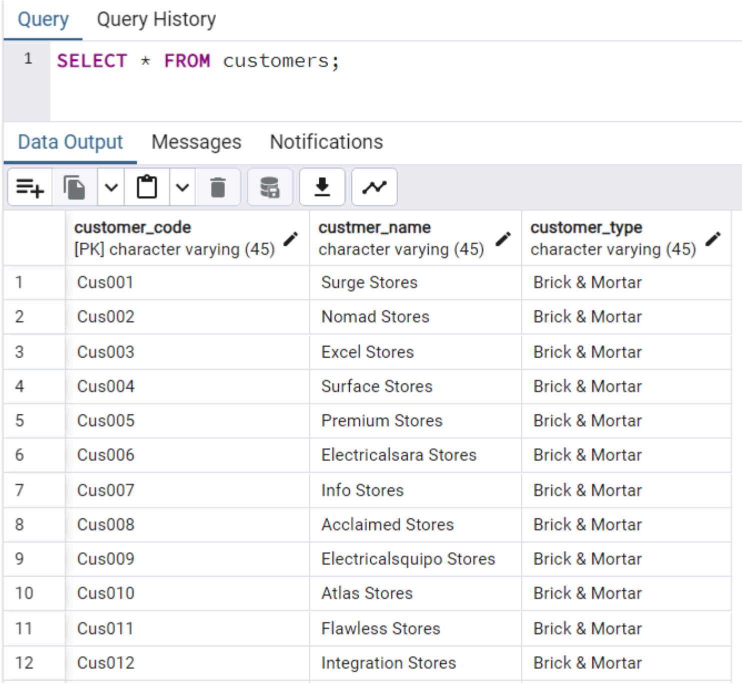
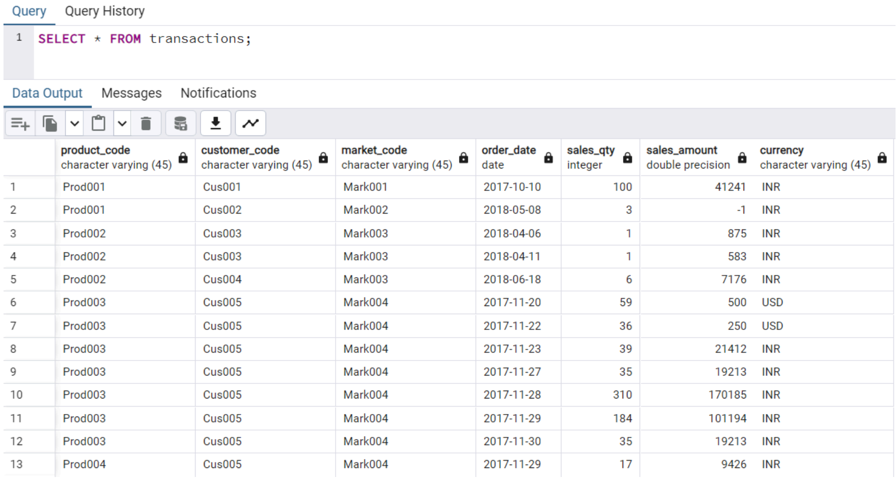
Sales insights using tableau and sql

First, create a sales dashboard in PostgreSQL with five different tables: dates, transactions, markets, customers, and products, then insert the desired data into the respective tables.



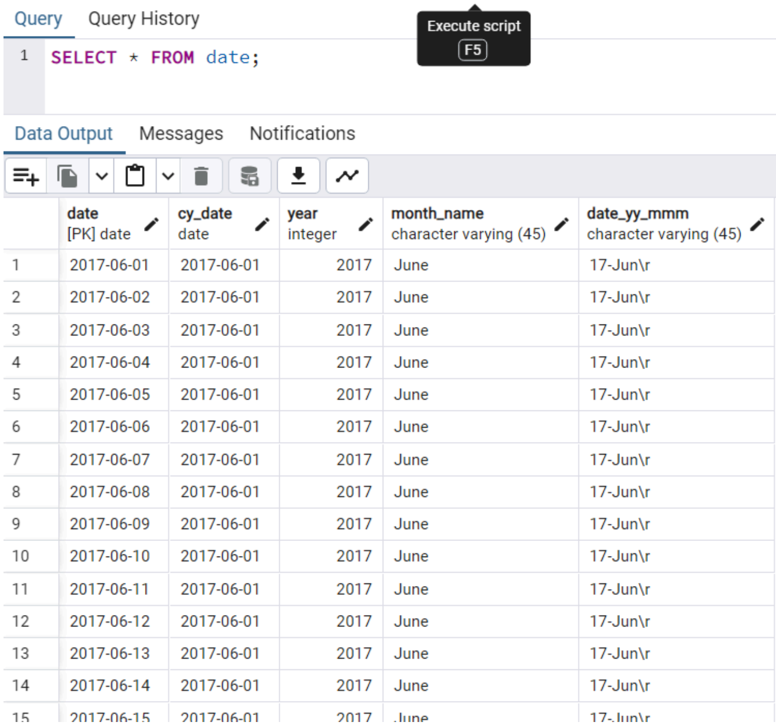
```
SELECT * FROM customers;
```

	customer_code [PK] character varying (45)	custmer_name character varying (45)	customer_type character varying (45)
1	Cus001	Surge Stores	Brick & Mortar
2	Cus002	Nomad Stores	Brick & Mortar
3	Cus003	Excel Stores	Brick & Mortar
4	Cus004	Surface Stores	Brick & Mortar
5	Cus005	Premium Stores	Brick & Mortar
6	Cus006	Electricalsara Stores	Brick & Mortar
7	Cus007	Info Stores	Brick & Mortar
8	Cus008	Acclaimed Stores	Brick & Mortar
9	Cus009	Electricalsquipo Stores	Brick & Mortar
10	Cus010	Atlas Stores	Brick & Mortar
11	Cus011	Flawless Stores	Brick & Mortar
12	Cus012	Integration Stores	Brick & Mortar



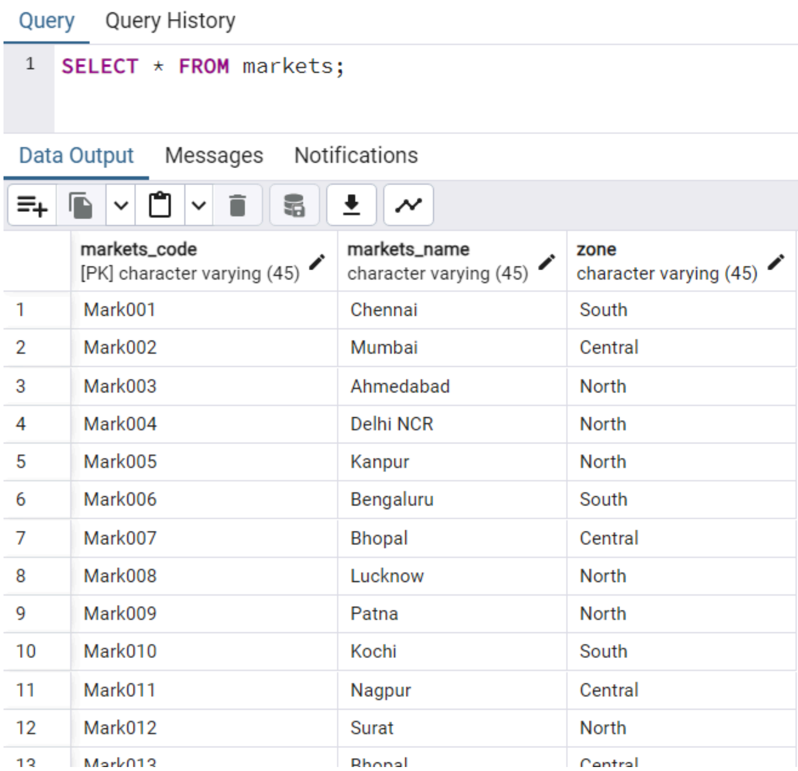
```
SELECT * FROM transactions;
```

	product_code character varying (45)	customer_code character varying (45)	market_code character varying (45)	order_date date	sales_qty integer	sales_amount double precision	currency character varying (45)
1	Prod001	Cus001	Mark001	2017-10-10	100	41241	INR
2	Prod001	Cus002	Mark002	2018-05-08	3	-1	INR
3	Prod002	Cus003	Mark003	2018-04-06	1	875	INR
4	Prod002	Cus003	Mark003	2018-04-11	1	583	INR
5	Prod002	Cus004	Mark003	2018-06-18	6	7176	INR
6	Prod003	Cus005	Mark004	2017-11-20	59	500	USD
7	Prod003	Cus005	Mark004	2017-11-22	36	250	USD
8	Prod003	Cus005	Mark004	2017-11-23	39	21412	INR
9	Prod003	Cus005	Mark004	2017-11-27	35	19213	INR
10	Prod003	Cus005	Mark004	2017-11-28	310	170185	INR
11	Prod003	Cus005	Mark004	2017-11-29	184	101194	INR
12	Prod003	Cus005	Mark004	2017-11-30	35	19213	INR
13	Prod004	Cus005	Mark004	2017-11-29	17	9426	INR



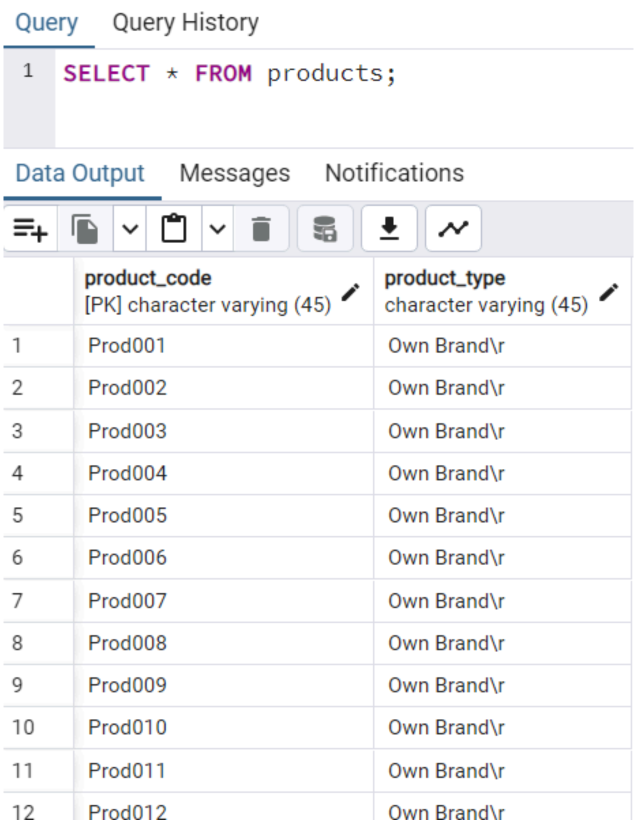
```
SELECT * FROM date;
```

	date [PK] date	cy_date date	year integer	month_name character varying (45)	date_yy_mmm character varying (45)
1	2017-06-01	2017-06-01	2017	June	17-Jun\r
2	2017-06-02	2017-06-01	2017	June	17-Jun\r
3	2017-06-03	2017-06-01	2017	June	17-Jun\r
4	2017-06-04	2017-06-01	2017	June	17-Jun\r
5	2017-06-05	2017-06-01	2017	June	17-Jun\r
6	2017-06-06	2017-06-01	2017	June	17-Jun\r
7	2017-06-07	2017-06-01	2017	June	17-Jun\r
8	2017-06-08	2017-06-01	2017	June	17-Jun\r
9	2017-06-09	2017-06-01	2017	June	17-Jun\r
10	2017-06-10	2017-06-01	2017	June	17-Jun\r
11	2017-06-11	2017-06-01	2017	June	17-Jun\r
12	2017-06-12	2017-06-01	2017	June	17-Jun\r
13	2017-06-13	2017-06-01	2017	June	17-Jun\r
14	2017-06-14	2017-06-01	2017	June	17-Jun\r
15	2017-06-15	2017-06-01	2017	June	17-Jun\r



```
SELECT * FROM markets;
```

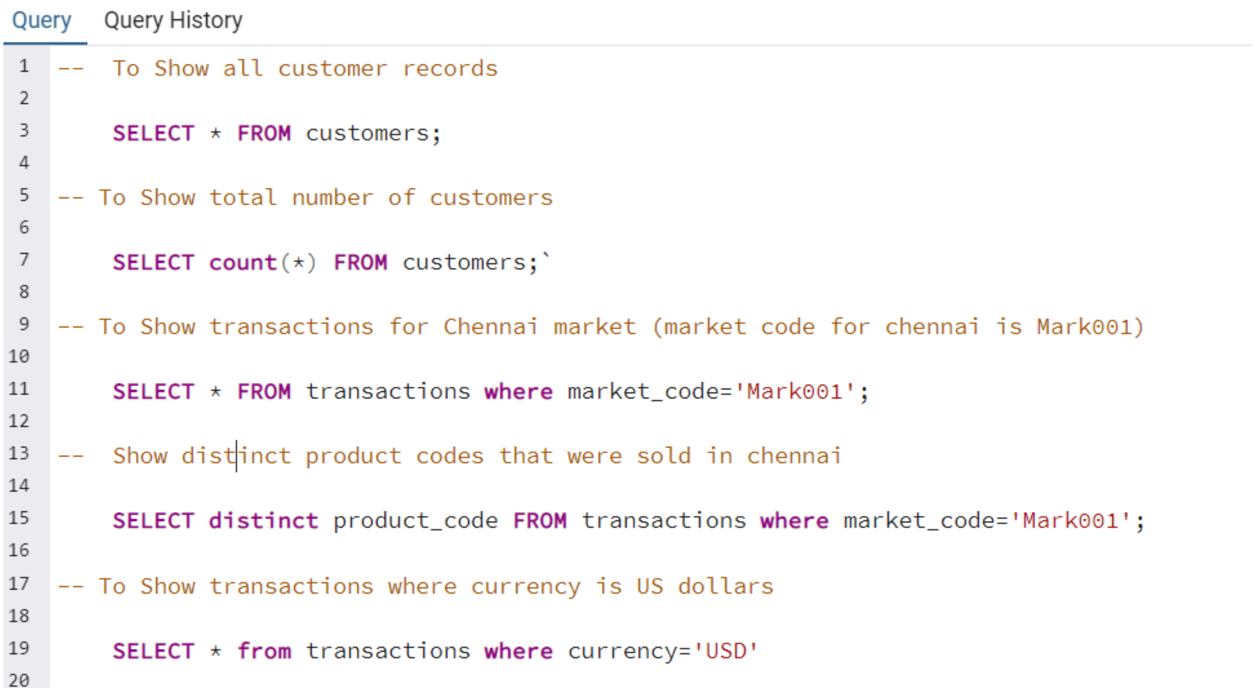
	markets_code [PK] character varying (45)	markets_name character varying (45)	zone character varying (45)
1	Mark001	Chennai	South
2	Mark002	Mumbai	Central
3	Mark003	Ahmedabad	North
4	Mark004	Delhi NCR	North
5	Mark005	Kanpur	North
6	Mark006	Bengaluru	South
7	Mark007	Bhopal	Central
8	Mark008	Lucknow	North
9	Mark009	Patna	North
10	Mark010	Kochi	South
11	Mark011	Nagpur	Central
12	Mark012	Surat	North
13	Mark013	Bhopal	Central



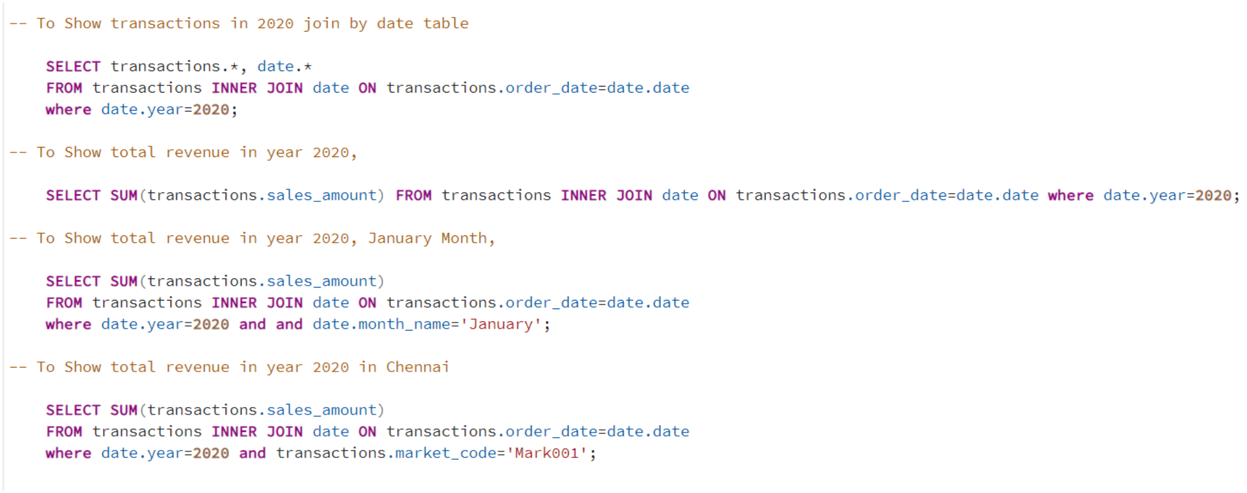
```
SELECT * FROM products;
```

	product_code [PK] character varying (45)	product_type character varying (45)
1	Prod001	Own Brand\r
2	Prod002	Own Brand\r
3	Prod003	Own Brand\r
4	Prod004	Own Brand\r
5	Prod005	Own Brand\r
6	Prod006	Own Brand\r
7	Prod007	Own Brand\r
8	Prod008	Own Brand\r
9	Prod009	Own Brand\r
10	Prod010	Own Brand\r
11	Prod011	Own Brand\r
12	Prod012	Own Brand\r

Then perform data analysis using SQL with various queries and algorithms.

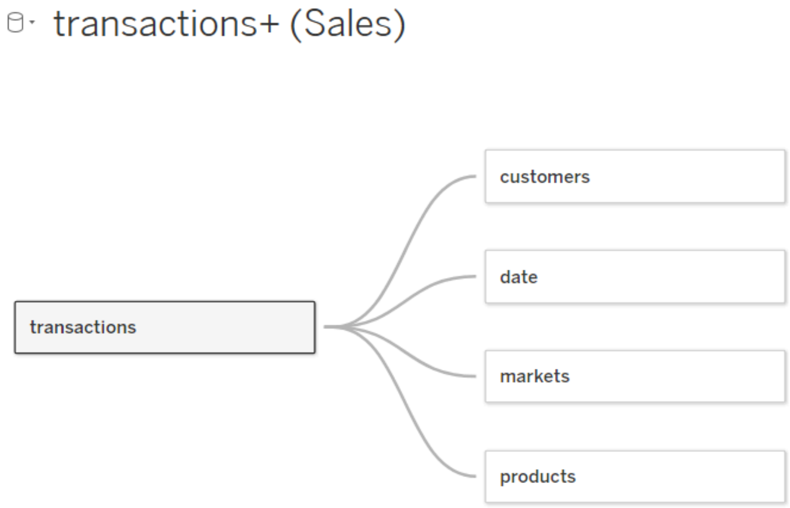


```
--  To Show all customer records

    SELECT * FROM customers;

-- To Show total number of customers

    SELECT count(*) FROM customers;`

-- To Show transactions for Chennai market (market code for chennai is Mark001)

    SELECT * FROM transactions where market_code='Mark001';

--  Show distinct product codes that were sold in chennai

    SELECT distinct product_code FROM transactions where market_code='Mark001';

-- To Show transactions where currency is US dollars

    SELECT * from transactions where currency='USD'

```



```
-- To Show transactions in 2020 join by date table

    SELECT transactions.*, date.*
    FROM transactions INNER JOIN date ON transactions.order_date=date.date
    where date.year=2020;

-- To Show total revenue in year 2020,

    SELECT SUM(transactions.sales_amount) FROM transactions INNER JOIN date ON transactions.order_date=date.date where date.year=2020;

-- To Show total revenue in year 2020, January Month,

    SELECT SUM(transactions.sales_amount)
    FROM transactions INNER JOIN date ON transactions.order_date=date.date
    where date.year=2020 and and date.month_name='January';

-- To Show total revenue in year 2020 in Chennai

    SELECT SUM(transactions.sales_amount)
    FROM transactions INNER JOIN date ON transactions.order_date=date.date
    where date.year=2020 and transactions.market_code='Mark001';
```

Then connect SQL with Tableau Desktop. The sales database and the five different tables are now present in Tableau. Initially, establish the relationships between these tables to create a star schema.



Analyzed the transactions table to identify and clean sales amounts by excluding values of -1 and 0, ensuring the column includes only values greater than or equal to 1.

Removed rows with null values in the market table.

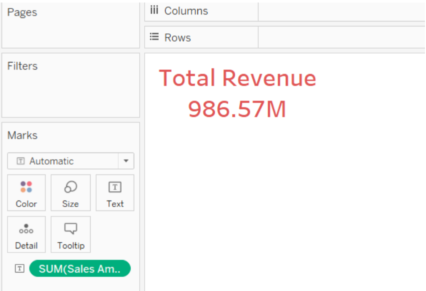
Standardized the currency column in the transactions table by creating a calculated field in Tableau to convert all USD values to INR, using the formula: IF [CURRENCY] == 'USD' THEN [SALES AMOUNT]*83.16 ELSE [SALES AMOUNT] END."

The data cleaning process is complete.

Creating a Dashboard in Tableau

Create a Text Box for Sum of Normalized Sales Amount

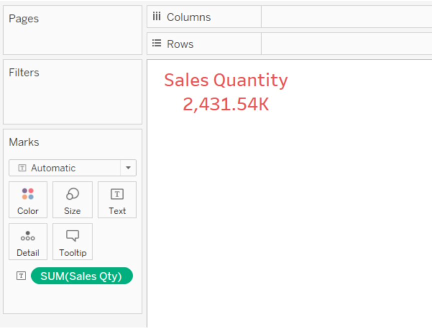
- Use the text option to display "Sum of Normalized Sales Amount."
- Customize the text's color, size, alignment, and format it as a custom currency (e.g., million).



Total Revenue
986.57M

Create a Text Box for Sum of Sales Quantity

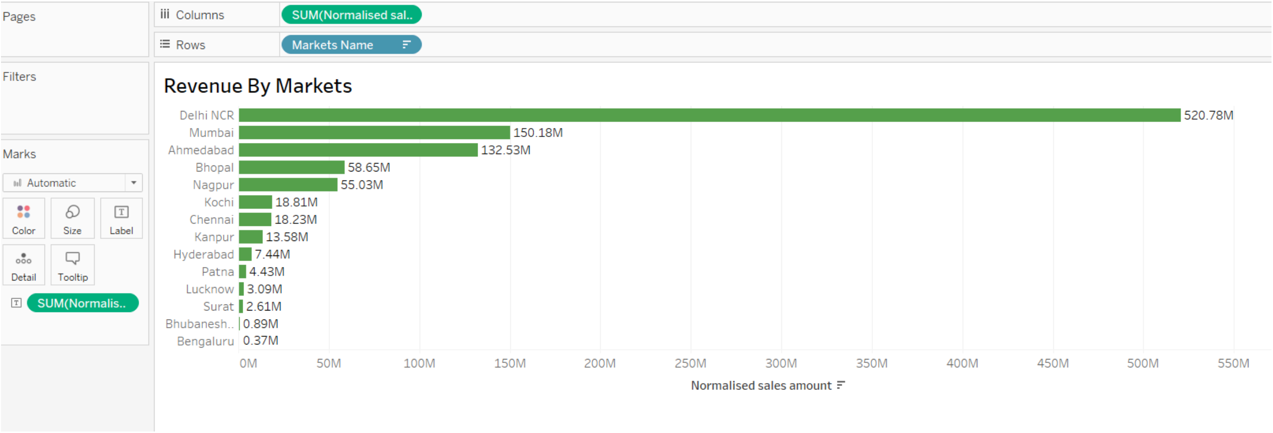
- Display the sum of sales quantity in a visually appealing format.



Sales Quantity
2,431.54K

Create a Bar Chart for Normalized Sales Amount by Market

- Show the sum of normalized sales amounts by market name.
- Use the transpose option to convert column values to rows and rows to columns.
- Use the transpose option to convert column values to rows and rows to columns.
- Customize the chart by increasing its size, changing colors to enhance visual appeal, adding labels, formatting labels to display amounts in millions, removing the prefix from the labels, and displaying the labels inside the bars for better readability.



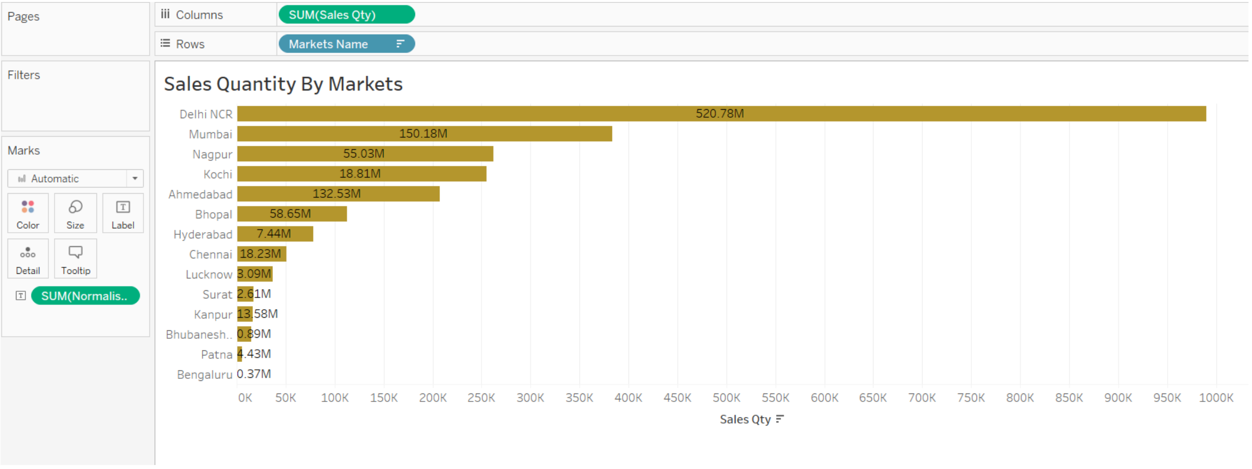
Columns SUM(Normalised sal..

Rows Markets Name

Marks: Automatic, Label SUM(Normalis..

Create a Bar Chart for Sales Quantity by Market

- Show the sum Sales Quantity by market name.
- Use the transpose option to convert column values to rows and rows to columns.
- Use the transpose option to convert column values to rows and rows to columns.
- Customize the chart by increasing its size, changing colors to enhance visual appeal, adding labels, formatting labels to display amounts in millions, removing the prefix from the labels, and displaying the labels inside the bars for better readability.



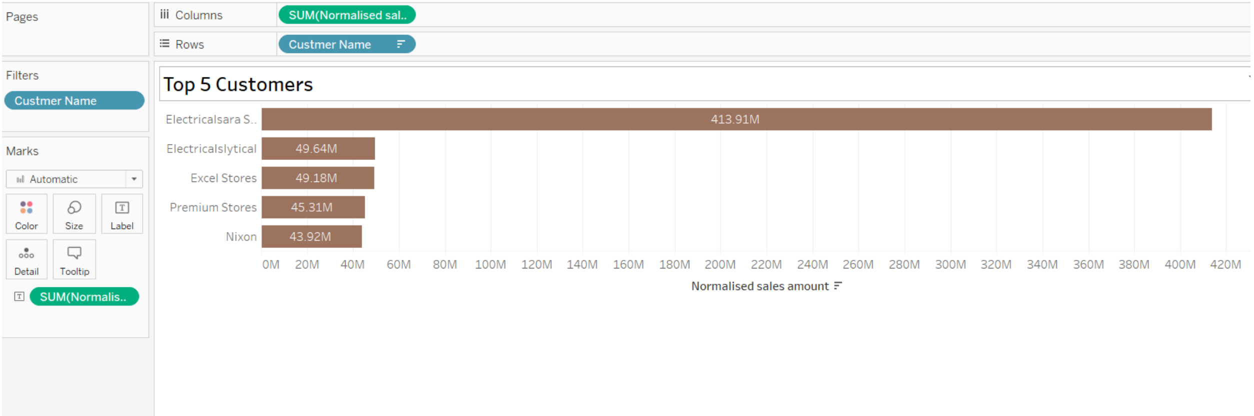
Columns SUM(Sales Qty)

Rows Markets Name

Marks: Automatic, Label SUM(Normalis..

Create a Bar Chart for Sales Amount by Product

- Customize the chart by increasing its size, changing colors to enhance visual appeal, adding labels, formatting the labels to display amounts in millions, removing the prefix from the labels, and displaying the labels inside the bars for better readability.
- Filter the chart to show the top market names.



Columns: SUM(Normalised sal..

Rows: Custmer Name

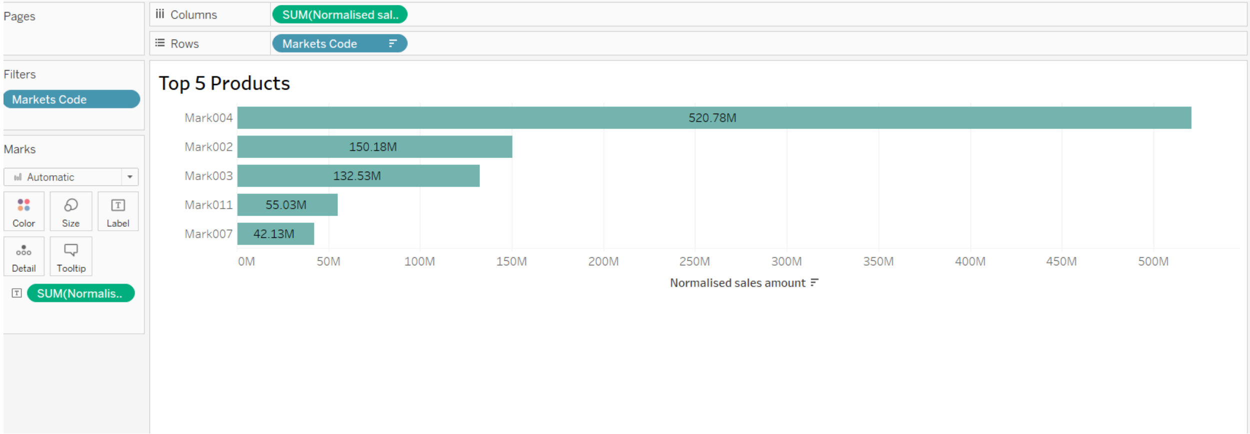
Filters: Custmer Name

Marks: Automatic

SUM(Normalis..

Create a Bar Chart for Sales Amount by Customer Name

- Customize the chart by increasing its size, changing colors to enhance visual appeal, adding labels, formatting the labels to display amounts in millions, removing the prefix from the labels, and displaying the labels inside the bars for better readability.
- Filter the chart to show the top market names.



Columns: SUM(Normalised sal..

Rows: Markets Code

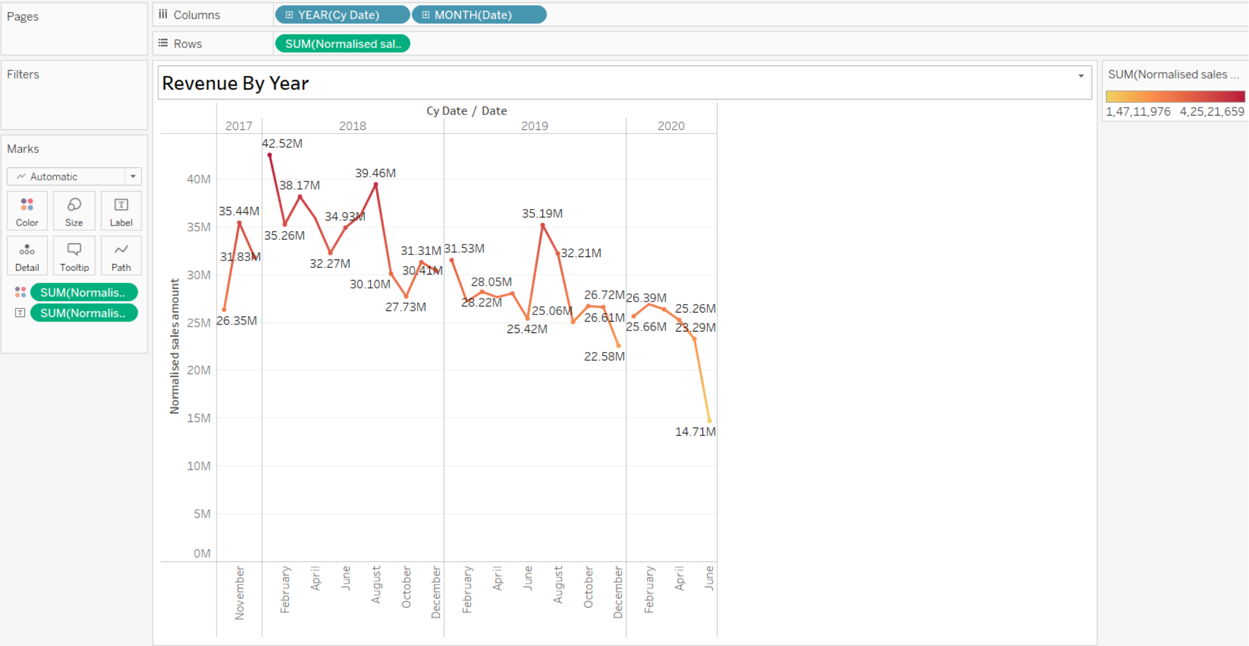
Filters: Markets Code

Marks: Automatic

SUM(Normalis..

Create a Bar Chart for Normalized Amount by Date

- Include both date (MONTH) and cyc date (YEAR) to create a hierarchy-like appearance.
- Sort the data for clear visualization.
- Hide the column names to reduce clutter.
- Add colors to differentiate the data.
- Add labels for better understanding and readability.



Columns: YEAR(Cy Date) MONTH(Date)

Rows: SUM(Normalised sal..

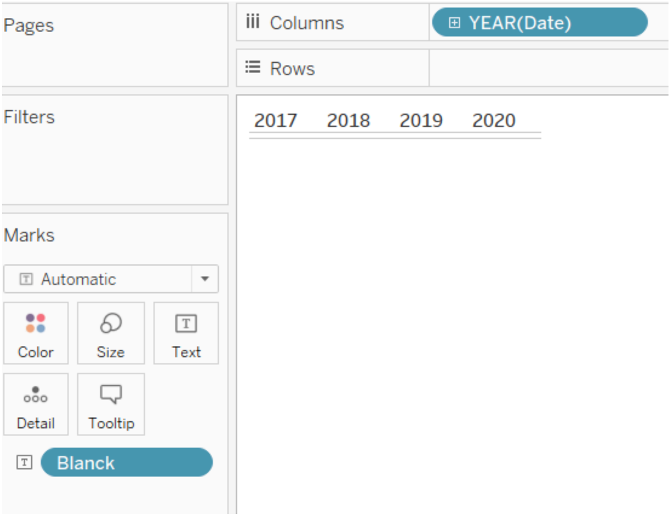
Marks: Automatic

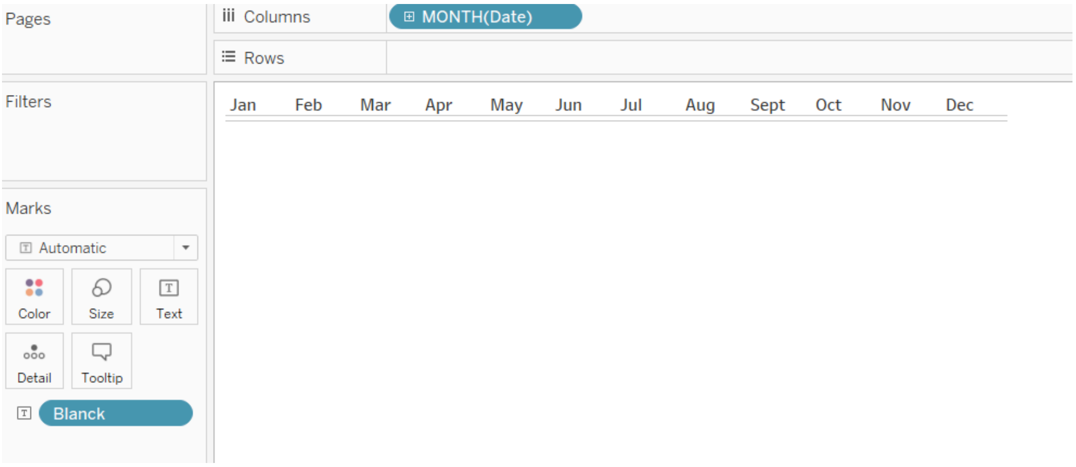
SUM(Normalis..

SUM(Normalis..

Create a Dropdown Filter for Date and Month Name

- Drag the date column to the column field.
- Add a blank field to the column field.
- Add the blank field to the label to create filters for both year and month.





.