Logical2)
OR	Checks whether any argument is TRUE to return TRUE, otherwise returns FALSE	=OR(Logical1, Logical2)

Note: Use the && and || operators to include more than two conditions

Sales Category

Example 1: In our dataset, if we want to classify customers into High and Low Sales Value categories, we can use the following conditions: Price of item is > 300 and Quantity Ordered >= 10.

```
Sales Category = IF('Sales DAX'[Price of Item] > 300 && 'Sales DAX'[Quantity Ordered] >= 10, "High Sales", "Low Sales")
```

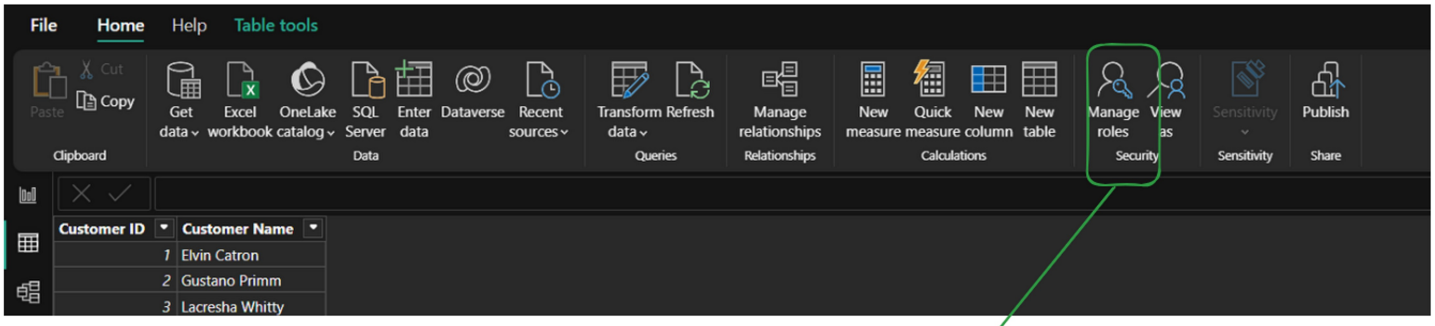
Example 2: From our dataset, for an advertisement campaign we do not want to target people who are either below 20 years of age or whose region is East or West, then we can use the conditions below.

```
Target Customer = IF('Sales DAX'[Customer Age] < 20 || 'Sales DAX'[Customer Region] IN {"East","West"},"Not a Target Customer", "Target Customer")
```

Customer Region	Revenue	East Revenue	North Revenue
		$1,85,011.999	$1,43,889.9995
East	$1,85,011.999	$1,85,011.999	$1,43,889.9995
North	$1,43,889.9995	$1,85,011.999	$1,43,889.9995
South	$1,14,829.001	$1,85,011.999	$1,43,889.9995
West	$64,950.0021	$1,85,011.999	$1,43,889.9995
Total	**$5,08,681.0016**	**$1,85,011.999**	**$1,43,889.9995**

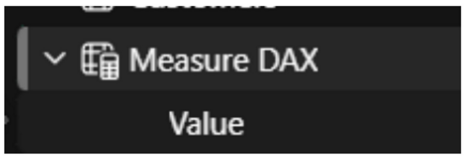
Creating a Measure Table

1st Option - Table View / Data View :

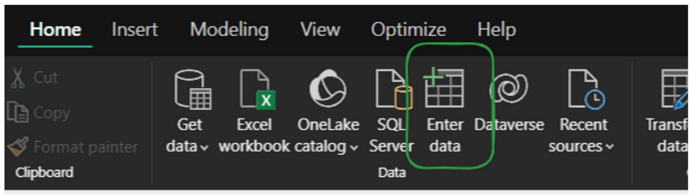


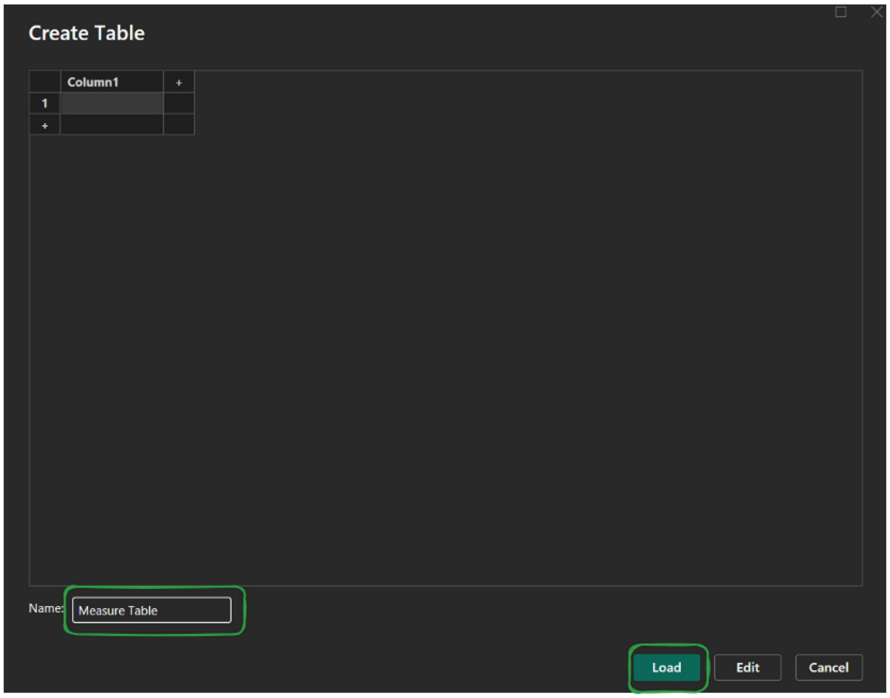

```
1 Measure DAX = {""}
```

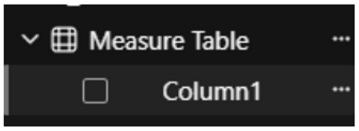
Measure Table = {""}

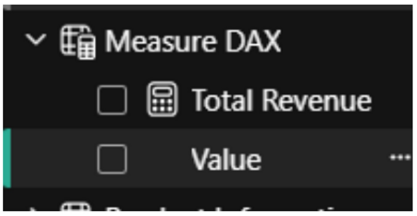


2nd Option : Report View










```
Total Revenue = SUM('Sales DAX'[Sales])
```

SUMX → Iterative Functions

Customer ID	Customer Region	Customer Age	Quantity Ordered	Price of Item
21	North	10	24	$272
22	North	23	19	$419
23	South	56	30	$34
24	East	38	12	$485
25	East	22	14	$404
26	West	22	7	$459
27	West	58	28	$297
28	East	11	15	$238

(24 * 272) + (19 * 419) + (30 * 34) + ...

26	West	22	7	$459
27	West	58	28	$297
28	East	11	15	$238
29	East	43	5	$1,492
30	North	45	19	$85
31	North	14	15	$405
32	South	27	3	$3,022
33	East	57	17	$260
34	East	17	26	$33
35	East	53	24	$154
36	East	59	3	$1,958
37	North	49	29	$151
38	North	33	23	$416
39	South	38	10	$232
40	East	59	12	$407
41	North	18	4	$1,539
42	North	36	28	$334
43	South	31	21	$121
44	East	50	3	$399

Sales

Time Complexity
Space Complexity

```
Total Revenue = SUMX('Sales DAX','Sales DAX'[Quantity Ordered] * 'Sales DAX'[Price of Item])
```

Report view

```
Calendar Table = CALENDAR("01/01/2022", TODAY())
```

Date	Year
Saturday, 1 January, 2022	2022
Sunday, 2 January, 2022	2022
Monday, 3 January, 2022	2022
Tuesday, 4 January, 2022	2022
Wednesday, 5 January, 2022	2022
Thursday, 6 January, 2022	2022
Friday, 7 January, 2022	2022
Saturday, 8 January, 2022	2022
Sunday, 9 January, 2022	2022
Monday, 10 January, 2022	2022
Tuesday, 11 January, 2022	2022
Wednesday, 12 January, 2022	2022
Thursday, 13 January, 2022	2022
Friday, 14 January, 2022	2022
Saturday, 15 January, 2022	2022
Sunday, 16 January, 2022	2022
Monday, 17 January, 2022	2022
Tuesday, 18 January, 2022	2022
Wednesday, 19 January, 2022	2022
Thursday, 20 January, 2022	2022
Friday, 21 January, 2022	2022
Saturday, 22 January, 2022	2022
Sunday, 23 January, 2022	2022
Monday, 24 January, 2022	2022
Tuesday, 25 January, 2022	2022
Wednesday, 26 January, 2022	2022