Practical 10

Write a python program to perform data visualization trough Matplotlib.

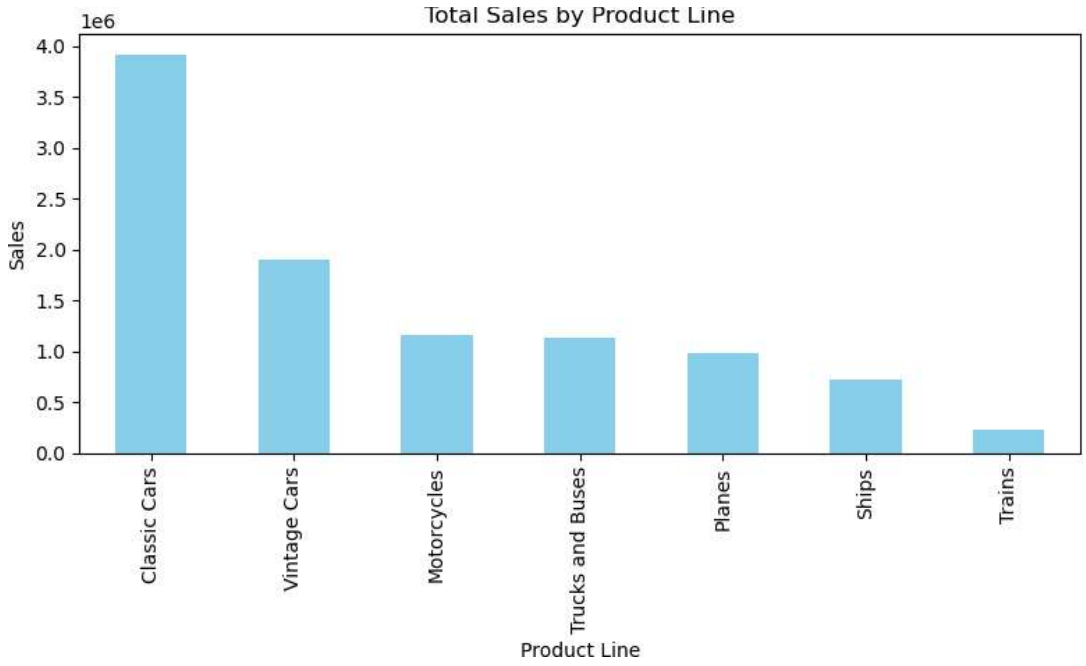
In [3]:
```
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv("sales_dataset.csv")
```

In [4]:
```
df.head()
```

Out[4]:
```
<bound method NDFrame.head of          SALES    ORDERDATE     STATUS  YEAR_ID  PRO DUCTLINE  \
0      2871.00   08/03/2008   Shipped      2008  Motorcycles
1      2765.90   15/03/2008   Shipped      2008  Motorcycles
2      3884.34   22/03/2008   Shipped      2008  Motorcycles
3      3746.70   13/01/2008   Shipped      2008  Motorcycles
4      5205.27   07/04/2008   Shipped      2008  Motorcycles
```

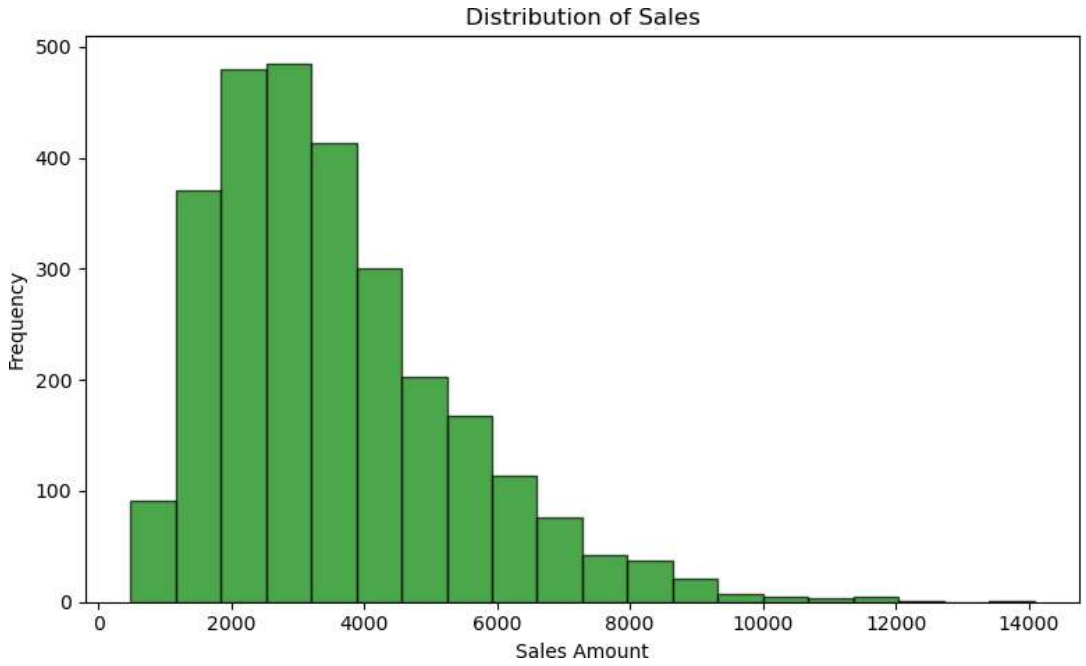
Bar Plot: Total Sales by Product Line

```
product_sales =

df.groupby('PRODUCTLINE')['SALES'].sum().sort_values(ascending=False)

plt.figure(figsize=(8,5))
product_sales.plot(kind='bar', color='blue')
plt.title("Total Sales by Product Line")
plt.xlabel("Product Line")
plt.ylabel("Sales")
plt.tight_layout()
plt.show()
```



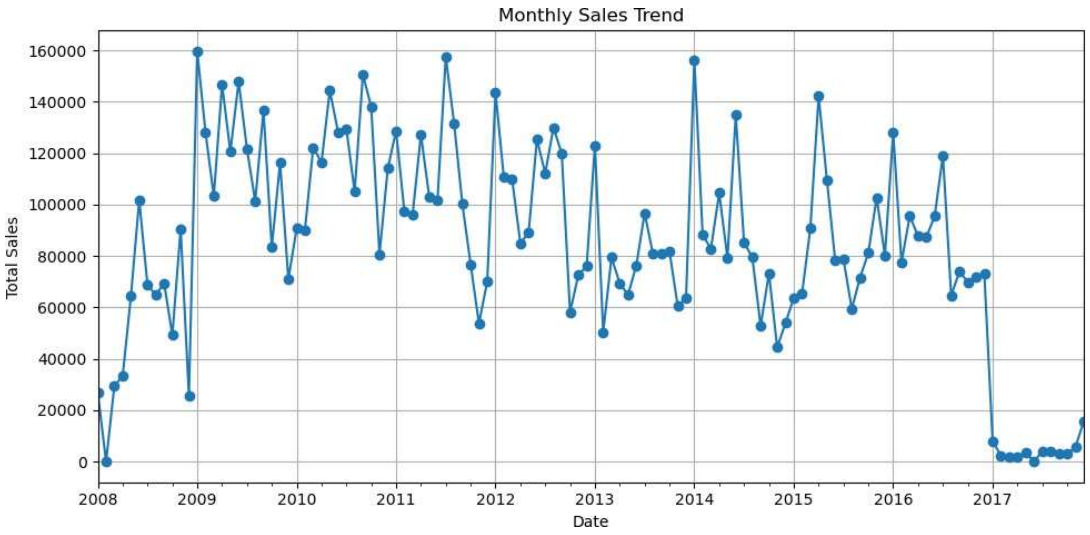
Histogram: Distribution of Sales Amount

In [15]:
```
plt.figure(figsize=(8,5))
plt.hist(df['sales'], bins=20, color='blue', edgecolor='black', alpha=0.7)
plt.title("Distribution of Sales")
plt.xlabel("Sales Amount")
plt.ylabel("Frequency")
plt.tight_layout()
plt.show()
```



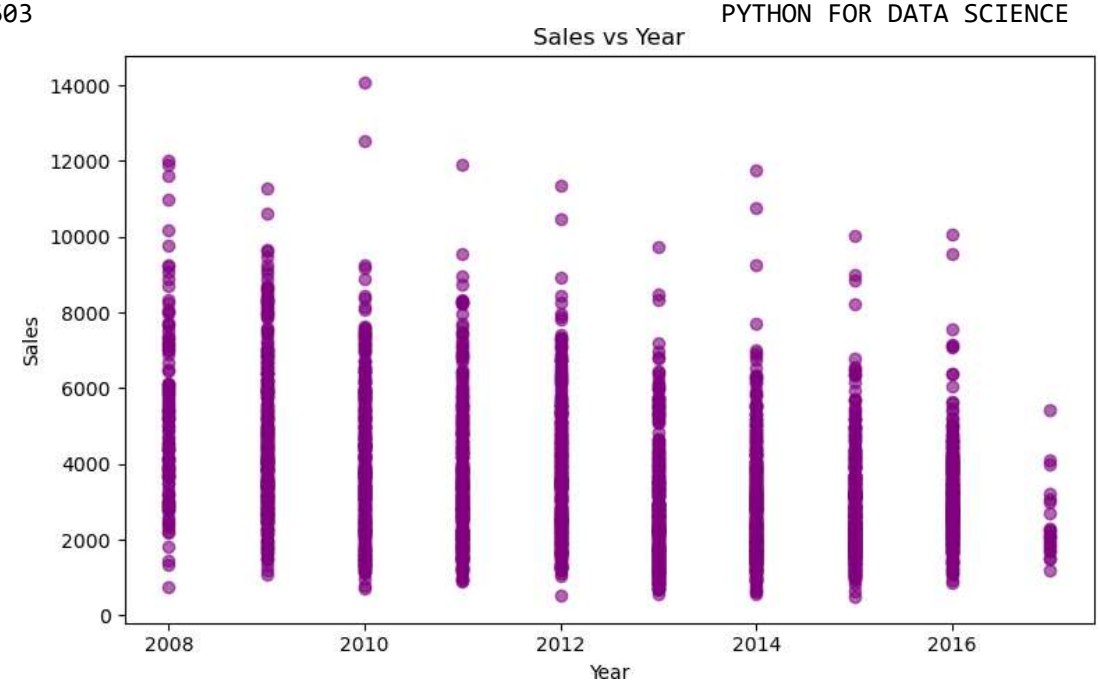
Line Plot: Monthly Sales Trend

In [16]:
```
df['ORDERDATE'] = pd.to_datetime(df['ORDERDATE'], format="%d/%m/%Y")

df_time = df.set_index('ORDERDATE').resample('M')['SALES'].sum()

plt.figure(figsize=(10,5))
df_time.plot(marker='o')
plt.title("Monthly Sales Trend")
plt.xlabel("Date")
plt.ylabel("Total Sales")
plt.grid(True)
plt.tight_layout()
plt.show()
```



Scatter Plot: Sales vs Year

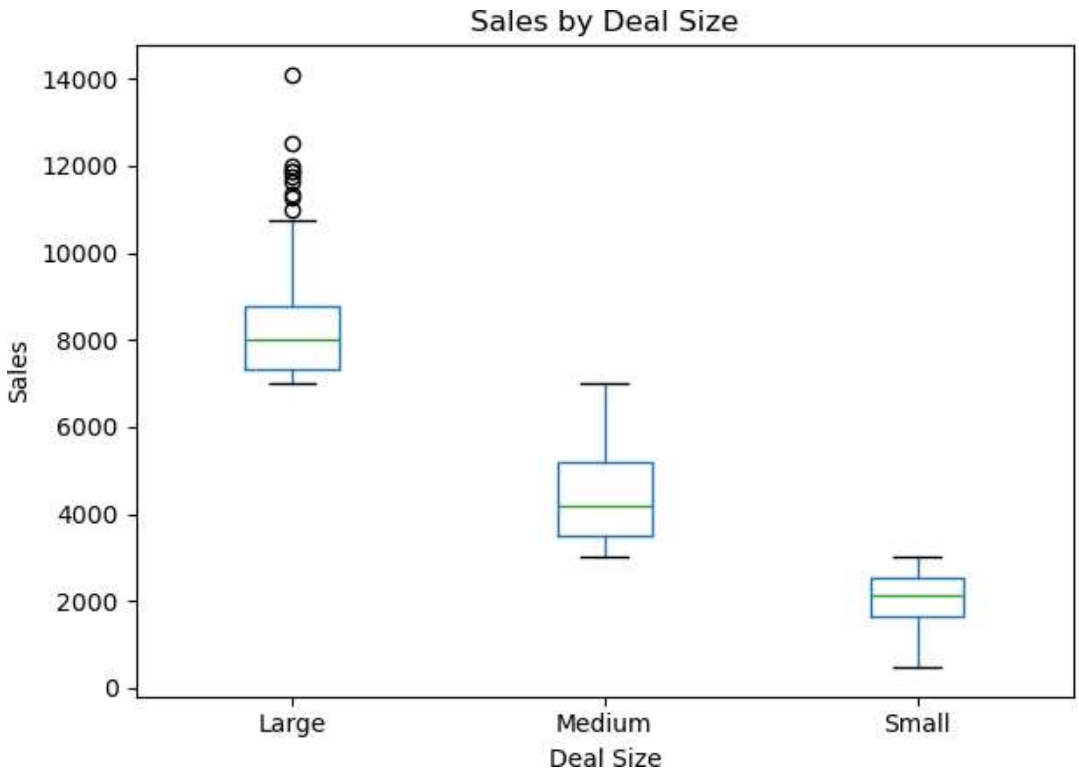
In [17]:
```
plt.figure(figsize=(8,5))
plt.scatter(df['YEAR_ID'], df['SALES'], alpha=0.6, c='purple')
plt.title("Sales vs Year")
plt.xlabel("Year")
plt.ylabel("Sales")
plt.tight_layout()
plt.show()
```



Box Plot: Sales by Deal Size

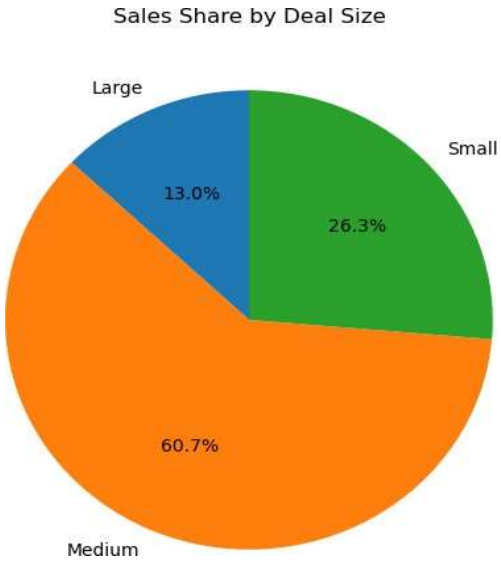
In [18]:
```
plt.figure(figsize=(8,5))
df.boxplot(column='SALES', by='DEALSIZE', grid=False)
plt.title("Sales by Deal Size")
plt.suptitle("")  # Remove automatic title
plt.xlabel("Deal Size")
plt.ylabel("Sales")
plt.tight_layout()
plt.show()
```

```
<Figure size 800x500 with 0 Axes>
```



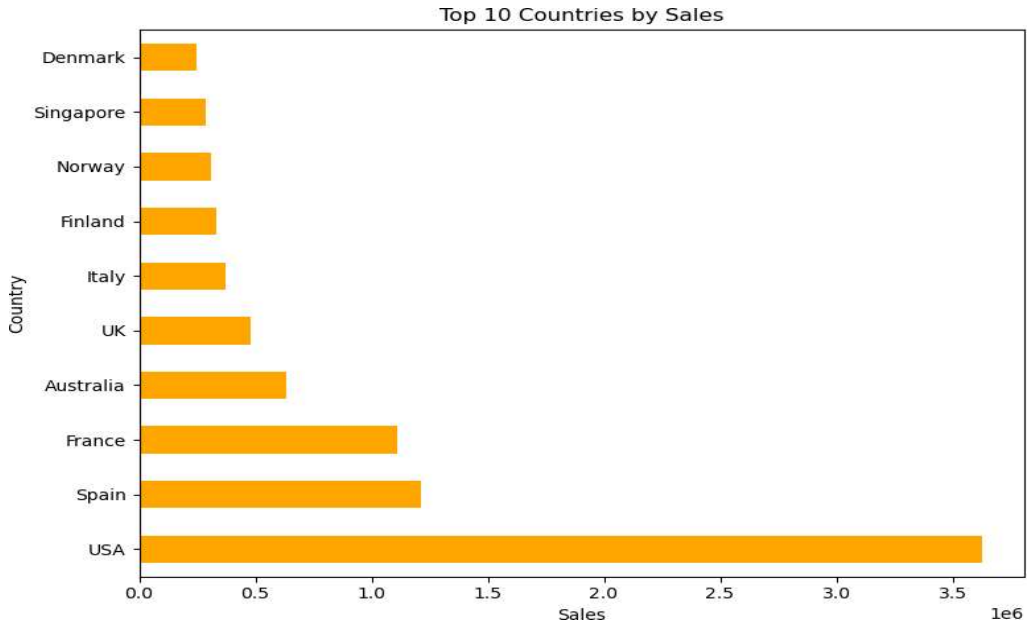
Pie Chart – Sales by Deal Size

In [19]:
```
deal_sales = df.groupby('DEALSIZE')['SALES'].sum()

plt.figure(figsize=(6,6))
plt.pie(deal_sales, labels=deal_sales.index, autopct='%1.1f%%', startangle=90)
plt.title("Sales Share by Deal Size")
plt.show()
```



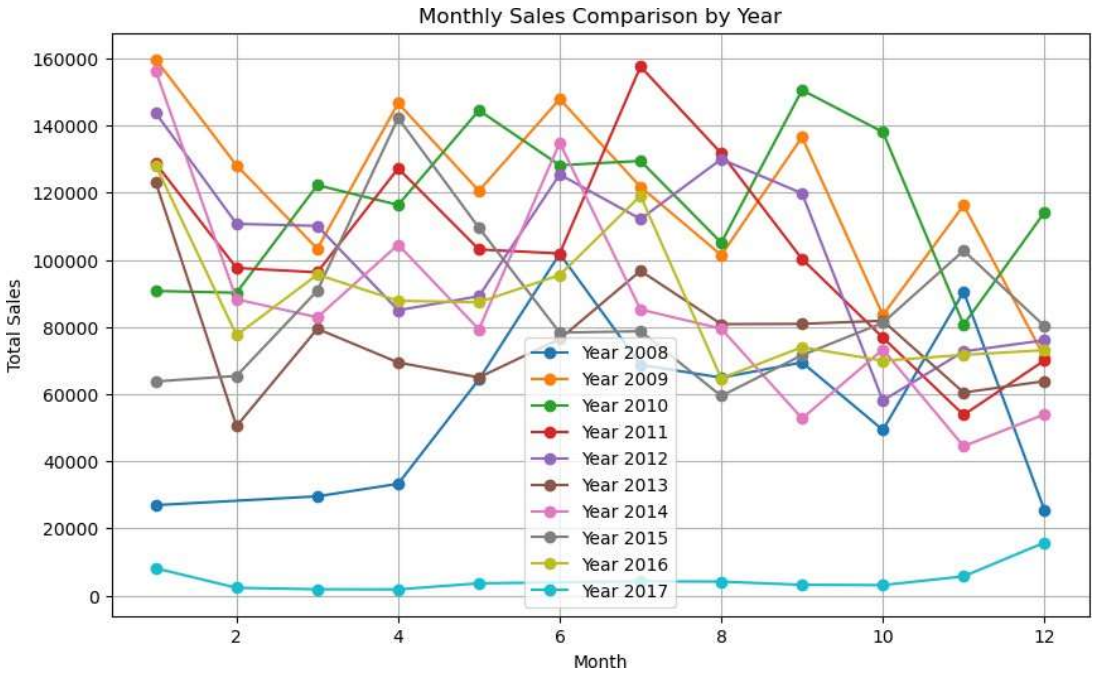
Horizontal Bar Chart – Sales by Country

In [20]:
```
country_sales  =  df.groupby('COUNTRY')['SALES'].sum().sort_values(ascending=False

plt.figure(figsize=(8,6))
country_sales.plot(kind='barh', color='cyan')
plt.title("Top 10 Countries by Sales")
plt.xlabel("Sales")
plt.ylabel("Country")
plt.tight_layout()
plt.show()
```



Multiple Line Plots – Sales per Year

In [21]:
```
df['MONTH'] = df['ORDERDATE'].dt.month
year_month_sales = df.groupby(['YEAR_ID', 'MONTH'])['SALES'].sum().reset_index()

plt.figure(figsize=(10,6))
for year in sorted(df['YEAR_ID'].unique()):
    subset = year_month_sales[year_month_sales['YEAR_ID'] == year]
    plt.plot(subset['MONTH'], subset['SALES'], marker='o', label=f"Year {year}")

plt.title("Monthly Sales Comparison by Year")
plt.xlabel("Month")
plt.ylabel("Total Sales")
plt.legend()
plt.grid(True)
plt.show()
```



Subplots – Two Graphs Side by Side

In [22]:
```
fig, axes = plt.subplots(1, 2, figsize=(12,5))

# Histogram
axes[0].hist(df['SALES'], bins=20, color='green', edgecolor='black')
axes[0].set_title("Sales Distribution")
axes[0].set_xlabel("Sales")
axes[0].set_ylabel("Frequency")

# Boxplot
axes[1].boxplot(df['SALES'], vert=True)
axes[1].set_title("Box Plot of Sales")
axes[1].set_ylabel("Sales")

plt.tight_layout()
plt.show()
```