

REPORT & SCREENSESHOTS

Tool Used: Microsoft Power BI

Dataset: Brazilian E-Commerce Public Dataset by Olist (From Kaggle) **Link:**

<https://www.kaggle.com/datasets/olistbr/brazilian-e-commerce>

Reason for Choosing the dataset

The Brazilian E-Commerce Public Dataset by Olist from Kaggle is ideal for this assignment. It meets all the necessary requirements, featuring over 100,000 rows in the orders table, multiple related tables, and numeric measures such as price, freight value, and payment value. Additionally, it includes important date columns like purchase date, delivery date, and estimated delivery date, making it a comprehensive and realistic business dataset.

The screenshot shows the Kaggle website interface. On the left, there's a sidebar with navigation links: Create, Home, Competitions, Datasets (selected), Models, Benchmarks, Game Arena, Code, Discussions, Learn, More, Your Work, and VIEWED. The main content area displays the dataset details for "olist_customers_dataset.csv". The title is "Brazilian E-Commerce Public Dataset by Olist". Below the title, there are tabs for Data Card, Code (691), Discussion (67), and Suggestions (0). A large preview table for "olist_customers_dataset.csv" is shown, containing columns: customer_id, customer_unique_id, customer_zip_code_prefix, customer_city, and customer_state. The table includes a histogram for customer_zip_code_prefix and a breakdown of customer_state percentages (SP: 16%, RJ: 7%, Other: 77%). The Data Explorer section lists other datasets in the dataset version. The Summary section indicates 9 files and 52 columns.

customer_id	customer_unique_id	# customer_zip_code_prefix	customer_city	customer_state
99441 unique values	96096 unique values	first five digits of customer zip code	sao paulo rio de janeiro Other (77019)	SP 16% RJ 7% Other (44843) 42%
38e2577742c5	1141171f5801			
ea2196dc456ba36fe4f6b81dca4867d4	4a4de987b37555970ffc9688d858a72	44033	feira de santana	BA
09241c552e9fe2428997a6c535e9d408	44e9a1246448bd68a2e3bf0f1966c57a	04537	sao paulo	SP
e50a30de3c32f9406a7185f40ce6874d	b4d6e1b900d99b52e901860bc1f44e35	71540	brasilia	DF

Data Explorer
Version 2 (126.19 MB)
olist_customers_dataset.csv
olist_geolocation_dataset.csv
olist_order_items_dataset.csv
olist_order_payments_dataset.csv
olist_order_reviews_dataset.csv
olist_orders_dataset.csv
olist_products_dataset.csv
olist_sellers_dataset.csv
product_category_name_trans

Summary
9 files
52 columns

SECTION A — DATA PREPARATION (18 Marks)

Q1. Data Profiling and Quality Checks (4 Marks)

The Olist dataset was downloaded from Kaggle, extracted, and six CSV files were imported into Power BI using:

Home → Get Data → Text/CSV → Transform Data.

The following tables were renamed for proper star schema modelling:

Original Name	Renamed To
<u>olist_orders_dataset</u>	<u>Fact Orders</u>
<u>olist_order_items_dataset</u>	<u>Fact OrderItems</u>
<u>olist_order_payments_dataset</u>	<u>Fact Payments</u>
<u>olist_customers_dataset</u>	<u>Dim Customers</u>
<u>olist_products_dataset</u>	<u>Dim Products</u>
<u>olist_sellers_dataset</u>	<u>Dim Sellers</u>

The screenshot shows the Power Query Editor interface with the following details:

- File Bar:** Untitled - Power Query Editor, File, Home, Transform, Add Column, View, Tools, Help.
- Toolbars:** Close & Apply, New Source, Recent Sources, Enter Data, Data source settings, Manage Parameters, Export query results, Refresh Preview, Properties, Advanced Editor, Manage Columns, Manage Rows, Sort.
- Queries [6]:** olist_orders_dataset, olist_order_items_dataset, olist_order_payments_dataset, olist_customers_dataset, olist_products_dataset, olist_sellers_dataset. The olist_sellers_dataset is selected.
- Preview Area:** Shows a table with one column labeled "seller_id". The properties for this column are: Valid (100%), Error (0%), Empty (0%). It also states "1000 distinct, 1000 unique". Below the preview are 15 rows of sample data.
- Query Settings:**
 - PROPERTIES:** Name is set to "olist_sellers_dataset".
 - APPLIED STEPS:** Shows a step named "Changed Type".
- Bottom Status:** 4 COLUMNS, 999+ ROWS, Column profiling based on top 1000 rows, PREVIEW DOWNLOADED AT 2:25 PM.

Queries [6]

- Fact_Orders
- Fact_OrderItems
- Fact_Payments
- Dim_Customers
- Dim_Products
- Dim_Sellers**

Properties

Name: Dim_Sellers

Applied Steps

Source: Promoted Headers

Changed Type

Query Settings

PREVIEW DOWNLOADED AT 2:31 PM

Data profiling tools were enabled in Power Query Editor by activating:

- Column Quality
- Column Distribution
- Column Profile (Entire Dataset)

Queries [6]

- Fact_Orders
- Fact_OrderItems
- Fact_Payments
- Dim_Customers
- Dim_Products
- Dim_Sellers

Properties

Name: Fact_Orders

Applied Steps

Source: Promoted Headers

Changed Type

Query Settings

PREVIEW DOWNLOADED AT 2:28 PM

Identified Data Quality Issues

1. Missing Delivery Dates

Table: Fact_Orders

Column: order_delivered_customer_date

Column Quality showed null values in this column. These represent undelivered or cancelled orders.

Fix Applied:

No rows were deleted. The null values were preserved because they reflect real business scenarios. Instead, a calculated column was later created to compute delivery days only when the value is not null.

2. Blank Product Categories

Table: Dim_Products

Column: product_category_name

Column profiling showed missing values.

Fix Applied:

Transform → Replace Values

Null values were replaced with "Unknown".

Applied Step: Replaced Value

3. Messy City Names

Table: Dim_Customers

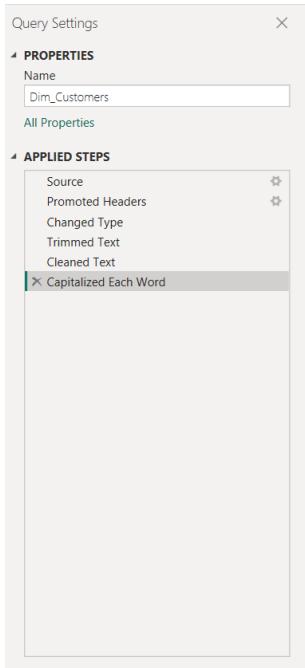
Column: customer_city

The column contained lowercase text and spacing inconsistencies.

Fix Applied (in order):

1. Transform → Format → Trim
2. Transform → Format → Clean
3. Transform → Capitalize Each Word

The column was renamed to:
Customer City



4. Incorrect Data Types

Several date and numeric columns were initially loaded as Text.

These were corrected in the next section.

Q2. Data Types and Locale Conversion (4 Marks)

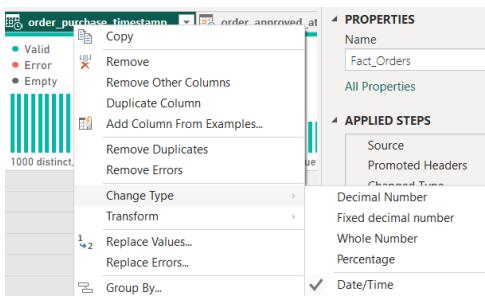
The following corrections were made:

Date Column

Table: Fact_Orders

Column: order_purchase_timestamp

Changed to: Date/Time



Whole Number

Table: Fact_Payments

Column: payment_installments

Changed to: Whole Number

The screenshot shows the Power BI Data Editor interface. On the left, there's a preview pane with a bar chart showing 12 distinct values. The main area shows the 'payment_installments' column with a dropdown menu open. The 'APPLIED STEPS' pane on the right shows a checked step for 'Whole Number' under the 'Change Type' section.

Decimal Number

Table: Fact_OrderItems

Column: price

Changed to: Decimal Number

The screenshot shows the Power BI Data Editor interface. On the left, there's a preview pane with a bar chart showing 470 distinct values. The main area shows the 'price' column with a dropdown menu open. The 'APPLIED STEPS' pane on the right shows a checked step for 'Decimal Number' under the 'Change Type' section.

Locale Conversion

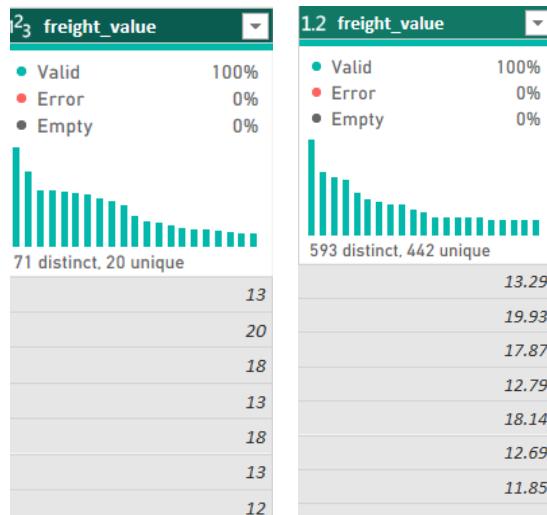
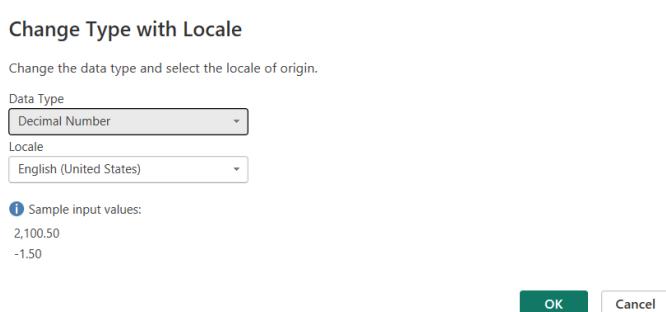
One numeric column that loaded as text was converted using:

Right Click → Change Type → Using Locale

Selected:

- Data Type: Decimal Number
- Locale: English (United States)

This ensured correct interpretation of decimal formatting.



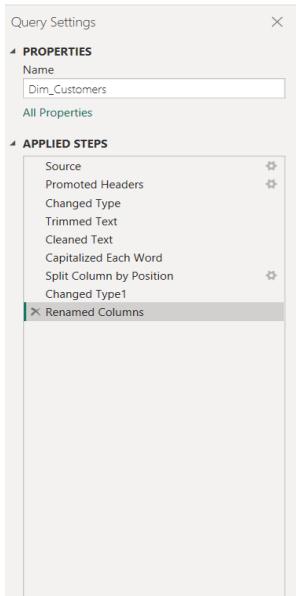
Q3. Text Standardization (4 Marks)

Text standardization was performed in the Dim_Customers table.

```
= Table.TransformColumnTypes(#"Split Column by Position",{{"customer_zip_code_prefix.1", Int64.Type}, {"customer_zip_code_prefix.2", Int64.Type}})
```

The following steps were applied:

1. Trim — removed leading and trailing spaces
2. Clean — removed hidden characters
3. Capitalize Each Word — standardized casing

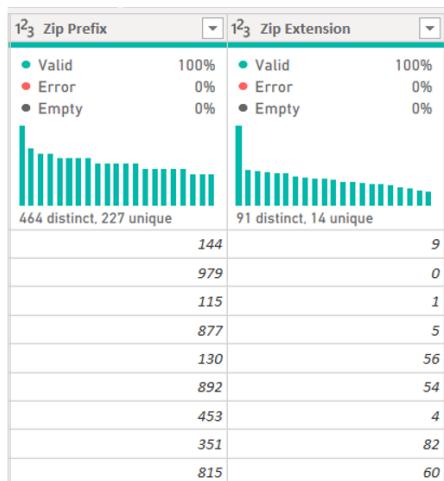


4. Additionally, the column customer_zip_code_prefix was split:

Transform → Split Column → By Number of Characters

The resulting columns were renamed:

- Zip Prefix
- Zip Extension



Q4. Conditional and Custom Columns (4 Marks)

1. Conditional Column

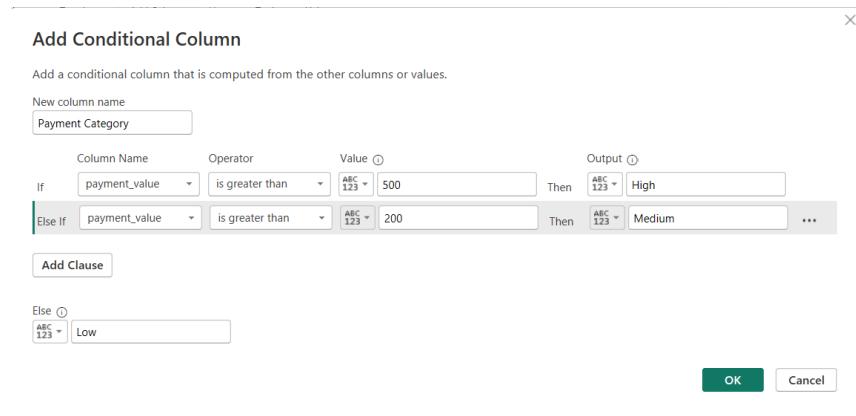
Created in Fact_Payments:

Column Name: Payment Category

Rules:

- $\text{payment_value} > 500 \rightarrow \text{High}$
- $\text{payment_value} > 200 \rightarrow \text{Medium}$
- Else $\rightarrow \text{Low}$

This categorizes payment size for analytical segmentation.



2. Custom Column (M Formula)

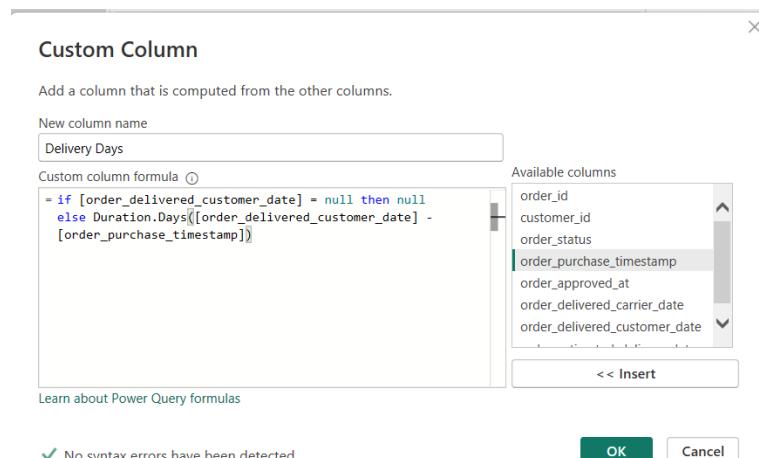
Created in Fact_Orders:

Column Name: Delivery Days

M Formula used:

```
if [order_delivered_customer_date] = null then null  
else Duration.Days([order_delivered_customer_date] - [order_purchase_timestamp])
```

This calculates delivery duration only for delivered orders.



Q5. Merge Queries (2 Marks)

A merge operation was performed between:

Fact_Orders and Dim_Customers

Join Type: Left Outer (All from Fact_Orders)

Join Key: customer_id

Merge

Select a table and matching columns to create a merged table.

Fact_Orders

order_id	customer_id	order_status	order_purchase_timestamp
e481f51cbdc54678b7cc49136f2d6af7	9ef432eb6251297304e76186b10a928d	delivered	10/2/2017 10:56:33 AM
53cdb2fc8bc7dce0b6741e2150273451	b0830fb4747a6c6d20dea0b8c802d7ef	delivered	7/24/2018 8:41:37 PM
47770eb9100c2d0c44946d9cf07ec65d	41ce2a54c0b03bf3443c3d931a367089	delivered	8/8/2018 8:38:49 AM
949d5b44dbf5de918fe9c16f97b45f8a	f88197465ea7920adcdbec7375364d82	delivered	11/18/2017 7:28:06 PM

Dim_Customers

customer_id	customer_unique_id	Zip Prefix	Zip Extension	Customer
06b8999e2fba1a1fbc88172c00ba8bc7	861eff4711a542e4b93843c6dd7febb0	144	9	Franca
18955e83d337fd6b2def6b18a428ac77	290c77bc529b7ac935b93aa66c333dc3	979	0	Sao Bernardo
4e7b3e00288586ebd08712fdd0374a03	060e732b5b29e181a18229c7b0b2b5e	115	1	Sao Paulo
b2b6027bc5c5109e529d4dc6358b12c3	259dac757896d24d7702b9acbbff3f3c	877	5	Mogi Das Cru

Join Kind

Left Outer (all from first, matching from second)

Use fuzzy matching to perform the merge

Fuzzy matching options

✓ The selection matches 99441 of 99441 rows from the first table.

OK Cancel

Only the following columns were expanded:

- Customer City & customer_state

Query Editor dialog showing the creation of a nested join:

```
= Table.NestedJoin(#"Added Custom", "Dim_Customers")
```

The table being joined is "Dim_Customers".

Search Columns to Expand:

Expand / Aggregate: Expand Aggregate

Selected Columns:

- (Select All Columns)
- customer_id
- customer_unique_id
- Zip Prefix
- Zip Extension
- Customer City
- customer_state

Properties Panel:

- Name: Fact_Orders
- All Properties

Applied Steps:

- Source
- Promoted Headers
- Changed Type
- Added Custom
- Merged Queries
- Expanded Dim_Customers

Buttons: OK, Cancel

A ^B Dim_Customers.Customer City	A ^B Dim_Customers.customer_state
● Valid 100%	● Valid 100%
● Error 0%	● Error 0%
● Empty 0%	● Empty 0%
	
4119 distinct, 1144 unique	27 distinct, 0 unique
Sao Paulo	SP
Franca	SP
Barreiras	BA
Sao Bernardo Do Campo	SP
Vianopolis	GO
Sao Paulo	SP
Sao Goncalo Do Amarante	RN
Mogi Das Cruzes	SP
Santo Andre	SP
Campinas	SP
Congonhinhas	PR
Jaragua Do Sul	SC
Santa Rosa	RS
Sao Paulo	SP
Nilopolis	RJ
Timoteo	MG
Faxinalzinho	RS
Curitiba	PR
Sorocaba	SP
Belo Horizonte	MG
Rio De Janeiro	RJ
Montes Claros	MG

SECTION B — DATA MODELLING (12 Marks)

Q6. Star Schema Identification (3 Marks)

The final data model follows a Star Schema design.

Fact Tables:

- Fact_Orders
- Fact_OrderItems
- Fact_Payments

Dimension Tables:

- Dim_Customers
- Dim_Products
- Dim_Sellers

Fact tables contain measurable numeric values and foreign keys, while dimension tables contain descriptive attributes.

For example:

Fact_OrderItems is classified as a Fact table because it contains measurable fields such as price and freight_value and links to multiple dimensions.

Dim_Customers is classified as a Dimension table because it contains descriptive customer attributes such as city and state.

Q7. Relationships (5 Marks)

Relationships were created in Model View with:

Edit relationship

Select tables and columns that are related.

From table
Fact_Payments

order_id	Payment Date...	payment_inst...	payment_seq...	payment_type	payment_value
a9810da8291...	Low	1	1	credit_card	24.39
25e8ea4e933...	Low	1	1	credit_card	65.71
771ea366b00...	Low	1	1	credit_card	81.16

To table
Fact_Orders

pprov...	order_deliver...	order_deliver...	order_estimat...	order_id	order_purcha...	order_status
117 8:1...	7/14/2017 6:4...	7/19/2017 2:0...	8/8/2017 12:0...	34513ce0c4fa...	7/13/2017 7:5...	delivered
117 3:0...	6/16/2017 2:5...	6/19/2017 6:5...	7/6/2017 12:0...	2edfd6d1f0b4...	6/13/2017 9:1...	delivered
8 10:3...	5/10/2018 1:3...	5/14/2018 6:5...	5/23/2018 12:...	e37797aedc7...	5/8/2018 10:1...	delivered

Cardinality: Many to one (*:1) **Cross-filter direction**: Single

Make this relationship active Apply security filter in both directions Assume referential integrity

Save **Cancel**

Relationship 1

Fact_Orders (order_id) → Fact_OrderItems (order_id)

Cardinality: One-to-Many

Cross Filter: Single

Active: Yes

Justification: One order can contain multiple order items.

Relationship 2

Fact_Orders (customer_id) → Dim_Customers (customer_id)

Cardinality: One-to-Many

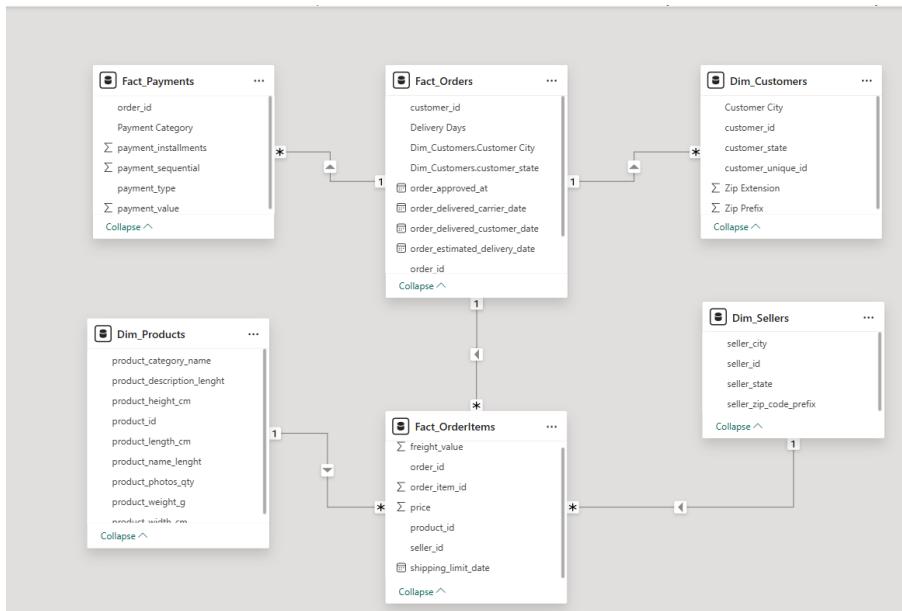
Cross Filter: Single

Active: Yes

Justification: One customer may place multiple orders.

Modelling Mistake Example

If relationships were set to Many-to-Many or Bi-Directional filtering, it could create ambiguous filter paths, leading to double counting of revenue and incorrect totals.



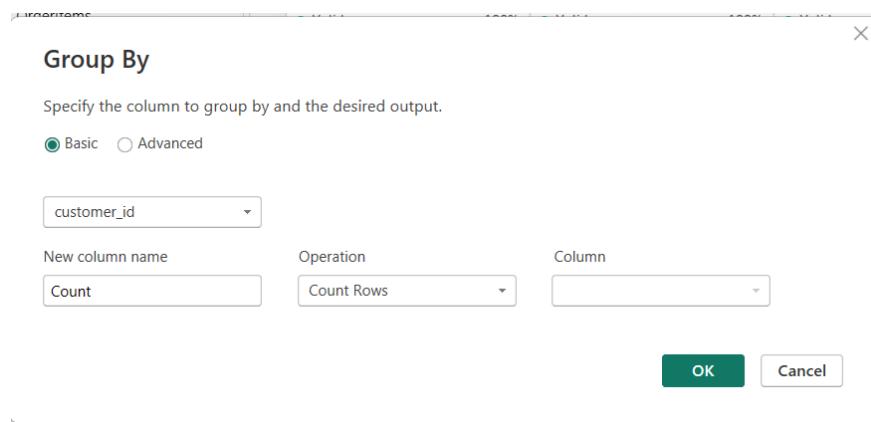
Q8. Key Uniqueness Verification (2 Marks)

The column `customer_id` in `Dim_Customers` was tested using:

Transform → Group By → Count Rows

No duplicate keys were found.

If duplicates had existed, they would have been removed to preserve referential integrity.



The screenshot shows the Power BI Data View interface. On the left is a table named "customer_id" with one column. The first row shows summary statistics: 1000 distinct, 1000 unique. The second row shows detailed counts for three categories: Valid (100%), Error (0%), and Empty (0%). The main body of the table lists 13 rows of customer IDs, each with a count of 1. To the right of the table is a "APPLIED STEPS" pane, which lists several data processing steps: Source, Promoted Headers, Changed Type, Trimmed Text, Cleaned Text, Capitalized Each Word, Split Column by Position, Changed Type1, Renamed Columns, and Grouped Rows.

Q9. Date Table Decision (2 Marks)

Should a date table exist? Yes

A Date Table was created using DAX:

The table was marked as a Date Table and related to Fact_Orders.

The screenshot shows the Power BI Data View interface. On the left is a code editor window displaying DAX code for creating a Date table. The code defines a variable Dim_Date and adds columns for Year, Month Number, Month Name, and Quarter. On the right is a "Data" pane, which contains a search bar and a list of tables. The "Dim_Date" table is selected and has a checkmark next to its "Date" column, indicating it is marked as a Date table. Other tables listed include Dim_Customers, Dim_Products, Dim_Sellers, Fact_OrderItems, Fact_Orders, and Fact_Payments.

```

1 Dim_Date =
2 ADDCOLUMNS(
3     CALENDAR(
4         MIN(Fact_Orders[order_purchase_timestamp]),
5         MAX(Fact_Orders[order_purchase_timestamp])
6     ),
7     "Year", YEAR([Date]),
8     "Month Number", MONTH([Date]),
9     "Month Name", FORMAT([Date], "MMMM"),
10    "Quarter", "Q" & FORMAT([Date], "Q")
11 )
12

```

Reasons for creating a Date Table:

1. Enables time intelligence calculations (YTD, MTD, MoM).
2. Ensures consistent time-based filtering across the model.

Edit relationship

Select tables and columns that are related.

From table

Fact_Orders

pprov...	order_deliver...	order_deliver...	order_estimat...	order_id	order_purcha...	order_status
17 8:1...	7/14/2017 6:4...	7/19/2017 2:0...	8/8/2017 12:0...	34513ce0cfa...	7/13/2017 7:5...	delivered
17 3:0...	6/16/2017 2:5...	6/19/2017 6:5...	7/6/2017 12:0...	2edfd6d1f0b4...	6/13/2017 9:1...	delivered
8 10:3...	5/10/2018 1:3...	5/14/2018 6:5...	5/23/2018 12:...	e37797aedc7...	5/8/2018 10:1...	delivered

X

To table

Dim_Date

Date	Month Name	Month Number	Quarter	Year
7/1/2017 12:0...	July	7	Q3	2017
7/2/2017 12:0...	July	7	Q3	2017
7/3/2017 12:0...	July	7	Q3	2017

Cardinality

Many to one (*:1)

Cross-filter direction

Single

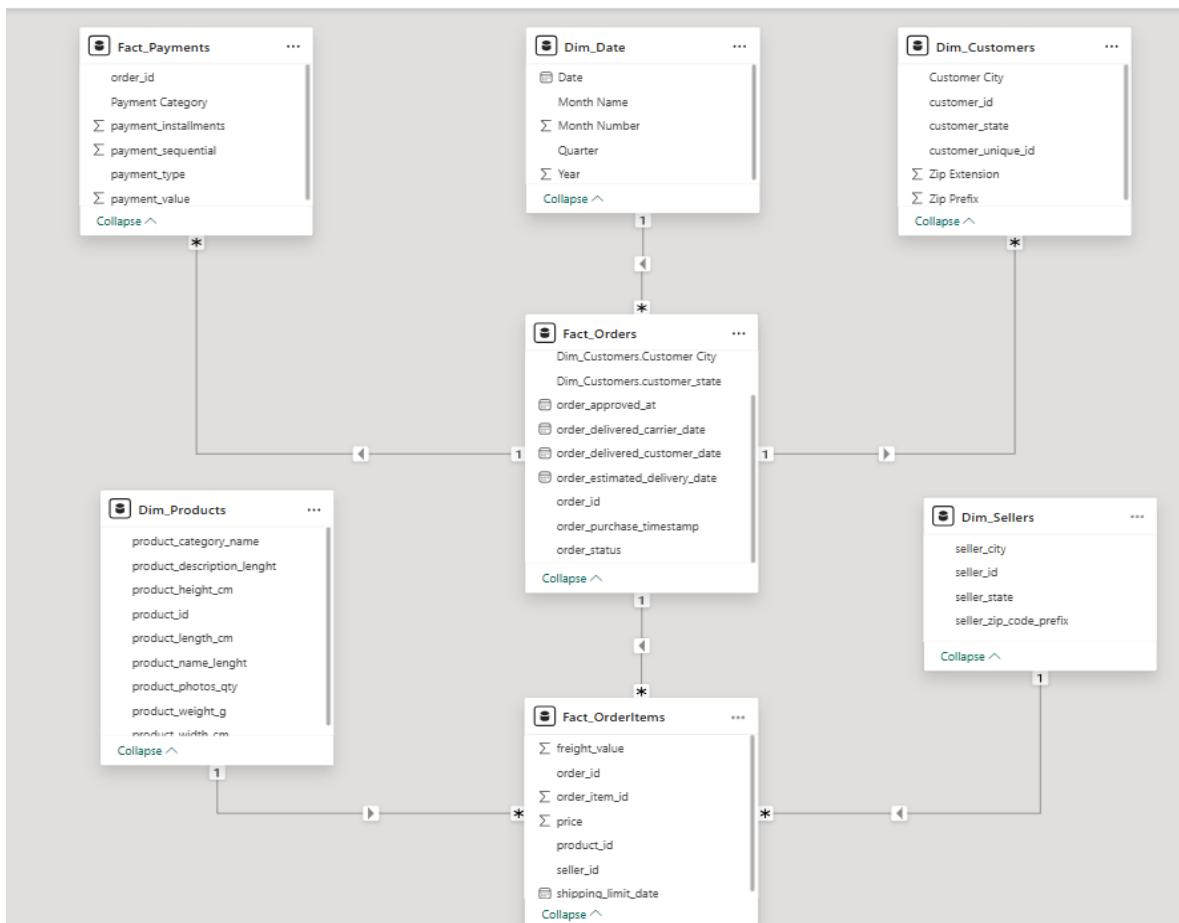
Make this relationship active

Apply security filter in both directions

Assume referential integrity

Save

Cancel



DATA DICTIONARY

Table	Column	Meaning	Type	Key/Attribute
Fact_Orders	order_id	Unique order identifier	Text	Primary Key
Fact_OrderItems	price	Item selling price	Decimal	Measure
Fact_Payments	payment_value	Total payment value	Decimal	Measure
Dim_Customers	customer_id	Unique customer ID	Text	Primary Key
Dim_Products	product_id	Unique product ID	Text	Primary Key
Dim_Sellers	seller_id	Unique seller ID	Text	Primary Key
Fact_Orders	order_purchase_timestamp	Purchase date	DateTime	Attribute
Fact_Orders	Delivery Days	Delivery duration	Whole Number	Calculated