E-commerce Sales Analysis

Contents-

1. **Project Overview**
2. **Project Objective**
3. **Project Dataset**
4. **Tool Used:**
   - Python
   - Power Bi
5. **Project Workflow:**
   - Loading
   - Data Cleaning
   - Transformation
   - EDA(Statistical Analysis)
   - Dashboard Reports
6. **Graph Analysis**
7. **Challenges and Solution**

Power BI

python™

E-commerce Sales Analysis

❑ 1. Project Overview:

In today’s data-driven digital economy, analyzing e-commerce sales data has become crucial for businesses to optimize operations, identify sales trends, and increase profitability. This project focuses on extracting actionable insights from e-commerce transactions using a combination of **Exploratory Data Analysis (EDA)** in Python and interactive visualizations in Power BI. Unlike dashboard-only approaches, this project highlights the essential role of EDA in cleaning, understanding, and shaping the data before visualization. Using the **CRISP-DM framework**, the analysis explored key sales trends, regional performance, and product demand, based on features like Order Date, Product, Quantity Ordered, Price Each, and Address. The insights derived from EDA guided the design of a business-focused Power BI dashboard.

E-commerce Sales Analysis

- **2. Project Objective:**

The objective of this project is to extract accurate, business-relevant insights from e-commerce sales data by following the CRISP-DM framework, with a strong emphasis on **Exploratory Data Analysis (EDA)** as the foundation for precision-driven reporting. The project begins with consolidating and standardizing 12 months of sales data from separate CSV files into a unified, clean dataset using Python. Through detailed EDA, patterns in product performance, regional demand, and monthly sales trends were uncovered. These insights were then used to design **a dynamic Power BI dashboard** that enables stakeholders to make informed, data-driven decisions based on thoroughly validated and structured data.

E-commerce Sales Analysis

❑ 3. Dataset Description:

The dataset contains e-commerce sales transactions with the following columns:

- **Order ID**: Unique identifier for each purchase transaction.
- **Product**: Name of the product sold.
- **Order Date**: Date and time when the order was placed.
- **Quantity Ordered**: Number of units ordered for a product in a transaction.
- **Price Each**: Price of a single unit of the product.
- **Address**: Full shipping address, which includes street, city, state, and ZIP code .

3. A Glance Of The Dataset

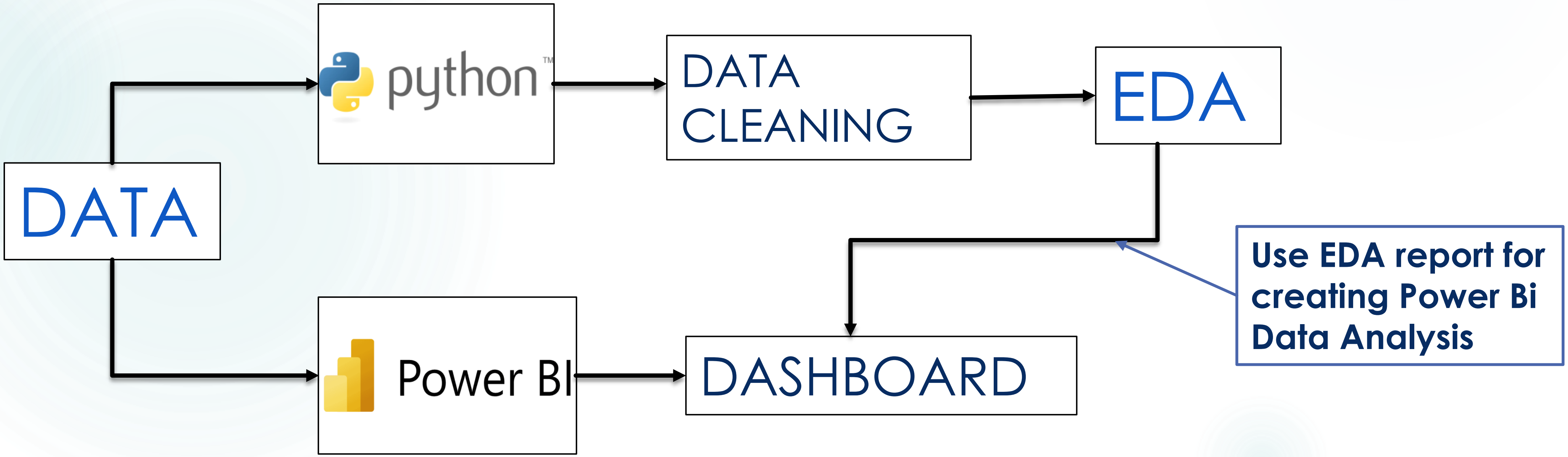
```
e_com.head()
```

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address
0	141234	iPhone	1	700	01/22/19 21:25	944 Walnut St, Boston, MA 02215
1	141235	Lightning Charging Cable	1	14.95	01/28/19 14:15	185 Maple St, Portland, OR 97035
2	141236	Wired Headphones	2	11.99	01/17/19 13:33	538 Adams St, San Francisco, CA 94016
3	141237	27in FHD Monitor	1	149.99	01/05/19 20:33	738 10th St, Los Angeles, CA 90001
4	141238	Wired Headphones	1	11.99	01/25/19 11:59	387 10th St, Austin, TX 73301

E-commerce Sales Analysis

❑ 4. Tools Used:

- **Python**: was used for **data preprocessing and exploratory data analysis (EDA)**. It helped in loading and merging 12-month CSV files, cleaning and standardizing the data, creating new columns (e.g., Sales, Month, City), and uncovering trends and patterns using libraries like Pandas, Matplotlib, and Seaborn.
- **Power BI**: was used to build an **interactive dashboard** based on the insights gathered from EDA. The cleaned and prepared dataset was loaded into Power BI to create visuals such as bar charts, line graphs, and slicers, allowing users to explore sales performance by time, product, and region.



E-commerce Sales Analysis

❑ 5. Project Workflow:- (Loading)

Python

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings

warnings.filterwarnings('ignore')
```

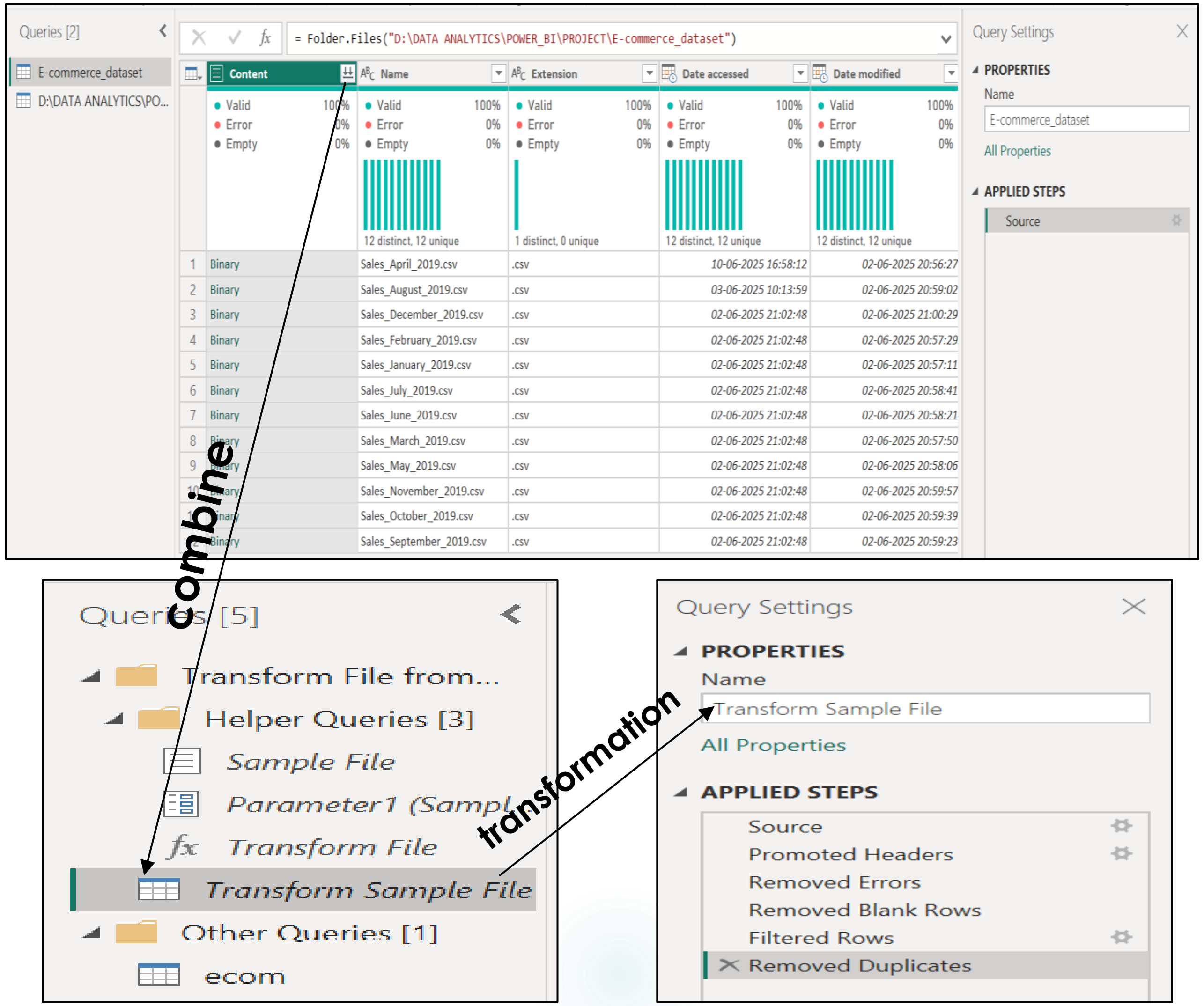
1. Load Dataset [A monthly Data of One Year]

```python
January = pd.read_csv("Sales_January_2019.csv")
February = pd.read_csv("Sales_February_2019.csv")
March = pd.read_csv("Sales_March_2019.csv")
April = pd.read_csv("Sales_April_2019.csv")
May = pd.read_csv("Sales_May_2019.csv")
June = pd.read_csv("Sales_June_2019.csv")
July = pd.read_csv("Sales_July_2019.csv")
August = pd.read_csv("Sales_August_2019.csv")
September = pd.read_csv("Sales_September_2019.csv")
October = pd.read_csv("Sales_October_2019.csv")
November = pd.read_csv("Sales_November_2019.csv")
December = pd.read_csv("Sales_December_2019.csv")
```

2. Append the Data

```python
e_com = pd.concat([January , February , March , April , May , June ,
                   July , August , September , October , November , December] , axis = 0 )
```

Power Bi



E-commerce Sales Analysis

1. Null Values And Duplicates

5. Project Workflow:- (Data Cleaning)

```python
# 2. Remove the null values(Rows):
ecom1 = e_com.dropna(how = 'all' , inplace = False)
```

4.1 NULL VALUE TREATMENT

```python
# 1.1. Count of nulls:
e_com.isna().sum()
```

```
Order ID            545
Product             545
Quantity Ordered    545
Price Each          545
Order Date          545
Purchase Address    545
dtype: int64
```

```python
# 3. After removing nulls:
ecom1.isnull().sum()
```

```
Order ID            0
Product             0
Quantity Ordered    0
Price Each          0
Order Date          0
Purchase Address    0
dtype: int64
```

Nulls

Duplicates

4.2 DUPLICATES

```python
# 1.1. Count of Duplicate(Row):
ecom1.duplicated().sum()
```

```
618
```

```python
# 2. Drop  the duplicate rows:
ecom2 = ecom1.drop_duplicates()
```

```python
# 3. check Duplicates after modification
ecom2.duplicated().sum()
```

```
0
```

E-commerce Sales Analysis

Standardization

5. Project Workflow:- (Data Cleaning)

```
ecom2["Quantity Ordered"].unique()
```

```
array(['1', '2', '3', '5', '4', '7', 'Quantity Ordered', '6', '9', '8'],
      dtype=object)
```

```
ecom2["Price Each"].unique()
```

```
array(['700', '14.95', '11.99', '149.99', '2.99', '389.99', '11.95',
       '99.99', '150', '1700', '300', '400', '3.84', '600', '109.99',
       '379.99', '999.99', '600.0', 'Price Each', '700.0', '150.0',
       '300.0', '1700.0', '400.0'], dtype=object)
```

```
# 2. Drop  the column index 1073:
ecom3 = ecom2[ecom2["Order ID"].str.isnumeric()]
```

```
# 2. Now reset  the index:
ecom3.reset_index(drop = True , inplace = True)
```

```
# 4. Correct  the datatype of the 'Order Date' column:
ecom3["Order Date"] = pd.to_datetime(ecom3["Order Date"])
```

```
# 3. Change the datatypes of these column:
ecom3["Quantity Ordered"] = ecom3["Quantity Ordered"].astype(int)
ecom3["Price Each"] = ecom3["Price Each"].astype(float)
```

Not Numeric

Change in Datatypes

4.3 Standardisation the data

```
# 1. show all the columns datatypes:
ecom2.dtypes
```

```
Order ID            object
Product             object
Quantity Ordered    object
Price Each          object
Order Date          object
Purchase Address    object
dtype: object
```

```
# 5. check the datatypes after modification:
ecom3.dtypes
```

```
Order ID                    object
Product                     object
Quantity Ordered             int32
Price Each                 float64
Order Date          datetime64[ns]
Purchase Address            object
dtype: object
```

E-commerce Sales Analysis

Add Features

❑ 5. <u>Project Workflow</u> : (Transformation)

5. Add some features

```python
# Add day and month
ecom3['Day'] = ecom3["Order Date"].dt.day
ecom3['Month'] = ecom3["Order Date"].dt.month

# add day and month name
ecom3['Day_name'] = ecom3["Order Date"].dt.day_name()
ecom3['Month_name'] = ecom3["Order Date"].dt.month_name()
```

```python
# make a category column:
Product = ['Charging Cable' , 'Batteries' , 'Headphones' ,
           'Monitor' , 'Phone' , 'Laptop' , 'TV' ,
           'Washing Machine' , 'Dryer']

# make a function to convert the the product to category
def product_category(Product_name):
    for p in Product:
        if p in Product_name:
          Product_name = p
    return Product_name

ecom3['Category']= ecom3['Product']
ecom3['Category']= ecom3['Category'].apply(product_category)
```

```python
# Extract the street name , city name , and pct code from 'Purchase Address' column:
ecom3[['Street' , 'City' , 'CPT_code' ]]= ecom3['Purchase Address'].str.split(',' , expand = True)
```

```python
# show the city column content:
ecom3['City'].unique()
```

```
array([' Boston', ' Portland', ' San Francisco', ' Los Angeles',
       ' Austin', ' Atlanta', ' Seattle', ' New York City', ' Dallas'],
      dtype=object)
```

```python
# Make Total Price column:
ecom3['Total Price'] = ecom3['Quantity Ordered'] * ecom3['Price Each']
```

```python
ecom3['City'].value_counts()
```

```
City
San Francisco    44662
Los Angeles      29564
New York City    24847
Boston           19901
Atlanta          14863
Dallas           14797
Seattle          14713
Portland         12449
Austin            9890
Name: count, dtype: int64
```

```python
ecom3['Category'].value_counts()
```

```
Category
Headphones         47672
Charging Cable     43469
Batteries          41170
Monitor            23995
Phone              14427
Laptop              8847
TV                  4794
Washing Machine      666
Dryer                646
Name: count, dtype: int64
```

```python
ecom3['Total Price']
```

```
0         700.00
1          14.95
2          23.98
3         149.99
4          11.99
           ...
185681     14.95
185682      7.68
185683    400.00
185684     11.99
185685     99.99
```

E-commerce Sales Analysis

❑ 5. Project Workflow:- (Exploratory Data Analysis)

❑ **Exploratory Data Analysis (EDA)** is the process of analyzing datasets to summarize their main characteristics using statistical methods and visualizations. It is often the first and crucial step before building predictive models, as it helps in understanding the structure of data, detecting patterns, identifying anomalies, and selecting relevant features. EDA guides effective data preprocessing and model selection, ultimately improving model performance and interpretability.

1. Profile of Data

- Understand data types and structure (e.g., numerical, categorical)
- Identify missing values, duplicates, or incorrect formats
- Assess data completeness and consistency

2. Statistical Analysis

- Calculate key metrics like mean, median, mode, standard deviation
- Analyze distributions and detect outliers
- Explore relationships and correlations between variables

3. Graphical Analysis

- **Univariate Analysis**: Focuses on visualizing individual variables to understand their distribution, central tendency, and outliers using charts like histograms, bar plots, or box plots.
- **Bivariate Analysis**: Examines the relationship between two variables through visual tools such as scatter plots or line plots, helping identify trends, correlations, or group differences.
- **Multivariate Analysis**: Involves analyzing and visualizing interactions among three or more variables using heatmaps, pair plots, or facet grids to uncover deeper patterns and variable dependencies.

E-commerce Sales Analysis

❑ 5. Project Workflow:- (Exploratory Data Analysis)

1. Profile of Data

```
# 1. To show the number of rows and columns
ecom3.shape
```

```
(185686, 15)
```

```
# 2. To show the total number of data points
ecom3.size
```

```
2785290
```

```
# 7. To show the number of duplicate rows
ecom3.duplicated().sum()
```

```
0
```

```
# 8. To show the index
ecom3.index
```

```
RangeIndex(start=0, stop=185686, step=1)
```

```
# 3. To show the column names
ecom3.columns
```

```
Index(['Order ID', 'Product', 'Quantity Ordered', 'Price Each', 'Order Date',
       'Purchase Address', 'Day', 'Month', 'Day_name', 'Month_name', 'Street',
       'City', 'CPT_code', 'Total Price', 'Category'],
      dtype='object')
```

```
# 5. To  show the datatypes of the columns
ecom3.dtypes
```

```
Order ID                    object
Product                     object
Quantity Ordered             int32
Price Each                 float64
Order Date          datetime64[ns]
Purchase Address            object
Day                          int32
Month                        int32
Day_name                    object
Month_name                  object
Street                      object
City                        object
CPT_code                    object
Total Price                float64
Category                    object
dtype: object
```

```
# 6.Null values in the columns
ecom3.isna().sum()
```

```
Order ID            0
Product             0
Quantity Ordered    0
Price Each          0
Order Date          0
Purchase Address    0
Day                 0
Month               0
Day_name            0
Month_name          0
Street              0
City                0
CPT_code            0
Total Price         0
Category            0
dtype: int64
```

```
# 9.Columns unique values count
ecom3.nunique()
```

```
Order ID            178437
Product                 19
Quantity Ordered         9
Price Each              17
Order Date          142395
Purchase Address    140787
Day                     31
Month                   12
Day_name                 7
Month_name              12
Street               46837
City                     9
CPT_code                10
Total Price             54
Category                 9
dtype: int64
```

```
# 10. Five point summary:
ecom3[['Quantity Ordered' , 'Price Each' , 'Total Price']].describe()
```

	Quantity Ordered	Price Each	Total Price
count	185686.000000	185686.000000	185686.000000
mean	1.124544	184.519255	185.611936
std	0.443069	332.843838	333.032118
min	1.000000	2.990000	2.990000
25%	1.000000	11.950000	11.950000
50%	1.000000	14.950000	14.950000
75%	1.000000	150.000000	150.000000
max	9.000000	1700.000000	3400.000000

E-commerce Sales Analysis

2. Statistical Analysis

Categorical Data

```
categorical_features.dtypes

Order ID            object
Product             object
Purchase Address    object
Day_name            object
Month_name          object
Street              object
City                object
CPT_code            object
Category            object
dtype: object
```

Numerical Data

```
numerical_features.dtypes

Quantity Ordered      int32
Price Each          float64
Day                   int32
Month                 int32
Total Price         float64
dtype: object
```

Price bin Stats

	Price Each	sum	count	max
0	2.99	30986	20612	9
1	3.84	27615	20558	7
2	11.95	23931	21859	6
3	11.99	20524	18849	4
4	14.95	23169	21610	4
5	99.99	13430	13298	3
6	109.99	4126	4098	2
7	149.99	7541	7498	2
8	150.00	15637	15525	3
9	300.00	4813	4794	2
10	379.99	6192	6174	2
11	389.99	6239	6225	2
12	400.00	2068	2065	2
13	600.00	6841	6834	2
14	700.00	6847	6840	2
15	999.99	4128	4126	2
16	1700.00	4725	4721	2

```
# city wise Orders:
ecom3['City'].value_counts()

City
San Francisco    44662
Los Angeles      29564
New York City    24847
Boston           19901
Atlanta          14863
Dallas           14797
Seattle          14713
Portland         12449
Austin            9890
Name: count, dtype: int64
```

```
# category wise no. of orders:
ecom3['Category'].value_counts()

Category
Headphones          47672
Charging Cable      43469
Batteries           41170
Monitor             23995
Phone               14427
Laptop               8847
TV                   4794
Washing Machine       666
Dryer                 646
Name: count, dtype: int64
```

```
# Total number of product Orders
ecom3.groupby(['Product'])['Quantity Ordered'].sum()

Product
20in Monitor                   4126
27in 4K Gaming Monitor         6239
27in FHD Monitor               7541
34in Ultrawide Monitor         6192
AA Batteries (4-pack)         27615
AAA Batteries (4-pack)        30986
Apple Airpods Headphones      15637
Bose SoundSport Headphones    13430
Flatscreen TV                  4813
Google Phone                   5529
LG Dryer                        646
LG Washing Machine              666
Lightning Charging Cable      23169
Macbook Pro Laptop             4725
ThinkPad Laptop                4128
USB-C Charging Cable          23931
Vareebadd Phone                2068
Wired Headphones              20524
iPhone                         6847
Name: Quantity Ordered, dtype: int32
```

E-commerce Sales Analysis

2. Statistical Analysis

❑ **5. Project Workflow:- (Exploratory Data Analysis)**

Category vs Total order price stats

Category	sum	median	mean	std	min	max
Batteries	198689.74	3.84	4.826081	2.620210	2.99	26.91
Charging Cable	632352.00	14.95	14.547195	4.284343	11.95	71.70
Dryer	387600.00	600.00	600.000000	0.000000	600.00	600.00
Headphones	3934498.46	99.99	82.532691	60.176281	11.99	450.00
Laptop	12160458.72	1700.00	1374.529074	351.892158	999.99	3400.00
Monitor	6370939.02	379.99	265.511107	126.352025	109.99	779.98
Phone	8937500.00	600.00	619.498163	102.797811	400.00	1400.00
TV	1443900.00	300.00	301.188986	18.850897	300.00	600.00
Washing Machine	399600.00	600.00	600.000000	0.000000	600.00	600.00

City vs total order price stats

City	sum	median	mean	std	min	max
Atlanta	2794199.07	14.95	187.996977	334.876275	2.99	1700.0
Austin	1818044.33	14.95	183.826525	331.628368	2.99	1700.0
Boston	3658627.65	14.95	183.841397	329.352885	2.99	3400.0
Dallas	2765373.96	14.95	186.887474	334.446402	2.99	1700.0
Los Angeles	5448304.28	14.95	184.288468	332.243107	2.99	1700.0
New York City	4661867.14	14.95	187.622938	336.359270	2.99	3400.0
Portland	2319331.94	14.95	186.306686	337.153437	2.99	1700.0
San Francisco	8254743.55	14.95	184.827002	332.675745	2.99	3400.0
Seattle	2745046.02	14.95	186.572828	329.165595	2.99	1700.0

Monthly revenue stats

Month_name	sum	median	mean	std
April	3389217.98	14.95	185.639370	332.141314
August	2241083.37	14.95	187.648277	332.193945
December	4608295.70	14.95	184.745658	332.820367
February	2200078.08	14.95	183.999170	325.988793
January	1821413.16	14.95	187.793913	331.092526
July	2646461.32	14.95	185.391336	333.024103
June	2576280.15	14.95	190.342087	337.582608
March	2804973.35	14.95	185.416007	331.556813
May	3150616.23	14.95	190.346558	342.964978
November	3197875.05	14.95	182.277420	330.306825
October	3734777.86	14.95	184.442583	334.363393
September	2094465.69	14.95	180.510703	328.478786

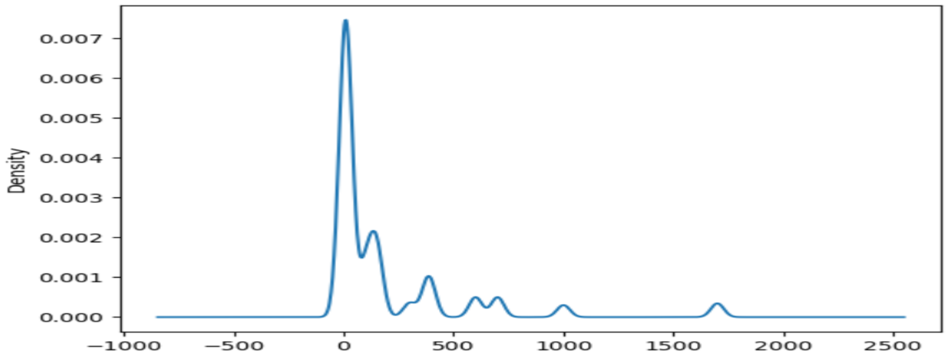
E-commerce Sales Analysis

3. <u>Graphical Analysis</u>

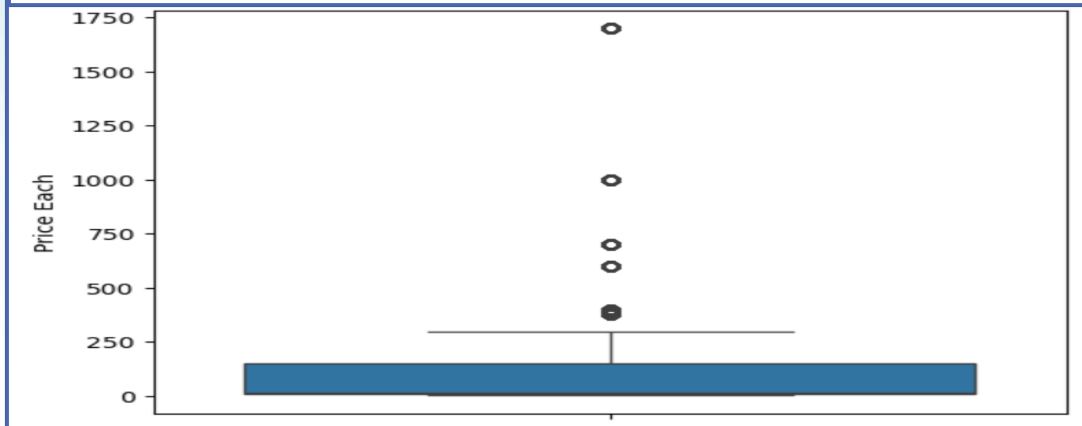
❑ **5. <u>Project Workflow</u>:- (Exploratory Data Analysis)**

Product Price Distribution Plot

```
[66]:  ecom3['Price Each'].plot.kde()
[66]:  <Axes: ylabel='Density'>
```

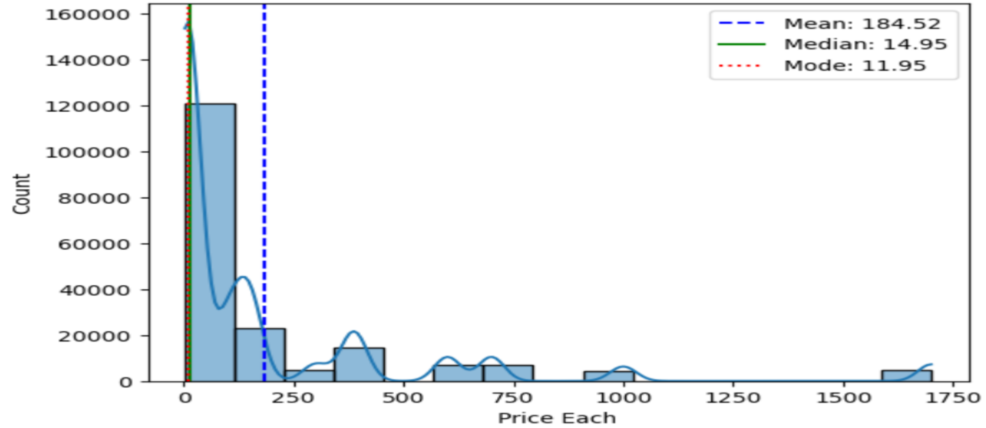


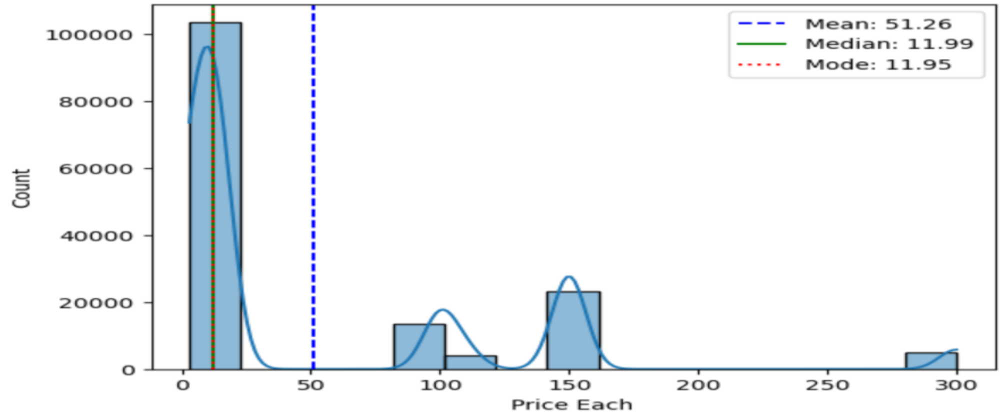
Boxplot




```
Standard deviation before= 332.8438383899937
Standard deviation after= 71.1309982733632
```

Product Price stats: count , mean ,median, mode (before and after outlier treatment)



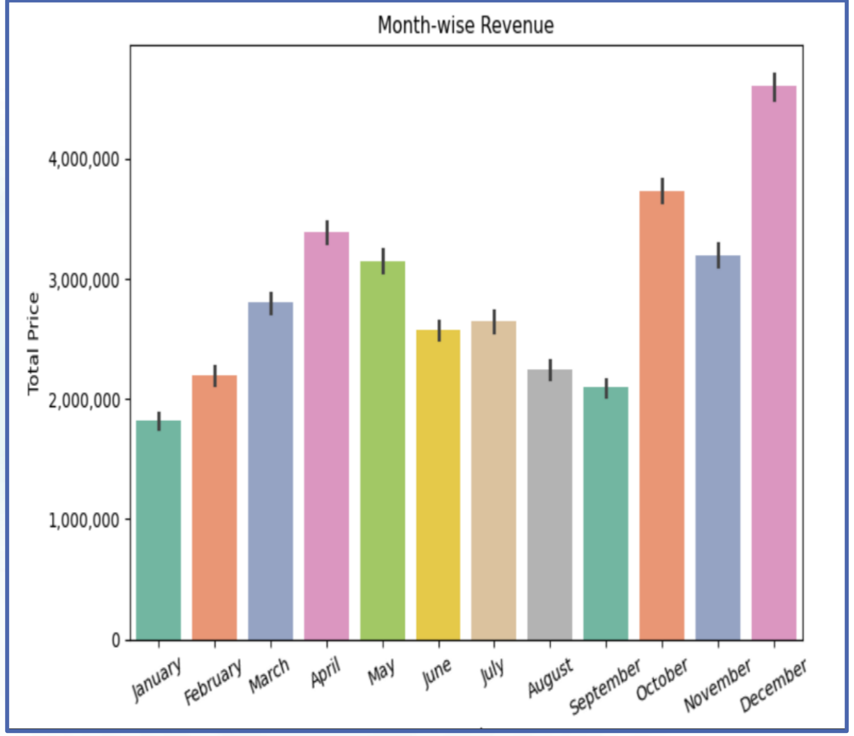


E-commerce Sales Analysis

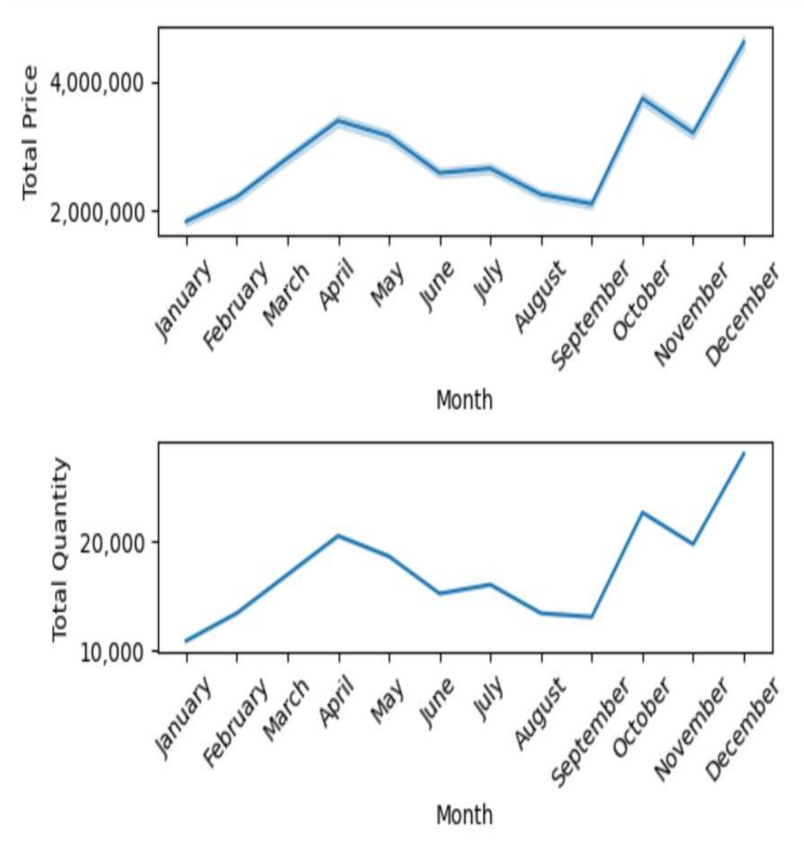
❑ **5. <u>Project Workflow</u>:- (Exploratory Data Analysis)**

3. <u>Graphical Analysis</u>

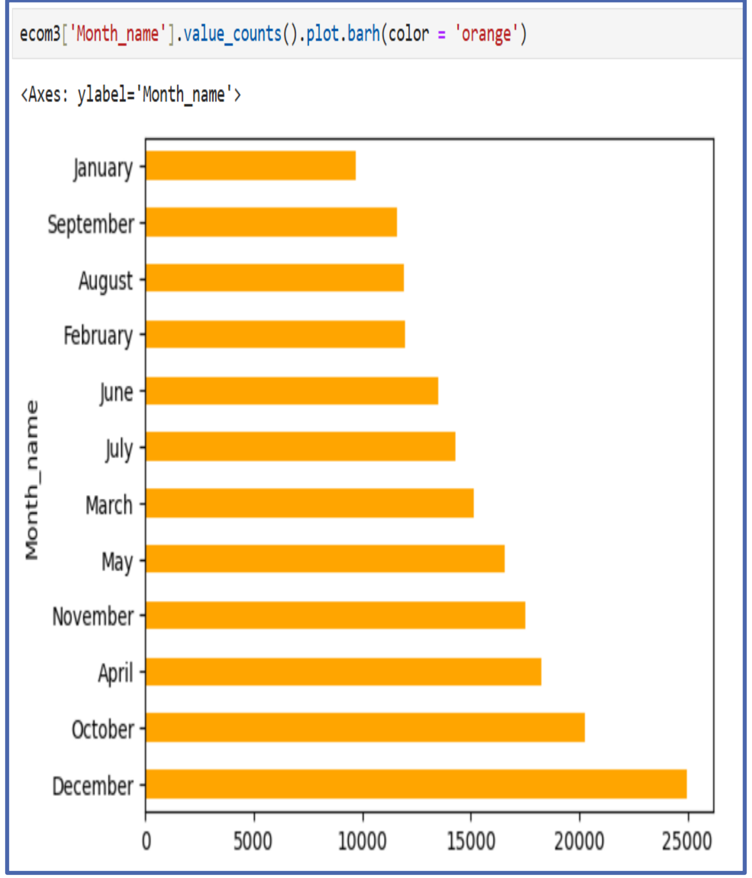
Monthly Revenue



Revenue vs. orders count



Monthly Order Counts

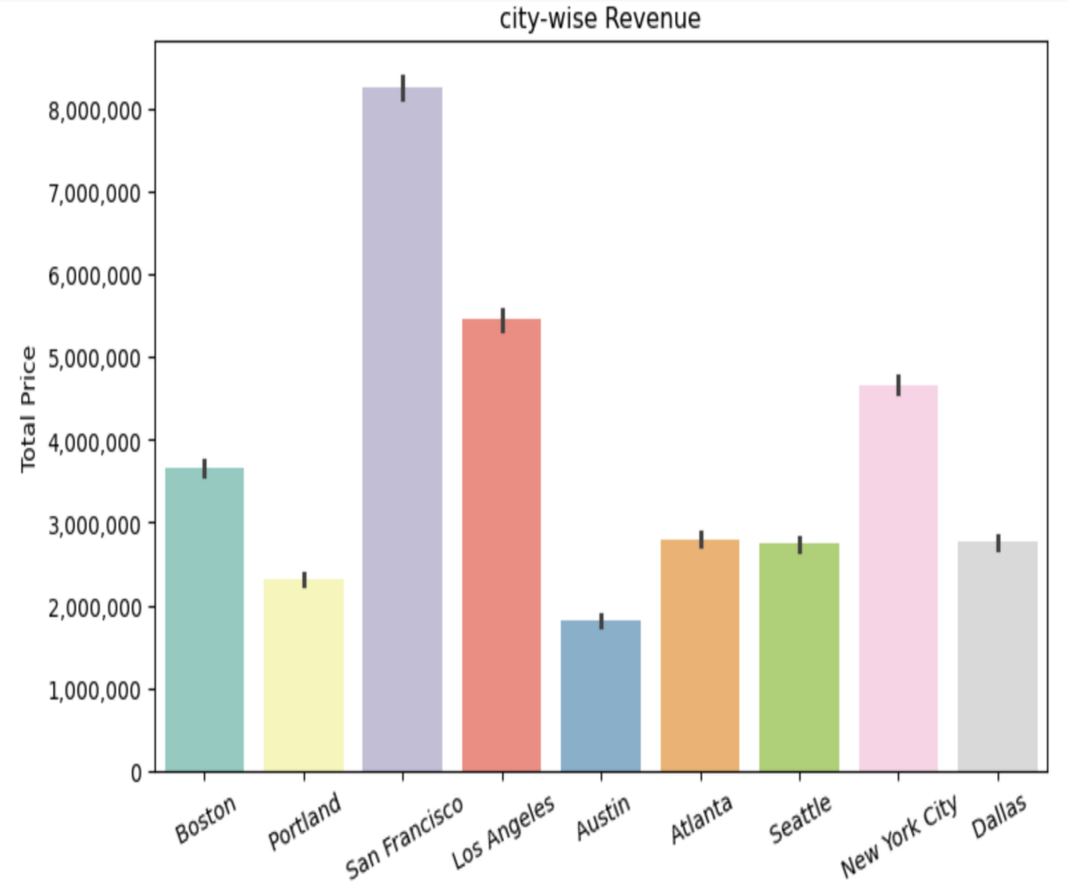


E-commerce Sales Analysis

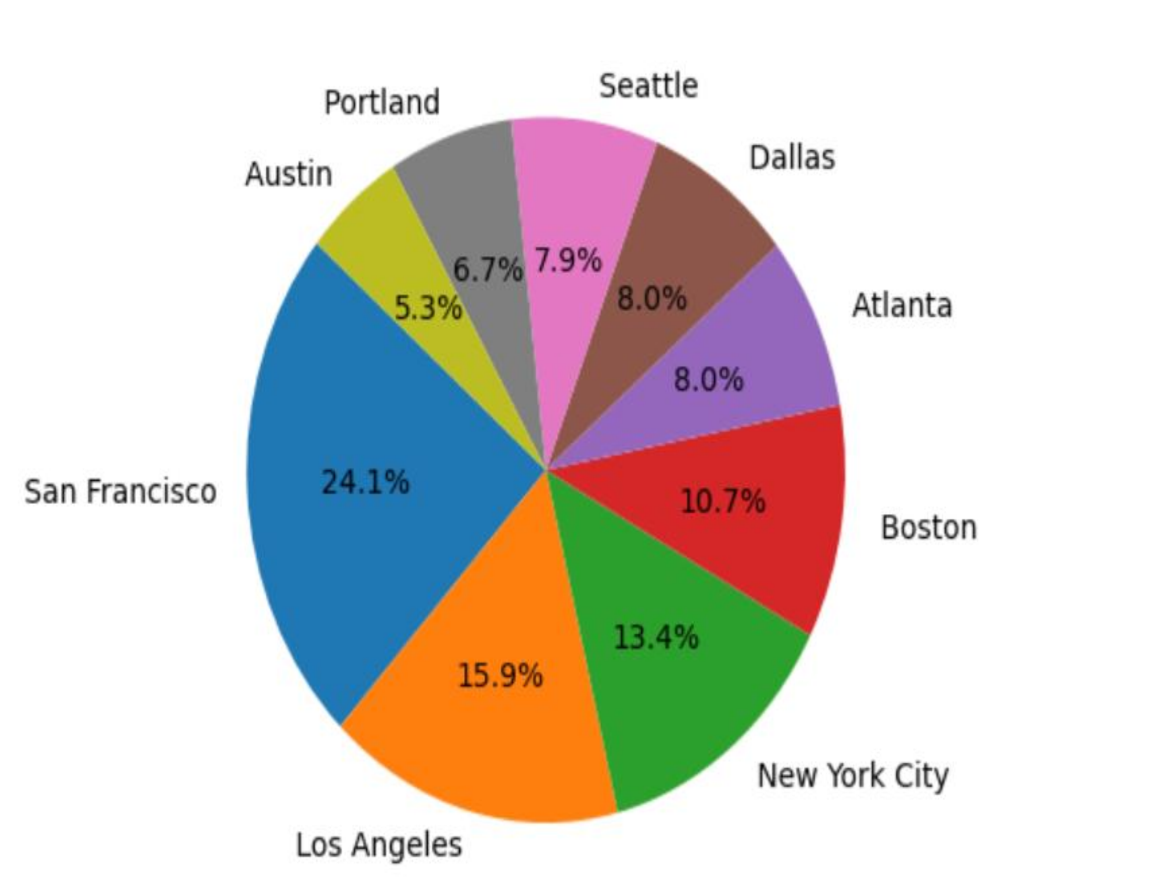
3. <u>Graphical Analysis</u>

❑ **5. <u>Project Workflow</u>:- (Exploratory Data Analysis)**

City Wise Revenue



City wise Order Percentage

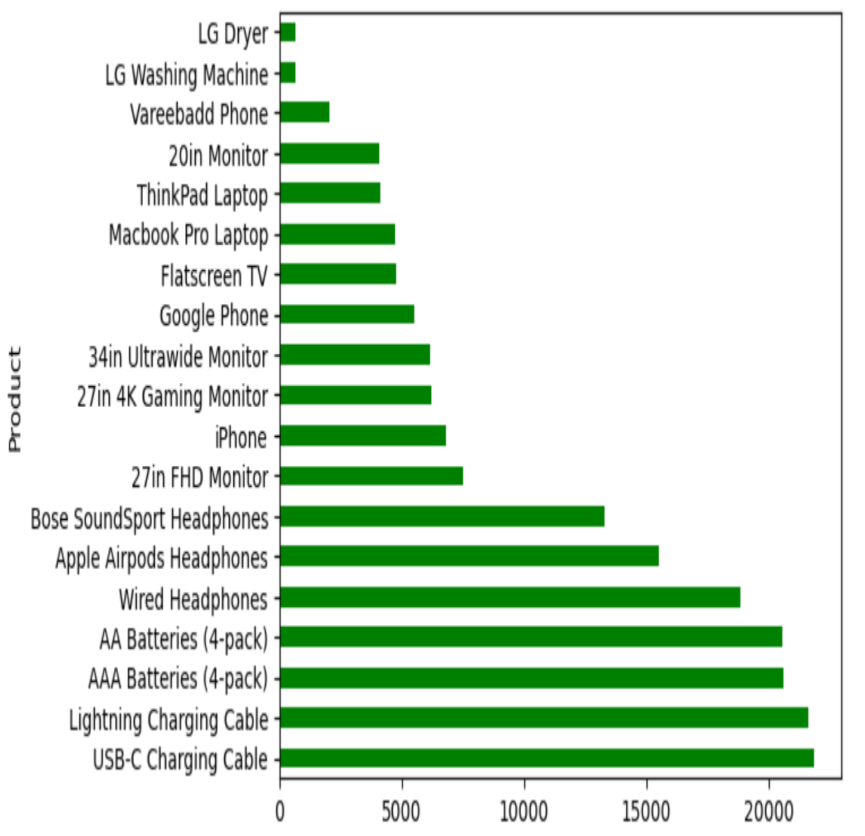


E-commerce Sales Analysis

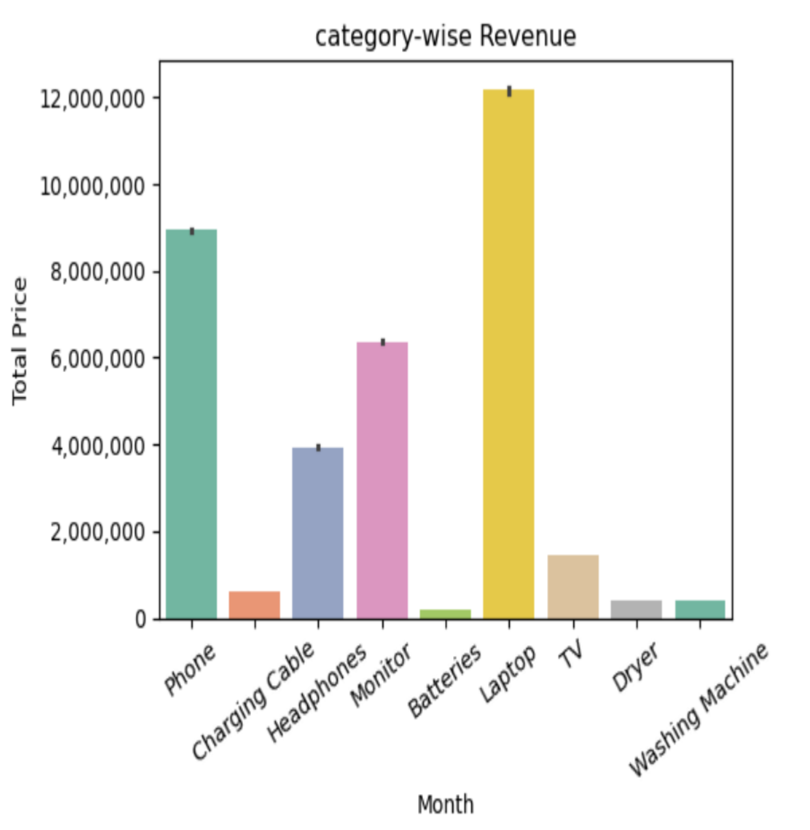
3. Graphical Analysis

❑ **5. Project Workflow:- (Exploratory Data Analysis)**

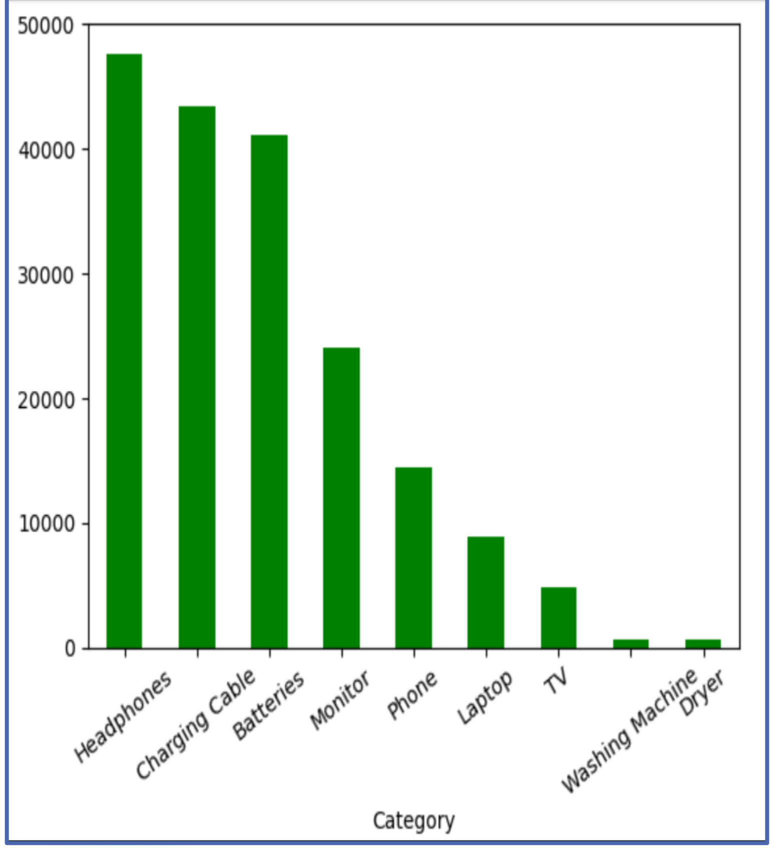
Product Wise Orders Count



Category wise Revenue



Category wise Orders count



E-commerce Sales Analysis

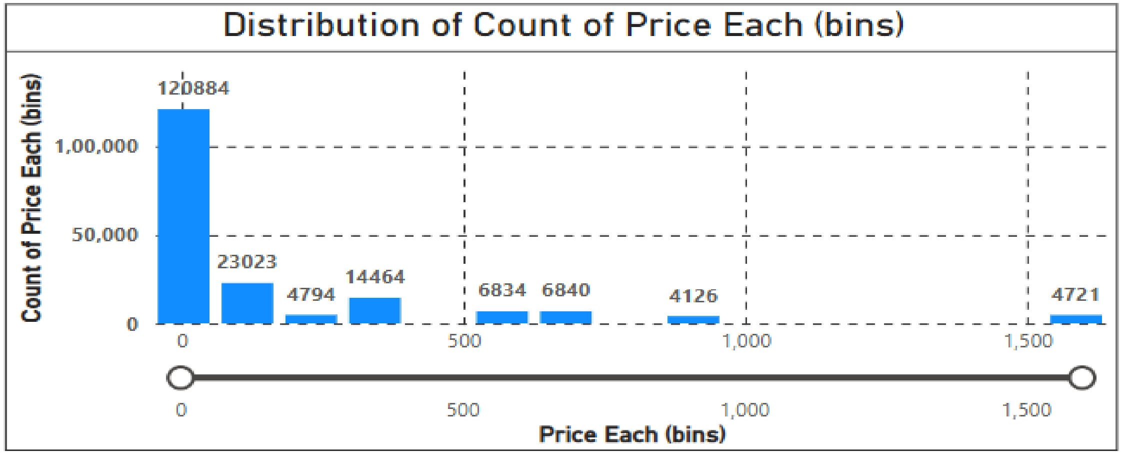
Dashboard-(Distribution)

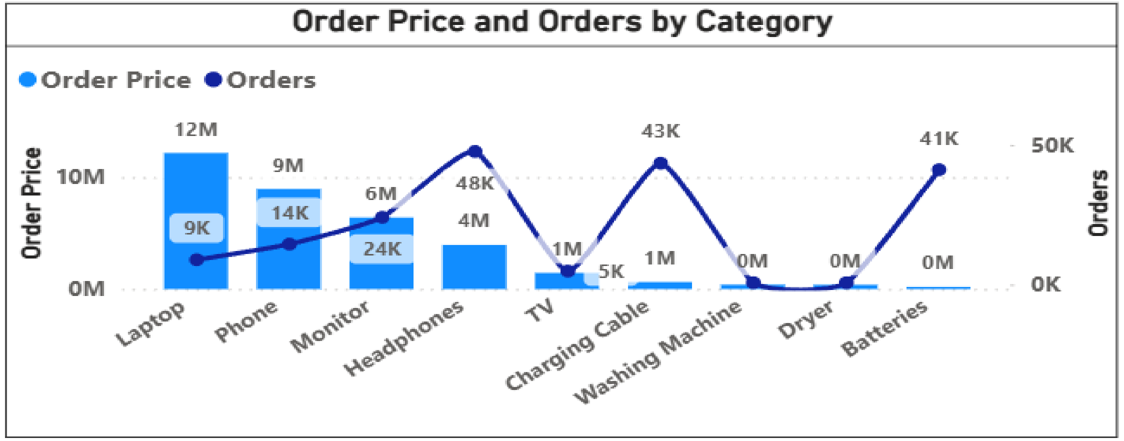
❑ 5.<u>Project Workflow</u> : (Dashboard)

Ecommerce Sales Analysis

Month	Category	City	Quarter
All	All	All	All

185686	208812	2.87M
Total Orders	Total Quantity	Revenue/Month



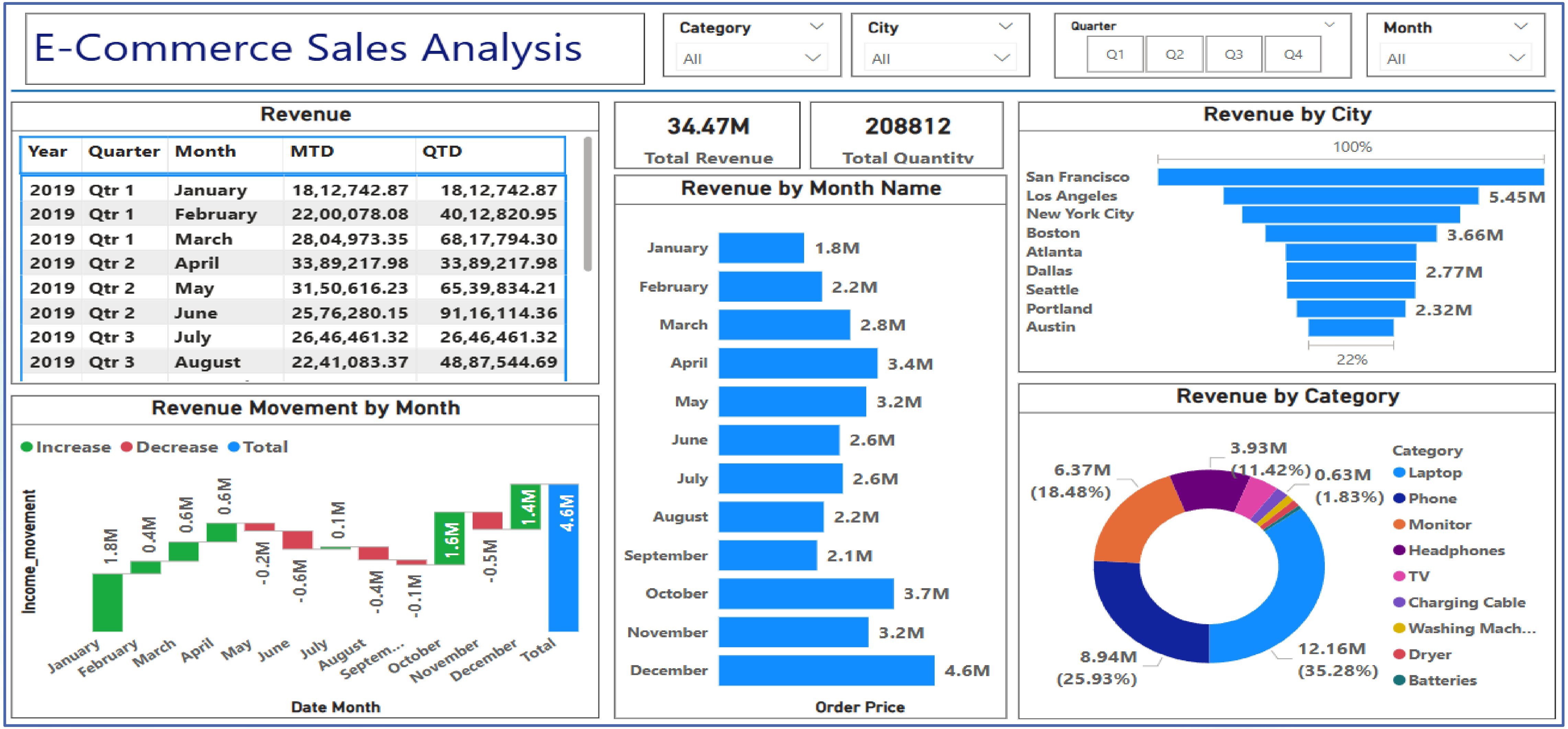


Category	Product	Price Each	Quantity
Batteries	AAA Batteries (4-pack)	2.99	30986
Batteries	AA Batteries (4-pack)	3.84	27615
Charging Cable	USB-C Charging Cable	11.95	23931
Headphones	Wired Headphones	11.99	20524
Charging Cable	Lightning Charging Cable	14.95	23169
Headphones	Bose SoundSport Headphones	99.99	13430
Monitor	20in Monitor	109.99	4126
Monitor	27in FHD Monitor	149.99	7541
Headphones	Apple Airpods Headphones	150.00	15637
TV	Flatscreen TV	300.00	4813
Monitor	34in Ultrawide Monitor	379.99	6192
Monitor	27in 4K Gaming Monitor	389.99	6239
Phone	Vareebadd Phone	400.00	2068
Phone	Google Phone	600.00	5529
Dryer	LG Dryer	600.00	646
Washing Machine	LG Washing Machine	600.00	666
Phone	iPhone	700.00	6847
Laptop	ThinkPad Laptop	999.99	4128
Laptop	Macbook Pro Laptop	1,700.00	4725

E-commerce Sales Analysis

Dashboard-(Revenue)

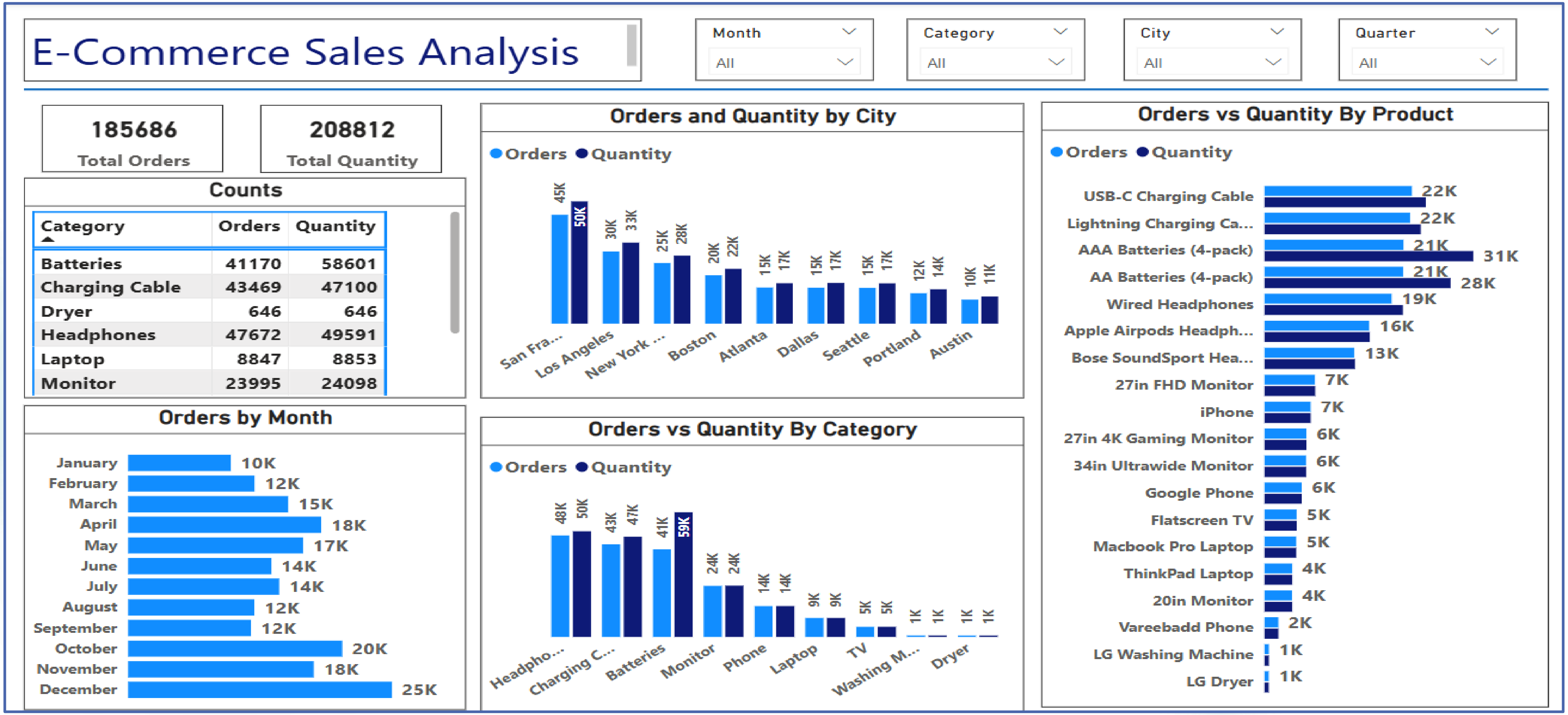
❑ **<u>5.Project Workflow</u> : (Dashboard)**



E-commerce Sales Analysis

Dashboard-(Counts)

❑ 5. **Project Workflow** : (Dashboard)



Category	Orders	Quantity
Batteries	41170	58601
Charging Cable	43469	47100
Dryer	646	646
Headphones	47672	49591
Laptop	8847	8853
Monitor	23995	24098

E-commerce Sales Analysis

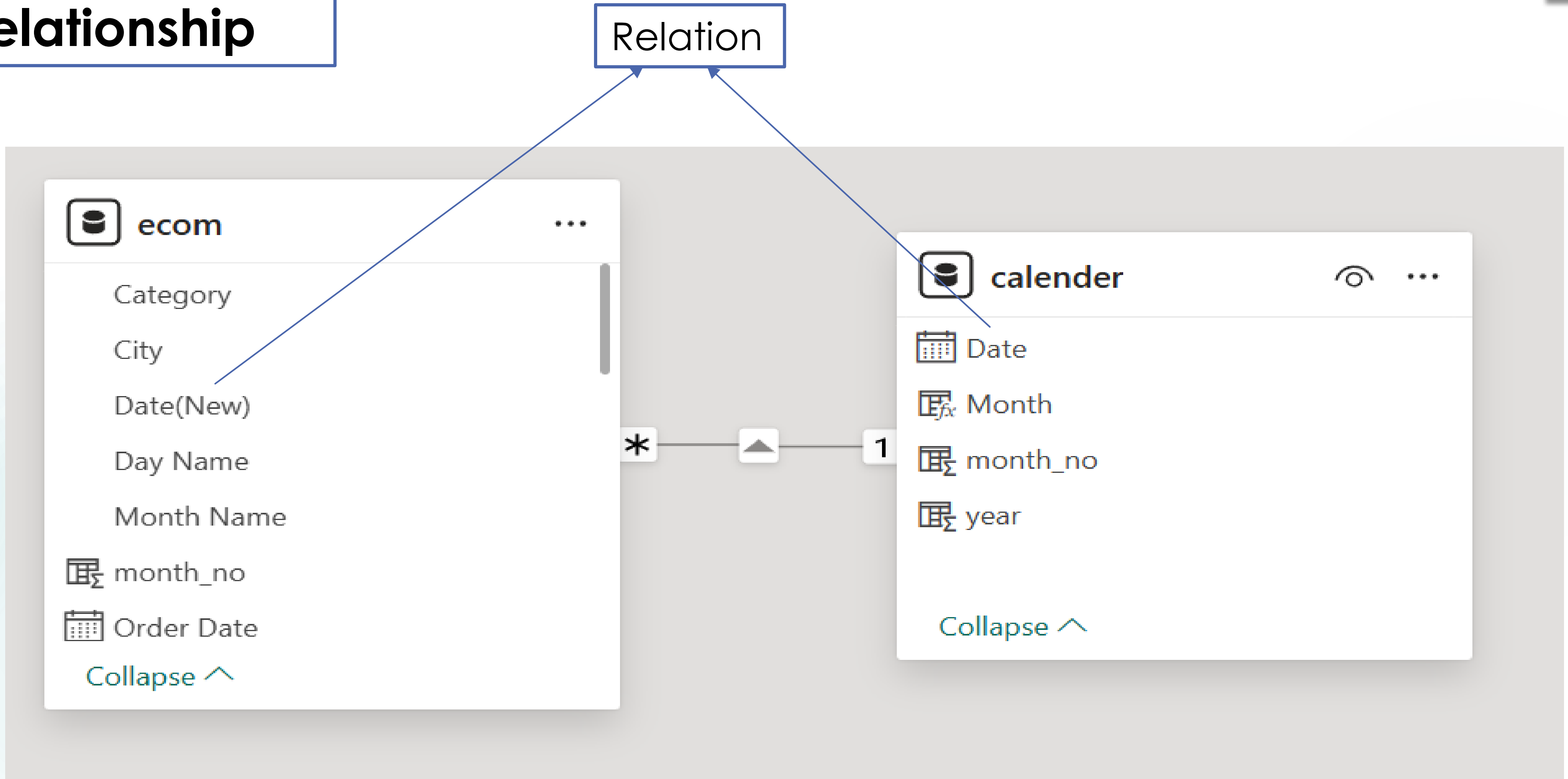
❑Dashboard (DAX)

1. calender = CALENDAR("2019-01-01","2020-01-01")
2. Month = FORMAT(calender[Date].[Date] , "MMMM")
3. month_no = MONTH(calender[Date].[Date])
4. year = YEAR(calender[Date])
5. Income_movement = SUM(ecom[Order Price])-[Prior_month_revenue]
6. month_avg_revenue= CALCULATE(AVERAGEX(VALUES(calender[Month]),CALCULATE(SUM(ecom[Order Price]))) ,

   calender[year] = 2019)
7. MTD = TOTALMTD(SUM(ecom[Order Price]) ,calender[Date].[Date])
8. QTD = TOTALQTD(SUM(ecom[Order Price]) ,calender[Date].[Date])
9. Order Price = ecom[Quantity Ordered] * ecom[Price Each]
10. Prior_month_revenue = CALCULATE(SUM(ecom[Order Price]) , DATEADD(calender[Date].[Date],-1, MONTH))
11. Quarter = "Q" & QUARTER(ecom[Date(New)])

E-commerce Sales Analysis

❑ Relationship

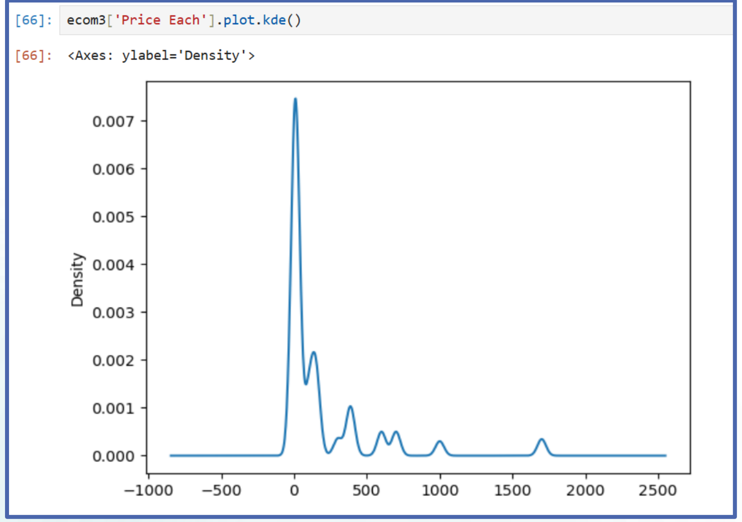


E-commerce Sales Analysis

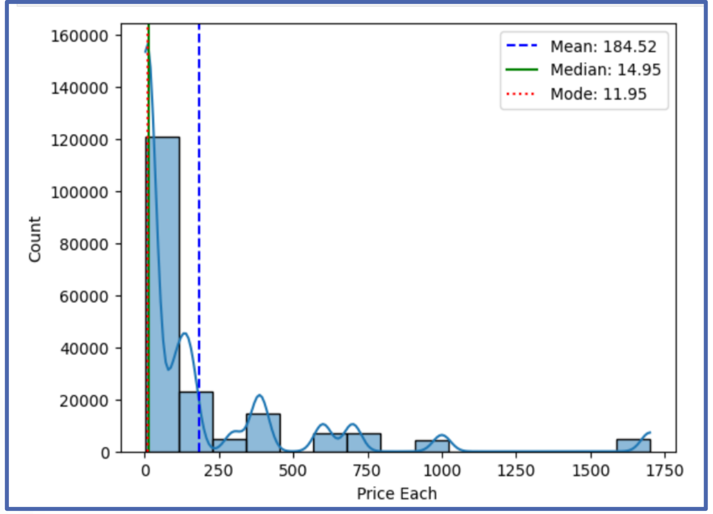
Distribution

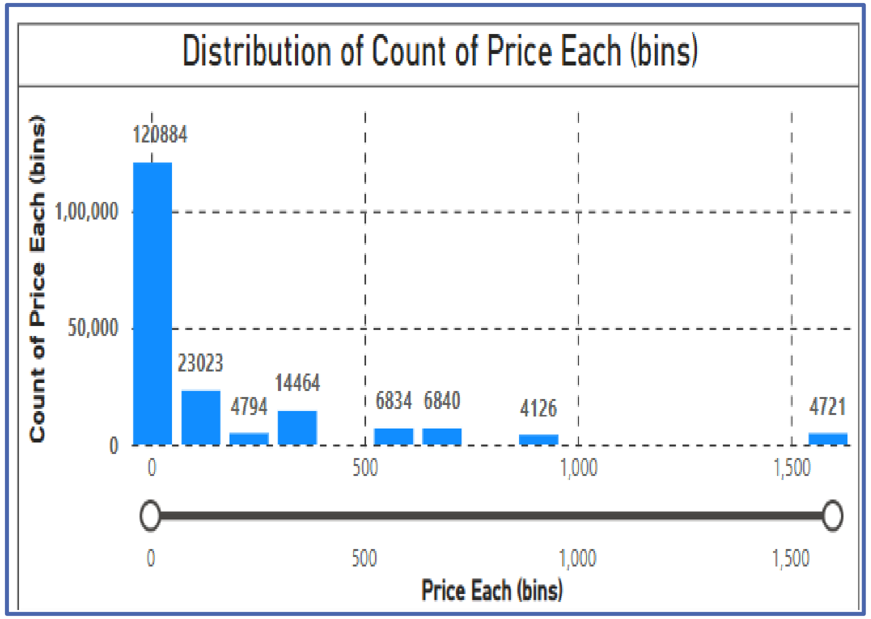
❑6. Graphical Analysis

Product Price Distribution Plot



Product Price stats: count , mean ,median, mode (before and after outlier treatment)






```
Standard deviation before= 332.8438383899937
Standard deviation after= 71.1309982733632
```

- What is Distribution of the Product Price ?
- Mean , Median , Standard Deviation- how far the data point lies around the mean?
- What is the count of Product Price in a certain range?
- What is the difference between mean and median?

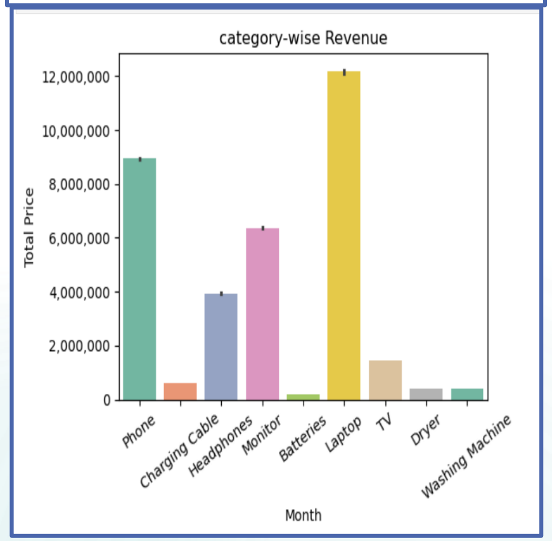
1. What is the Product Price Range?
2. How many product are ordered in a certain range?
3. What is the Product name and ordered quantity (form the Product list)?

E-commerce Sales Analysis

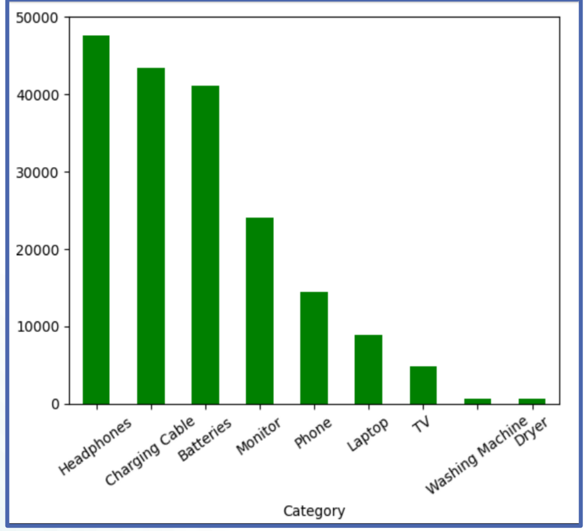
Distribution

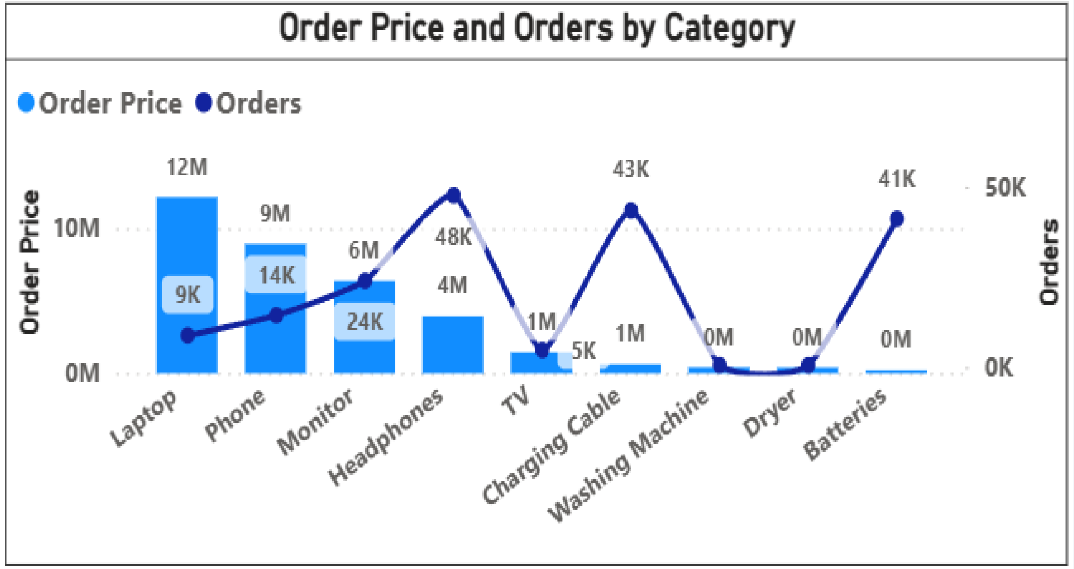
❑6. Graphical Analysis

Category wise Revenue



Category wise Orders count





Category vs Total order price stats

Category	sum	median	mean	std	min	max
Batteries	198689.74	3.84	4.826081	2.620210	2.99	26.91
Charging Cable	632352.00	14.95	14.547195	4.284343	11.95	71.70
Dryer	387600.00	600.00	600.000000	0.000000	600.00	600.00
Headphones	3934498.46	99.99	82.532691	60.176281	11.99	450.00
Laptop	12160458.72	1700.00	1374.529074	351.892158	999.99	3400.00
Monitor	6370939.02	379.99	265.511107	126.352025	109.99	779.98
Phone	8937500.00	600.00	619.498163	102.797811	400.00	1400.00
TV	1443900.00	300.00	301.188986	18.850897	300.00	600.00
Washing Machine	399600.00	600.00	600.000000	0.000000	600.00	600.00

```
ecom3['Category'].value_counts()
```

```
Category
Headphones        47672
Charging Cable    43469
Batteries         41170
Monitor           23995
Phone             14427
Laptop             8847
TV                 4794
Washing Machine     666
Dryer               646
Name: count, dtype: int64
```

1. Is the Product Ordered count and Revenue ratio is same? (comparison)
2. What is the order count and Revenue for a Particular category?

E-commerce Sales Analysis

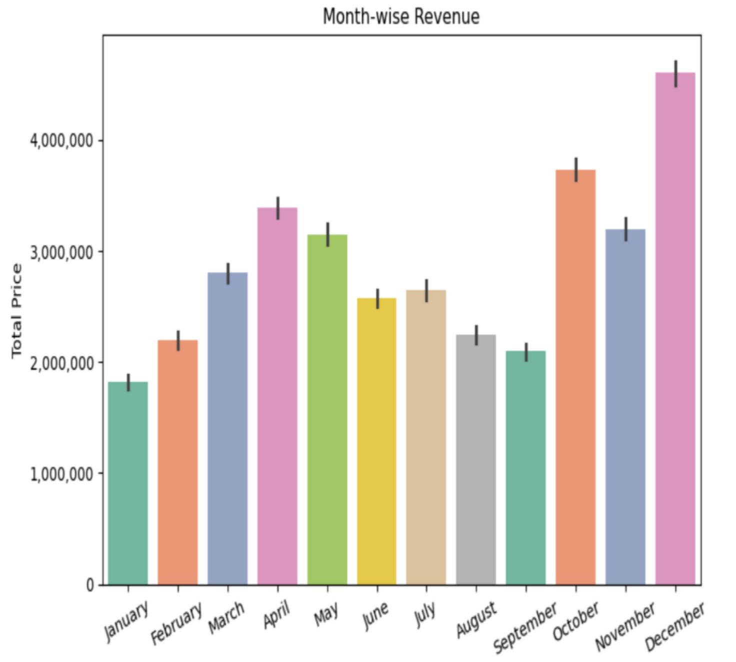
Revenue

❑6. Graphical Analysis

Monthly revenue stats

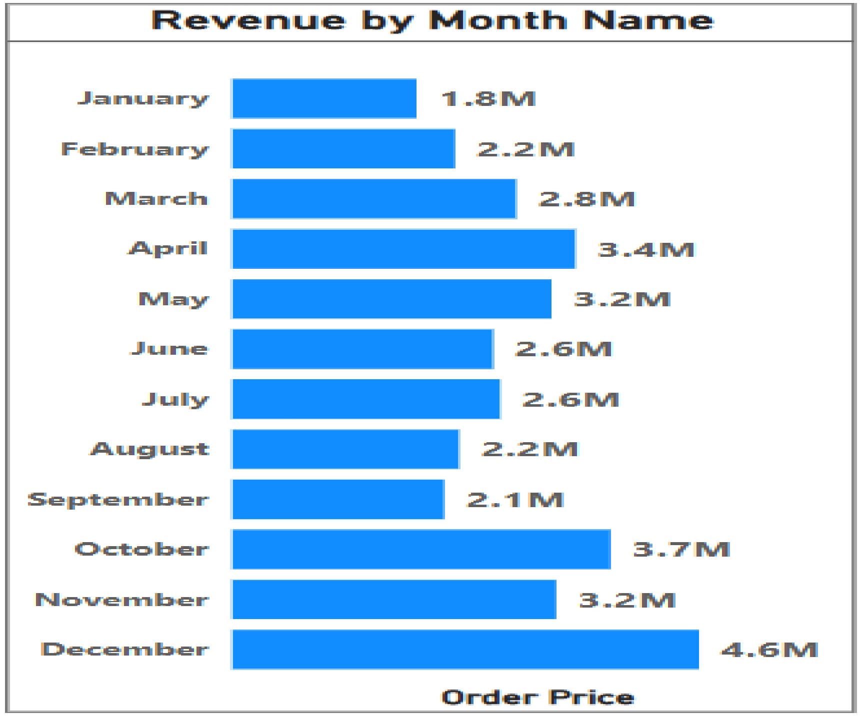
Month_name	sum	median	mean	std
April	3389217.98	14.95	185.639370	332.141314
August	2241083.37	14.95	187.648277	332.193945
December	4608295.70	14.95	184.745658	332.820367
February	2200078.08	14.95	183.999170	325.988793
January	1821413.16	14.95	187.793913	331.092526
July	2646461.32	14.95	185.391336	333.024103
June	2576280.15	14.95	190.342087	337.582608
March	2804973.35	14.95	185.416007	331.556813
May	3150616.23	14.95	190.346558	342.964978
November	3197875.05	14.95	182.277420	330.306825
October	3734777.86	14.95	184.442583	334.363393
September	2094465.69	14.95	180.510703	328.478786

Monthly Revenue



- What is the mean , median and standard deviation of monthly revenue? (help to understand the data distribution)
- How the monthly revenue bar graph look like(about distribution)?

34.47M	208812
Total Revenue	Total Quantity



1. What is the total revenue?
2. What is the monthly revenue trends?

E-commerce Sales Analysis

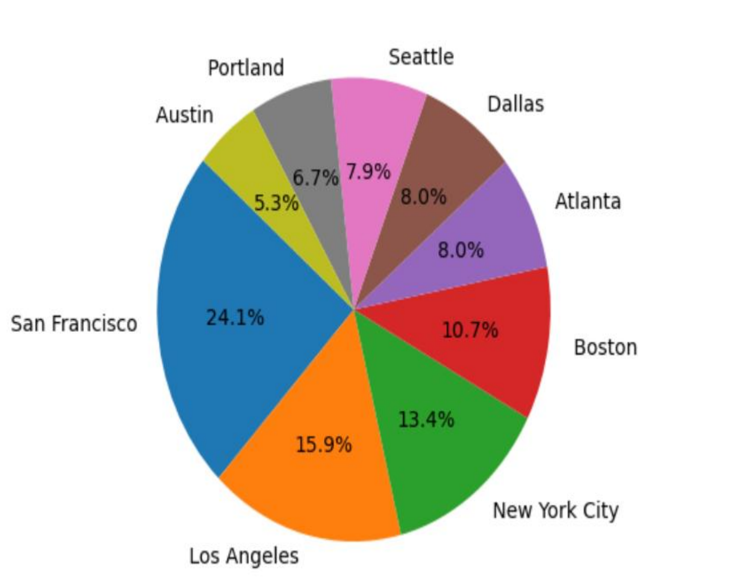
Revenue

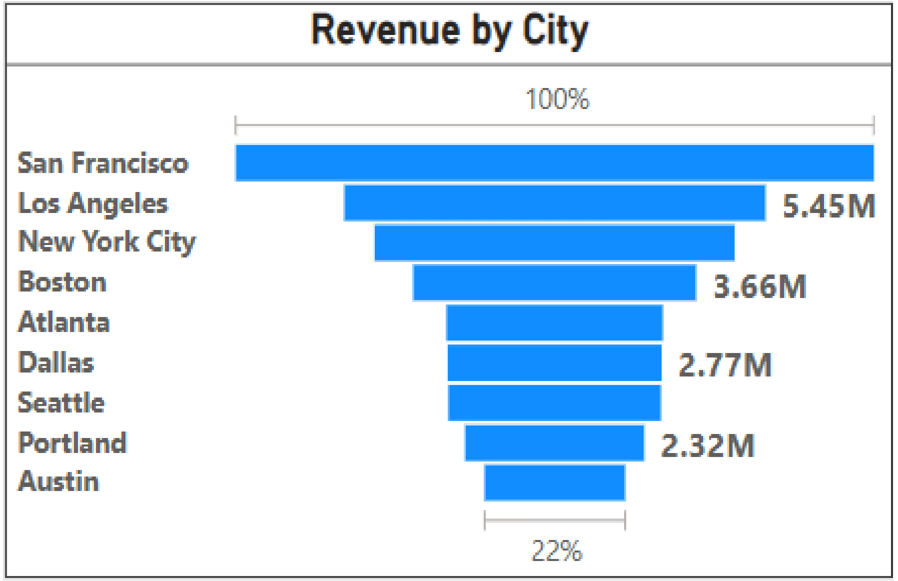
❑6. Graphical Analysis

City vs total order price stats

City	sum	median	mean	std	min	max
Atlanta	2794199.07	14.95	187.996977	334.876275	2.99	1700.0
Austin	1818044.33	14.95	183.826525	331.628368	2.99	1700.0
Boston	3658627.65	14.95	183.841397	329.352885	2.99	3400.0
Dallas	2765373.96	14.95	186.887474	334.446402	2.99	1700.0
Los Angeles	5448304.28	14.95	184.288468	332.243107	2.99	1700.0
New York City	4661867.14	14.95	187.622938	336.359270	2.99	3400.0
Portland	2319331.94	14.95	186.306686	337.153437	2.99	1700.0
San Francisco	8254743.55	14.95	184.827002	332.675745	2.99	3400.0
Seattle	2745046.02	14.95	186.572828	329.165595	2.99	1700.0

City wise Order Percentage





1. Which city causes the max revenue and min revenue?
2. What is the Top and button revenue cities?
3. What are the revenue generated by each city?

- What is the mean , median , standard deviation , min , max of revenue by city?
- What is the share percentage of total revenue of each city?

E-commerce Sales Analysis

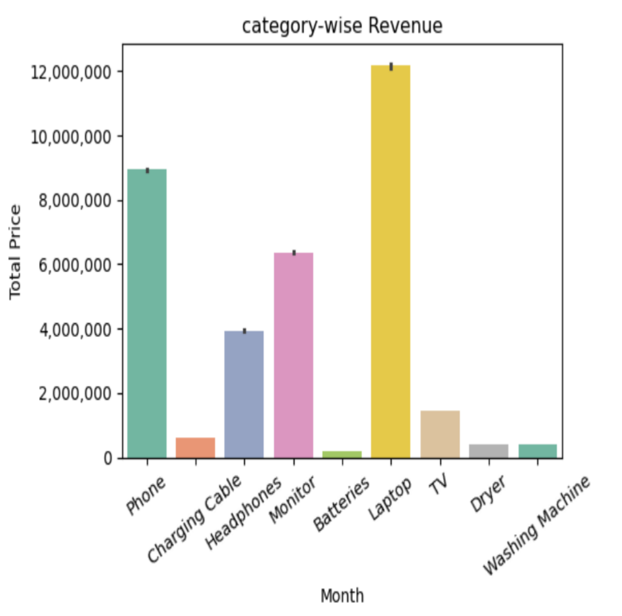
Revenue

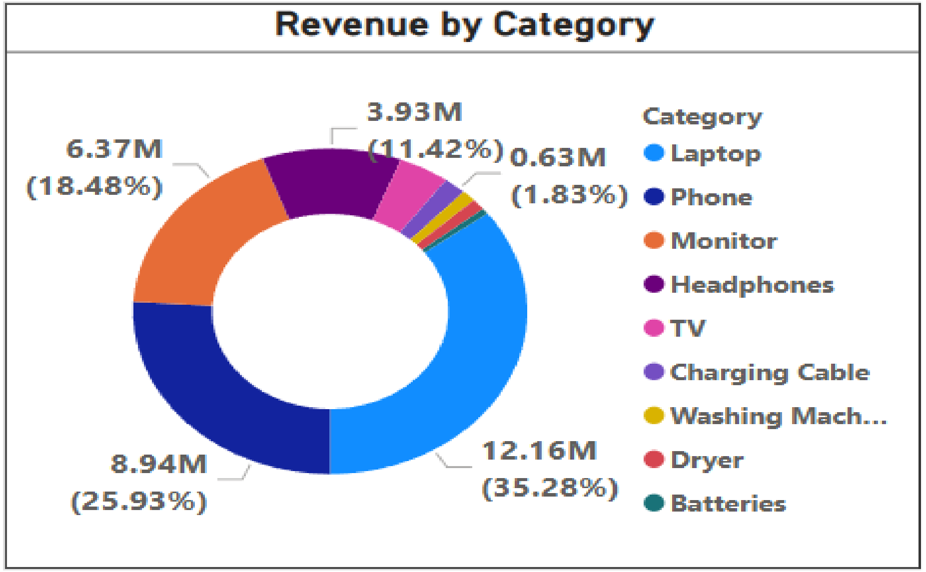
❑6. Graphical Analysis

Category vs Total order price stats

	sum	median	mean	std	min	max
Category						
Batteries	198689.74	3.84	4.826081	2.620210	2.99	26.91
Charging Cable	632352.00	14.95	14.547195	4.284343	11.95	71.70
Dryer	387600.00	600.00	600.000000	0.000000	600.00	600.00
Headphones	3934498.46	99.99	82.532691	60.176281	11.99	450.00
Laptop	12160458.72	1700.00	1374.529074	351.892158	999.99	3400.00
Monitor	6370939.02	379.99	265.511107	126.352025	109.99	779.98
Phone	8937500.00	600.00	619.498163	102.797811	400.00	1400.00
TV	1443900.00	300.00	301.188986	18.850897	300.00	600.00
Washing Machine	399600.00	600.00	600.000000	0.000000	600.00	600.00

Category wise Revenue





- What is the revenue distribution of product category?
- Min , max price of order product price by each category?
- What is the distribution of revenue by category?

1. What is the revenue generated by each category?
2. What is the share percentage of revenue generation by each category?
3. High and low revenue generated category?

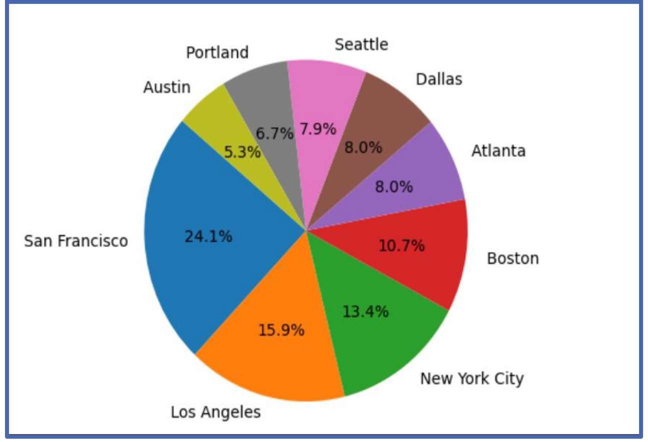
E-commerce Sales Analysis

Counts

❑6. Graphical Analysis

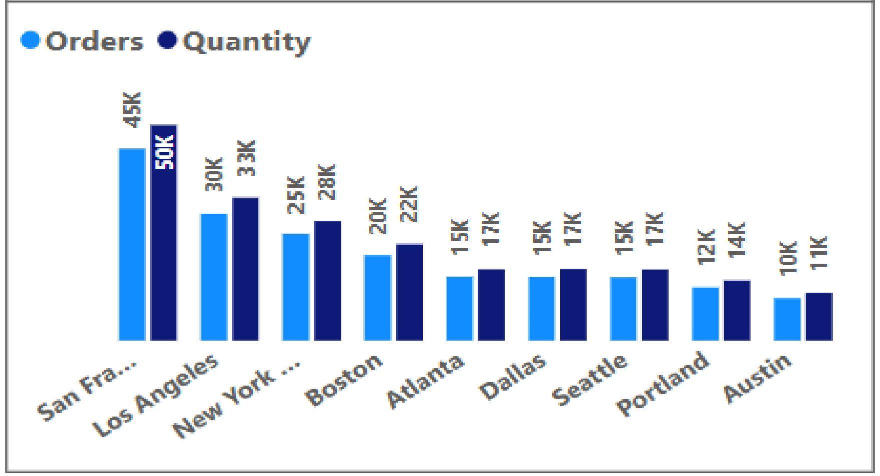
```
# city wise Orders:
ecom3['City'].value_counts()

City
San Francisco     44662
Los Angeles       29564
New York City     24847
Boston            19901
Atlanta           14863
Dallas            14797
Seattle           14713
Portland          12449
Austin             9890
Name: count, dtype: int64
```

City wise Order Percentage



- What is the city wise order count?
- What are the percentage orders by each city?

Orders and Quantity by City



1. Orders vs quantity Orders by each city?
2. What are the orders count and ordered quantity for each city? (helps in stock and inventory)

E-commerce Sales Analysis

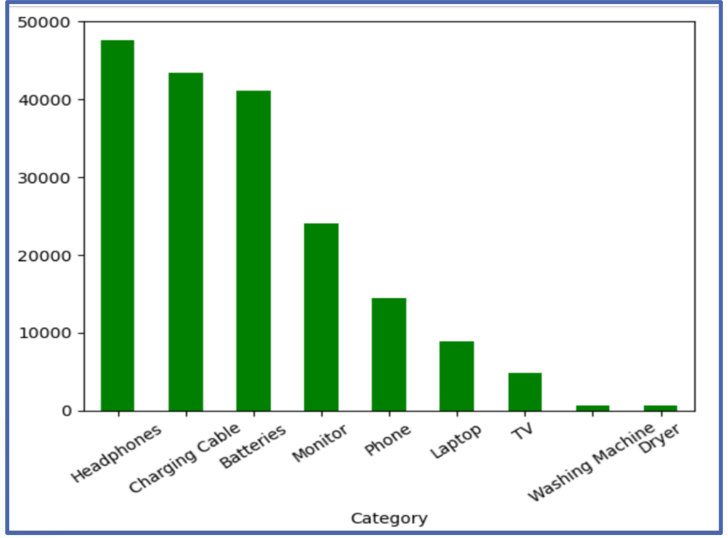
Counts

❑6. Graphical Analysis

```python
# category wise no. of orders:
ecom3['Category'].value_counts()
```

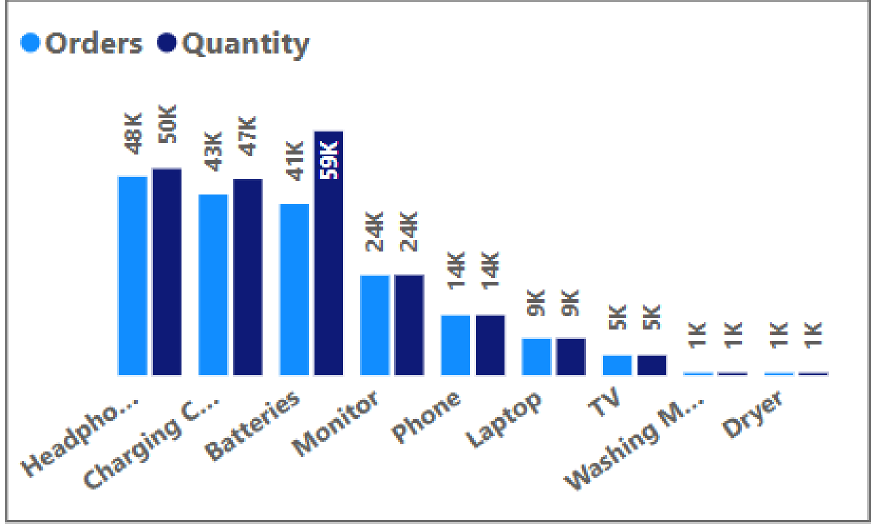
```
Category
Headphones          47672
Charging Cable      43469
Batteries           41170
Monitor             23995
Phone               14427
Laptop               8847
TV                   4794
Washing Machine       666
Dryer                 646
Name: count, dtype: int64
```

Category wise Orders count



- What is the order count of each category?
- What is the distribution of orders by product category?

Orders vs Quantity By Category



1. Orders vs orders quantity by category?
2. What is the orders and orders quantity of a product category? (help in stock and inventory)

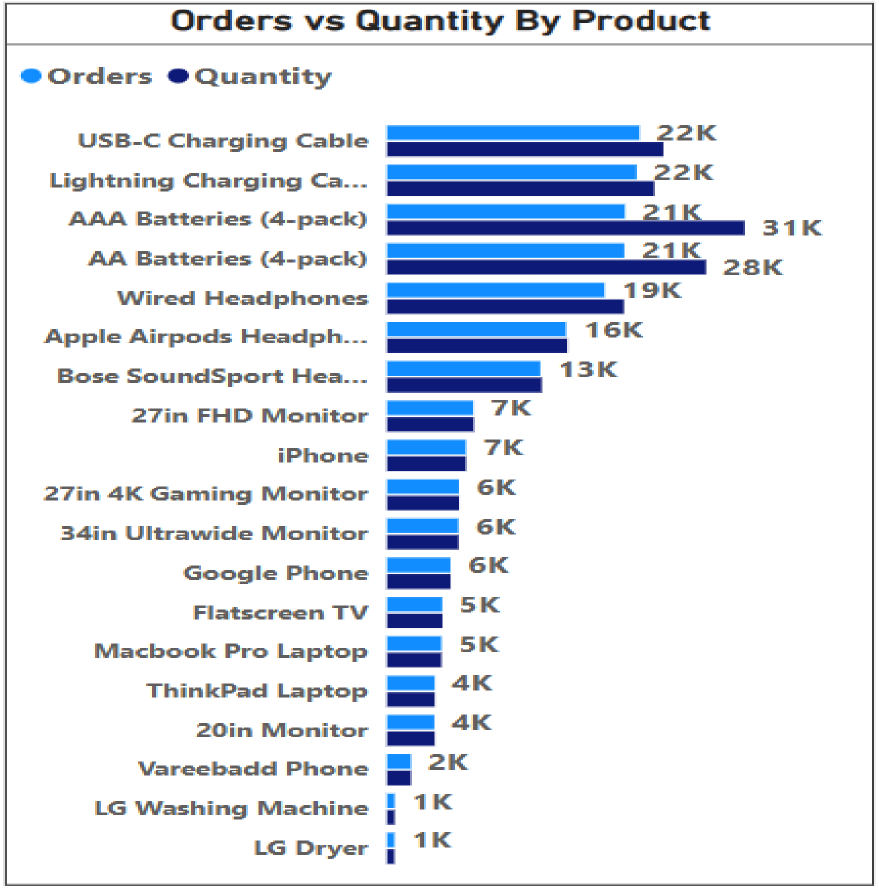
E-commerce Sales Analysis

Count

❑6. Graphical Analysis

```python
# Total number of product Orders
ecom3.groupby(['Product'])['Quantity Ordered'].sum()
```

```
Product
20in Monitor                  4126
27in 4K Gaming Monitor        6239
27in FHD Monitor              7541
34in Ultrawide Monitor        6192
AA Batteries (4-pack)        27615
AAA Batteries (4-pack)       30986
Apple Airpods Headphones     15637
Bose SoundSport Headphones   13430
Flatscreen TV                 4813
Google Phone                  5529
LG Dryer                       646
LG Washing Machine             666
Lightning Charging Cable     23169
Macbook Pro Laptop            4725
ThinkPad Laptop               4128
USB-C Charging Cable         23931
Vareebadd Phone               2068
Wired Headphones             20524
iPhone                        6847
Name: Quantity Ordered, dtype: int32
```



1. What is the product ordered count and quantity ordered for each product?
2. Orders vs quantity orders comparison

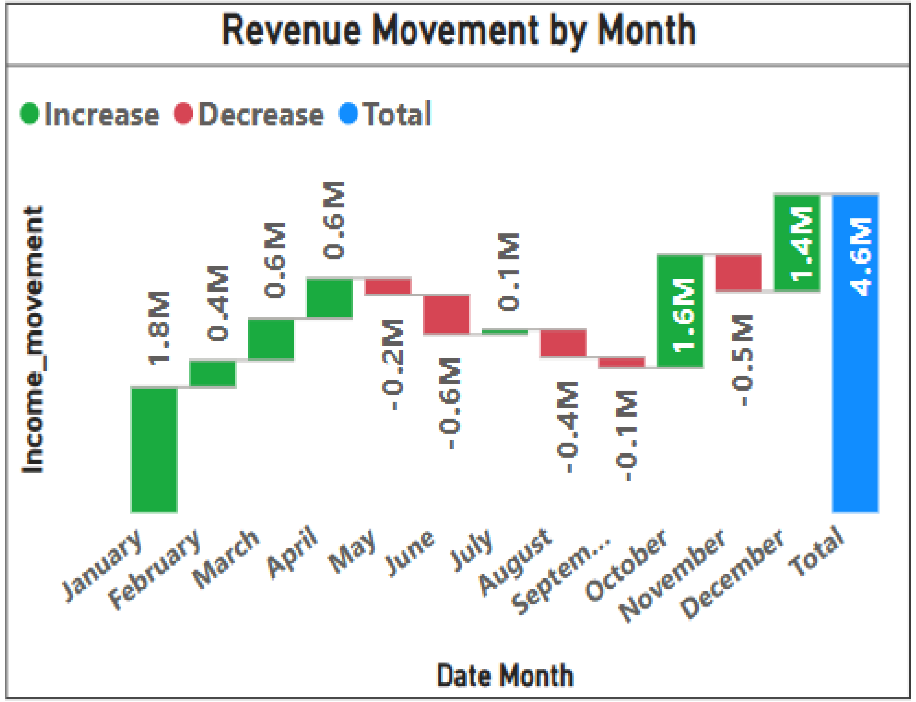
E-commerce Sales Analysis

Monthly Stats

❑6. Graphical Analysis

Monthly revenue stats

Month_name	sum	median	mean	std
April	3389217.98	14.95	185.639370	332.141314
August	2241083.37	14.95	187.648277	332.193945
December	4608295.70	14.95	184.745658	332.820367
February	2200078.08	14.95	183.999170	325.988793
January	1821413.16	14.95	187.793913	331.092526
July	2646461.32	14.95	185.391336	333.024103
June	2576280.15	14.95	190.342087	337.582608
March	2804973.35	14.95	185.416007	331.556813
May	3150616.23	14.95	190.346558	342.964978
November	3197875.05	14.95	182.277420	330.306825
October	3734777.86	14.95	184.442583	334.363393
September	2094465.69	14.95	180.510703	328.478786



1. Which month revenue increases from the last month or decreases?
   - month wise revenue movement

E-commerce Sales Analysis

❑7. Challenges & Solution

1. **While using Combine Files in Power BI to append monthly e-commerce data, inconsistent column structures caused load errors by using append data method** .
2. **To match data counting between Exploratory Data Analysis (EDA) in tools like Python/Excel and Power BI, it's important to ensure consistency** .
3. **Handled inconsistent and incomplete data across multiple CSVs.**
4. **Created custom time intelligence logic without built-in date columns.**
5. **Managed sorting of textual month names in visuals for correct chronological order.**
6. **Built dynamic filtering and KPI visuals to allow category-wise comparison.**
7. **Optimized large dataset for faster refresh and responsiveness in Power BI.**

E-commerce Sales Analysis