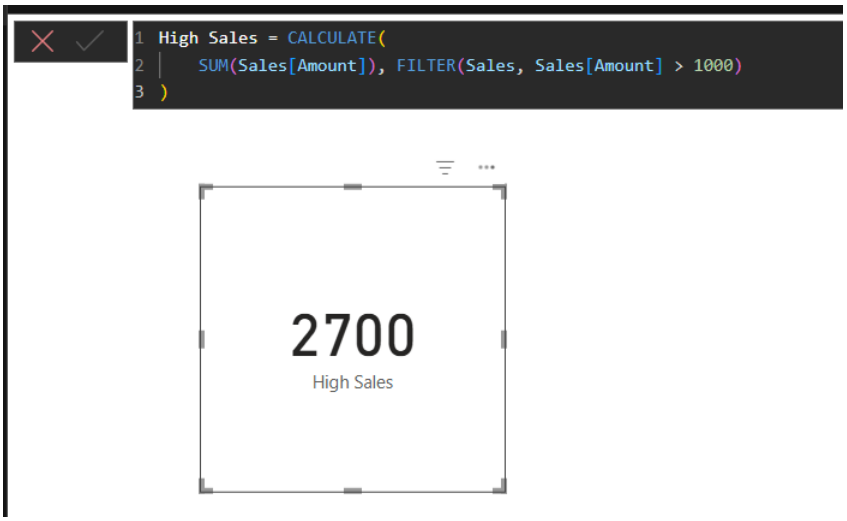
1. What does FILTER(Sales, Sales[Amount] > 1000) return?

   FILTER function returns a tabular value. From the given source, it returns this result.

Results | Result 1 of 1 | Copy

	Sales[SaleID]	Sales[ProductID]	Sales[Amount]	Sales[Region]	Sales[SaleDate]
1	1	P1	1200	North	1/5/2023 12:00:00 AM
2	3	P1	1500	North	1/15/2023 12:00:00 AM

2. Write a measure High Sales that sums Amount where Amount > 1000 using FILTER.



3. How does ALLEXCEPT(Sales, Sales[Region]) differ from ALL(Sales)?

   ALLEXCEPT(Sales, Sales[Region]) removes all the filters from the Sales table, but keeps the existing filters active for Sales[Region] column. It returns Sales[Region] column filtered but other column are unfiltered table.

   ALL(Sales) removes all filters from the entire Sales table. It returns the complete, unfiltered Sales table.

```
Total Sales = CALCULATE(
    SUM(Sales[Amount]),
    ALL(Sales)
)
```

```
Total Sales except region = CALCULATE(
    SUM(Sales[Amount]),
    ALLEXCEPT(Sales, Sales[Region])
)
```

4. Use SWITCH to categorize Amount:
   "Medium" if 500–1000
   "High" if > 1000 ```

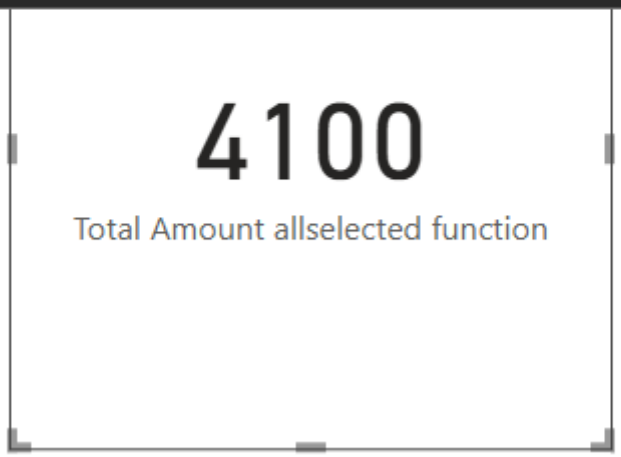
```
Total Amount = SWITCH(
    TRUE(),
    Sales[Amount] > 0 && Sales[Amount] < 500, "Low",
    Sales[Amount] >= 500 && Sales[Amount] < 1000, "Medium",
    "High"
)
```

SaleID	ProductID	Amount	Region	SaleDate	Total Amount
1	P1	1200	North	Thursday, January 5, 2023	High
2	P2	800	South	Tuesday, January 10, 2023	Medium
3	P1	1500	North	Sunday, January 15, 2023	High
4	P2	600	East	Friday, January 20, 2023	Medium

5. What is the purpose of ALLSELECTED?

   This function is used to remove filter context from the visual or report, but only for the filters that are currently applied within that specific visual. It ignores filters within the current context while respecting other filter applied at higher level.

```
Total Amount allselected function = CALCULATE(
    SUM(Sales[Amount]),
    ALLSELECTED(Sales)
)
```



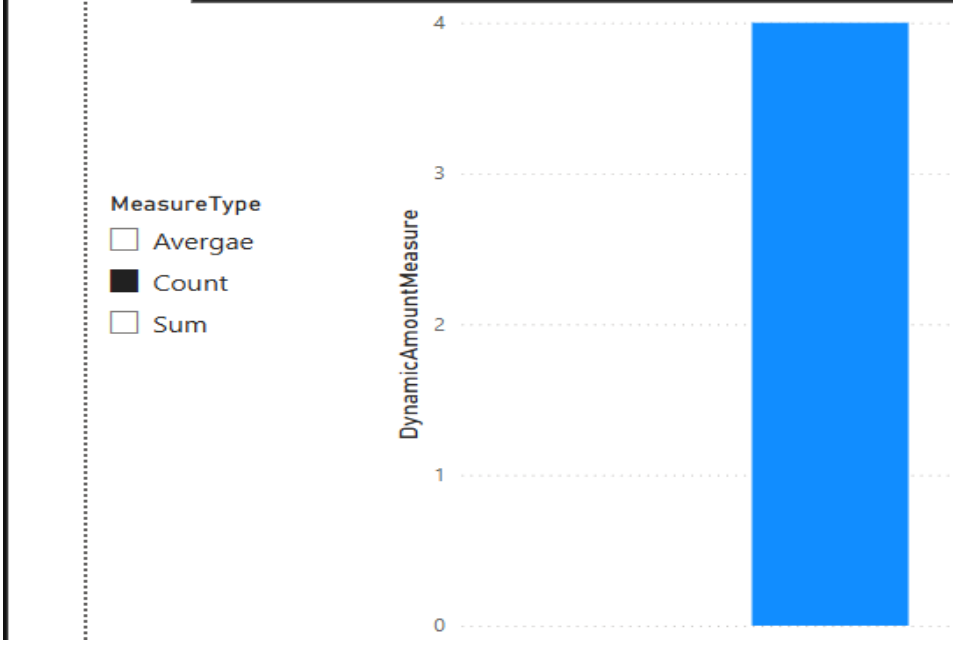
6. Write a measure Regional Sales % showing each sale's contribution to its region's total (use ALLEXCEPT).

```
Regional Sales % =
DIVIDE(
    SUM(Sales[Amount]),
    CALCULATE(
        SUM(Sales[Amount]),
        ALLEXCEPT(Sales, Sales[Region])
    )
)
```

Region	ProductID	Amount	Regional Sales %
East	P2	600	100.00%
North	P1	1200	44.44%
North	P1	1500	55.56%
South	P2	800	100.00%
Total			**100.00%**

7. Create a dynamic measure using SWITCH to toggle between SUM, AVERAGE, and COUNT of Amount.

```
DynamicAmountMeasure =
SWITCH(
    SELECTEDVALUE('ToggleTable'[MeasureType]),
    "Sum", SUM('Sales'[Amount]),
    "Average", AVERAGE('Sales'[Amount]),
    "Count", COUNT('Sales'[Amount]),
    SUM('Sales'[Amount])
)
```



8. Use FILTER inside CALCULATE to exclude "Furniture" sales (Products[Category] = "Furniture").

```
TotalSalesExcludingFurniture =
CALCULATE(
    SUM(Sales[Amount]),
    FILTER(
        VALUES(Products[ProductID]),  // Preserves oth
        RELATED(Products[Category]) <> "Furniture"
    )
)
```

Category	Total Sales	Sum of TotalSalesExcludingFurniture
Electronics	4100	2700
Furniture	4100	
Total	**4100**	**2700**

9. Why might ALLSELECTED behave unexpectedly in a pivot table?

   There are multiple reasons why this function might behave unexpectedly.
   1) Ignores Row/Column header. It bypasses filters from the pivot table rows, using only slicer selections.
   2) Bidirectional relationship. It might create circular logic.
   3) Ignores filters applied directly to the pivot table.

10. Write a measure that calculates total sales and ignores filters from region

```
Total Sales ignoring region = CALCULATE(
    SUM(Sales[Amount]),
    ALLSELECTED(Sales[Region])
)
```

Region	Category	Sum of Amount	Total Sales ignoring region
East	Electronics		2700
North	Electronics	2700	2700
South	Electronics		2700
East	Furniture	600	1400
North	Furniture		1400
South	Furniture	800	1400
Total		**4100**	**4100**

11. Optimize this measure:
    High Sales = CALCULATE(SUM(Sales\[Amount]), FILTER(Sales, Sales\[Amount] > 1000)) (Hint: Replace FILTER with a Boolean filter inside CALCULATE.)

    High Sales =

    CALCULATE(

        SUM(Sales[Amount]),

        Sales[Amount] > 1000 // Boolean filter replacement

)

12. Write a measure Top 2 Products using TOPN and FILTER to show the highest-grossing products.

```
Top 2 product sales =
VAR Top_products =
    TOPN(2,
    SUMMARIZE(
        ALLSELECTED(Sales),
        Sales[ProductID],
        "Product Sales",
        SUM(Sales[Amount])),
        [Product Sales],
        DESC
    )
RETURN CALCULATE(
    SUM(Sales[Amount]),
    FILTER(
        ALLSELECTED(Sales),
        Sales[ProductID] IN SELECTCOLUMNS(Top_products, "PID", Sales[ProductID])
    )
)
```

13. Use ALLSELECTED with no parameters to respect slicers but ignore visual-level filters.

```
Total Sales Respect Slicers =
CALCULATE(
    SUM(Sales[Amount]),
    ALLSELECTED()  // No parameters
)
```

14. Debug: A SWITCH measure returns incorrect values when fields are added to a matrix visual.

```
Revenue Category =
VAR CurrentSales = [Total Sales]
RETURN
SWITCH(
   TRUE(),
   CurrentSales > 5000, "High",
   CurrentSales > 2000, "Medium",
   "Low"
)
```

15. Simulate a "reset filters" button using ALL in a measure.

```
Reset All Filters =
VAR AllTables = UNION(
    VALUES('Sales'[SaleID]),
    VALUES('Products'[ProductID]),
    VALUES('Sales'[Region])
)
RETURN
CALCULATE( SUM(Sales[Amount]), ALL(AllTables) )
```