Key Metrics Calculation in DAX

▼ Power BI Report, Page 1

🌟 Total revenue:

```
Total_revenue = SUM(Listings[Price_in_usd])
```

🌟 Total average price:

```
total_price_avg = CALCULATE([price_usd_avg], ALL(Listings))
```

🌟 Most visited city:

```
Most_visited_city =
VAR TopCityTable = TOPN(1, ALL(DimCity), [Accomodates_total], DESC)
RETURN
CONCATENATEX(TopCityTable, DimCity[city], ", ")
```

🌟 Quantity of new listings:

```
New_Listing Qty = COUNT(Listings[listing_id])
```

🌟 Quantity of new hosts:

```
New_Host Quantity = DISTINCTCOUNT(Listings[host_id])
```

🌟 New Listings Quantity depends on the selected year (for title):

```
New_Listing Qty for title =
VAR user_year = SELECTEDVALUE('Calendar'[Year])
VAR last_year_list_qty = CALCULATE(COUNT(Listings[listing_id]),
                                   FILTER('Calendar', 'Calendar'[Year] = 2021))
RETURN
IF(user_year = BLANK(), last_year_list_qty, [New_Listing Qty])
```

🌟 New Host Quantity depends on the selected year (for title):

```
New_Host Qty for title =
VAR user_year = SELECTEDVALUE('Calendar'[Year])
VAR last_year_host_qty = CALCULATE(DISTINCTCOUNT(Listings[host_id]),
                                   FILTER('Calendar', 'Calendar'[Year] = 2021))
```

```
RETURN
IF(user_year = BLANK(), last_year_host_qty, [New_Host Quantity])
```

🌟 Total listings quantity as a cumulative quantity:

```
Comul_listing_qty =
CALCULATE([New_Listing Qty],FILTER(ALL('Calendar'[Date]),
            'Calendar'[Date]<=MAX('Calendar'[Date])))
```

🌟 Total hosts quantity as a cumulative quantity:

```
Comul_host_qty =
CALCULATE([New_Host Quantity],FILTER(ALL('Calendar'[Date]),
                'Calendar'[Date]<=MAX('Calendar'[Date])))
```

🌟 Total guests:

```
Accomodates_total = SUM(Listings[accommodates])
```

🌟 Quantity of old (already existing) listings for the selected year:

```
Old_comul_list_qty = [Comul_listing_qty] - [New_Listing Qty]
```

🌟 Average price:

```
price_usd_avg = DIVIDE([price_usd],[New_Listing Qty])
```

▼ **Power BI Report, Page 2**

🌟 Quantity of hosts who have a profile picture:

```
Host_profile_pic_qty =
CALCULATE([Comul_host_qty], FILTER(DimHost, DimHost[host_has_profile_pic]="t"))
```

🌟 Quantity of hosts who have a verified id:

```
Host_id_verified_qty =
CALCULATE([Comul_host_qty], FILTER(DimHost, DimHost[host_identity_verified]="t"))
```

🌟 Quantity of hosts who are a superhost:

```
Host_is_superhost_qty =
CALCULATE([Comul_host_qty], FILTER(DimHost, DimHost[host_is_superhost]="t"))
```

🌟 Quantity of hosts filtered by Response time (withing an hour, withing a few hours, withing a few days, a few days or more):

```
Response_time_host_qty =
VAR response_time = SELECTEDVALUE('DimHost'[host_response_time])
RETURN
IF(ISBLANK(response_time), CALCULATE([New_Host Quantity], ALL(DimHost)),
CALCULATE([New_Host Quantity],
                FILTER(DimHost, DimHost[host_response_time] = response_time)))
```

🌟 Host Response Rate by Rates or Cities:

```
Response_rate_host_qty =
VAR user_rate = SELECTEDVALUE('Rate'[rates])
RETURN
IF(ISBLANK(user_rate), CALCULATE([New_Host Quantity], ALL(DimHost)),
CALCULATE([New_Host Quantity],
FILTER(DimHost, DimHost[host_response_rate] <= VALUE(user_rate) &&
            DimHost[host_response_rate] > VALUE(user_rate - 0.1))))
```

🌟 Host Response Rate by Rates or Cities:

```
Response_rate_host_qty_optim =
VAR user_rate = SELECTEDVALUE('Rate'[rates])
VAR lower_bound = VALUE(user_rate - 0.1)
RETURN
IF(
ISBLANK(user_rate),
CALCULATE([New_Host Quantity], ALL(DimHost)),
CALCULATE(
[New_Host Quantity],
FILTER(DimHost, DimHost[host_response_rate] <= user_rate &&
            DimHost[host_response_rate] > lower_bound)))
```

🌟 Hosts' Acceptance Rate by Rates or Cities:

```
Acceptance_rate_host_qty =
VAR user_rate = SELECTEDVALUE('Rate'[rates])
RETURN
IF(ISBLANK(user_rate), CALCULATE([New_Host Quantity], ALL(DimHost)),
CALCULATE([New_Host Quantity],
FILTER(DimHost, DimHost[host_acceptance_rate] <= VALUE(user_rate) &&
        DimHost[host_acceptance_rate] > VALUE(user_rate - 0.1))))
```

🌟 Scores Accuracy (for Ratings):

```
Scores_accuracy =
DIVIDE(SUM(Listings[review_scores_accuracy]), CALCULATE([New_Listing Qty],
            FILTER(Listings, Listings[review_scores_accuracy] > 0)))
```

🌟 Scores Checkin (for Ratings):

```
Scores_checkin =
DIVIDE(SUM(Listings[review_scores_checkin]), CALCULATE([New_Listing Qty],
                FILTER(Listings, Listings[review_scores_checkin] > 0)))
```

🌟 Scores Cleanliness (for Ratings):

```
Scores_cleanliness =
DIVIDE(SUM(Listings[review_scores_cleanliness]), CALCULATE([New_Listing Qty],
              FILTER(Listings, Listings[review_scores_cleanliness] > 0)))
```

🌟 Scores Communication (for Ratings):

```
Scores_communication =
DIVIDE(SUM(Listings[review_scores_communication]), CALCULATE([New_Listing Qty],
          FILTER(Listings, Listings[review_scores_communication] > 0)))
```

🌟 Scores Location (for Ratings):

```
Scores_location =
DIVIDE(SUM(Listings[review_scores_location]), CALCULATE([New_Listing Qty],
          FILTER(Listings, Listings[review_scores_location] > 0)))
```

🌟 Scores Rating (for Ratings):

```
Scores_rating =
DIVIDE(SUM(Listings[review_scores_rating]), CALCULATE([New_Listing Qty],
          FILTER(Listings, Listings[review_scores_rating] > 0)))
```

▼ **Power BI Report, Page 3**

🌟 Max listing's price:

```
MAX_price = MAX(Listings[Price_in_usd])
```

🌟 City with max listing's price:

```
MAX_price_city =
VAR maxprice = MAX(Listings[Price_in_usd])
RETURN
"City: " & CALCULATE(MAX(DimCity[city]),
FILTER(Listings, Listings[Price_in_usd] = maxprice))
```

🌟 Listing_id with max listing's price:

```
MAX_price_listing_id =
VAR maxprice = MAX(Listings[Price_in_usd])
RETURN
"Listing_id: " & CALCULATE(MAX(Listings[listing_id]),
FILTER(Listings, Listings[Price_in_usd] = maxprice))
```

🌟 Year with max listing's price:

```
MAX_price_year =
VAR maxprice = MAX(Listings[Price_in_usd])
RETURN
"Year: " & CALCULATE(MAX('Calendar'[Year]),
FILTER(Listings, Listings[Price_in_usd] = maxprice))
```

🌟 Dynamic TOP Amenities:

- Created a calculated table:

  ```
  TOP = GENERATESERIES(10, 50, 5)
  ```

- Created a slicer with a field `TOP[Value]`
- Quantity of listings in Listings-Amenities table:

  ```
  Listing_amenit = COUNT('Listings-Amenities'[listing_id])
  ```

- Listing's amenity %:

  ```
  Listing_amenit % = [Listing_amenit] / [New_Listing Qty]
  ```

- Amenity Rank:

  ```
  amenit_rank =
  RANKX(ALL(DimAmenities[Amenity]), [Listing_amenit],,DESC,Dense)
  ```

- Listing's amenity % for parameter:

  ```
  Listing_amenit_%_param =
  VAR param = SELECTEDVALUE('TOP'[Value])
  RETURN IF([amenit_rank] <= param, [Listing_amenit %], BLANK())
  ```

- Listing's amenity parameter:

  ```
  Listing_amenit_param =
  VAR param = SELECTEDVALUE('TOP'[Value])
  RETURN
  IF([amenit_rank] <= param, [Listing_amenit], BLANK())
  ```

- Average price by amenities:

```
price_avg_by_amenities =
VAR amenit = SELECTEDVALUE(DimAmenities[Amenity])
VAR total_price_by_amenit = CALCULATE(SUM(Listings[Price_in_usd]),
      FILTER('Listings-Amenities','Listings-Amenities'[amenity_cleaned]=amenit))
VAR total_list_by_amenit = CALCULATE(COUNT(Listings[Price_in_usd]),
      FILTER('Listings-Amenities','Listings-Amenities'[amenity_cleaned]=amenit))
VAR avg_price = DIVIDE(total_price_by_amenit, total_list_by_amenit)
VAR param = SELECTEDVALUE('TOP'[Value])
RETURN
IF([amenit_rank] <= param, avg_price, BLANK())
```

🌟 Dynamic visual titles depends on selected year and/or selected city :

```
Amenity_type_avg_price =
IF(
ISBLANK(SELECTEDVALUE(DimCity[city])) && ISBLANK(SELECTEDVALUE('Calendar'[Year])),
"Amenity Type and Amenities Average price",
IF(ISBLANK(SELECTEDVALUE(DimCity[city])),
"Amenity Type and Amenities Average price in " & (SELECTEDVALUE('Calendar'[Year])),
IF(ISBLANK(SELECTEDVALUE('Calendar'[Year])),
"Amenity Type and Amenities Average price in " & (SELECTEDVALUE(DimCity[city])),
"Amenity Type and Amenities Average price in " &
(SELECTEDVALUE('Calendar'[Year]) & " in " & (SELECTEDVALUE(DimCity[city])))))
```

▼ **Power BI Report, Page 4**

🌟 Number of reviews:

```
Review_number = COUNT(Reviews[review_id])
```

🌟 Number of reviewers:

```
Reviewers_number = DISTINCTCOUNT(Reviews[reviewer_id])
```

🌟 Total listings with reviews:

```
review_listing_number = DISTINCTCOUNT(Reviews[listing_id])
```

🌟 A measure for dynamically color-coding the top season for each city in the visual **City's Top Season by Number of Reviews**:

```
Color_SeasonForCity =
VAR MaxReviewPerSeason = CALCULATE(MAXX(ALL(Calendar_review[Season]),
            [Review_number]),VALUES(DimCity[city]))
VAR CurrentValue = [Review_number]
```

```
RETURN
IF(CurrentValue = MaxReviewPerSeason, 1, 0)
```

🌟 A measure for dynamically color-coding the Number of reviews in **Number of reviews Heatmap**:

```
Normalized_Review =
VAR CurrentYear = SELECTEDVALUE(Calendar_review[Year])
VAR MonthlyReviewCounts = SUMMARIZE(FILTER(ALL(Calendar_review),
      Calendar_review[Year] = CurrentYear), Calendar_review[Month],
      "ReviewCount", CALCULATE(COUNT(Reviews[review_id])))
VAR MinValue = MINX(MonthlyReviewCounts, [ReviewCount])
VAR MaxValue = MAXX(MonthlyReviewCounts, [ReviewCount])
VAR CurrentValue = [Review_number]
VAR Scale =
IF(
MaxValue = MinValue,
0,
DIVIDE(CurrentValue - MinValue, MaxValue - MinValue))
RETURN
SWITCH(
TRUE(),
ISBLANK(Scale), BLANK(),
Scale <= 0.05, "#FFD2D6",
Scale <= 0.15, "#FCC6C9",
Scale <= 0.30, "#ffb3b6",
Scale <= 0.45, "#ff7a80",
Scale <= 0.60, "#ff5a5f",
Scale <= 0.75, "#d94d54",
Scale <= 0.90, "#b83e44",
"#A1343C")
```

▼ **Power BI Report, Page 5**

🌟 Calculated column `Price_per_guest` in `Listings`:

```
Price_per_guest = DIVIDE(Listings[Price_in_usd], Listings[accommodates])
```

🌟 Average price per guest:

```
Price_per_guest_AVG = AVERAGE(Listings[Price_per_guest])
```

🌟 Average review scores value:

```
Review_scores_value_AVG = AVERAGE(Listings[review_scores_value])
```

🌟 Min price per guest:

```
MIN_price_per_guest = MIN(Listings[Price_per_guest])
```

🌟 Max price per guest:

```
MAX_price_per_guest = MAX(Listings[Price_per_guest])
```

🌟 City rating: create a composite "value" metric that gives a normalized score of rating per unit of cost:

```
Norm_value_score = DIVIDE([Review_scores_value_AVG],[Price_per_guest_AVG])
```

🌟 Best city for travel:

```
Best_city_for_travel =
VAR TopCityTable = TOPN(1, ALL(DimCity), [Norm_value_score], DESC)
RETURN
CONCATENATEX(TopCityTable, DimCity[city], ", ")
```

🌟 Dynamic title for Best city for travel by years:

```
Best_city_for_travel_by_years =
IF(SELECTEDVALUE('Calendar'[Year])=BLANK(),
"Best City for Travel: ",
"Best City for Travel in " & (SELECTEDVALUE('Calendar'[Year]) & ": "))
```

🌟 Dynamic title for Average price per guest by years:

```
Average_price_per_guest_by_years =
IF(SELECTEDVALUE('Calendar'[Year])=BLANK(),
"Average Price per Guest",
"Average Price per Guest in " & (SELECTEDVALUE('Calendar'[Year])))
```

🌟 Dynamic formatting for scatter plot

- City with MAX Review_scores_value_AVG:

```
TOP MAX City by Review_scores_value_AVG =
CALCULATE(MAXX(
TOPN(
1,
ADDCOLUMNS(
SUMMARIZE(ALLSELECTED(Listings), Listings[city]),
"AvgValue", [Review_scores_value_AVG]),
```

```
[AvgValue], DESC),
Listings[city]), ALLSELECTED(Listings))
```

- City with MIN Price_per_guest_AVG:

```
TOP MIN City by Price_per_guest_AVG =
CALCULATE(MAXX(
TOPN(
1,
ADDCOLUMNS(
SUMMARIZE(ALLSELECTED(Listings), Listings[city]),
"AvgValue", [Price_per_guest_AVG]),
[AvgValue], ASC),
Listings[city]), ALLSELECTED(Listings))
```

- Measure to assign colors to data points in the 'Markers' section of the Scatter Plot formatting pane:

```
TOP_2_categories =
IF(
SELECTEDVALUE(Listings[city]) = [TOP MIN City by Price_per_guest_AVG] ||
SELECTEDVALUE(Listings[city]) = [TOP MAX City by Review_scores_value_AVG],
1, 0)
```