1. Edit interactions in power BI:

Edit interaction allows the control visual elements on a report page to interact with one another. When a filter from one visual is selected it must also filter other visuals based on the selected data.

We have 3 types of interactions between visuals:

1. **Filter**: When a visual is set to filter, selecting a data point in one visual will filter other visuals on the page to display only the data related to that selection.
2. **Highlight**: When a visual is set to Highlight, selecting a data point in one visual will highlight related data points in other visuals, while still displaying the complete dataset.
3. **None**: When a visual is set to None, selecting data points in other visuals will not affect it. The visual remains unchanged regardless of interactions from other visuals.
4. Difference between Merge and Append:

|  |  |
| --- | --- |
| **Append** | **Merge** |
| Combine rows from multiple tables into one table | Join tables based on a common column |
| Requires similar column structures | Can have different structures |
| Adds rows vertically | Adds columns horizontally |
| Consolidates similar data sets | Enriching data with related information |

1. Why it is not preferable to use many to many relationships:

* It is because of their complexity and potential impact on performance and data accuracy.
* It adds complexity in data modeling.
* Many-to-many relationships increase the complexity of DAX calculations, leading to performance degradation, and lack accuracy.
* Many-to-many relationships can lead to ambiguous or unexpected results when aggregating data. For example, a single data point might be counted multiple times if the relationships are not handled properly.
* Many-to-many relationships complicate the filter context, making it harder to predict how filters will propagate through the model and affect results in visuals.

To solve this, it is preferable to use a bridge table between those two tables.

1. How do we handle the values if we do not have the same number of columns in append queries?

When appending queries in Power BI, if the tables involved do not have the same number of columns, Power BI aligns columns based on their names and handles any discrepancies by filling in missing values with null.

1. Use case of fuzzy match in merge query:

* Fuzzy matching in Power BI’s merge query operation allows to join tables based on similar, but not exactly matching, text strings. This can be extremely useful in scenarios where data may not be consistently formatted, such as when dealing with user inputs, imported data from multiple sources, or legacy systems.
* Fuzzy matching helps to overcome inconsistencies and variations in data entry, leading to more accurate data integration.
* Automates the process of finding similar records, reducing the need for manual reconciliation.
* By combining records that would otherwise remain unlinked, fuzzy matching enhances the completeness of your datasets.

1. What is pivot and unpivot in power BI? Are they exact reverse of each other?

Pivot and unpivot are data transformation techniques.

**Pivot**: Pivot is a process where rows are transformed into columns. In pivot operation, unique values of columns are turned into new columns.

Ex:

|  |  |  |
| --- | --- | --- |
| **Product** | **Month** | **Sales** |
| A | Jan | 100 |
| A | Feb | 150 |
| B | Jan | 200 |
| B | Feb | 250 |

Table after pivoting:

|  |  |  |
| --- | --- | --- |
| **Product** | **Jan** | **Feb** |
| A | 100 | 150 |
| B | 200 | 250 |

**Unpivot**: It is the opposite of pivoting, where columns are transformed into rows. Unpivoting involves taking multiple columns and converting them into attribute-value pairs, which helps in creating a more compact and easier-to-analyze format.

|  |  |  |
| --- | --- | --- |
| **Product** | **Jan** | **Feb** |
| A | 100 | 150 |
| B | 200 | 250 |

After unpivoting:

|  |  |  |
| --- | --- | --- |
| **Product** | **Month** | **Sales** |
| A | Jan | 100 |
| A | Feb | 150 |
| B | Jan | 200 |
| B | Feb | 250 |

* While pivot and unpivot are conceptually opposite operations, they are not always exact reverses of each other. This is because unpivoting a pivoted table might not yield the original table if any aggregations or transformations were applied during the pivot process.
* For example, if the pivot operation involved aggregating data (like summing up values), unpivoting the result won't retrieve the individual data points that were aggregated.
* However, if no aggregations or transformations are applied, and all columns involved in the pivot and unpivot processes are included, then they can act as reverse operations.

1. Difference between measure and calculated columns.

Measure:

* Measures are dynamic calculations used to aggregate data. They are evaluated in the context of the filters applied to a report or visual.
* Measures are calculated based on the filter context of the report or visual where they are used. It means their values can change depending on how data is filtered or sliced.
* Measures are typically used for aggregations such as sum, average, count etc.
* Measures do not physically add data to the table, they are calculated whenever the report is rendered or filtered.

Ex: Total Sales = SUMX(Sales, Sales[Quantity] \* Sales[Price])

This measure dynamically calculates the total sales based on the current filters applied to the report.

Calculated Columns:

* Calculated columns are static calculations added to the table to create new data that can be used in the reports.
* Calculated row by row, their values remain constant regardless of the filter context.
* The values of the calculated columns store in the mode. Therefore, increases model’s size.
* Calculated columns are used for categorizing or grouping data. They can be used as a filter or a slicer in report.
* Since they are calculated during data refresh, their values won’t change until the model is refreshed.

Ex: Total Price = Sales[Quantity] \* Sales[Price]

This calculated column adds a new "Total Price" column to the table, with a static value for each row.

### **When to Use Measures vs. Calculated Columns**

**Use Measures When:**

* You need calculations that change based on the report context (e.g., dynamic aggregations).
* You want to keep the data model efficient and avoid increasing its size unnecessarily.
* The calculation involves large datasets or complex aggregations.

**Use Calculated Columns When:**

* You need static calculations or additional columns for categorization and grouping.
* The calculation is simple and does not require context-based evaluation.
* You plan to use the result in slicers, filters, or as a category in visuals.