

```
!pip install fpdf plotly openpyxl
```

```
➡ Collecting fpdf
  Downloading fpdf-1.7.2.tar.gz (39 kB)
  Preparing metadata (setup.py) ... done
Requirement already satisfied: plotly in /usr/local/lib/python3.11/dist-packages (5.24.1)
Requirement already satisfied: openpyxl in /usr/local/lib/python3.11/dist-packages (3.1.5)
Requirement already satisfied: tenacity>=6.2.0 in /usr/local/lib/python3.11/dist-packages (from plotly)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-packages (from plotly) (24.1)
Requirement already satisfied: et-xmlfile in /usr/local/lib/python3.11/dist-packages (from openpyxl) (1.0.1)
Building wheels for collected packages: fpdf
  Building wheel for fpdf (setup.py) ... done
  Created wheel for fpdf: filename=fpdf-1.7.2-py2.py3-none-any.whl size=40704 sha256=17aa8e8eccdb3bc905
  Stored in directory: /root/.cache/pip/wheels/65/4f/66/bbda9866da446a72e206d6484cd97381cbc7859a7068541
Successfully built fpdf
Installing collected packages: fpdf
Successfully installed fpdf-1.7.2
```

```
import pandas as pd
import networkx as nx
import matplotlib.pyplot as plt
import plotly.express as px
from fpdf import FPDF
import os

def create_sample_csv(filename="transactions.csv"):
    data = {
        'Sender': ['Alice', 'Bob', 'Alice', 'Charlie', 'David', 'Eve', 'Bob', 'Frank', 'Grace', 'Heidi'],
        'Receiver': ['Bob', 'Charlie', 'Charlie', 'Alice', 'Eve', 'Frank', 'Grace', 'Heidi', 'Alice', 'Bob'],
        'Amount': [1200, 1500, 800, 2000, 2200, 1800, 950, 1750, 3000, 1250]
    }
    df_sample = pd.DataFrame(data)
    df_sample.to_csv(filename, index=False)
    print(f"✅ Sample data written to `{filename}`")

def graph_analytics(df):
    G = nx.from_pandas_edgelist(df, 'Sender', 'Receiver', ['Amount'])
    centrality = nx.degree_centrality(G)
    return G, centrality

def generate_charts(df):
    os.makedirs("charts", exist_ok=True)
    os.makedirs("reports", exist_ok=True)

    sender_group = df.groupby('Sender')['Amount'].sum().reset_index()
    fig = px.bar(sender_group, x='Sender', y='Amount', title='Total Sent by Sender')
    fig.write_html("charts/sender_bar.html")

    cross_tab = pd.crosstab(df['Sender'], df['Receiver'], values=df['Amount'], aggfunc='sum', dropna=False)
    cross_tab.to_csv("reports/crosstab.csv")

    stats = df['Amount'].describe()
    stats.to_csv("reports/statistics.csv")

    return sender_group, stats, cross_tab
```

```

def filter_sort_group(df):
    df_filtered = df[df['Amount'] > 1000]
    df_sorted = df_filtered.sort_values(by='Amount', ascending=False)
    df_grouped = df_sorted.groupby('Sender').agg({'Amount': ['sum', 'mean']}).reset_index()
    return df_grouped

def add_summary_line(df):
    total = df['Amount'].sum()
    return f"Total Transaction Value: ₹{total:.2f}"

def export_reports(df, summary):
    # Resetting MultiIndex columns to single-level columns
    df.columns = ['_'.join(col).strip() for col in df.columns.values]
    # Now it should work as intended
    df.to_excel("reports/report.xlsx", index=False)
    df.to_csv("reports/report.csv", index=False)
    df.to_xml("reports/report.xml", index=False)

    pdf = FPDF()
    pdf.add_page()
    pdf.set_font("Arial", size=12)
    pdf.cell(200, 10, txt="Transactional Report", ln=True, align='C')
    pdf.ln(10)
    for i, row in df.iterrows():
        pdf.cell(200, 10, txt=str(row.to_dict()), ln=True)
    pdf.ln(10)
    pdf.set_font("Arial", style='B')
    # Encode the summary string to UTF-8 to support the Rupee symbol.
    pdf.cell(200, 10, txt=summary.encode('utf-8').decode('latin-1'), ln=True)
    pdf.output("reports/report.pdf")

def run_report():
    create_sample_csv()
    df = pd.read_csv("transactions.csv")
    G, centrality = graph_analytics(df)
    sender_group, stats, crosstab = generate_charts(df)
    processed_data = filter_sort_group(df)
    summary = add_summary_line(df)
    export_reports(processed_data, summary)
    print("✅ Reports generated! Check the left file browser in Colab to download.")

```

run_report()



- ✅ Sample data written to `transactions.csv`
- ✅ Reports generated! Check the left file browser in Colab to download.

Start coding or [generate](#) with AI.

Start coding or [generate](#) with AI.