

1) Data cleaning (Power Query) — recommended steps

Open **Power Query Editor** (Transform Data) and do the following **before** creating DAX measures:

1. Change column data types:
 - o Release Date → Date
 - o Budget, BoxOffice (or Box Office) → Decimal Number
 - o RunTime (or Runtime) → Whole Number
 - o Nominations, OscarWins (or similar) → Whole Number
 2. Handle missing Budget / Box Office:
 - o If you want **0** for missing values: select column → Transform → Replace Values → null → 0.
 - o If you prefer **median**:
 - Group By nothing? Simpler: duplicate column → remove nulls → Home → Transform → Statistics → Median → use that value with Replace Values (or use M to compute median dynamically).
 - o Save changes. (Either approach is fine; 0 is simplest and often safest for totals.)
 3. Trim/clean text columns (Title, Director, Genre): Transform → Trim / Clean → Detect Data Type.
 4. Remove duplicates (if necessary): Home → Remove Rows → Remove Duplicates (e.g., on Title + Release Date).
 5. Once clean, Close & Apply.
-

2) Create the Date table (DAX)

Create a new table in the Model view using the following DAX. This uses `CALENDARAUTO()`, `ADDCOLUMNS`, `FORMAT`, and `FILTER` to remove future dates and add useful columns:

```
Date =
VAR AllDates = CALENDARAUTO()
VAR AddCols =
    ADDCOLUMNS(
        AllDates,
        "Year", YEAR([Date]),
        "Quarter", "Q" & FORMAT(QUARTER([Date]), "0"),
        "QuarterNumber", QUARTER([Date]),
        "MonthNumber", MONTH([Date]),
        "MonthName", FORMAT([Date], "MMMM"),
        "MonthYear", FORMAT([Date], "yyyy-MM"),
        "DayOfWeek", FORMAT([Date], "dddd")
    )
-- optional: filter to only dates up to today to avoid future noise
RETURN
    FILTER( AddCols, [Date] <= TODAY() )
```

After creating this table:

- Mark it as the **Date table**: Table tools → Mark as date table → select Date column.

- Create a relationship: `Date[Date] ⇌ Movies[Release Date]` (single, inactive unless you need multiple).

3) Calculated columns (Movies table)

Profit (column) — uses `COALESCE` to handle nulls (replace with 0 if null):

```
Profit =  
COALESCE( Movies[BoxOffice], 0 ) - COALESCE( Movies[Budget], 0 )
```

Run Time Category (column) — segments runtime:

```
RunTime Category =  
VAR RT = Movies[RunTime]  
RETURN  
SWITCH(  
    TRUE(),  
    RT < 90, "Short (<90)",  
    RT >= 90 && RT < 120, "Medium (90-119)",  
    RT >= 120, "Long (120+)",  
    "Unknown"  
)
```

(Adjust column names if your sheet uses `Runtime` / `Run Time` etc.)

4) Core measures (copy & paste)

Note: assume `Movies` table name is `Movies` and `Date` table is `Date`. Replace field names if different.

Total Box Office

```
Total Box Office =  
SUM( Movies[BoxOffice] )
```

Average Budget

```
Average Budget =  
AVERAGE( Movies[Budget] )
```

Average Margin (%) — average movie margin as $(\text{BoxOffice} - \text{Budget}) / \text{Budget}$; safe with `DIVIDE` to avoid divide-by-zero:

```
Average Margin % =  
AVERAGEX(  
    FILTER( Movies, NOT( ISBLANK( Movies[Budget] ) ) && Movies[Budget] <> 0  
) ,  
    DIVIDE( COALESCE( Movies[BoxOffice], 0 ) - COALESCE( Movies[Budget], 0 ),  
    Movies[Budget] )  
)
```

Return format in report: set to Percentage with 1 decimal.

Total Movies with Oscars

(assuming OscarWins column: count movies where OscarWins > 0)

```
Movies with Oscars =  
CALCULATE(  
    COUNTROWS( Movies ),  
    FILTER( Movies, COALESCE( Movies[OscarWins], 0 ) > 0 )  
)
```

If you have a boolean WonOscar column, use CALCULATE (COUNTROWS (Movies, Movies[WonOscar] = TRUE())).

Top Genre by Box Office (returning genre name)

This returns the single top genre in the current filter context:

```
Top Genre by BoxOffice =  
VAR SummaryTable =  
    SUMMARIZE(  
        Movies,  
        Movies[Genre],  
        "GenreBox", SUM( Movies[BoxOffice] )  
    )  
VAR Top1 =  
    TOPN( 1, SummaryTable, [GenreBox], DESC )  
RETURN  
CONCATENATEX( Top1, Movies[Genre], ", " )
```

YoY Box Office Growth % (Year-over-Year % change)

This uses the Date table for proper time intelligence:

```
YoY Box Office % =  
VAR ThisYear = [Total Box Office]  
VAR LastYear =  
    CALCULATE( [Total Box Office], SAMEPERIODLASTYEAR( 'Date'[Date] ) )  
RETURN  
DIVIDE( ThisYear - LastYear, LastYear )
```

Display as Percentage. For an absolute difference instead of percent, return ThisYear - LastYear.

Average Nominations per Director

This calculates, for the current filter context, the average of total nominations per director:

```
Avg Nominations per Director =  
AVERAGEX(  
    VALUES( Movies[Director] ),  
    CALCULATE( SUM( Movies[Nominations] ) )  
)
```

If Nominations is blank for some directors, ensure you decide how to treat them (they'll count as 0 if Nominations = 0).

5) Additional useful time-intelligence measures (recommended)

Box Office - Last Year (for comparisons)

```
BoxOffice LY =  
CALCULATE( [Total Box Office], SAMEPERIODLASTYEAR( 'Date'[Date] ) )
```

YoY KPI value (for the KPI visual)

```
YoY Box Office % (for KPI) = [YoY Box Office %]
```

Set KPI target to 0% (meaning growth greater than 0 is positive).

6) Flag if movie's director is in top 5 by total box office — example (calculated column or measure)

If you want a product-level flag (per Director) as a column in `Movies` that indicates whether the director is in top 5 by total `BoxOffice`:

Top 5 Director Flag (measure recommended) — measure approach (better than column, respects filters):

```
Top 5 Director Flag (Measure) =  
VAR ThisDirector = SELECTEDVALUE( Movies[Director] )  
VAR DirectorRanks =  
    RANKX(  
        ALL( Movies[Director] ),  
        CALCULATE( SUM( Movies[BoxOffice] ) ),  
        ,  
        DESC,  
        DENSE  
    )  
RETURN  
IF( DirectorRanks <= 5, "Yes", "No" )
```

If you *must* store as a calculated column (static snapshot), you can create a column with similar logic, but measures are preferred for dynamic reporting.

7) Model / relationship & performance notes

- Ensure `Date[Date]` is related to `Movies[Release Date]` (one-to-many: `Date` → `Movies`).

- If you have very large data, create aggregated tables in Power Query (pre-aggregation) and avoid expensive iterators over the entire Movies table inside visuals showing many rows.
 - Mark Date table as Date Table.
-

8) Visualization layout & configuration

Page 1 — Overview Dashboard

- **Card:** Total Box Office (format as currency).
- **Card:** Average Margin % (format as percentage).
- **Card:** Movies with Oscars.
- **Bar Chart:** Axis = Movies[Genre], Value = Total Box Office. Add Legend = Movies[Certificate] (to stack by certificate). Sort by value descending.
- **Line Chart:** Axis = Date[Year] (or Date[Date] with Year granularity), Value = Total Box Office. Ensure continuous X axis if you want trendline.
- **Slicers:** Movies[Country] (list) and Date[Date] (between/range).
- **KPI Visual:** Indicator = Total Box Office (or current year value) ; Trend = BoxOffice LY or show YoY Box Office %. Set Target = 0% or create a target measure TargetYoY = 0.
 - Configure KPI to display green when YoY > 0.

Page 2 — Director Analysis

- **Treemap:** Group by Movies[Director], Values = SUM(Budget) ; Color by SUM(OscarWins) (or Movies[OscarWins] aggregated).
- **Table:** Columns: Director, Total Nominations (use CALCULATE(SUM(Nominations)...) or create measure), Total Oscars (SUM(OscarWins)), Avg Nominations per Director (the measure above).
- **Slicer:** Movies[Genre].
- **Donut Chart:** RunTime Category (calculated column) for selected director; Value = Count of movies or sum of BoxOffice.

Page 3 — Genre & Country Insights

- **Matrix:** Rows = Movies[Genre], Columns = Movies[Country], Values = Total Box Office.
 - Conditional formatting: Format by color scale on values (Home → Conditional formatting → Color scale).
 - **Pie Chart:** Movies[Certificate] share by Total Box Office.
 - **Word Cloud:** Install custom Word Cloud from marketplace → use Genre and size by Total Box Office or count.
 - **Slicer:** RunTime Category.
-

9) Small UX / formatting tips

- Use **tooltips**: add measures like Profit, Average Budget to tooltip fields of charts.
 - Sort bar charts by measure (Total Box Office) rather than alphabetically.
 - Use **Drill-through** page to open Director or Movie details.
 - Use **Bookmarks** to store filter combinations for presentations.
-

10) Quick optimization tips specific to this model

1. Use **Measures not Calculated Columns** for aggregations that must obey filters — measures are evaluated in engine context and are usually faster/more flexible.
 2. Use **SUM / AVERAGE rather than iterators** where possible. Use `AVERAGEX` only when you must compute per-row or per-entity expressions.
 3. **Avoid row-by-row operations on visuals** showing many rows — pre-aggregate in Power Query when possible.
 4. **Mark Date table** and use built-in time intelligence functions (`SAMEPERIODLASTYEAR`, `DATESINPERIOD`) to leverage engine optimizations.
 5. **Minimise use of ALL** and large FILTERs in visuals where possible — prefer `REMOVEFILTERS` or scoped filters.
-

11) Example extra measures you may need for visuals

Total Nominations (measure)

```
Total Nominations =  
SUM( Movies[Nominations] )
```

Total Oscars (measure)

```
Total Oscars =  
SUM( Movies[OscarWins] )
```

Profit (measure)

```
Profit (Measure) =  
SUMX( Movies, COALESCE( Movies[BoxOffice], 0 ) - COALESCE( Movies[Budget], 0 ) )
```

Profit Margin % (measure for KPI or card)

```
Profit Margin % (Measure) =  
DIVIDE( [Profit (Measure)], SUM( Movies[BoxOffice] ) )
```

12) Final checklist before publishing

- Verify relationships and set cross-filter direction to single (Date → Movies).
- Format all measures (currency, percent, thousands).
- Test slicers and interactions (Edit interactions to show/hide visuals if needed).
- Add descriptive titles and tooltips for each visual.