Name: Saniya Deshpande

Email: [saniyadeshpande05@gmail.com](mailto:saniyadeshpande05@gmail.com)

Project: financial analysis

Without dashboard:

import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns  
  
df = pd.read\_csv("Financial Analytics data.csv", encoding='unicode\_escape')  
print("printing rows and col")  
print(df.shape)  
print("printing first five entries")  
print(df.head())  
  
*# Drop unnamed columns with all NaN values*df\_cleaned = df.dropna(axis=1, how='all')  
  
print("\nchecking null values")  
print(pd.isnull(df\_cleaned).sum())  
print(df\_cleaned.shape)  
  
*# Convert to numeric*df\_cleaned['Mar Cap - Crore'] = pd.to\_numeric(df\_cleaned['Mar Cap - Crore'], errors='coerce')  
df\_cleaned['Sales Qtr - Crore'] = pd.to\_numeric(df\_cleaned['Sales Qtr - Crore'], errors='coerce')  
  
print("describing ")  
print(df\_cleaned[['Mar Cap - Crore', 'Sales Qtr - Crore']].describe())  
print("the columns are")  
print(df.columns)  
  
*# Assuming df\_cleaned has been processed as in your previous code*plt.figure(figsize=(10, 6))  
sns.histplot(df\_cleaned['Mar Cap - Crore'], bins=30, kde=True, color='blue')  
plt.title('Distribution of Mar Cap - Crore')  
plt.xlabel('Mar Cap - Crore')  
plt.ylabel('Frequency')  
plt.show()  
  
*# Histogram for 'Sales Qtr - Crore'*plt.figure(figsize=(10, 6))  
sns.histplot(df\_cleaned['Sales Qtr - Crore'], bins=30, kde=True, color='green')  
plt.title('Distribution of Quarterly Sales')  
plt.xlabel('Sales (Crore)')  
plt.ylabel('Frequency')  
plt.show()  
  
*# Bar chart for top N companies based on market capitalization*top\_n = 10  
top\_market\_cap = df\_cleaned.nlargest(top\_n, 'Mar Cap - Crore')  
plt.figure(figsize=(12, 6))  
sns.barplot(x='Mar Cap - Crore', y='Name', data=top\_market\_cap, palette='viridis')  
plt.title(f'Top {top\_n} Companies by Market Capitalization')  
plt.xlabel('Market Capitalization (Crore)')  
plt.ylabel('Company Name')  
plt.show()  
  
correlation\_matrix = df\_cleaned[['Mar Cap - Crore', 'Sales Qtr - Crore']].corr()  
sns.heatmap(correlation\_matrix, annot=True, cmap='coolwarm', linewidths=.5)  
plt.title('Correlation Matrix')  
plt.show()

With Dashboard

import dash  
from dash import dcc, html  
from dash.dependencies import Input, Output  
import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns  
  
from file1 import correlation\_matrix, top\_market\_cap, top\_n  
  
*# Load the dataset*df = pd.read\_csv("Financial Analytics data.csv", encoding='unicode\_escape')  
  
*# Drop unnamed columns with all NaN values*df\_cleaned = df.dropna(axis=1, how='all')  
  
*# Convert to numeric*df\_cleaned['Mar Cap - Crore'] = pd.to\_numeric(df\_cleaned['Mar Cap - Crore'], errors='coerce')  
df\_cleaned['Sales Qtr - Crore'] = pd.to\_numeric(df\_cleaned['Sales Qtr - Crore'], errors='coerce')  
  
*# Initialize the Dash app*app = dash.Dash(\_\_name\_\_)  
  
*# Define the layout of the dashboard*app.layout = html.Div([  
 html.H1("Financial Analytics Dashboard"),  
  
 *# Distribution of Mar Cap - Crore* dcc.Graph(  
 id='mar-cap-histogram',  
 figure={  
 'data': [  
 {  
 'x': df\_cleaned['Mar Cap - Crore'],  
 'type': 'histogram',  
 'name': 'Mar Cap - Crore',  
 'marker': {'color': 'blue'}  
 }  
 ],  
 'layout': {  
 'title': 'Distribution of Mar Cap - Crore',  
 'xaxis': {'title': 'Mar Cap - Crore'},  
 'yaxis': {'title': 'Frequency'}  
 }  
 }  
 ),  
  
 *# Distribution of Quarterly Sales* dcc.Graph(  
 id='sales-histogram',  
 figure={  
 'data': [  
 {  
 'x': df\_cleaned['Sales Qtr - Crore'],  
 'type': 'histogram',  
 'name': 'Sales Qtr - Crore',  
 'marker': {'color': 'green'}  
 }  
 ],  
 'layout': {  
 'title': 'Distribution of Quarterly Sales',  
 'xaxis': {'title': 'Sales (Crore)'},  
 'yaxis': {'title': 'Frequency'}  
 }  
 }  
 ),  
  
 *# Top N Companies by Market Capitalization* dcc.Graph(  
 id='top-market-cap-bar',  
 figure={  
 'data': [  
 {  
 'x': top\_market\_cap['Mar Cap - Crore'],  
 'y': top\_market\_cap['Name'],  
 'type': 'bar',  
 'orientation': 'h',  
 'marker': {'color': 'viridis'}  
 }  
 ],  
 'layout': {  
 'title': f'Top {top\_n} Companies by Market Capitalization',  
 'xaxis': {'title': 'Market Capitalization (Crore)'},  
 'yaxis': {'title': 'Company Name'}  
 }  
 }  
 ),  
  
 *# Correlation Matrix* dcc.Graph(  
 id='correlation-heatmap',  
 figure={  
 'data': [  
 {  
 'z': correlation\_matrix.values,  
 'x': correlation\_matrix.columns,  
 'y': correlation\_matrix.index,  
 'type': 'heatmap',  
 'colorscale': 'coolwarm'  
 }  
 ],  
 'layout': {  
 'title': 'Correlation Matrix'  
 }  
 }  
 ),  
])  
  
*# Run the app*if \_\_name\_\_ == '\_\_main\_\_':  
 app.run\_server(debug=True)