1. What does FILTER(Sales, Sales[Amount] > 1000) return?
It returns a table that includes only rows from the Sales table where Sales[Amount] is greater than 1000. It’s used inside functions like CALCULATE to modify the row context.

---

2. Write a measure High Sales that sums Amount where Amount > 1000 using FILTER.
DAX
КопироватьРедактировать
```
High Sales =
CALCULATE(
   SUM(Sales[Amount]),
   FILTER(Sales, Sales[Amount] > 1000)
)
```

---

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

ALL(Sales) removes all filters from the Sales table.

ALLEXCEPT(Sales, Sales[Region]) removes all filters except those on Sales[Region].

---

4. Use SWITCH to categorize Amount: "Medium" if 500–1000, "High" if > 1000
DAX
КопироватьРедактировать
```
Amount Category =
SWITCH(
   TRUE(),
   Sales[Amount] > 1000, "High",
   Sales[Amount] >= 500, "Medium",
   "Low"
)
```

---

5. What is the purpose of ALLSELECTED?
It retains only slicer and filter selections made by the user, ignoring visual-level filters. Useful for calculating values relative to a selected subset, not the entire dataset.

---

6. Write a measure Regional Sales % using ALLEXCEPT
DAX
КопироватьРедактировать
```
Regional Sales % =
DIVIDE(
   SUM(Sales[Amount]),
   CALCULATE(SUM(Sales[Amount]), ALLEXCEPT(Sales, Sales[Region]))
)
```

---

7. Create a dynamic measure using SWITCH to toggle between SUM, AVERAGE, and COUNT of Amount.
Assume a disconnected table MeasureType[Selection] with values "SUM", "AVERAGE", "COUNT":
DAX
КопироватьРедактировать
```
Dynamic Measure =
SWITCH(
   SELECTEDVALUE(MeasureType[Selection]),
   "SUM", SUM(Sales[Amount]),
   "AVERAGE", AVERAGE(Sales[Amount]),
   "COUNT", COUNT(Sales[Amount])
)
```

8. Use FILTER inside CALCULATE to exclude "Furniture" sales
DAX
КопироватьРедактировать
```
Non-Furniture Sales =
CALCULATE(
   SUM(Sales[Amount]),
   FILTER(Products, Products[Category] <> "Furniture")
)
```

---

9. Why might ALLSELECTED behave unexpectedly in a pivot table?
Because ALLSELECTED keeps slicer filters but not visual (row/column) filters, leading to unexpected totals or percentages when you expect it to reflect the matrix visual. It can behave unpredictably when combined with row context.

---

10. Write a measure that calculates total sales and ignores filters from region
DAX
КопироватьРедактировать
```
Total Sales Ignore Region =
CALCULATE(
   SUM(Sales[Amount]),
   ALL(Sales[Region])
)
```

---

11. Optimize this measure (Boolean filter inside CALCULATE):
DAX
КопироватьРедактировать
```
-- Original:
High Sales = CALCULATE(SUM(Sales[Amount]), FILTER(Sales, Sales[Amount] > 1000))

-- Optimized:
High Sales = CALCULATE(SUM(Sales[Amount]), Sales[Amount] > 1000)
```
Using a Boolean expression instead of FILTER is faster and more efficient.

---

12. Write a measure Top 2 Products using TOPN and FILTER
DAX
КопироватьРедактировать
```
Top 2 Products Sales =
CALCULATE(
   SUM(Sales[Amount]),
   FILTER(
      TOPN(2, SUMMARIZE(Sales, Sales[Product], "Total", SUM(Sales[Amount])), [Total], DESC),
      TRUE()
   )
)
```

---

13. Use ALLSELECTED() with no parameters to respect slicers but ignore visual-level filters
DAX
КопироватьРедактировать
```
Selected Total Sales =
CALCULATE(
   SUM(Sales[Amount]),
   ALLSELECTED()
)
```

---

14. Debug: A SWITCH measure returns incorrect values in a matrix
This happens if:

SELECTEDVALUE() returns blank because multiple values are selected.

Row context interferes with expected results.

Fix:
Wrap SELECTEDVALUE() in a COALESCE() or use HASONEVALUE() to check before SELECTEDVALUE().

---

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

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