1. What is row context? Give an example in a calculated column.
Answer:
Row context is the "current row" that DAX processes when calculating a value in a calculated column. In row context, DAX evaluates expressions based on the values of the current row, unless another filter or function modifies the context.

Example:

DAX

CopyEdit

TotalPrice = Sales[Quantity] * Sales[UnitPrice]

Here, each row in the **Sales** table gets its own TotalPrice value using its own Quantity and UnitPrice.

---

2. Write a measure that finds total sales.

DAX

CopyEdit

Total Sales = SUM(Sales[Quantity] * Sales[UnitPrice])

---

3. Use RELATED to fetch the name from the Customers table into the Sales table.

DAX

CopyEdit

CustomerName = RELATED(Customers[Name])

Works if there is an active relationship between **Sales** and **Customers** via CustomerID.

---

4. What does CALCULATE(SUM(Sales[Quantity]), Sales[Category] = "Electronics") return?
Answer:
It returns the sum of Quantity **only** for rows where Category = "Electronics", ignoring any other category filters that may be applied.

---

5. Explain the difference between VAR and RETURN in DAX.
Answer:

- VAR creates a temporary variable to store a value or table for reuse in the formula.

- RETURN specifies the final result that should be output by the measure or column.

Example:

DAX

CopyEdit

Example =

VAR TotalQty = SUM(Sales[Quantity])

RETURN TotalQty / COUNTROWS(Sales)

---

6. Create a calculated column in Sales called TotalPrice using row context (Quantity * UnitPrice).

DAX

CopyEdit

TotalPrice = Sales[Quantity] * Sales[UnitPrice]

---

7. Write a measure Electronics Sales using CALCULATE to sum sales only for the "Electronics" category.

DAX

CopyEdit

Electronics Sales =

CALCULATE(

  SUM(Sales[Quantity] * Sales[UnitPrice]),

  Sales[Category] = "Electronics"

)

---

8. Use ALL(Sales[Category]) in a measure to show total sales ignoring category filters.

DAX

CopyEdit

Total Sales Ignore Category =

CALCULATE(

  SUM(Sales[Quantity] * Sales[UnitPrice]),

ALL(Sales[Category])

)

---

9. Fix this error: A calculated column in Sales uses RELATED(Customers[Region]) but returns blanks.
Answer:
The most likely reason is that there is no active relationship between the **Sales** and **Customers** tables on CustomerID.
To fix:

- Check the data model.
- Ensure the relationship exists and is active.
- Make sure filter direction allows data to flow from Customers to Sales.

---

10. Why does CALCULATE override existing filters?
Answer:
Because CALCULATE changes the filter context before performing calculations. If a filter is specified inside CALCULATE, it replaces or adds to the current filters.

---

11. Write a measure that returns the average unit price of products.

DAX

CopyEdit

Average UnitPrice = AVERAGE(Sales[UnitPrice])

---

12. Use VAR to store a temporary table of high-quantity sales (Quantity > 2), then count rows.

DAX

CopyEdit

HighQtySales Count =

VAR HighQtySales = FILTER(Sales, Sales[Quantity] > 2)

RETURN COUNTROWS(HighQtySales)

---

13. Write a measure % of Category Sales that shows each sale's contribution to its category total.

DAX

CopyEdit

```
% of Category Sales =
VAR CategoryTotal =
  CALCULATE(
    SUM(Sales[Quantity] * Sales[UnitPrice]),
    ALLEXCEPT(Sales, Sales[Category])
  )
RETURN
DIVIDE(
  SUM(Sales[Quantity] * Sales[UnitPrice]),
  CategoryTotal
)
```

---

14. Simulate a "Remove Filters" button using ALL in a measure.

DAX

CopyEdit

```
Total Sales No Filters =
CALCULATE(
  SUM(Sales[Quantity] * Sales[UnitPrice]),
  ALL(Sales)
)
```

---

15. Troubleshoot: A CALCULATE measure ignores a slicer. What's the likely cause? Answer:

- The measure uses ALL or REMOVEFILTERS inside CALCULATE, which removes the slicer's filter.
- Or the filter argument inside CALCULATE explicitly overrides the slicer selection.