Assignment: Visualizing Sales Data for a Retail Store Chain

Objective: The purpose of this assignment is to help you learn how to use Matplotlib to create meaningful visualizations from real-world datasets. You will analyze and visualize sales data of a fictional retail store chain, applying the concepts of various plot types, customizations, and subplots.

Scenario:

You are a data analyst at a retail chain, "RetailMart," that operates in multiple cities. The management has tasked you with analyzing and presenting insights from the monthly sales data. Your goal is to use Matplotlib to create professional-grade visualizations for their quarterly report.

Dataset Description:

The dataset contains the following columns:

1. **City**: Name of the city where the store is located.
2. **Month**: Month of the sales.
3. **Revenue**: Total revenue generated by the store.
4. **Number of Transactions**: Count of transactions during the month.
5. **Product Categories**: Categories of products sold (e.g., Electronics, Clothing, Groceries).
6. **Customer Satisfaction Score**: Average satisfaction rating (1 to 10).

Tasks:

Task 1: Install and Setup Matplotlib

1. Install Matplotlib using pip.
2. Import the necessary libraries (`matplotlib.pyplot` and `numpy`).
3. Verify the installation and version of Matplotlib.

Task 2: Visualizing Monthly Revenue

1. Create a **line chart** showing the revenue trend across months for a selected city.
    - Use markers to indicate individual data points.
    - Customize the line style and color.
    - Add grid lines, axis labels, and a title.

Example:

```
plt.plot(months, revenue, marker='o', linestyle='--', color='b')
plt.title("Monthly Revenue Trend")
plt.xlabel("Month")
plt.ylabel("Revenue ($)")
plt.grid()
plt.show()
```

Task 3: Comparing Product Categories

1. Use a **bar chart** to compare the total revenue generated by each product category for a specific city.
2. Add labels to the bars to display exact revenue values.
3. Customize bar width and colors.

Task 4: Scatter Plot of Transactions vs. Revenue

1. Create a **scatter plot** to show the relationship between the number of transactions and revenue for all cities.
2. Customize the dot size to represent customer satisfaction scores and the dot color to represent cities.
3. Add a legend to explain the sizes and colors.

Task 5: City-Wise Revenue Distribution

1. Create a **pie chart** to represent the percentage contribution of revenue from each city for the quarter.
2. Use the `explode` parameter to highlight the city with the highest revenue.
3. Add labels, a legend, and shadows to enhance the visualization.

Task 6: Customer Satisfaction Analysis

1. Create a **histogram** of customer satisfaction scores across all cities.
2. Set the number of bins to 10 to represent score ranges (1–10).

3. Add appropriate titles, axis labels, and grid lines.

Task 7: Subplots for Comparative Analysis

1. Use **subplots** to display:
    - A line chart for revenue trends.
    - A bar chart for product category revenue.
    - A scatter plot for transactions vs. revenue.
2. Add a **supertitle** for the entire figure summarizing the analysis.

Submission Guidelines:

1. Submit the Python script or Jupyter Notebook used for the assignment.
2. Include comments explaining each step of the visualization process.
3. Present the results to the professor.

Evaluation Criteria:

1. Correct implementation of Matplotlib concepts.
2. Quality and clarity of visualizations.
3. Creativity in customization and presentation.
4. Proper use of titles, labels, legends, and annotations.
5. Code readability and documentation.

Dataset

For the dataset required for the assignment, you can either generate your own synthetic dataset or find a relevant dataset online. Here's how you can proceed:

Option 1: Generate a Synthetic Dataset

You can generate a synthetic dataset using Python with libraries like `pandas` and `numpy`. Here's an example:

```
import pandas as pd
import numpy as np

# Generate synthetic data
data = {
```

```
    "City": np.random.choice(
        ["New York", "Los Angeles", "Chicago", "Houston", "Phoenix"], size=100
    ),
    "Month": np.random.choice(
        ["January", "February", "March", "April", "May", "June", "July", "August", "September",
"October", "November", "December"], size=100
    ),
    "Revenue": np.random.randint(5000, 50000, size=100),
    "Number of Transactions": np.random.randint(50, 500, size=100),
    "Product Categories": np.random.choice(
        ["Electronics", "Clothing", "Groceries", "Books", "Furniture"], size=100
    ),
    "Customer Satisfaction Score": np.random.randint(1, 11, size=100),
}

# Create DataFrame
df = pd.DataFrame(data)

# Save to CSV
df.to_csv("sales_data.csv", index=False)

print(df.head())
```

This script creates a CSV file (`sales_data.csv`) with 100 rows of random data matching the required fields. Students can use this dataset for analysis.

Option 2: Find an Existing Dataset

You can find similar datasets from public data repositories. Some suggested sources include:

1. **Kaggle**:
    - Search for "retail sales dataset," "store performance dataset," or "monthly sales dataset."
    - Example: *Retail Sales Data*, *Superstore Dataset*.
2. **Google Dataset Search**:
    - Look for retail or sales datasets.
3. **Data.gov**:
    - Check for datasets related to business and sales.
4. **UCI Machine Learning Repository**:
    - Look for datasets in the "Business" category.

Option 3: Use Example Dataset

If you want an example dataset immediately, I can provide a small CSV file with the necessary fields. Let me know if you'd like me to generate and provide this file.