Power BI - lesson 14

Lesson 14 DAX Optimization Solutions

1. % Growth in Sales Compared to Last Year (with VAR)

```
Sales Growth % =
VAR SalesCY = [Total Sales]
VAR SalesLY = CALCULATE([Total Sales], SAMEPERIODLASTYEAR(DateTable
[Date]))
RETURN DIVIDE(SalesCY - SalesLY, SalesLY, 0)
```

---

2. Difference: Current Month vs Previous Month

```
Sales Diff MoM =
VAR CurrMonth = [Total Sales]
VAR PrevMonth = CALCULATE([Total Sales], PREVIOUSMONTH(DateTable[Da
te]))
RETURN CurrMonth - PrevMonth
```

---

3. Total Boxes + Avg Monthly Boxes (in one measure)

Assuming you have a column `Sales[Boxes]`:

```
Boxes and Avg =
VAR TotalBoxes = SUM(Sales[Boxes])
VAR AvgMonthly = AVERAGEX(VALUES(DateTable[Month]), [Total Sales])
RETURN TotalBoxes & " Boxes, Avg " & FORMAT(AvgMonthly, "0.00")
```

---

4. Return Only Average Monthly Boxes

```
Avg Monthly Boxes =
VAR AvgMonthly = AVERAGEX(VALUES(DateTable[Month]), [Total Sales])
RETURN AvgMonthly
```

---

5. Growth % From Last Month

```
Sales Growth MoM % =
VAR CurrMonth = [Total Sales]
VAR PrevMonth = CALCULATE([Total Sales], PREVIOUSMONTH(DateTable[Da
te]))
RETURN DIVIDE(CurrMonth - PrevMonth, PrevMonth, 0)
```

---

6. Moving Average (Last 3 Months)

```
Sales Moving Avg 3M =
AVERAGEX(
    DATESINPERIOD(DateTable[Date], MAX(DateTable[Date]), -3, MONTH),
    [Total Sales]
)
```

---

7. Dynamic Card Message (Rank + YoY)

```
Sales Performance Message =
VAR Prod = SELECTEDVALUE(Sales[Product])
VAR RankProd = RANKX(ALL(Sales[Product]), [Total Sales], , DESC)
VAR SalesCY = [Total Sales]
VAR SalesLY = CALCULATE([Total Sales], SAMEPERIODLASTYEAR(DateTable
[Date]))
```

```
VAR YoY = DIVIDE(SalesCY - SalesLY, SalesLY, 0)
RETURN
SWITCH(
    TRUE(),
    RankProd <= 3 && YoY > 0, Prod & " - Top Performer (Sales up " & FORMAT
(YoY, "0%") & ")",
    YoY > -0.05 && YoY < 0.05, Prod & " - Consistent Performer",
    Prod & " - Needs Improvement"
)
```

---

8. Top 5 Manual Optimization Tips

1. **Use VAR for repeated expressions** → Reduces recalculations.
2. **Minimize row context inside measures** (prefer `CALCULATE`, `SUMX` with filters).
3. **Avoid FILTER when Boolean filter works** → e.g., `Sales[Amount] > 1000` inside `CALCULATE`.
4. **Use proper Date Table** → ensures DAX time intelligence runs efficiently.
5. **Limit high-cardinality columns in visuals** → avoids query bloat.

---

9. Benefits of Tools (DAX Studio, Performance Analyzer, Tabular Editor)

- **DAX Studio** → Analyze query plans, measure execution time, spot bottlenecks.
- **Performance Analyzer** → See which visuals/measures take the longest.
- **Tabular Editor** → Advanced modeling, optimization, dependency tracing.

---

10. Top 5 Flag with RANKX (using VAR)

```
Top 5 Product Flag =
VAR ProdRank = RANKX(ALL(Sales[Product]), [Total Sales], , DESC)
RETURN IF(ProdRank <= 5, "Yes", "No")
```