



**Module Code & Module Title**

**CC4057NP Introduction to Information Systems**

**Assessment Weightage & Type**

**30% Individual Coursework**

**Year and Semester**

**2021-22 Autumn**

**Student Name: Yurisha Acharya**

**Group: c2**

**London Met ID: 21050026**

**College ID:** **NP04CP4A210102**

**Assignment Due Date:** **4th May 2023**

**Assignment Submission Date:** **4th May 2023**

*I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.*

Acknowledgement

I am appreciative to Mr. Badri, our honorable teacher, for helping me complete this project effectively. He provided us with his invaluable insight and understanding which helped us immensely with our reports and we were able able to complete our project in due time.

I also want to express our gratitude to our college for giving us all the tools we needed for the project. Overall, we would want to thank everyone who contributed to this project and offered comments that helped us improve it.

Last but not least, I'm not sure whether we would have arrived here today without their assistance along the way, so I want to extend my gratitude to everyone else who helped us finish this amazing project together by contributing in some manner along the way and finally, we want to express our gratitude to our parents and friends for being there for us no matter what.

Table of Contents

[1. Introduction 1](#_Toc134115284)

[1.1 Smart Data 1](#_Toc134115285)

[1.2 Business Data and Data processing 4](#_Toc134115286)

[1.3 Data Processing Cycle 5](#_Toc134115287)

[1.4 Conclusion 8](#_Toc134115288)

[2. Objectives 9](#_Toc134115289)

[3. Overview of Coursework 10](#_Toc134115290)

[4. Task Development 12](#_Toc134115291)

[4.1 Data Understanding 12](#_Toc134115292)

[4.2 Data preparation 13](#_Toc134115293)

[4.3 Data Analysis 22](#_Toc134115294)

[4.4 Data Exploration 25](#_Toc134115295)

[5. Conclusion 33](#_Toc134115296)

**List of Figures**

Figure 1: Importing Library 13

Figure 2: Merging multiple CSV files into one and removing unused rows 13

Figure 3: Checking missing value in data set 15

Figure 4: Calculating missing value in data set 16

Figure 5: filtering out NaN values in a dataset 16

Figure 6: Calculating missing value in dataset after filtering NaN values 17

Figure 7: Output after filtering NaN values 17

Figure 8: Convert Quantity Ordered and Price each in numeric 18

Figure 9: creates a new column called Month in the DataSet 19

Figure 10: Returning the first five rows 20

Figure 11: Creating new column City from Purchase Address 21

Figure 12: To show summary statistics of sum, mean, standard deviation, skewness, and kurtosis of Quantity Ordered 22

Figure 13: Calculating the correlation matrix 23

Figure 14: To calculating and show correlation of all variables 24

Figure 15: Best Selling and Earning Month 25

Figure 16: Bar graph of Sales 26

Figure 17: Calculating the sum of sales of city 27

Figure 18: City which sold the highest product 28

Figure 19: Bar graph City which sold the highest Product 29

Figure 20: Calculating the total quantity of each product 31

Figure 21: Creating a list Product and short in ascending order 32

# Introduction

## 1.1 Smart Data

First of all, Smart data is digital information that is formatted so it can be acted upon at the collection point before being sent to a downstream analytics platform for further data consolidation and analytics. The term smart data is often associated with the Internet of Things (IoT) and the data that smart sensors embedded in physical objects produce.

The label smart is directly related to a data entry point being intelligent enough to make some types of decisions on incoming data immediately, without requiring processing power from a centralized system. Basically, Smart data refers to smaller sets of valuable and actionable information. So, if big data is about volume, speed, and variety then smart data is more focused on creating value, meaning, and accuracy (veracity) for some sort of purpose or outcome.

Whereas, smart data discovery essentially means exploring data in a less-than-structured way, to discover hidden patterns and trends for maximizing business impact. This is a relatively new BI trend that empowers users to perform self-service analysis- data preparation, native language queries and automatic creation of visualizers. The term ‘smart’ comes into play because this functionality is provided based on artificial intelligence (example: IBM Watson). Smart data discovery is an encapsulation of predictive analytics, interactive data visualization, pattern matching and machine learning to produce automated decision support which comes in many avatars, namely search-based, visual-based and graph-based

Data discovery involves collecting and evaluating data from multiple sources to uncover hidden trends and patterns. The data discovery process allows business leaders to take a step back from individual data points and unlock broader trends from the combination of internal and external third-party data sources. Leaders and key stakeholders can make better, more strategic business decisions when they can see the big picture. In addition, since data discovery almost always involves data preparation and cleaning, this process also ensures that bad data doesn’t infiltrate the analysis and distort conclusions.

Data discovery has several benefits for businesses. The most important benefits to cover include the following:

* Complete overview of company data:

Data discovery gives organizations a macro-level view of all data streams. Data discovery ensures that business leaders have a comprehensive overview and understanding of their operational performance.

* Greater insight:

Data discovery provides a complete picture of a company’s data streams. However, when we discuss greater insight, we also refer to the fact that data discovery enables more people in your organization to understand data analysis regardless of their technical data literacy.

* Improved risk management:

Risk management has become a critical focus for organizations as the true costs of data breaches and attacks become more obvious. Data discovery can help companies identify outliers and potential threats in their collected data. As a result, businesses can proactively manage potential threats and prevent costly data breaches.

* Automatic classification:

Organizations are collecting data from more sources than ever before. The data discovery process helps businesses automatically classify the data they collect. This enables organizations to look closely at specific data points and streams. (Kazmi, 2022)

The Smart Data and Business Discovery module is designed to help students understand the role of data in business decision-making and the methods used to extract insights from data. The module covers a range of topics related to data analytics and business intelligence, with a focus on practical applications and real-world case studies.

Some of the key focus points of the Smart Data and Business Discovery module include:

1. Data management: This focus point covers the basics of data management, including data storage, retrieval, and organization. Students will learn how to collect and clean data, as well as how to store and manage it in a way that makes it easily accessible for analysis.
2. Data analysis: This focus point covers the methods and tools used to analyze data, including statistical techniques, data visualization, and machine learning algorithms. Students will learn how to use these tools to identify patterns and trends in data, as well as how to create visualizations and dashboards to communicate their findings.
3. Business intelligence: This focus point covers the use of data to support business decision-making. Students will learn how to use data to identify opportunities and risks, as well as how to create reports and dashboards that provide insights into business performance.
4. Data ethics: This focus point covers the ethical considerations involved in working with data, including issues related to privacy, security, and bias. Students will learn how to identify and address ethical concerns related to data collection, analysis, and use.

## 1.2 Business Data and Data processing

Business data is the totality of information pertaining to a business and its activities. Any statistical data, unprocessed analytical data, customer feedback data, sales figures, and other sets of information can be included in this. Businesses frequently gather as much information as they can about their operations in order to use that information to streamline operations and understand client wants more fully so they may better serve their audience. Customers can be surveyed, analytical software can be used, or information can simply be observed while gathering company data. Business data analysts use the results of data analysis to further the objectives of their organization. Business data analysts use the data analysis process to recognize, decipher, and forecast business patterns. They then apply these data-driven insights to improve their company's operations (Team, 2023).

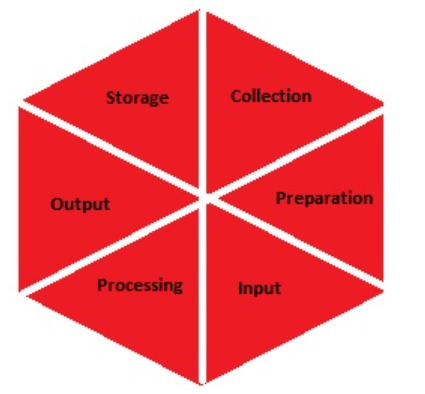
Data in its raw form is not useful to any organization. Data processing is the method of collecting raw data and translating it into usable information. Data processing means to process the data i.e. to convert its format. As we all know data is the very useful and when it is well presented and it becomes informative + useful. Data processing system is also referred as information system. It is also right to say that data processing becomes the process of converting information into data and also vice-versa. Data is generated every single second, whether you use the internet to order food, make financial transactions, or learn about a particular subject. Social media use, internet shopping, and the use of video streaming services have all contributed to the rise in data. According to a Domo study, by the year 2020, every person on the earth will produce 1.7MB of data every second. And data processing is necessary in order to make use of and gain insights from such a vast volume of data.

. In simple words, data processing can be expressed as

* Process of conversion of data in the computer understandable format.
* The sorting or processing of data by a computer (itskawal2000, 2020).

## 1.3 Data Processing Cycle

For businesses to improve their business strategy and gain a competitive edge, data processing is crucial. Employees across the organization can understand and use the data by turning it into readable representations like graphs, charts, and texts. Raw data is fed into a system as input during the data processing cycle, which results in actionable insights as output. The process is carried out in a predetermined order,however it is constantly cycled back on itself. According to the image below, the output of the initial data processing cycle can be saved and used as the initial input for the subsequent cycle (Duggal, 2023).



1. Gathering:

The initial stage of the data processing cycle is the gathering of raw data. The type of raw data gathered has a significant influence on the results generated. Therefore, in order for the subsequent findings to be reliable and applicable, raw data should be gathered from well-defined and precise sources. Financial data, cookies from websites, firm profit/loss accounts, user behavior, etc. are all examples of raw data (Duggal, 2023).

1. Preparation of Data:

Data preparation is an essential step in the data processing cycle that involves collecting, cleaning, and labeling raw data to make it suitable for further processing and analysis. The process of collecting data involves gathering information from various sources, including databases, sensors, and other devices. To prepare raw data for further analysis and processing, it is verified for errors, duplication, errors in computations, and missing data. This is done to make sure that the processing unit only receives the best data. In order to start building high-quality information that can be used in the best possible way for business intelligence, this stage aims to remove bad data (redundant, incomplete, or erroneous data) (Duggal, 2023).

1. Data Input:

The step being referred to here is data input, which is the process of converting raw data into a machine-readable format and feeding it into a processing unit. This can be achieved using various input sources, such as keyboards, scanners, or other devices. The primary goal of this step is to ensure that the data is accurate, complete, and in a format that can be processed by the computer system. Data input is a critical step in the data processing cycle, as the accuracy and completeness of the input data directly impact the quality of the output produced by the system. Therefore, it is essential to ensure that the data input process is reliable and efficient to produce high-quality output (Duggal, 2023).

1. Processing of Data:

In this step, machine learning and artificial intelligence algorithms are used to process the raw data in a variety of ways to produce a desired result. Depending on the data source being processed (data lakes, online databases, linked devices, etc.) and the intended use of the output, this stage may vary slightly from process to process (Duggal, 2023).

1. Output:

Finally, the user receives the data and it is presented to them in a readable format, such as graphs, tables, vector files, audio, video, papers, etc. The following cycle of data processing can store and further process this output (Duggal, 2023).

1. Storage:

The storage step of the data processing cycle involves the permanent storage of both the processed data and its associated metadata for future use. This step enables easy access and retrieval of information whenever required, and the stored data can also be used as input for the next data processing cycle directly. Proper storage of data is crucial for the long-term preservation and retrieval of information (Duggal, 2023).

## 1.4 Conclusion

Data is a crucial asset for organizations in the modern digital age since it offers insights into consumer behavior, market trends, and operational efficiency. Businesses may enhance operations, make wise decisions, and gain a competitive edge by gathering and analyzing data. The data processing cycle is essential for assisting firms in gleaning insights and useful data from raw data.

Making wise decisions is one of the main advantages of using data in company. Businesses can use data to study client behavior, spot patterns and trends, and improve operations. Businesses can use data to make decisions that are more accurate and dependable by basing them on facts and evidence rather than feelings or assumptions. The influence of data on innovation in company is another advantage. Businesses can find market gaps, fresh client requirements, and developing trends by evaluating data. New products and services that better fulfill client requirements and expectations can be created using this knowledge.

Data may have a big impact on business, and data-driven decisions can raise revenue, boost customer happiness, and improve operational efficiency. To keep customers' confidence and loyalty, businesses must make sure that data is gathered, handled, and used ethically and in accordance with privacy laws. In conclusion, firms that embrace data-driven decision-making and innovation are likely to prosper and stay one step ahead of the competition. The use of data in business is essential to success in the current digital era.

Overall the data processing cycle can also be utilized for research, such as spotting patterns and trends in the social sciences or in academic studies. In conclusion, companies may harness the power of data and use it to enhance decision-making, operations, and overall performance by utilizing the data processing cycle.

# Objectives

1. Understand the data resources and the characteristics of those resources.
2. To hone critical thinking and problem-solving abilities based on programming knowledge.
3. Calculate summary statistics, correlation, and the top performing month and city, as well as the most popular product, to analyze the data.
4. Explore the data by creating a histogram plot and visualizing the sales and product performance through bar graphs.
5. To work for Data analysis and analysing for the business prospective
6. Organize the technical report by following a proper structure.

# Overview of Coursework

The coursework is an individual assessment that carries a weightage of 60% for the module. It involves using programming knowledge and skills to perform data analysis tasks and showcase problem-solving and critical thinking abilities we must use their programming knowledge and skills to complete data analysis activities as part of this project, as well as show their aptitude for problem-solving and critical thought. The current work requires analysing sales data from the ABC Company for the year 2019 using the Python programming language. We must complete a variety of tasks, including data interpretation, data preparation, data exploration, and preliminary analysis, in order to get the data ready for future study. These tasks are crucial for ensuring that the data is well-structured, ready for analysis, and well-prepared. Understanding data requires recognizing its nature and features, such as its sources, type, format, and quality. we must comprehend the many characteristics that make up the data set at this step, including Order ID, Product, Quantity Ordered, and other pertinent details. Data cleansing, data removal, handling of missing data, and data transformation into an analytically-ready format are all parts of data preparation. we must use their programming knowledge at this level to alter the data and prepare it for future analysis. Finding patterns, trends, and relationships in the data is known as data exploration. The data will be analyzed by the students using statistical methods such descriptive statistics, correlation analysis, and regression analysis.

The final step of initial analysis is to make findings and suggestions based on the information gleaned from data exploration. we must apply critical thinking and problem-solving techniques at this stage in order to understand the data and give stakeholders insightful explanations. The assignment involves an organized procedure that includes data understanding, preparation, exploration, and preliminary analysis. The final goal is to prepare the data for future analysis utilizing data mining techniques

# Task Development

## 4.1 Data Understanding

Data understanding is the first step in the data processing cycle, which involves getting a deep understanding of the data resources and their characteristics. In this coursework, Upon examining the given CSV files, it is observed that the data pertains to the sales analysis of ABC company of the year 2019. The data is divided into separate CSV files, each representing the sales information for a particular month of the year. Each CSV file contains various columns providing information such as Order ID, Product, Quantity Ordered, Price Each, Purchase Address, and Order Date, among others.

* The Order ID column consists of alphanumeric values that uniquely identify each order placed by customers.
* The Product column contains the names of the products that were ordered by the customers. This column can be used to analyze the popularity of different products.
* The Quantity Ordered column contains the number of items of each product that were ordered by the customers. This column can be used to analyze the demand for different products.
* The Price Each column contains the price of each item ordered by the customers. This column can be used to calculate the total revenue generated by each product.
* The Order Date column contains the date and time when each order was placed. This column can be used to analyze sales trends over time.
* The Purchase Address column contains the address where each order was delivered. This column can be used to analyze sales trends across different locations.

Further investigation of the data reveals that the dataset has some missing values or NaNs. In addition, the object data type of the columns Quantity Ordered and Price Each makes them unsuitable for data analysis. In order to allow time-series analysis, the Order Date column, which is now in string format, needs to be transformed to a date format. Moreover, the city data for each purchase can be retrieved using the Purchase Address column. Overall, some preparation and cleaning of the provided data is necessary before performing any analysis.

## Data preparation

Here are the Python Program to prepare the data:

* Merging data from each month into one CSV and read in updated dataframe.

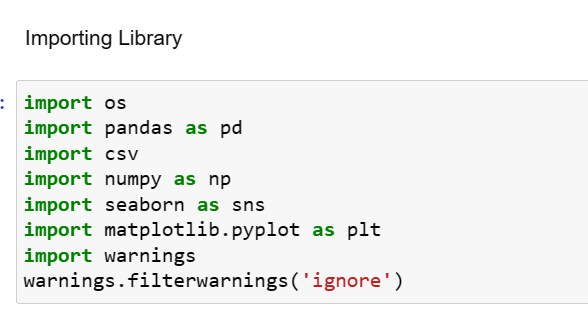


Figure 1: Importing Library

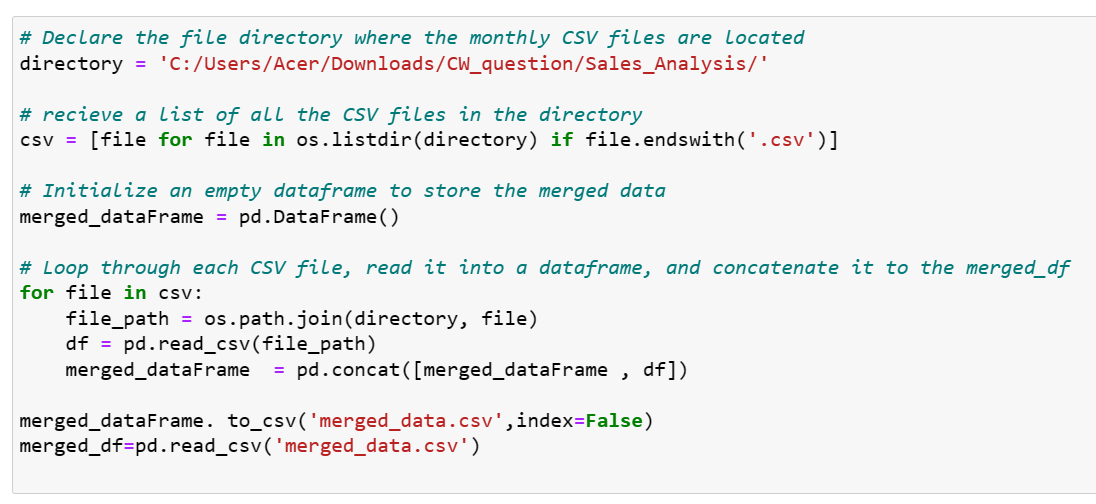


Figure 2: Merging multiple CSV files into one and removing unused rows

This code snippet creates a single dataframe from a lot monthly CSV files in the directory "C:/Users/Acer/Downloads/CW\_question/Sales\_Analysis/" and saves it as a CSV file with the name "merged\_data.csv" in the same location. The CSV files in the provided directory are first listed using the OS library, and they are then saved in the variable csv. To store the combined data, it next creates an empty Pandas dataframe called merged\_dataFrame. The code then iterates through every CSV file in the csv directory, reading each one into a Pandas dataframe with the read\_csv function before concatenating them into the merged\_dataFrame dataframe with the concat function. The merged dataframe is saved to a new CSV file called "merged\_data.csv" using the to\_csv function and the index=False option to prevent the index column from being included in the CSV file after all the CSV files have been combined. A new Pandas dataframe called merged\_df is created by reading the stored "merged\_data.csv" file into the code.

The result has practical importance in that it can give a thorough overview of sales trends over time and help firms see trends, make data-driven decisions, and enhance their sales tactics. This method can also be automated and used to update the dataset on a regular basis when new data become available. To create a more thorough framework for sales analysis, this approach can be expanded in the future to include new data sources including consumer demographics, marketing initiatives, and economic factors. Overall, several business stakeholders, such as sales teams, marketing analysts, and executive management, can benefit from the dataset that was produced.

* Removing the NaN missing values from updated dataframe.

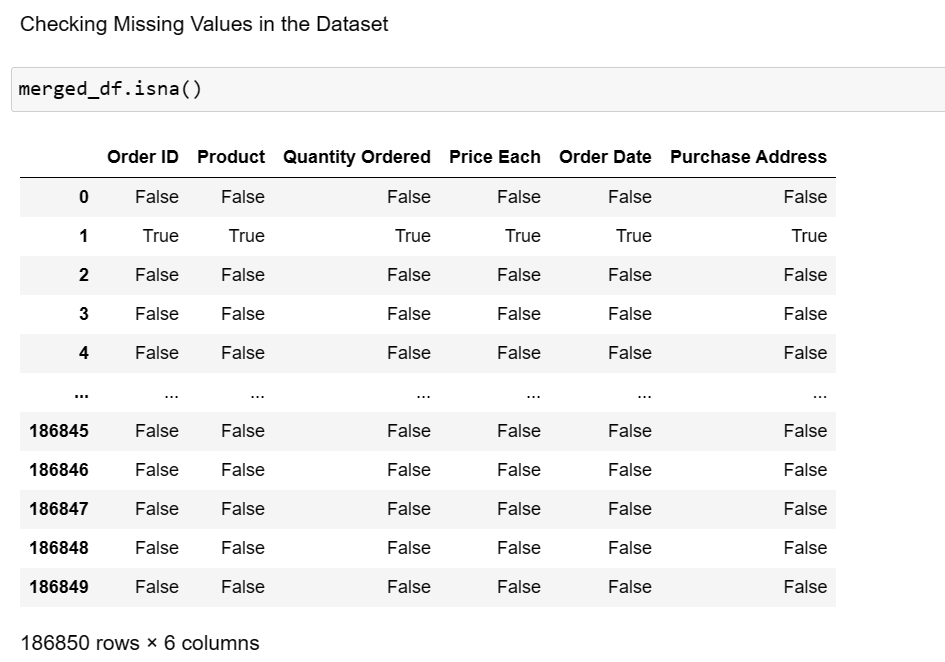


Figure 3: Checking missing value in data set

The merged\_df.isna() method provides a DataFrame with boolean values indicating if each element in the original DataFrame is missing or not, but with the same shape as merged\_df. Non-missing values are presented as False, whereas missing values are portrayed as True. The locations of missing values in the dataset can be found using this method. Using this knowledge, we can choose the best approaches to deal with missing data, such as imputation or removal of missing values. In conclusion, the merged\_df.isna() function can be used to find missing data in a pandas DataFrame.

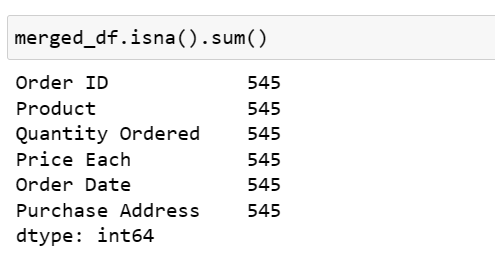


Figure 4: Calculating missing value in data set

merged\_df.isna() is the code. The DataFrame'merged\_df's' sum() function reports the number of missing values for each column. More specifically, it uses the DataFrame and the isna() method to generate a boolean mask with True denoting a missing item and False denoting a value that is present. The sum() method is then used on this boolean mask to get the total number of missing values by counting the True values in each column.

In order to verify the integrity of the data analysis, this information is useful for identifying columns with a significant number of missing values and determining whether imputation or removal of missing values is required.

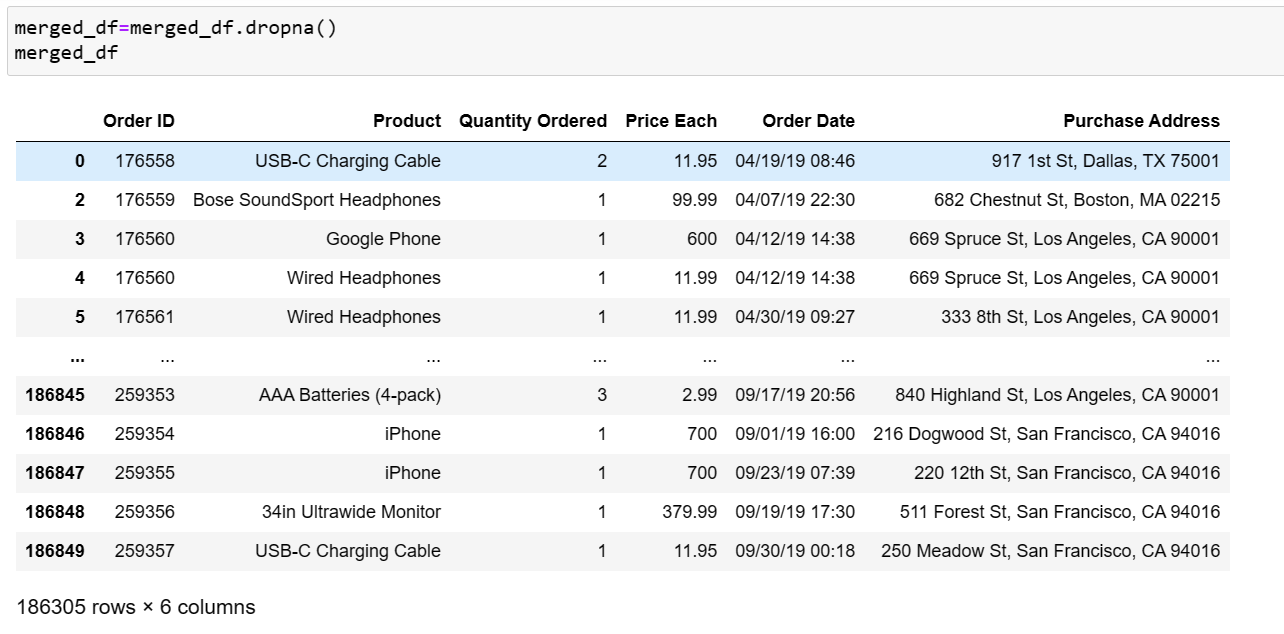


Figure 5: filtering out NaN values in a dataset

A new DataFrame with the remaining rows is returned by the function merged\_df.dropna(), which removes any rows from the DataFrame that have missing values. For the purposes of data analysis and modeling, the resulting DataFrame merged\_df wouldn't have any missing values. It is crucial to keep in mind that eliminating missing values may reduce the sample size and perhaps result in the loss of vital data, especially if the missing values are not absent at random.

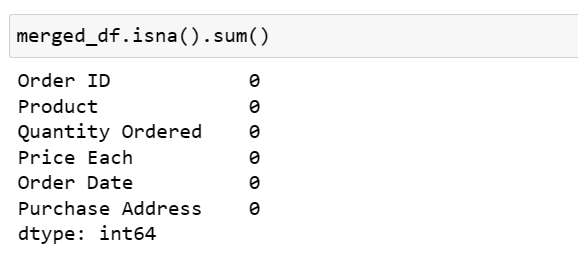


Figure 6: Calculating missing value in dataset after filtering NaN values

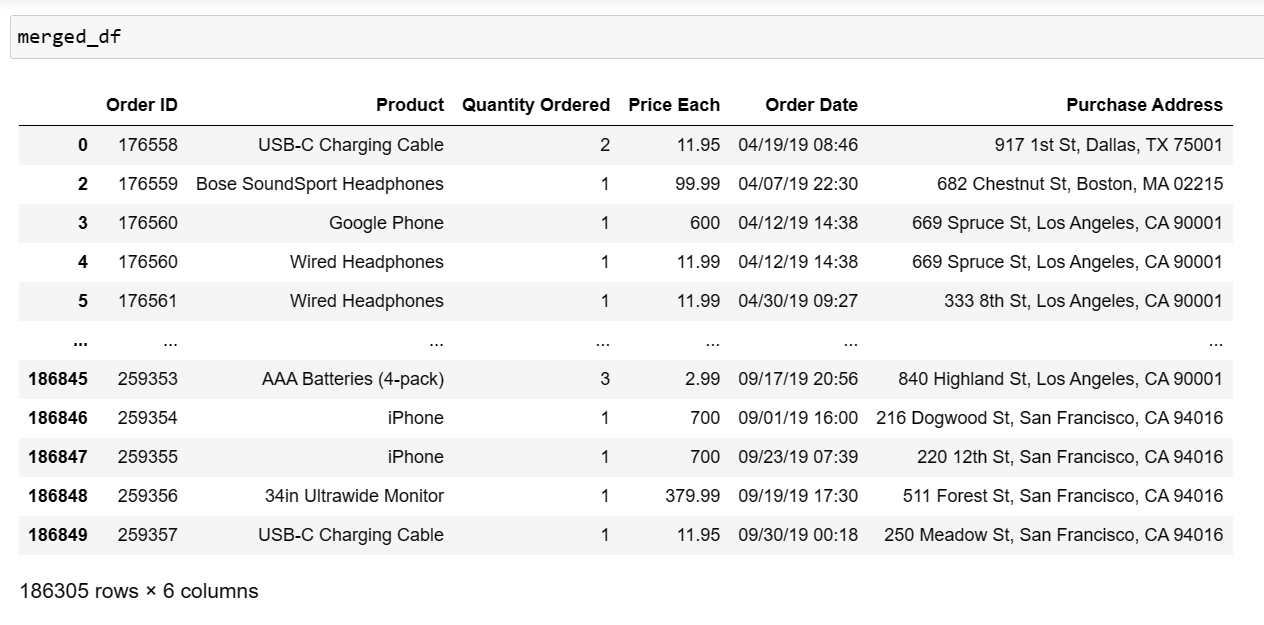


Figure 7: Output after filtering NaN values

This line first determines whether any missing values (NaN) exist in each column of the DataFrame using the 'isna()' method, and then uses the'sum()' method to tally the total number of missing values in each column. The outcome is a Pandas Series that displays the quantity of missing data for each DataFrame column.

Any analysis done on the dataset can be significantly impacted by the presence of missing values in the dataset. Therefore, depending on the unique context of the data, it is crucial to carefully examine and handle missing values. The significance of this discovery in real-world scenarios is that it aids in determining the level of data cleaning needed to increase the precision of any insights drawn from the dataset. Correct management of missing values can also aid in minimizing influence in the analysis.

In the future, this strategy can be expanded by applying different data imputation techniques to fill in the dataset's missing values. This can assist in maintaining the dataset's completeness and enhancing the precision of any analyses conducted using it.

* Converting **Quantity Ordered** and **Price Each** to numeric.

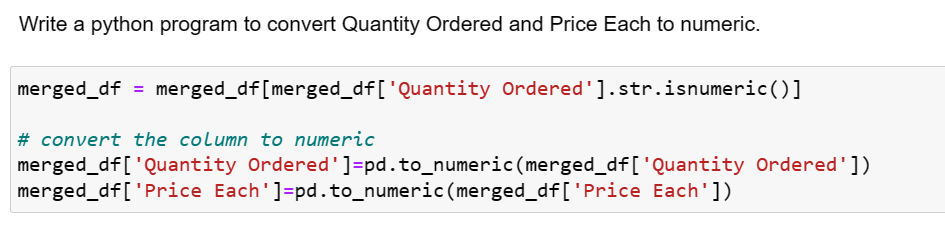


Figure 8: Convert Quantity Ordered and Price each in numeric

'Quantity Ordered' column values that are not numeric are excluded from the merged\_df in the first line of the code. This is crucial to eliminate any inconsistent or inaccurate data that might influence the analysis that follows. The next two lines use the pandas function pd.to\_numeric() to convert the object/string data types of the columns "Quantity Ordered" and "Price Each" to numeric data types. This is required so that the ensuing analysis can use these columns for arithmetic calculations.

In general, these changes to the dataset aid in making sure the information is accurate, consistent, and prepared for analysis. The outcome is regarded favorably because it raises the analysis's precision and dependability. In real-world scenarios, this outcome can assist firms in better understanding their sales patterns and trends, pinpointing the most lucrative goods and clientele, and making data-driven choices to boost sales performance. The use of additional data cleaning and preprocessing methods, such as dealing with missing data, addressing outliers, and scaling the data to make it uniform and comparable, can be added to this strategy in the future. Overall, the cleaned-up dataset produced might be a useful input for many machine learning and statistical modeling techniques.

* Creating a new column named **Month** from **Ordered Date** of updated dataframe and convert it to integer as data type.

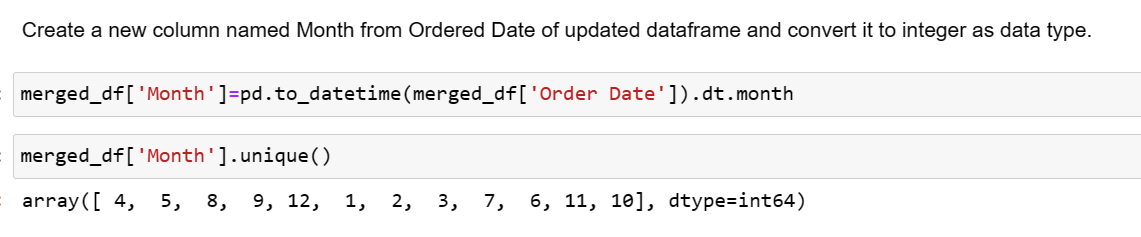


Figure 9: creates a new column called Month in the DataSet

By utilizing the pandas to\_datetime() and dt.month functions to extract the month value from the 'Order Date' column, the aforementioned code adds a new column called 'Month' to the combined DataFrame'merged\_df'. The distinct months that are present in the 'Month' column are then displayed using the unique() function. This data update is significant since it enables improved monthly analysis and visualization of sales data. The data is turned into a more relevant format that may be easily used for various analytical reasons by changing the "Order Date" column into a datetime format and extracting the month value.

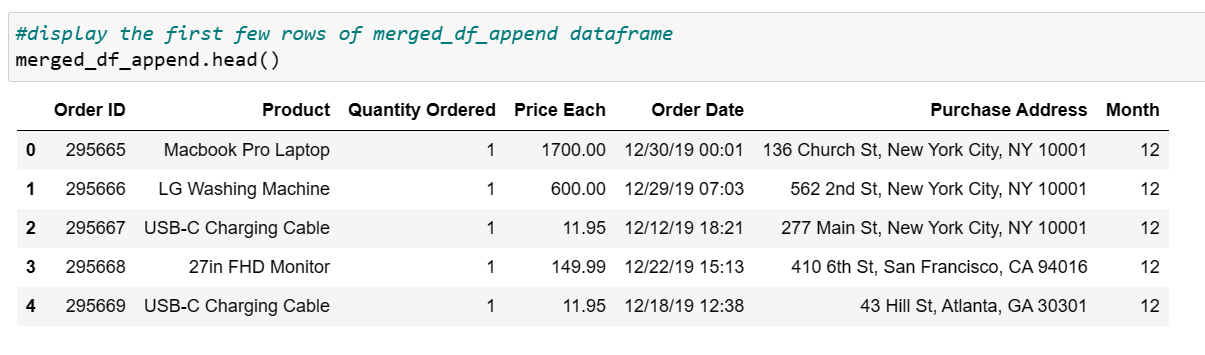


Figure 10: Returning the first five rows

The first few rows of the combined DataFrame'merged\_df' would be shown by using the function'merged\_df.head()'. A DataFrame's top rows can be examined using the 'head()' function, a pandas method, to ensure that the data has been properly merged and transformed.

In real-world scenarios, this data alteration can help firms pinpoint their busiest seasons, assess the effectiveness of their marketing initiatives, and allocate their resources wisely. To acquire a deeper understanding of sales patterns and trends in the future, this approach can be expanded to include additional time-related data like day of the week, hour of the day, and seasonality. Overall, the 'Month' column is a useful addition to the examination of sales data and can help firms make decisions more effectively.

* Creating a new column named **City** from **Purchase Address** based on the value in updated dataframe.

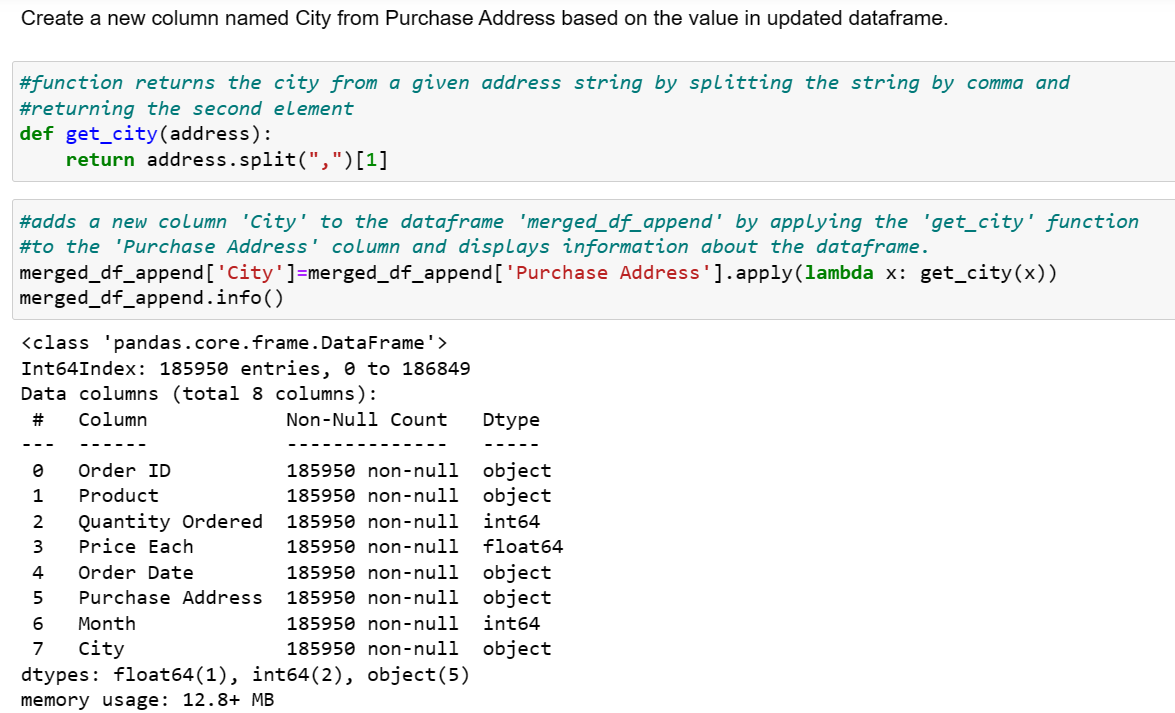


Figure 11: Creating new column City from Purchase Address

The code above defines a function named "get\_city" that accepts a string parameter called "address" and returns the second member of the list that is produced when the string is split using a comma. In the'merged\_df\_append' DataFrame, the 'Purchase Address' column contains the city name, which is extracted using this function. The 'get\_city' function is then applied to each item in the DataFrame's 'Purchase Address' column using the apply() method, and the resulting city name is stored in a new 'City' column. The changed DataFrame's summary information, including the number of rows, columns, and data types, is then shown using the info() method.

This result has practical significance since it can shed light on customer behavior and sales geography, which can help businesses make decisions about where to locate stores, what products to offer, and how to advertise. This method can be expanded in the future to incorporate more geographic data like zip codes, states, or nations, which can give more in-depth insights into local sales patterns. This strategy can also be used in conjunction with other data sources, such as demographics, meteorological information, or traffic patterns, to develop a more thorough picture of sales trends and consumer behavior. Overall, the produced dataset might be a useful tool for firms to enhance their bottom line and optimize their sales strategy.

## Data Analysis

* Write a Python program to show summary statistics of sum, mean, standard deviation, skewness, and kurtosis of any chosen variable.

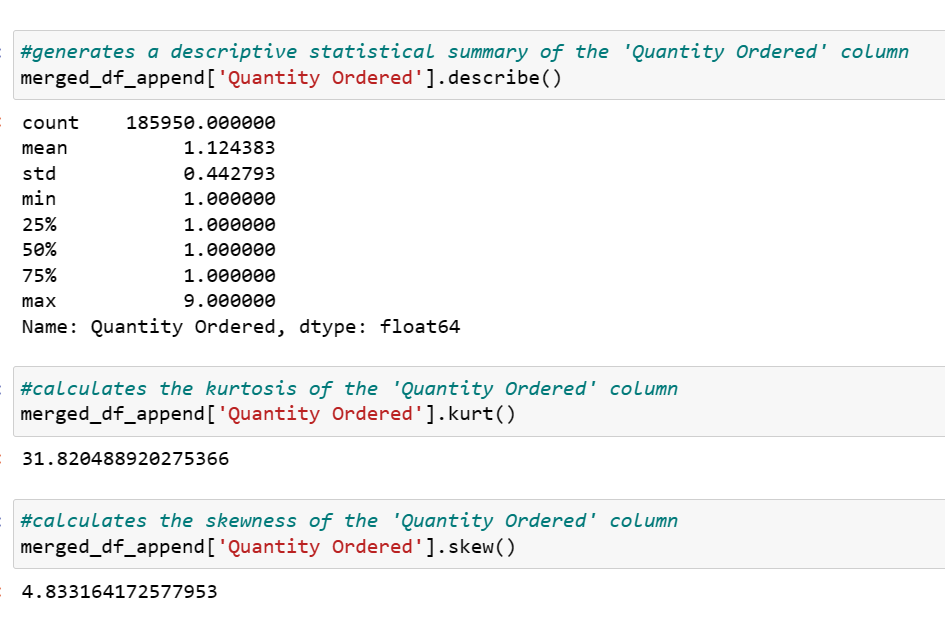


Figure 12: To show summary statistics of sum, mean, standard deviation, skewness, and kurtosis of Quantity Ordered

The distribution of the 'Quantity Ordered' values is thoroughly summarized by the describe() method, which includes the count, mean, standard deviation, minimum, maximum, and quartiles. This might help you grasp the values' range and central tendency as well as spot any outliers or odd trends in the data.

The degree to which the distribution is peaked or flat in comparison to a normal distribution is measured by the kurtosis of the 'Quantity Ordered' values, which is calculated using the kurt() methodThis can help uncover any non-normality in the data so that statistical analysis can be adjusted as necessary.

The skew() method determines the 'Quantity Ordered' values' skewness, which gauges the distribution's asymmetry. A right-skewed distribution with a longer tail on the right side has a skewness value of positive, whereas a left-skewed distribution with a longer tail on the left side has a skewness value of negative.

* Write a Python program to calculate and show correlation of all variables.

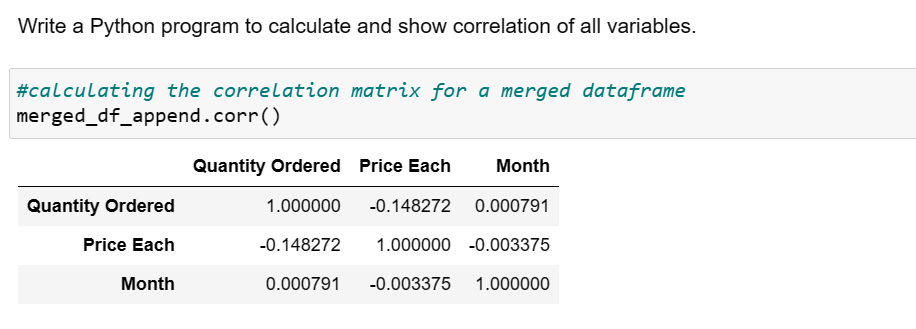


Figure 13: Calculating the correlation matrix

The'merged\_df\_append' DataFrame's corr() method is used in the code above to compute the correlation matrix for each pair of numerical columns. The resulting matrix, which can be used to investigate the connections between variables and spot any noteworthy correlations, shows the pairwise correlation coefficients between all columns.

The correlation coefficient ranges from -1 to 1, with values of 1 denoting a perfect positive correlation (i.e., when one variable increases, the other variable also increases), -1 denoting a perfect negative correlation (i.e., when one variable increases, the other variable decreases), and 0 denoting no correlation (i.e., there is no linear relationship between the variables).

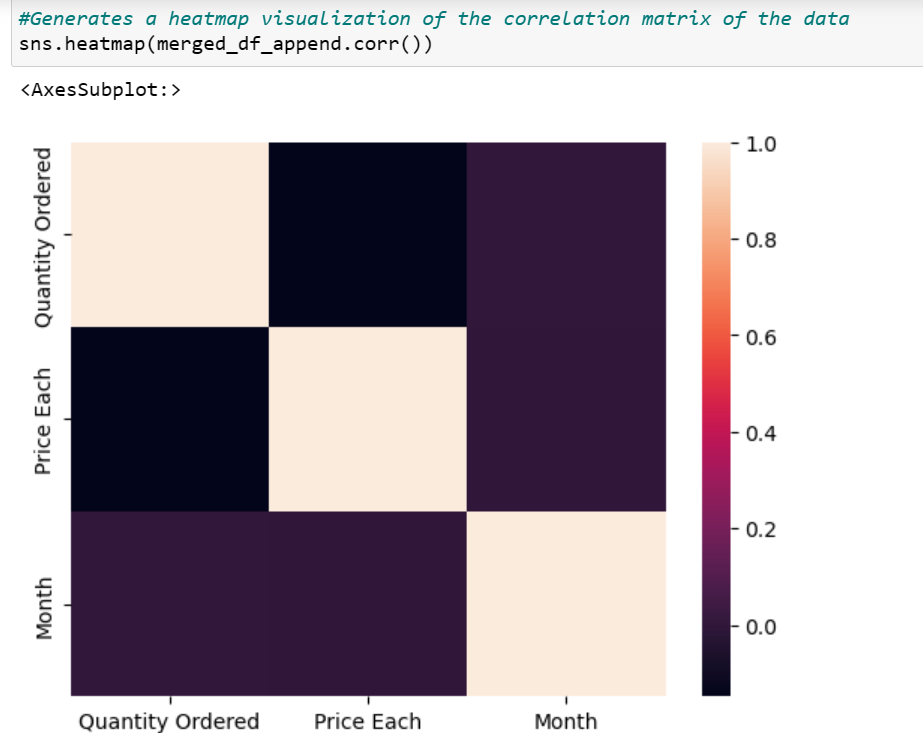


Figure 14: To calculating and show correlation of all variables

Using the Seaborn library, the aforementioned code generates a heatmap that displays the relationships between the various columns in the combined DataFrame. A negative correlation is shown by a correlation coefficient of -1, a zero indicates no association, and a positive correlation is indicated by a correlation coefficient of 1. The resulting heatmap gives the correlation matrix a visual representation, and it can be used to spot trends and connections between various variables. For instance, to reduce multicollinearity and enhance model performance, strongly correlated variables might be merged or eliminated from the dataset.

This discovery is significant because it can enable firms to pinpoint the variables that influence sales performance and create plans to optimize them. For instance, a company may decide to invest more resources in advertising if there is a positive association between advertising spending and sales. The resulting correlation matrix can then be fed into regression, classification, and clustering machine learning models in the future to forecast sales, identify consumer segments, and improve marketing efforts.

## Data Exploration

* Which Month has the best sales? and how much was the earning in that month? Make a bar graph of sales as well.

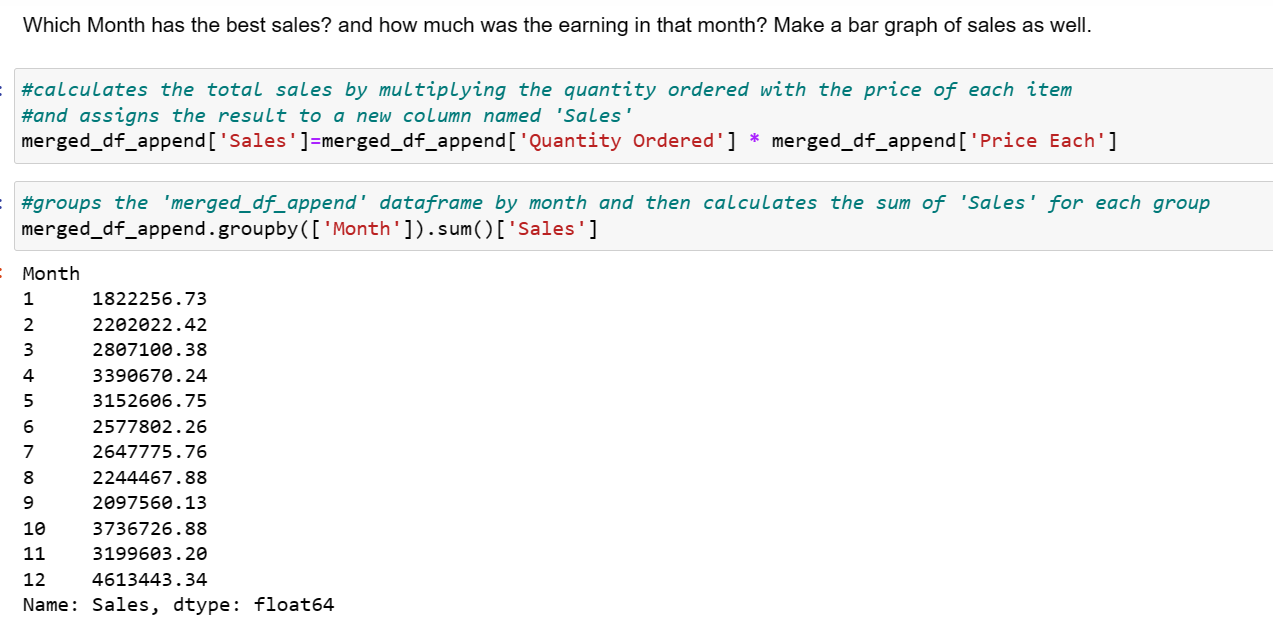
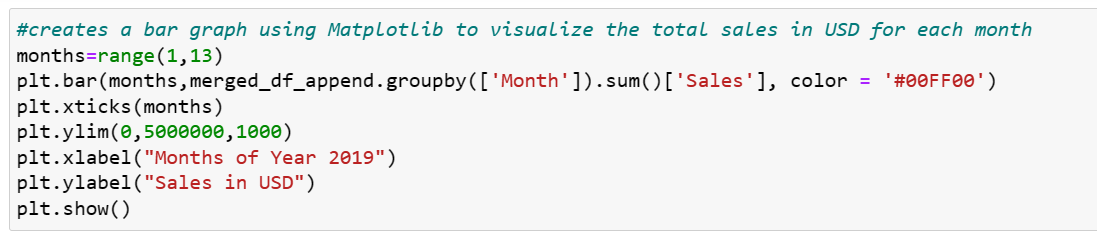


Figure 15: Best Selling and Earning Month

The code above multiplies the 'Quantity Ordered' and 'Price Each' columns of the combined DataFrame to determine the total sales for each month, then groups the results by month using the 'groupby' technique.

The output that is produced gives a summary of monthly sales trends and can be used to spot seasonal trends, evaluate performance over various time frames, and guide forecasting and budgeting decisions.



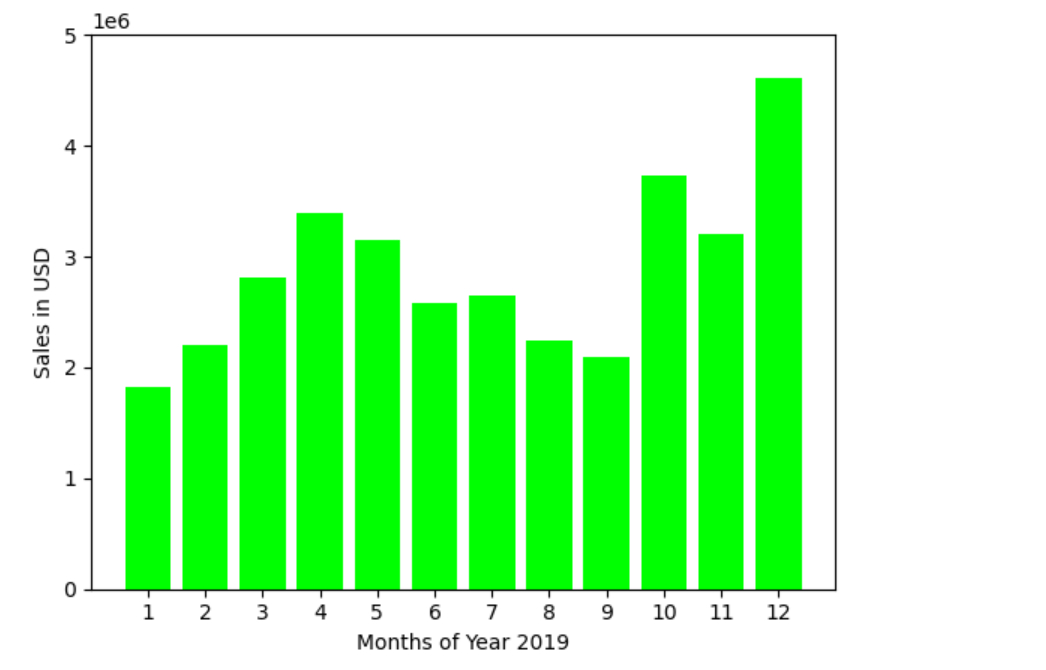


Figure 16: Bar graph of Sales

The aforementioned code creates a bar chart that displays the monthly sales for 2019 in a table format. The sales numbers are represented on the y-axis, while the months of the year are plotted on the x-axis. The ensuing graph displays the monthly sales amounts in US dollars. Finding trends and patterns in sales data over time is made easier with the help of this visualization. According to the graph, sales are greater in November and December, which is in line with the US holiday shopping season. On the other side, during the summer months of June, July, and August, sales are lower.

In a real-world scenario, this finding might be used to guide commercial choices like inventory control and marketing tactics. Businesses should, for instance, modify their marketing strategies to take advantage of the months with strong sales and make sure they have enough inventory to fulfill the increased demand during such months. In the future, this strategy can be expanded to incorporate more elements, like product categories or demographics of the client, to better analyze sales trends and pinpoint opportunities for development. Overall, the visualization produced can help firms make data-driven decisions to improve their sales performance by offering insightful information about sales patterns.

* Which city has sold the highest product?

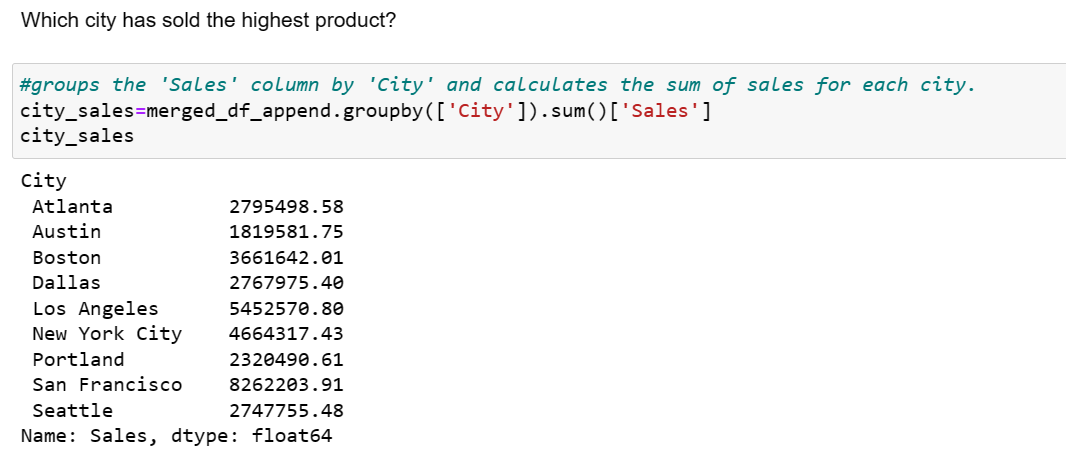


Figure 17: Calculating the sum of sales of city

In the code above, the total sales for each city in the merged\_df\_append DataFrame are calculated, and the result is saved in a pandas series called "city\_sales." We may use the describe() method, which provides a summary of the data, to explain the "city\_sales" series.

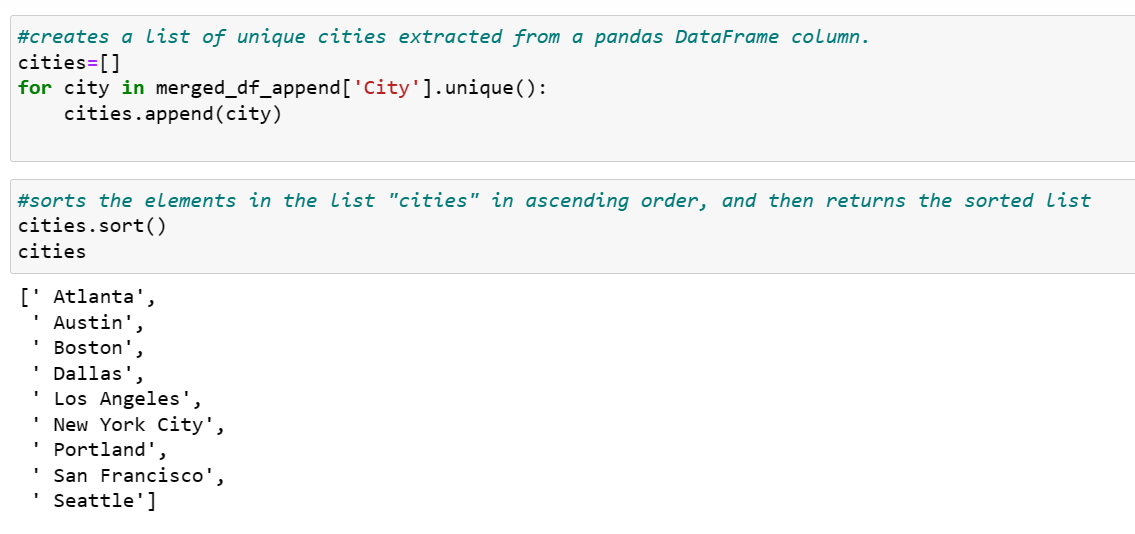


Figure 18: City which sold the highest product

By iterating through the distinct values in the 'City' column of the merged\_df\_append DataFrame, the code above constructs an empty list called "cities" and adds each distinct city name to the list. The sort() method is then used to arrange the list in alphabetical order.

This code builds a list of distinct city names in the DataFrame and arranges them alphabetically, to put it simply. This list can be used to filter the data by city, examine sales statistics for particular cities, or create infographics for particular cities.

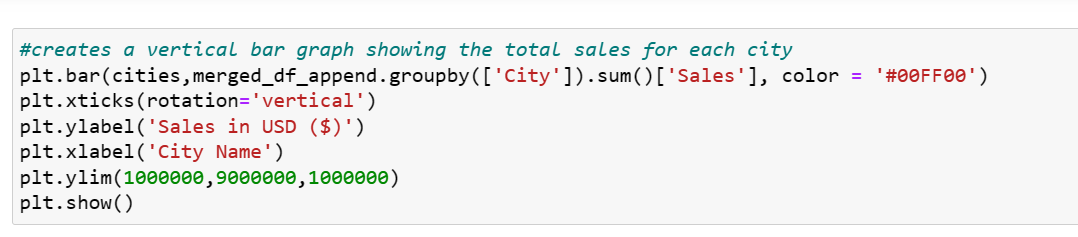
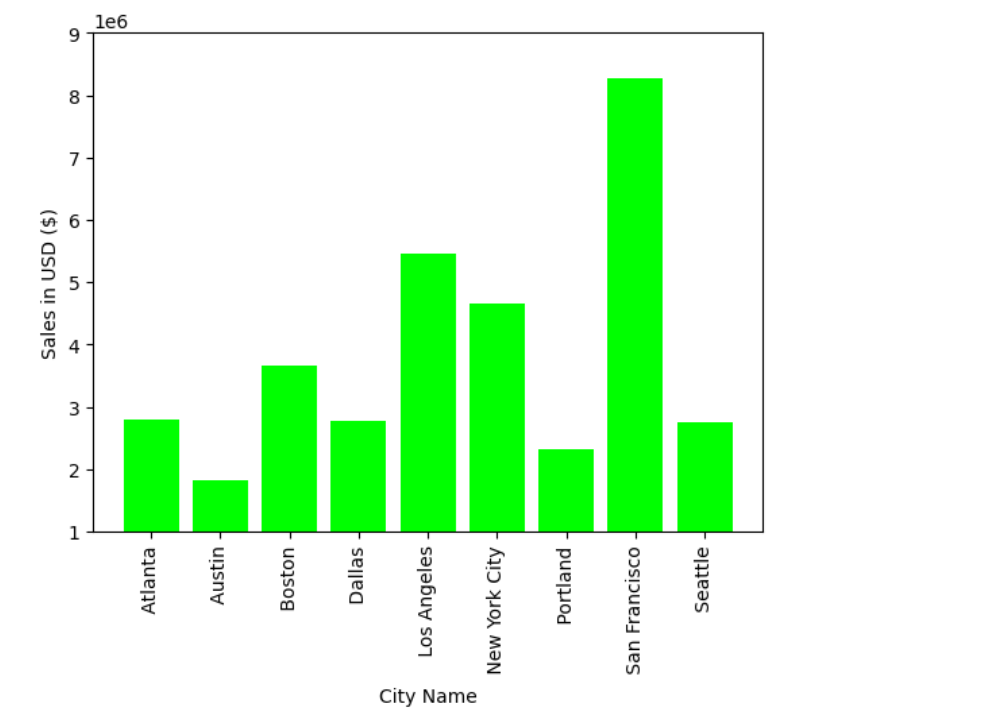
 

Figure 19: Bar graph City which sold the highest Product

The code shown above creates a bar graph that displays the total sales for each city in USD. The city names are represented on the x-axis, and the USD sales are shown on the y-axis. The merged\_df\_append DataFrame, which has been grouped by city and aggregated by adding the sales column, provided the data for this graph. The y-axis is constrained to the range between 1,000,000 and 9,000,000 USD in steps of 1,000,000, and the chart has been vertically rotated for easier reading.

By allocating resources to high-performing cities and highlighting areas for improvement in underperforming cities, this outcome can help organizations maximize their sales efforts. The outcome can also be used to determine regional sales patterns, preferences, and trends, which can guide the creation of new products and sales tactics in the future. In the future, this strategy can be expanded by incorporating further variables like demographics, economic statistics, or consumer feedback to better understand the elements influencing sales in various cities. In general, organizations can use the bar chart as a useful tool to assess and improve their sales strategy.

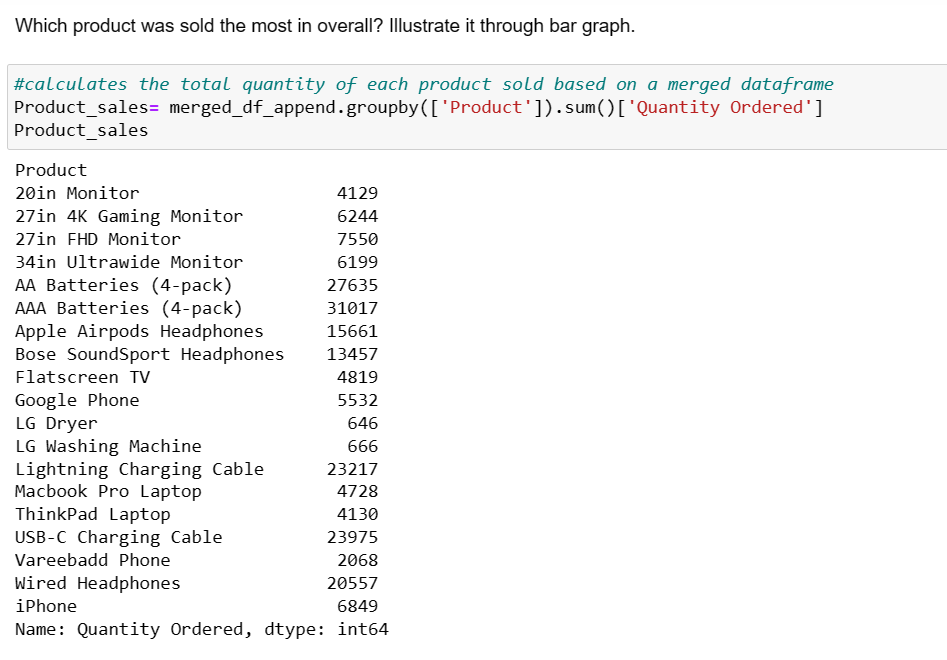
* Which product was sold the most in overall? Illustrate it through bar graph. 

Figure 20: Calculating the total quantity of each product

The merged\_df\_append DataFrame is grouped by product in the code above, and the amount ordered column is averaged to determine the overall quantity of each product sold. Product\_sales, the resulting variable, is a Pandas Series object that holds the total number of sales for each product in the dataset. You can get this Series object's summary statistics, including count, mean, standard deviation, minimum and maximum values, using the describe() method.

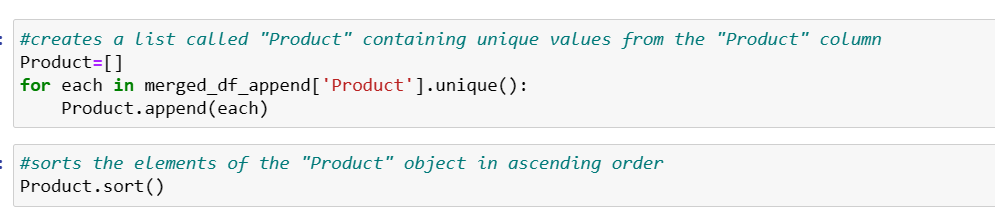


Figure 21: Creating a list Product and short in ascending order

In the code above, the "Product" column's unique values are looped through after which an empty list called "Product" is created. It adds a unique product value to the Product list for each one that exists. Finally, the sort() method is used to order the Product list in ascending order.

The Product list can be sorted to improve the legibility of the data and facilitate effective data processing, among other things. Sorting the list can make sure that the items are displayed or handled in a logical order, for instance, when plots are created or computations are based on the Product list.

* Write a Python program to show histogram plot of any chosen variables. Use proper labels in the graph.

# Conclusion

In conclusion, completing this coursework has been a valuable learning experience that has provided me with practical experience in applying programming knowledge and skills to real-world data analysis problems. By following the given requirements, I was able to demonstrate my abilities in problem-solving, critical thinking, and data analysis.

My familiarity with the data's features was necessary as part of the data understanding process, which was an essential ability for any data analysis effort.

I was able to identify any potential problems, biases, or restrictions that might have had an impact on the analysis's findings by understanding the data. I was able to learn more about the data resources, comprehend its properties, and prepare the data for further mining and analysis throughout the coursework. I created Python scripts to combine the data from each month, get rid of missing values, change the data types of columns to numbers, and make new columns based on the old ones.

Additionally, I carried out a number of data analysis and exploration tasks, such as calculating summary statistics, determining correlations, and using bar graphs and histograms to visualize the sales, product, and city data. These assignments assisted me in developing a deeper grasp of the data and obtaining insightful information that can be used to business decisions.

In business perspective, data analysis is a crucial aspect of decision-making processes. It enables businesses to gain insights and make informed decisions based on the data. By completing this coursework, I was able to gain practical experience in applying programming knowledge and skills to solve real-world data analysis problems, which is a valuable skill in today's dita-driven world. During the data exploration phase, I was able to apply my data analysis skills to explore the data and answer business-related questions. By doing so, I was able to identify trends, patterns, and relationships in the data that provided valuable insights for the business.

Overall, this coursework has provided me with a great opportunity to showcase my strengths in data analysis, critical thinking, and problem-solving. It has given me useful knowledge and abilities that I may use in a variety of professional settings.