CSE 587 Project Phase 1

Market Basket Analysis Recommendation Engine

Vivek Singh (vsingh28 | 50418786)

Sougato Bagchi (sougatob | 50411918)

Problem Statement:

Background: - Traditionally the idea of shopping meant buying only necessity items by the customers. This is the typical scenario before the onset of consumerism. But now a days with greater affordability, purchasing power and various kinds of items available in the market, it is necessity to close the bridge between the retailers/sellers and the customers. There are various examples to elaborate the problems we are trying to deal with in this project.

Nowadays people not only buy necessity items, but often do they buy items many items which may not be a necessity to them but are related to the item which they thought of buying. A person thinks of buying a smartphone which maybe his primary objective, but while buying if his eyes catch on to a Fast Charger which is better than the one included in-the-box of the phone then he might think of buying that too as a bundle. This is a win-win situation for both the parties i.e., the retailers as well as the customer. But here's a catch this is only possible if he/she is suggested with this secondary item, or he has a prior knowledge.

From the point of the retailer, its necessary for them to retain the customers who buy items from them such that it generates higher revenue. For that reason, the retailers need to deploy strategies. And it's better if they can analyze if their customers are leaving them for joining their competitors and if yes, then why?

We are trying to analyze our collected data from external sources to derive insights, patterns, relationships and how these can be used in designing, developing, and implementing a strategy (like understanding customer behavior of selecting the item, popular trends, and segmenting the profitable /non-profitable customers) to improve the retail business' profits, revenue, and overall operations.

Data Sources:

- Online retail data set: https://archive.ics.uci.edu/ml/datasets/online+retail
- This is a UK based transnational dataset which contains the records of all transactions happened between year 2010 and 2011 at non-store online retail.

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom

- Let's have a look at each attribute and its type:
- From above table we can observe that there is total 541,908 number of transactions happening on daily basis. We have:

```
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype
---  ------       --------------   -----
 0   InvoiceNo    541909 non-null  object
 1   StockCode    541909 non-null  object
 2   Description  540455 non-null  object
 3   Quantity     541909 non-null  int64
 4   InvoiceDate  541909 non-null  datetime64[ns]
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   406829 non-null  float64
 7   Country      541909 non-null  object
```

1. Invoice No: represent the transaction ID
2. Stock Code: product code details of what is the product.
3. Description: Product name
4. Quantity: number of items bought at each transaction.
5. Invoice Date: at what time or day the transaction took place
6. Unit Price: price of the product
7. Customer ID: represent the customer id who had bought the item.
8. Country: In which country this transaction has been made.

Data Cleaning/Processing:

- Before starting the cleaning process, we understood the dataset and found that, there are:
  - Total unique transaction: 25900
  - Total unique Products: 4070
  - Total unique Customers: 4372
  - Total unique Countries: 38
- Now have look into Numerical columns: Quantity and UnitPrice and check its statistics:

	Quantity	UnitPrice
count	541909.000000	541909.000000
mean	9.552250	4.611114
std	218.081158	96.759853
min	-80995.000000	-11062.060000
25%	1.000000	1.250000
50%	3.000000	2.080000
75%	10.000000	4.130000
max	80995.000000	38970.000000

- We can see that there are **some negative values** in Quantity and unit price column.
- After careful analysis, we found that these are the values coming **from cancellation in transactions** were recorded in our data.
- In below table we could see C in InvoiceNo which represent the cancellation.

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
141	C536379	D	Discount	-1	2010-12-01 09:41:00	27.50	14527.0	United Kingdom
154	C536383	35004C	SET OF 3 COLOURED FLYING DUCKS	-1	2010-12-01 09:49:00	4.65	15311.0	United Kingdom
235	C536391	22556	PLASTERS IN TIN CIRCUS PARADE	-12	2010-12-01 10:24:00	1.65	17548.0	United Kingdom
236	C536391	21984	PACK OF 12 PINK PAISLEY TISSUES	-24	2010-12-01 10:24:00	0.29	17548.0	United Kingdom
237	C536391	21983	PACK OF 12 BLUE PAISLEY TISSUES	-24	2010-12-01 10:24:00	0.29	17548.0	United Kingdom

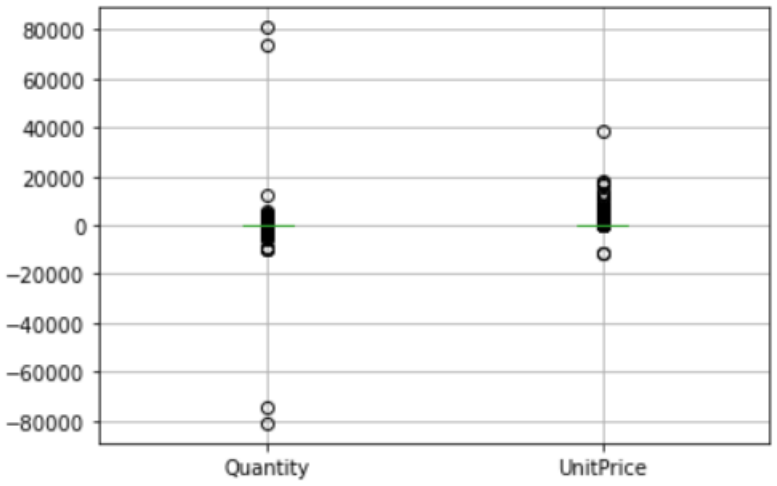
- With such details we are cleared about removing such transaction is necessary because this will lead to the inaccurate information to our model.
- Now, we have checked for **null values** in all the attributes and then replaced them with appropriate values during data pre-processing.
- Here are some of our findings and corresponding implemented strategies.
  - "**CustomerID**" has large volume of null values.
    - These null values are replaced with string "New_Customers". We have considered them to be new-customers or those customers who buy items without creating an account.

```
Total numbers of row in dataset:  541909

Count of null values:
 InvoiceNo          0
StockCode           0
Description      1454
Quantity            0
InvoiceDate         0
UnitPrice           0
CustomerID     135080
Country             0
```

  - "**Description**" has less numbers of null values (1,454) in a total of total 541909 entries.
    - We decided on dropping the following rows with null “Description”.

- For better understanding data and transaction in daily, weekly, and monthly basis we **separate the date columns into day, week, month, day of week, time of day.**
- **Removing duplicate entries and outliers** (mainly negative values from Quantity and UnitPrice).
  - Negative values in "UnitPrice" represents cancelled orders so these are removed.
  - Also, the negative values for the attribute “Quantity”, have been removed.



- **Creating new column called – “Sales_Revenue”** using “Quantity” and “UnitPrice" column to calculate monetary value for our analysis.

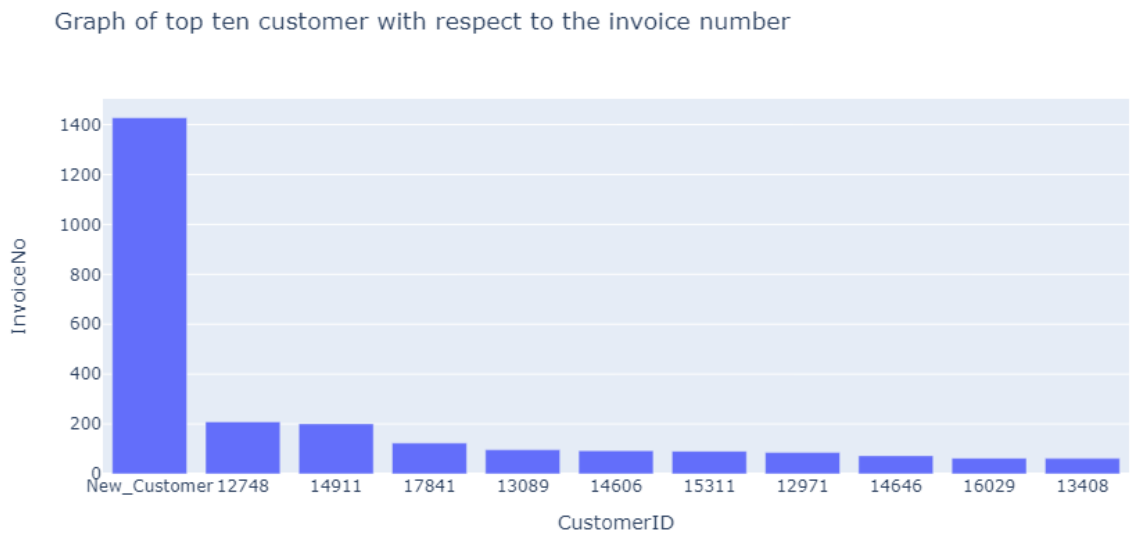
- **Dropping Rows –**
  - We have dropped some of the rows which we believed will not be fruitful for any kind of analysis. These are:
    - Where we have sales revenue = 0
    - Country name unspecified.

Exploratory Data Analysis:

- Finally, after all these pre-processing our dataset looks like.
- We have successfully added columns:
  - **Date column**: to understand transaction on daily basis.
  - **Month column:** to understand the transaction on monthly basis.
  - **Year column**: to understand transaction on yearly basis
  - **Day of Week**: on which day of week to understand the transaction.
  - **Sales revenue**: to capture total revenue made per every transaction.

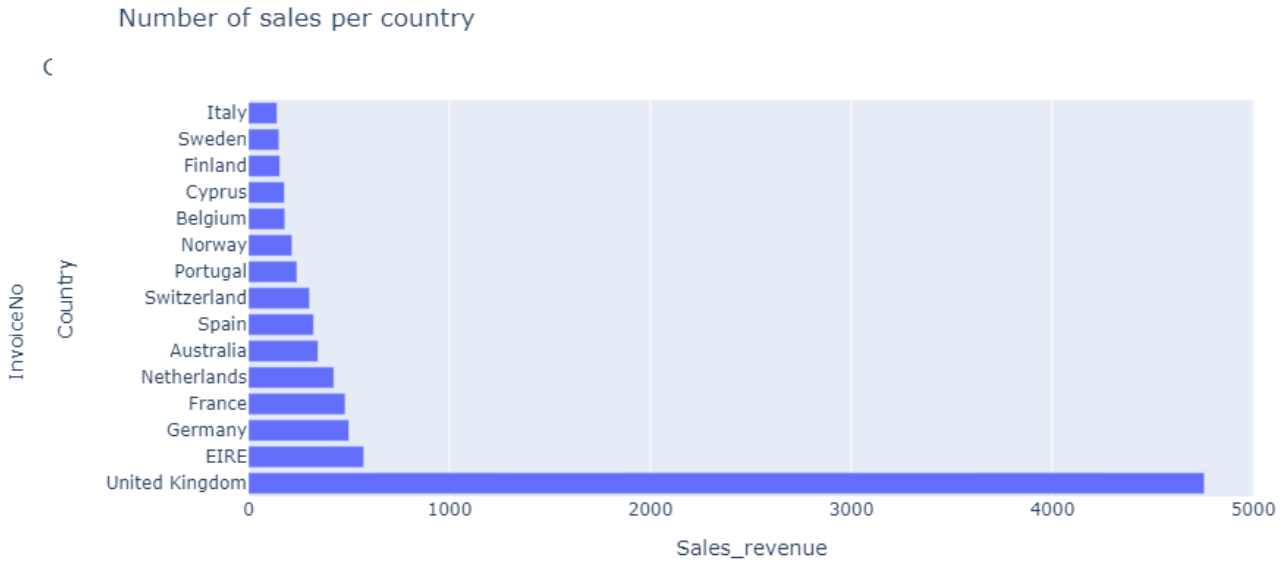
```
Data columns (total 13 columns):
 #   Column         Non-Null Count   Dtype
---  ------         --------------   -----
 0   InvoiceNo      524878 non-null  object
 1   StockCode      524878 non-null  object
 2   Description    524878 non-null  object
 3   Quantity       524878 non-null  int64
 4   InvoiceDate    524878 non-null  datetime64[ns]
 5   UnitPrice      524878 non-null  float64
 6   CustomerID     524878 non-null  object
 7   Country        524878 non-null  object
 8   Date           524878 non-null  object
 9   Month          524878 non-null  object
 10  Year           524878 non-null  int64
 11  Day of Week    524878 non-null  object
 12  Sales_revenue  524878 non-null  float64
```

Customer Analysis w.r.t their invoice number/no. of transactions.



	CustomerID	InvoiceNo
0	New_Customer	1428
1	12748	209
2	14911	201
3	17841	124
4	13089	97
5	14606	93
6	15311	91
7	12971	86
8	14646	73
9	16029	63
10	13408	62

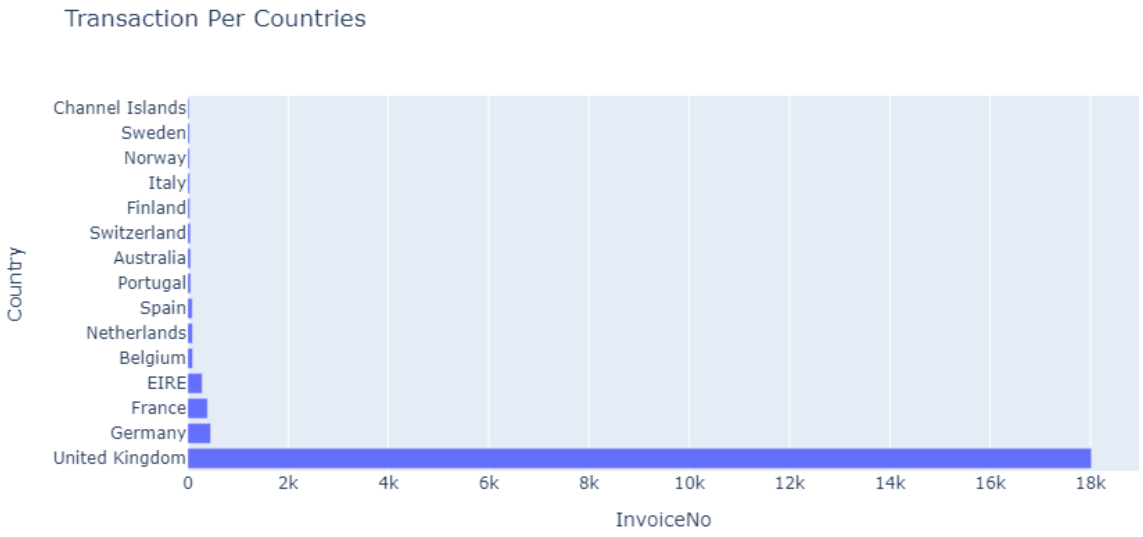
- Here our graphs look imbalanced as guest customers/New Customer has been included. That means there is a huge proportion of guest customers.
- This graph without the guest customers looks kind of balanced, with the highest no. of orders made by a customer in that retail company at around 200.



- Sales revenue per country:

	Country	Sales_revenue
0	United Kingdom	4757
1	EIRE	574
2	Germany	500
3	France	481
4	Netherlands	425
5	Australia	346
6	Spain	324
7	Switzerland	304
8	Portugal	241
9	Norway	217
10	Belgium	181
11	Cyprus	179

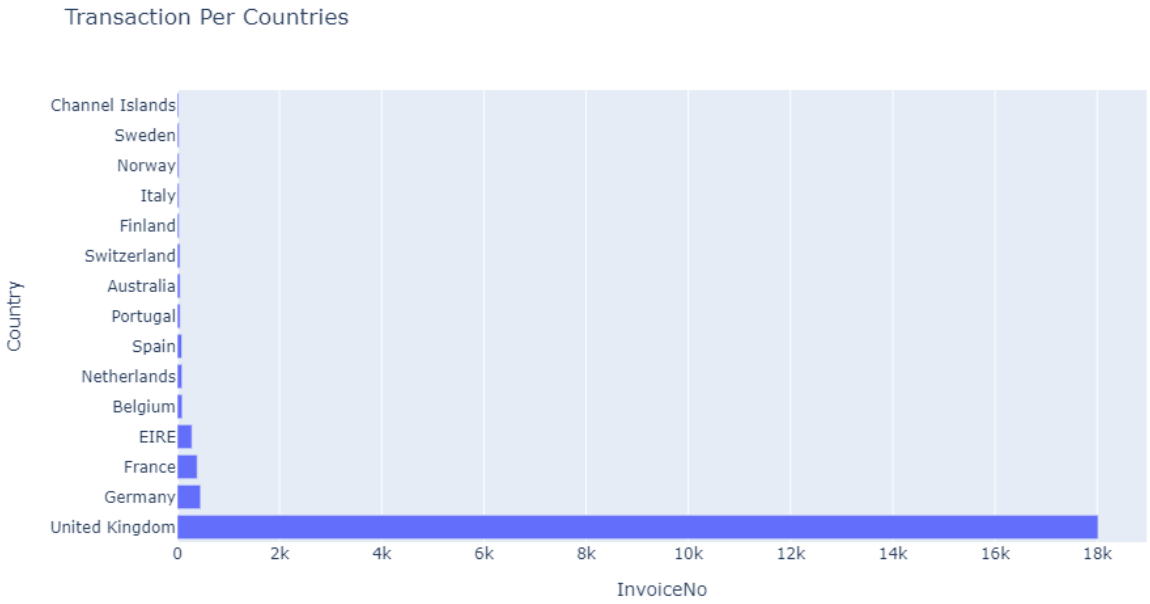
- Here the UK has very high revenue difference with other countries. This is since the retailer “4757” is a UK based retailer.



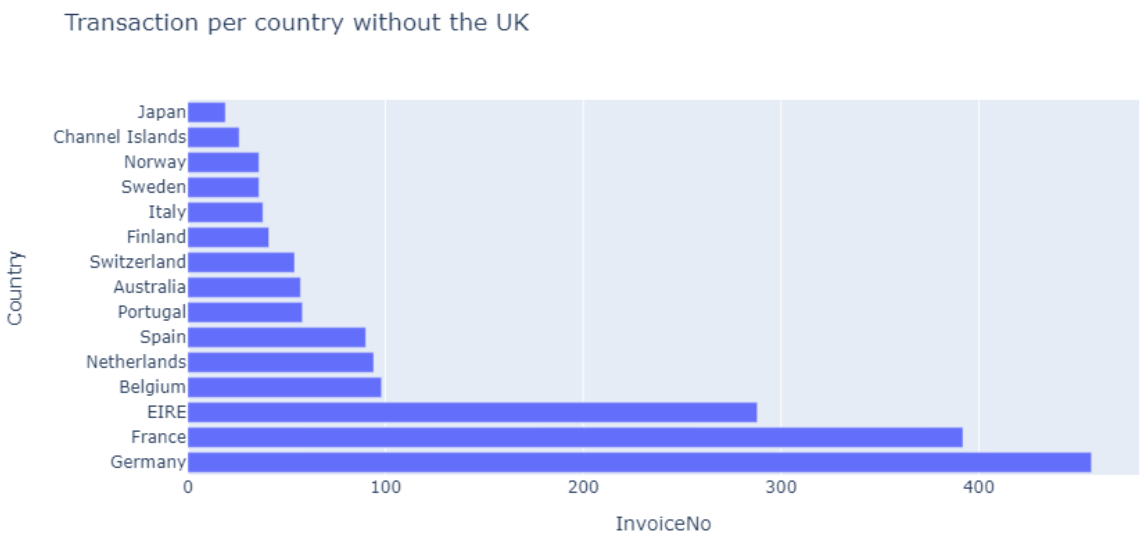
- **Transaction per Countries**

	Country	InvoiceNo
0	United Kingdom	18019
1	Germany	457
2	France	392
3	EIRE	288
4	Belgium	98
5	Netherlands	94
6	Spain	90
7	Portugal	58
8	Australia	57
9	Switzerland	54

  - The UK seems to be having a huge difference in the number of transactions. This is since the dataset is from a UK based retailer.



- **Transaction per Countries without UK**



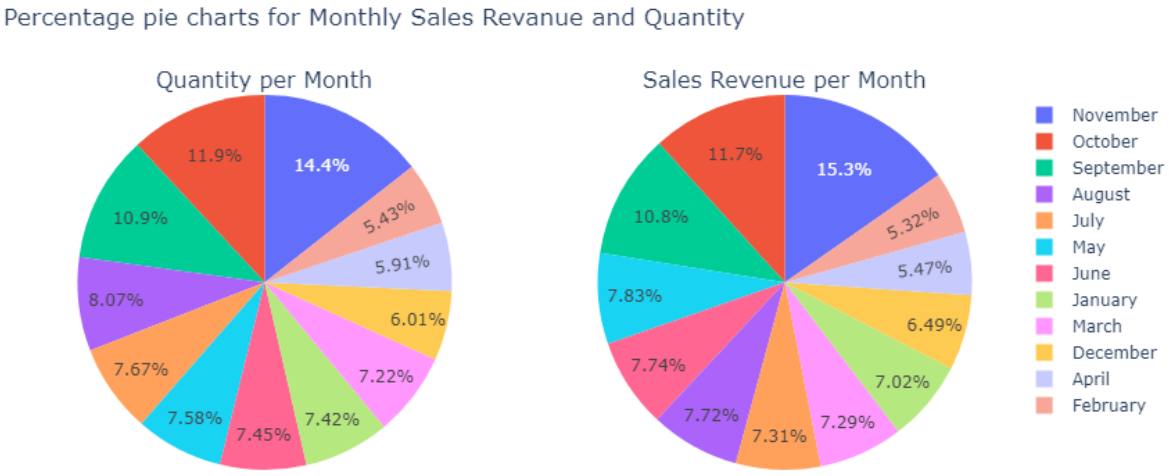
To analyze the relative no. of transactions between different countries in the EU we decide to not take UK into account.

- Checking Revenue per month for a particular year (2011)
  - We clearly see that revenue from sales is at peak for the months of November, October, and September.
  - This can be due to the fact of forthcoming holiday season in the European countries.

	Month	Quantity	Sales_revenue
0	November	751377	1503866.780
1	October	621029	1151263.730
2	September	569573	1056435.192
3	May	395001	769296.610
4	June	388511	760547.010
5	August	421020	757841.380
6	July	399693	718076.121
7	March	376599	716215.260
8	January	387099	689811.610
9	December	313612	637790.330
10	April	307953	536968.491
11	February	282934	522545.560

- Monthly Sales Revenue and Quantity:

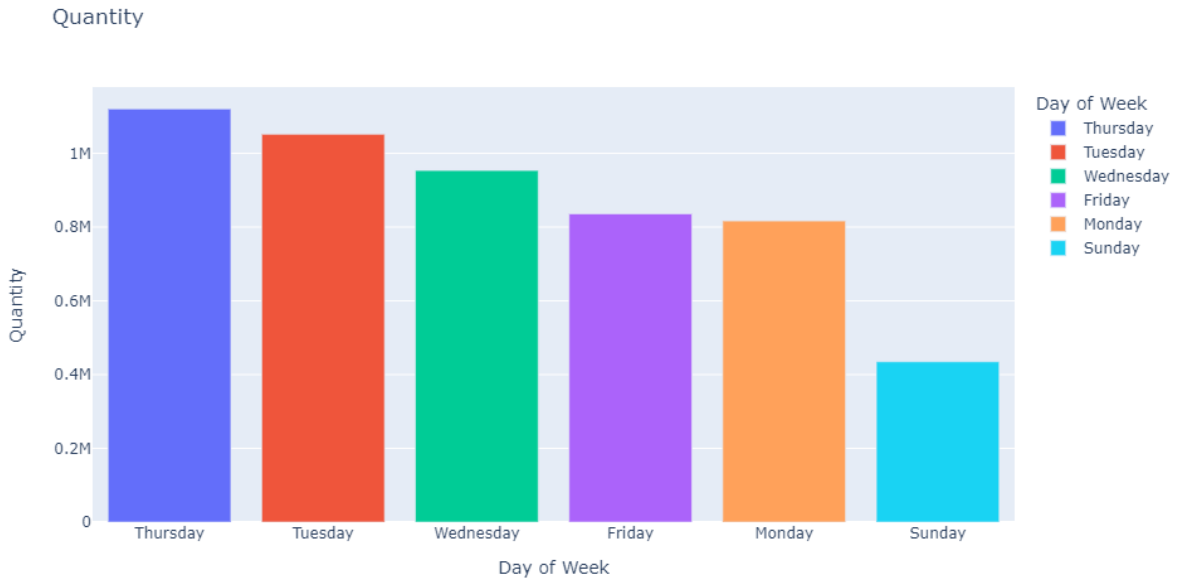
Here we see that both graphs are like each other. We can say that items with very high price is bought in less frequency as high quantity of sales generally is corresponding to high sales revenue.

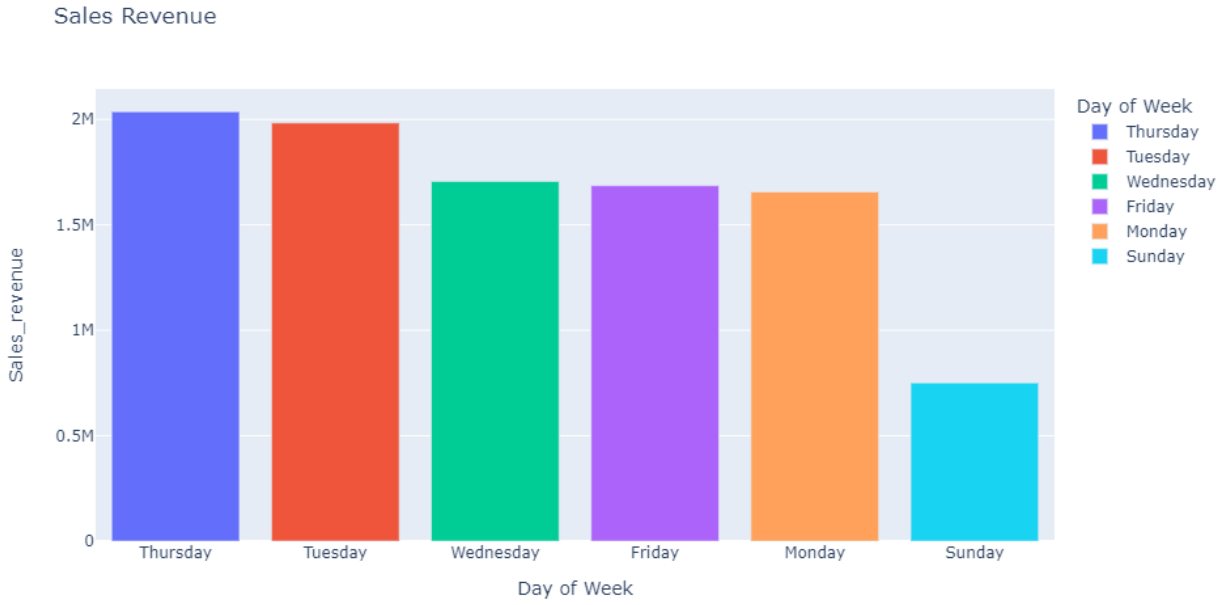


- **Sales Revenue & Quantity w.r.t days: -**
  - Here we have analyzed the proportions in terms of absolute numbers and in terms of percentage. As both types reveal some different insights.

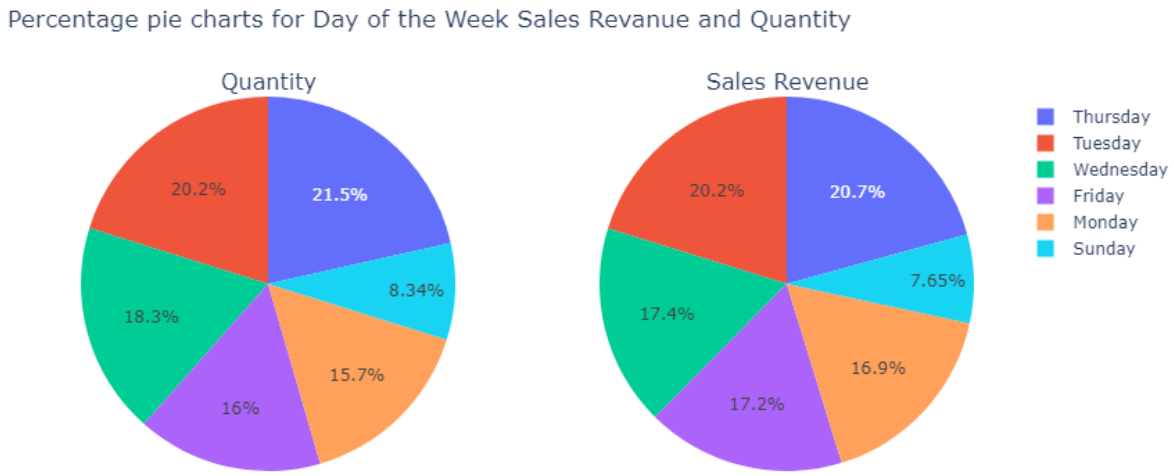
	Day of Week	Quantity	Sales_revenue
0	Thursday	1120610	2036928.910
1	Tuesday	1052047	1983699.211
2	Wednesday	953701	1706477.890
3	Friday	836133	1686212.711
4	Monday	816914	1656449.321
5	Sunday	434996	750890.031

- Product analysis w.r.t its frequency of being sold and the revenue these items generated individually: -





- Here we have seen that the item which is most frequently bought doesn't generate highest amount of revenue for the company. From this we may infer that the selling price of the item is less than other items and from the item "Dotcom Postage".



- **Top Product Transaction per unit sale**:
- Here we observed the top 5 product name, and the number of Quantity is sold, and the total revenue generated from them.

	Description	Quantity	Sales_revenue
0	PAPER CRAFT , LITTLE BIRDIE	80995	168469.60
1	MEDIUM CERAMIC TOP STORAGE JAR	78033	81700.92
2	WORLD WAR 2 GLIDERS ASSTD DESIGNS	49756	12639.88
3	JUMBO BAG RED RETROSPOT	46220	90140.66
4	POPCORN HOLDER	36749	34288.67

- Here we have recorded the name of the products and Quantity sold. Here is the top 5 products name which had generated the highest revenue.

	Description	Quantity	Sales_revenue
0	DOTCOM POSTAGE	652	181577.58
1	PAPER CRAFT , LITTLE BIRDIE	80995	168469.60
2	REGENCY CAKESTAND 3 TIER	11774	146461.78
3	PARTY BUNTING	18046	98237.49
4	WHITE HANGING HEART T-LIGHT HOLDER	34002	95002.50