DAX Statements

Navbar:

1. Last Dashboard Refresh:

   Power Query -> DateTime.LocalNow( ) [Returns present Date & Time]
   Date_Last_Refresh = VALUES(Last_Refresh[date_time])

2. Drop Down Date slicer:
   Selected_Date = SUMMARIZE(
   'Orders_T',
   'Orders_T'[order_date],
   'Orders_T'[Month_Year],
   'Orders_T'[Year]
)

3. Year Silcer:
   Year = FORMAT(Orders_T[order_date], "yyyy")

Sidebar:

4. Notifications:
   Tooltip ->
   New_Notification = CALCULATE(

   COUNT(Notifications_T[Sl_No]),
   Notifications_T[Date] > TODAY() - 3
) & " "

Dashboard:

5. Revenue:
   Total_Sales = SUM(Sales_T[overall_price])

6. Profit:
   Profit = [Total_Sales] - Products_T[Total_Expenses]

7. Orders:
   Total_Orders = DISTINCTCOUNT(Sales_T[order_id])

8. Customers:
   Total_Customers = COUNT(Customer_T[customer_id])

9. Savings:
   - Total_Expenses = SUMX(Products_T, Products_T[overall_price])

- Savings = [Total_Sales] - Products_T[Total_Expenses]
- Savings_Percentage = [Savings] / [Total_Sales]

10. Expense, Sales, Savings Chart:
    - Line_Selection_Table = DATATABLE("Type", STRING, "Serial No", INTEGER,
      {
        {"Sales", 1}, {"Expense", 2}, {"Savings", 3}
      }
)

    - Line_Chart_Measures =
    var selected_Val = SELECTEDVALUE(Line_Selection_Table[Serial No])
    RETURN
      SWITCH(selected_Val,1,[Sales_Percentage],2,[Expense_Percentage],3,[Savings_Percentage])

    - Dynamic_Title = SELECTEDVALUE(Line_Selection_Table[Type])&" by Date"

11. Time:
    - Current_Hour = HOUR([Current_Time])
    - Formatted_Hour = FORMAT([Current_Hour], "00")

    - Current_Minutes = MINUTE(NOW())
    - Formatted_Minutes = FORMAT([Current_Minutes], "00")

    - AM_PM = IF([Current_Hour] < 12, "AM", "PM")

    - Current_Seconds = SECOND(NOW())
    - Formatted_Seconds = FORMAT([Current_Seconds], "00")

    - Current_Month = FORMAT(TODAY(), "mmm")
    - Current_Day = FORMAT(TODAY(), "dddd")

    - Current_Date_Number = DAY(TODAY())
    - Current_Year = YEAR(TODAY())

    - Clock_Card = "
    <!DOCTYPE html>
    <html lang=""" & "en""" & ">
      <head>
        <meta charset=""" & "UTF-8""" & " />
        <meta name=""" & "viewport""" & " content=""" & "width=device-width,
    initial-scale=1.0""" & " />
        <title>Document</title>
        <link
          rel=""" & "stylesheet""" & "
          href=""" & "https://unicons.iconscout.com/release/v4.0.0/css/line.css""" &
    "
        />
        <link
          href=""" &
    "https://fonts.googleapis.com/css2?family=Poppins:ital,wght@0,100;0,200;0,300;0,

```
400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,80
0;1,900&display=swap""" & "
      rel=""" & "stylesheet""" & "
    />
  </head>
  <style>
    * {
      padding: 0;
      margin: 0;
      font-family: """ & "Poppins""" & ", sans-serif;
    }

    body {
      min-height: 100vh;
      display: flex;
      justify-content: center;
      align-items: center;
    }

    .digital-clock {
      position: relative;
      color: #fff;
      background: #ffffff;
      width: 402px;
      height: 179px;
      padding: 20px 45px;
      /* box-shadow: 0 5px 25px rgba(14, 21, 37, 0.8); */
      box-shadow: 0 5px 25px rgba(111, 111, 217, 0.4);
      border-radius: 10px;
      display: flex;
      justify-content: center;
      align-items: center;
      flex-direction: column;
    }

    .time {
      position: relative;
      display: flex;
      justify-content: center;
      align-items: center;
    }

    .hour,
    .minutes,
    .dots {
      display: flex;
      justify-content: center;
      align-items: center;
      font-weight: 600;
      padding: 0 10px;
      line-height: 125px;
```

```
    }

    .hour,
    .minutes {
      font-size: 6.5rem;
      width: 125px;
    }

    .dots {
      font-size: 5rem;
      color: #929292;
    }

    .hour {
      background: -webkit-linear-gradient(90deg, #6F6FD9, #5fd4ff);
      -webkit-text-fill-color: transparent;
      -webkit-background-clip: text;
    }

    .minutes {
      background: -webkit-linear-gradient(90deg, #e66c37, rgb(230, 108, 55,
0.4));
      -webkit-text-fill-color: transparent;
      -webkit-background-clip: text;
    }

    .right-side {
      position: relative;
      display: flex;
      justify-content: center;
      align-items: center;
      flex-direction: column;
      margin-left: 10px;
    }

    .period,
    .seconds {
      font-size: 1.2em;
      font-weight: 500;
    }

    .period {
      transform: translateY(-20px);
      background: -webkit-linear-gradient(90deg, #f8f5f0, #404040);
      -webkit-text-fill-color: transparent;
      -webkit-background-clip: text;
    }

    .seconds {
      transform: translateY(16px);
      background: -webkit-linear-gradient(90deg, #634dff, #5fd4ff);
```

```
      -webkit-text-fill-color: transparent;
      -webkit-background-clip: text;
    }

    .calendar {
      display: flex;
      justify-content: center;
      align-items: center;
      font-size: 1.3em;
      margin-bottom: 5px;
      background: -webkit-linear-gradient(90deg, #E044A7, #E044A7);
      -webkit-text-fill-color: transparent;
      -webkit-background-clip: text;
    }

    .day-name,
    .day-number,
    .year {
      margin-left: 8px;
    }
  </style>
  <body>
    <div class=""" & "digital-clock""" & ">
      <div class=""" & "time""" & ">
        <span class=""" & "hour""" & ">" & [Formatted_Hour] & "</span>
        <span class=""" & "dots""" & ">:</span>
        <span class=""" & "minutes""" & ">" & [Formatted_Minutes] & "</span>
        <div class=""" & "right-side""" & ">
          <span class=""" & "period""" & ">" & [AM_PM] & "</span>
          <span class=""" & "seconds""" & ">" & [Formatted_Seconds] & "</span>
        </div>
      </div>
      <div class=""" & "calendar""" & ">
        <span class=""" & "month-name""" & ">" & [Current_Month] & "</span>,
        <span class=""" & "day-name""" & ">" & [Current_Day] & "</span>
        <span class=""" & "day-number""" & ">" & [Current_Date_Number] &
"</span>
        <span class=""" & "year""" & ">" & [Current_Year] & "</span>
      </div>
    </div>
  </body>
</html>
"
```

-

Stats:

12. Categories:
    - Category_A = COUNTROWS(FILTER(Sales_T, Sales_T[category] = "Category A"))
    - CategoryA_Percentage = DIVIDE([Category_A], [Total_Items])

- Category_B = COUNTROWS(FILTER(Sales_T, Sales_T[category] = "Category B"))
- CategoryB_Percentage = DIVIDE([Category_B], [Total_Items])

- Category_C = COUNTROWS(FILTER(Sales_T, Sales_T[category] = "Category C"))
- CategoryC_Percentage = DIVIDE([Category_C], [Total_Items])

13. Radar Chart:
    - Profit = [Total_Sales] - Products_T[Total_Expenses]

14. Budget Allocation:
    - Total_Allocated_Budget = SUM(Department_T[allocated_budget])
    - Target_Percentage_Allocation = [Total_Allocated_Budget] / 320000000
    - Target_Difference_Allocation = 1 - [Target_Percentage_Allocation]

15. Gross Profit Margin:
    - Gross_Profit_margin = ([Total_Sales] - [Total_Expenses]) / [Total_Sales]
    - Formatted_Gross_Profit_KPI =

```
VAR Add_Arrow =
  SWITCH(
    TRUE(),
    [Gross_Profit_margin] > 0, "▲",
    [Gross_Profit_margin] < 0, "▼",
    " "
  )
VAR _metrics = [Gross_Profit_margin]
RETURN
  Add_Arrow & " " & FORMAT(_metrics, "0%")
```

16. Most Orders By State:
    - Most_Orders_CY =

```
CALCULATE(
  MAXX(
    SUMMARIZE(Customer_T, Customer_T[state], "Total Orders", COUNTROWS(Orders_T)),
    [Total Orders]
  ),
  VALUE(Orders_T[Year]) = YEAR(TODAY())
)
```

    - Most_Orders_PY =

```
CALCULATE(
  MAXX(
    SUMMARIZE(
      FILTER(Orders_T, VALUE(Orders_T[Year]) = YEAR(TODAY()) - 1),
      Customer_T[state],
      "Total Orders",
      COUNTROWS(Orders_T)
    ),
    [Total Orders]
  )
)
```

    - Order_Rank_CY =

```
RANKX(
  ALL(Customer_T[state]),
  Orders_T[Most_Orders_CY],
  Orders_T[Most_Orders_CY],
  ,
  Dense
)
```

- Order_Rank_PY =

```
RANKX(
    ALL(Customer_T[state]),
    Orders_T[Most_Orders_PY],
    Orders_T[Most_Orders_PY],
    ,
    Dense
)
```

- Ranking_Label =

```
VAR Rank_Diff = Orders_T[Order_Rank_CY] - Orders_T[Order_Rank_PY]
VAR emptyChar = REPT(" ",3)
VAR Label =
        SWITCH(
          TRUE,
          Rank_Diff = 0, "#" & Orders_T[Order_Rank_CY] & emptyChar & "—",
          Rank_Diff > 0, "#" & Orders_T[Order_Rank_CY] & emptyChar & "▼" & Rank_Diff,
          Rank_Diff < 0, "#" & Orders_T[Order_Rank_CY] & emptyChar & "▲" & ABS(Rank_Diff)
        )
RETURN
        Label
```

- Ranking_Label_Color =

```
VAR Rank_Diff = Orders_T[Order_Rank_CY] - Orders_T[Order_Rank_PY]
VAR Label =
SWITCH(
  TRUE,
  Rank_Diff = 0, "grey",
  Rank_Diff > 0, "red",
  Rank_Diff < 0, "green"
)
RETURN
Label
```