

Question 1

12  
Meas\_ceiling

10  
Meas\_Floor

87.1...  
Measure\_fact


4  
Measure\_Mod

Question 2: SumX with filter Condition category - "Furniture"

742K  
Measure\_sumx

11.68  
Measure\_round

Question 3: Measure with calculation and unicode

316932   
Meas\_concat

Question 4: Convert 20201030(yyyymmdd) to 30 Oct 2020(date)

October 30, 2020  
Earliest ChangedDate

Question: 5,6,7  
length, Upper and Lower case

Meas_len_string	Category	Meas_lower_case	Meas_upper_case
53	Furniture	furniture	FURNITURE
26	Office Supplies	office supplies	OFFICE SUPPLIES
33	Technology	technology	TECHNOLOGY
26		technology	TECHNOLOGY

Question9: Distnct function

- Category
- Furniture
- Office Supplies
- Technology

Question 8: Related Function: Column related category is created using related function

Product Subcategory

EnglishProductSubcategoryName	ProductSubcategoryKey	Column_rel_category	ProductCategoryKey
Bike Racks	26	Accessories	
Bike Stands	27	Accessories	
Bottles and Cages	28	Accessories	
Cleaners	29	Accessories	
Fenders	30	Accessories	
Helmets	31	Accessories	
Hydration Packs	32	Accessories	
Lights	33	Accessories	
Locks	34	Accessories	
Panniers	35	Accessories	
Pumps	36	Accessories	
Tires and Tubes	37	Accessories	
Mountain Bikes	1	Bikes	
Road Bikes	2	Bikes	

Product Category

ProductCategoryKey	EnglishProductCategoryName
4	Accessories
1	Bikes
3	Clothing
2	Components

Question 11: Combine two table using union function

TabeA

Sum of id	Name
1	Product1
2	Product2
3	product3
4	product4
15	

TableB

id	Name	Value
12	Product12	1200
13	product13	2300
14	product14	4500
15	Product15	899
54		8899

TableAunionB

id	Name	Value
1	Product1	100
12	Product12	1200
13	product13	2300
14	product14	4500
15	Product15	899
2	Product2	200
3	product3	300
4	product4	400
5	product5	500
69		10399

Question 10: Slicer with Default as "Not Selected"

Category

Furniture

Office Supplies

Technology

Fur

Meas\_deflt\_slicer

Question 12: Combine two tables using union and summarize the result

Union of CA and DE Sales

ProductID	Revenue	Units	Zip	Country
194	102,732.80	203	10115	Germany
392	287,594.61	398	10117	Germany
521	279,651.77	613	10119	Germany
103	67,360.98	104	10178	Germany
6	3,815.87	6	10179	Germany
923	488,773.05	961	10243	Germany
639	304,993.95	666	10245	Germany
486	263,418.91	488	10247	Germany
5	5,273.53	5	10249	Germany
271	151,594.13	273	10315	Germany
11	4,555.29	11	10317	Germany
607	386,809.65	612	10318	Germany
570	367,517.04	603	10319	Germany
943	488,786.02	966	10365	Germany
216	137,300.51	219	10367	Germany
309390	166,047,005.83	321874		

Summary table according to country and Rank

Country	Year	Revenue_by_country
Germany	2006	10,111,268.63
Germany	2007	9,887,454.31
Germany	2004	9,775,034.46
Germany	2005	9,759,472.17
Germany	2003	9,746,123.65
Germany	2008	9,348,343.69
Germany	2002	9,145,549.77
Germany	2001	7,829,050.38
Germany	2009	7,161,470.99
Germany	2014	6,984,804.82
Germany	2000	6,907,838.85
Germany	2013	6,566,484.36
Germany	2010	6,435,329.07
Germany	2012	6,170,555.92
Germany	2011	6,082,311.96
Germany	1999	5,971,731.12
Total		166,047,005.83

Question 13: Variable and unicode operation

6/30/2015   
Measure\_unicode

Question 14: Measure to compare columns containing odd even, text value

id	Measure_isNUm	Number_value	Text_value
	Not a number		
1	Not a number	13	1
2	Not a number	11	Some text
3	Not a number	45	34
4	Not a number	56	56
5	Not a number	64	67
Total	Not a number	189	

Min Of  
Nmber\_value is  
ODD  
is\_odd

Question 16: DistinctCount, Count, and  
DistinctRowCount.

10K  
Meas\_Count

Not a  
number  
Measure\_isNUm

Max Of  
Number\_value  
is Even  
is\_Even

5009  
Meas\_Distinct\_Count

Is Text  
Measure\_isText

10K  
Meas\_Row\_Count

Question 15: Skip Rank and Dense Rank

Id	Name	Price	Column_skipRank	Column_DenseRank
3	Plums		1	1
7	Apple		1	1
1	Apricot	10	3	2
2	Raspberry	12	4	3
4	Peach	12	4	3
5	Banana	15	6	4
10	Tangerine	15	6	4
6	Mango	20	8	5
8	Grapes	23	9	6
9	Cherry	56	10	7
Total		163	52	36

Question 17 :TOP N Table

Country	ProductID	Date	Revenue
Canada	2	6/13/2014 12:00:00 AM	0.00
Canada	2	3/31/2011 12:00:00 AM	47.20
Canada	2	1/15/1999 12:00:00 AM	52.45
Canada	4	2/15/1999 12:00:00 AM	52.45
Canada	6	3/15/1999 12:00:00 AM	52.45
Canada	6	4/15/1999 12:00:00 AM	52.45
Canada	2	5/15/1999 12:00:00 AM	52.45
Canada	4	6/15/1999 12:00:00 AM	52.45
Canada	2	7/15/1999 12:00:00 AM	52.45
Canada	12	8/15/1999 12:00:00 AM	52.45
Canada	2	9/15/1999 12:00:00 AM	52.45
Total		48	

Question18: Dim\_Date Table

Column_Year	Column_quarter	Column_month_no	Column_week_No	Column_week_day	Column_rank
2017	1	1	2	1	1
2017	1	1	3	1	1
2017	1	1	4	1	1
2017	1	1	5	1	1
2017	1	1	1	2	2
2017	1	1	2	2	2
2017	1	1	3	2	2
2017	1	1	4	2	2
2017	1	1	5	2	2
2017	1	1	1	3	3
2017	1	1	2	3	3
2017	1	1	3	3	3
2017	1	1	4	3	3
2017	1	1	5	3	3
2017	1	1	1	4	4
2017	1	1	2	4	4
2017	1	1	3	4	4
2017	1	1	4	4	4
2017	1	1	5	4	4
2017	1	1	1	5	5
2017	1	1	2	5	5
2017	1	1	3	5	5
2017	1	1	4	5	5
2017	1	1	5	5	5
2017	1	1	1	6	6
2017	1	1	2	6	6
2017	1	1	3	6	6
2017	1	1	4	6	6
2017	1	1	5	6	6
2017	1	1	1	7	7
2017	1	1	2	7	7
2017	1	1	3	7	7
2017	1	1	4	7	7
2017	1	1	5	7	7
2017	1	1	1	8	8
2017	1	1	2	8	8
2017	1	1	3	8	8
2017	1	1	4	8	8
2017	1	1	5	8	8
2017	1	1	1	9	9
2017	1	1	2	9	9
2017	1	1	3	9	9
2017	1	1	4	9	9
2017	1	1	5	9	9
2017	1	1	1	10	10
2017	1	1	2	10	10
2017	1	1	3	10	10
2017	1	1	4	10	10
2017	1	1	5	10	10
2017	1	1	1	11	11
2017	1	1	2	11	11
2017	1	1	3	11	11
2017	1	1	4	11	11
2017	1	1	5	11	11
2017	1	1	1	12	12
2017	1	1	2	12	12
2017	1	1	3	12	12
2017	1	1	4	12	12
2017	1	1	5	12	12
2017	1	1	1	1	13
2017	1	1	2	1	13
2017	1	1	3	1	13
2017	1	1	4	1	13
2017	1	1	5	1	13
2017	1	1	1	2	14
2017	1	1	2	2	14
2017	1	1	3	2	14
2017	1	1	4	2	14
2017	1	1	5	2	14
2017	1	1	1	3	15
2017	1	1	2	3	15
2017	1	1	3	3	15
2017	1	1	4	3	15
2017	1	1	5	3	15
2017	1	1	1	4	16
2017	1	1	2	4	16
2017	1	1	3	4	16
2017	1	1	4	4	16
2017	1	1	5	4	16
2017	1	1	1	5	17
2017	1	1	2	5	17
2017	1	1	3	5	17
2017	1	1	4	5	17
2017	1	1	5	5	17
2017	1	1	1	6	18
2017	1	1	2	6	18
2017	1	1	3	6	18
2017	1	1	4	6	18
2017	1	1	5	6	18
2017	1	1	1	7	19
2017	1	1	2	7	19
2017	1	1	3	7	19
2017	1	1	4	7	19
2017	1	1	5	7	19
2017	1	1	1	8	20
2017	1	1	2	8	20
2017	1	1	3	8	20
2017	1	1	4	8	20
2017	1	1	5	8	20
2017	1	1	1	9	21
2017	1	1	2	9	21
2017	1	1	3	9	21
2017	1	1	4	9	21
2017	1	1	5	9	21
2017	1	1	1	10	22
2017	1	1	2	10	22
2017	1	1	3	10	22
2017	1	1	4	10	22
2017	1	1	5	10	22
2017	1	1	1	11	23
2017	1	1	2	11	23
2017	1	1	3	11	23
2017	1	1	4	11	23
2017	1	1	5	11	23
2017	1	1	1	12	24
2017	1	1	2	12	24
2017	1	1	3	12	24
2017	1	1	4	12	24
2017	1	1	5	12	24
2017	1	1	1	1	25
2017	1	1	2	1	25
2017	1	1	3	1	25
2017	1	1	4	1	25
2017	1	1	5	1	25
2017	1	1	1	2	26
2017	1	1	2	2	26
2017	1	1	3	2	26
2017	1	1	4	2	26
2017	1	1	5	2	26
2017	1	1	1	3	27
2017	1	1	2	3	27
2017	1	1	3	3	27
2017	1	1	4	3	27
2017	1	1	5	3	27
2017	1	1	1	4	28
2017	1	1	2	4	28
2017	1	1	3	4	28
2017	1	1	4	4	28
2017	1	1	5	4	28
2017	1	1	1	5	29
2017	1	1	2	5	29
2017	1	1	3	5	29
2017	1	1	4	5	29
2017	1	1	5	5	29
2017	1	1	1	6	30
2017	1	1	2	6	30
2017	1	1	3	6	30
2017	1	1	4	6	30
2017	1	1	5	6	30
2017	1	1	1	7	31
2017	1	1	2	7	31
2017	1	1	3	7	31
2017	1	1	4	7	31
2017	1	1	5	7	31
2017	1	1	1	8	32
2017	1	1	2	8	32
2017	1	1	3	8	32
2017	1	1	4	8	32
2017	1	1	5	8	32
2017	1	1	1	9	33
2017	1	1	2	9	33
2017	1	1	3	9	33
2017	1	1	4	9	33
2017	1	1	5	9	33
2017	1	1	1	10	34
2017	1	1	2	10	34
2017	1	1	3	10	34
2017	1	1	4	10	34
2017	1	1	5	10	34
2017	1	1	1	11	35
2017	1	1	2	11	35
2017	1	1	3	11	35
2017	1	1	4	11	35
2017	1	1	5	11	35
2017	1	1	1	12	36
2017	1	1	2	12	36
2017	1	1	3	12	36
2017	1	1	4	12	36
2017	1	1	5	12	36
2017	1	1	1	1	37
2017	1	1	2	1	37
2017	1	1	3	1	37
2017	1	1	4	1	37
2017	1	1	5	1	37
2017	1	1	1	2	38
2017	1	1	2	2	38
2017	1	1	3	2	38
2017	1	1	4	2	38
2017	1	1	5	2	38
2017	1	1	1	3	39
2017	1	1	2	3	39
2017	1	1	3	3	39
2017	1	1	4	3	39
2017	1	1	5	3	39
2017	1	1	1	4	40
2017	1	1	2	4	40
2017	1	1	3	4	40
2017	1	1	4	4	40
2017	1	1	5	4	40
2017	1	1	1	5	41
2017	1	1	2	5	41
2017	1	1	3	5	41
2017	1	1	4	5	41
2017	1	1	5	5	41
2017	1	1	1	6	42
2017	1	1	2	6	42
2017	1	1	3	6	42
2017	1	1	4	6	42
2017	1	1	5	6	42
2017	1	1	1	7	43
2017	1	1	2	7	43
2017	1	1	3	7	43
2017	1	1	4	7	43
2017	1	1	5	7	43
2017	1	1	1	8	44
2017	1	1	2	8	44
2017	1	1	3	8	44
2017	1	1	4	8	44
2017	1	1	5	8	44
2017	1	1	1	9	45
2017	1	1	2	9	45
2017	1	1	3	9	45
2017	1	1	4	9	45
2017	1	1	5	9	45
2017	1	1	1	10	46
2017	1	1	2	10	46
2017	1	1	3	10	46
2017	1	1	4	10	46
2017	1	1	5	10	46
2017	1	1	1	11	47
2017	1	1	2	11	47
2017	1	1	3	11	47
2017	1	1	4	11	47
2017	1	1	5	11	47
2017	1	1	1	12	48
2017	1	1	2	12	48
2017	1	1	3	12	48
2017	1	1	4	12	48
2017	1	1	5	12	48
2017	1	1	1	1	49
2017	1	1	2	1	49
2017	1	1	3	1	49
2017	1	1	4	1	49
2017	1	1	5	1	49
2017	1	1	1	2	50
2017	1	1	2	2	50
2017	1	1	3	2	50
2017	1	1	4	2	50
2017	1	1	5	2	50
2017	1	1	1	3	51
2017	1	1	2	3	51

## Question 20: Current Date time

3/21/2022

10:14:30

AM

Meas\_today

## **Power BI DAX**

Use "Test Data.xlsx" (Sheet1) to perform the following operations.

1. Create a Measure to calculate the Floor/Ceiling/Fact/Mod/Round values of a number.

```
Meas_ceiling = CEILING(11.65,1)
Meas_Floor = FLOOR(11.95,2)
Measure_fact = FACT(14)
Measure_Mod = MOD(34,5)
Measure_round = ROUND(11.677787,2)
```

2. Create a measure to calculate the Sum value based on the filter condition using the SUMX function. (Filter condition: Category= "Furniture")

```
Measure_sumx = SUMX(Filter(sheet1,[Category]="Furniture"),Sheet1[Sales])
```

3. Create a measure to concatenate a Unicode character with a calculation.

```
Meas_concat = CONCATENATE(MAX(Sheet1[Quantity])*MAX(Sheet1[Sales]),UNICHAR(9940))
```

4. Convert 20201030(yyyymmdd) to 30 Oct 2020(date datatype) using LEFT/RIGHT function.

```
ChangedDate = DATE(LEFT([Old_Date],4),MID([Old_Date],5,2),RIGHT([Old_Date],2))
```

5. Create a measure to calculate the length of a string.

```
Meas_len_string = LEN(Max(Sheet1[Product Name]))
```

6. Create a measure to convert lower case value to upper case.

```
Meas_lower_case = LOWER(MAX(Sheet1[Category]))
```

7. Create a measure to convert upper case to lower case.

```
Meas_upper_case = UPPER(MAX(Sheet1[Category]))
```

8. Copy a particular column from the other table which is having many to one relationship between the table, using Related function.



9. Select the distinct value from a table using the Distinct function.

```
tab_dis_col = DISTINCT(Sheet1[Category])
```

10. Capture the selected value from the slicer, default shows it as "Not Selected". Use card visuals to display the value.

```
Meas_deflt_slicer = IF(SELECTEDVALUE(Sheet1[Category])=Blank(),"Not Selected",  
    IF(MAX('Sheet1'[Category])="Office Supplies","OS",  
    IF(MAX(Sheet1[Category])="Technology","Tec",  
    IF(MAX(Sheet1[Category])="Furniture","Fur"))))
```

11. Combine two table which has the same numbers of columns in both tables, using the Union function.
12. Combine two tables which having is different sets of columns in both the tables, using the Union and Summarize function. (Use "CA Sales.csv" and "DE Sales.csv" to perform the operation)
13. Create a measure to concatenate a Unicode character with a calculation, use Variable to perform the operation.
14. Create a measure to compare columns containing Odd/Even/ISNUMBER/TEXT Values.

```
is_Even = IF(ISEVEN(MAX(Different_data_table[Number_value])), "Max Of Nmber_value is Even", "Not Even" )
```

```
is_odd = IF(ISODD(MIN(Different_data_table[Number_value])), "Min Of Nmber_value is ODD", "Min Of Nmber_value is NOT ODD" )
```

```
Measure_isNUm = If(ISNUMBER(Max(Different_data_table[Text_value])), "Number", "Not a number")
```

15. Create a calculated column to calculate DenseRANK and SKIPRANK values.

```
Column_DenseRank = RANKX(Rank_demo,Rank_demo[Price],,ASC,Dense)  
Column_skipRank = RANKX(Rank_demo,Rank_demo[Price],,ASC,SKIP)
```

16. Create a measure to count the row as DistinctCount, Count, and DistinctRowCount.

17. Create a table to calculate TOP 10 values from another table

```
Top_Table = TOPN(10, 'CA Sales', 'CA Sales'[Revenue], ASC) .
```

18. Create a DimDate table using the CALENDAR () function and add the following column into it-

```
Dim_Date_Table = CALENDAR(MIN(Sheet1[Order Date]),MAX(Sheet1[Order Date]))
```

- Year ()

```
Column_Year = YEAR(Dim_Date_Table[Date])
```

- MonthNo ()

```
Column_month_no = Month(Dim_Date_Table[Date])
```

- Quarter ()

```
Column_quarter = QUARTER(Dim_Date_Table[Date])
```

- WeekNo ()

```
Column_week_No = WEEKNUM(Dim_Date_Table[Date])
```

- WeekDay ()

```
Column_week_day = WEEKDAY(Dim_Date_Table[Date])
```

- Day ()

```
Column_day = DAY(Dim_Date_Table[Date])
```

- 

19. Create a dynamic title using the if Else statement.

```
Meas_dynamic_Cat = IF(MAX('Sheet1'[Category])="Office Supplies", "OS",  
    IF(MAX(Sheet1[Category])="Technology", "Tec",  
    IF(MAX(Sheet1[Category])="Furniture", "Fur",  
        "Other"))))
```

20. Capture Current date in date and date-time format.

```
Meas_today = Now()
```

References:

<https://radacad.com/get-a-field-value-from-a-related-table-in-power-bi-dax-related-function-explained>