Project Report: Sales Data Analysis Dashboard

1. Project Overview The goal of this project was to build an interactive sales dashboard using Streamlit, providing insights into sales data while I learn how to create interactive data visualizations in Python. This project marks my first experience developing a dashboard on Streamlit.
2. Objectives

- Load and explore raw sales data.
- Visualize key metrics, such as monthly revenue trends and revenue distribution by country.
- Create an interactive, user-friendly dashboard that allows users to explore the data easily.
- Gain hands-on experience with Streamlit, pandas, matplotlib, seaborn, and Plotly.

3. Tools and Technologies

- Python – programming language used for data analysis and visualization.
- Streamlit – for building the interactive dashboard.
- Pandas & NumPy – data manipulation and analysis.
- Matplotlib & Seaborn – static visualizations.
- Plotly – interactive charts and graphs.
- GitHub & Codespaces – version control and cloud development.

4. Data

- Dataset: Sales records containing the following fields: InvoiceNo, StockCode, Description, Quantity, InvoiceDate, UnitPrice, CustomerID, Country.
- Data preprocessing steps:
    - Converted InvoiceDate to datetime format.
    - Aggregated data for monthly revenue and country-level revenue distribution.
    - Checked for missing values and handled them appropriately.

5. Dashboard Features

- Raw Data Table: Users can view the full dataset.
- Summary Statistics: Displays key metrics like total quantity, average sales, and revenue distribution.
- Monthly Revenue Bar Chart: Shows sales trends over time using Seaborn.
- Revenue by Country Pie Chart: Interactive visualization with Plotly for comparing sales across countries.

6. Challenges and Learning

- This was my first Streamlit project, so I learned how to structure the dashboard and manage interactive plots.
- Handling dataset paths and ensuring all files are included for deployment on Streamlit Cloud was a key challenge.
- Learned to combine static and interactive visualizations to make insights easier to understand.

7. Future Improvements

- Add filters for date ranges, products, or countries.
- Include additional charts such as Quantity vs Revenue scatter plots.
- Improve dashboard aesthetics and layout for better user experience.

8. Conclusion This project served as an excellent hands-on introduction to Streamlit and data visualization. It helped me understand how to turn raw sales data into insights via an interactive dashboard and gave me practical experience in Python-based data analysis.

Project Link: https://portfoliomesh-lniczkzr7moncv4gfnmnkk.streamlit.app/