**Sales Data Analysis Report**

**Introduction**

This report details the steps taken to clean, analyze, and visualize sales data. The dataset contained sales transactions, including product details, purchase addresses, order dates, and revenue information. The goal was to process the raw data using Python (Pandas), analyze it using SQL Server, and create visualizations in Power BI for better insights.

**Data Overview**

**Dataset**

The dataset contains **371,900 rows and 6 columns**, representing transaction records from an online retail business. The data spans multiple products and customer locations, providing insights into sales trends and patterns.

**Columns in the dataset:**

* **Order ID:** Unique identifier for each sale.
* **Product:** The name of the product sold.
* **Quantity Ordered:** The number of units sold per transaction.
* **Price Each:** The price per unit of the product.
* **Order Date:** The date and time when the transaction occurred.
* **Purchase Address:** Customer purchase location details.

**Data type**

* order id – int
* product- object
* quantity ordered- int
* price each – int
* order date – date/time
* purchase address – object

**Sample Data**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Order ID** | **Product** | **Quantity Ordered** | **Price Each** | **Order Date** | **Purchase Address** |
| 176558 | USB-C Charging Cable | 2 | 11.95 | 19/04/2019 08:46 | 917 1st St, Dallas, TX 75001 |
| 176559 | Bose SoundSport Headphones | 1 | 99.99 | 07/04/2019 22:30 | 682 Chestnut St, Boston, MA 02215 |
| 176560 | Google Phone | 1 | 600 | 12/04/2019 14:38 | 669 Spruce St, Los Angeles, CA 90001 |

**Data Cleaning and Preprocessing (Pandas in Jupyter Notebook)**

To ensure data quality, the following preprocessing steps were performed using Pandas in Jupyter Notebook:

* **Loading the Data:** The dataset was imported using Pandas.
* import pandas as pd
* df = pd.read\_csv('sales\_data1.csv')
* **Handling Duplicates:** Duplicate records were identified and removed to prevent overestimation of sales.
* df = df.drop\_duplicates()
* **Handling Missing Values:** Missing values were either filled with appropriate values or dropped when necessary.
* df = df.dropna()
* **Updating Date/Time Format:** The order date column was converted to a standard datetime format.
* df['order\_date'] = pd.to\_datetime(df['order\_date'])

**Data Analysis Using SQL Server**

The cleaned dataset was loaded into SQL Server for deeper analysis. The following queries were used:

* **Total Sales Revenue:**

SELECT SUM(Revenue) AS TotalRevenue FROM salesdata1;

* **Sales by Product:**

SELECT Product, SUM(Revenue) AS TotalRevenue

FROM salesdata1

GROUP BY Product

ORDER BY TotalRevenue DESC;

* **Best-Selling Products:**

SELECT Product, SUM(Quantity\_Ordered) AS TotalQuantitySold

FROM salesdata1

GROUP BY Product

ORDER BY TotalQuantitySold DESC;

* **Monthly Sales Trend:**

SELECT FORMAT(Order\_Date, 'yyyy-MM') AS SalesMonth, SUM(Revenue) AS TotalRevenue

FROM salesdata1

GROUP BY FORMAT(Order\_Date, 'yyyy-MM')

ORDER BY SalesMonth;

* **Top Selling Locations (Cities):**

SELECT City, SUM(Revenue) AS TotalRevenue

FROM salesdata1

GROUP BY City

ORDER BY TotalRevenue DESC;

* **Peak Sales Hours:**

SELECT DATEPART(HOUR, Order\_Date) AS OrderHour, SUM(Revenue) AS TotalRevenue

FROM salesdata1

GROUP BY DATEPART(HOUR, Order\_Date)

ORDER BY TotalRevenue DESC;

**4. Data Visualization in Power BI**

After analyzing the data, key insights were visualized using Power BI:

* Total Sales Revenue & Quantity Ordered: Displayed using card visuals for quick insights.
* Sales by Product: A bar chart showing revenue contribution of each product.
* Best-Selling Products: A table visual listing products with the highest quantity sold.
* Monthly Sales Trend: A line chart depicting fluctuations in revenue.
* Sales by City: A bar chart comparing total revenue across different cities.
* Peak Sales Hours: A column chart to show the most profitable hours for sales.
* Filters & Slicers:
  + Slicers were added for product, sales month, and city to allow dynamic exploration.
  + Sorting was applied to tables and charts to display data in descending order for better insights.

**5. Key Findings**

* Highest revenue-generating product: MacBook Pro Laptop.
* Most sold product by quantity: AAA Batteries (4-pack).
* City with the highest revenue: San Francisco.
* Peak sales hour: 7 PM.
* Monthly revenue trends showed sales peaks in certain months, indicating seasonal demand.

**6. Conclusion**

This analysis provided key business insights by cleaning and processing raw data using Pandas, analyzing trends with SQL queries, and visualizing results in Power BI. These insights can help businesses optimize their sales strategies, inventory management, and marketing efforts.

The monthly sales trend showed a decline over time, particularly from mid-2019 to early 2020. This suggests a possible shift in consumer behavior or external factors impacting sales. Further analysis can help determine the cause, whether it's seasonal patterns, economic conditions, or product availability issues. The recommendation is to investigate these patterns further and adjust business strategies accordingly.

By leveraging these findings, businesses can enhance decision-making, streamline operations, and implement data-driven marketing strategies for sustained growth.

**7. Recommendations**

* Stock popular products in high-demand locations to maximize revenue.
* Target high-sales hours (e.g., 7 PM) for promotions and ads.
* Analyze seasonal trends further to prepare for peak months.
* Improve product bundling and discount strategies for best-selling products to boost sales further.