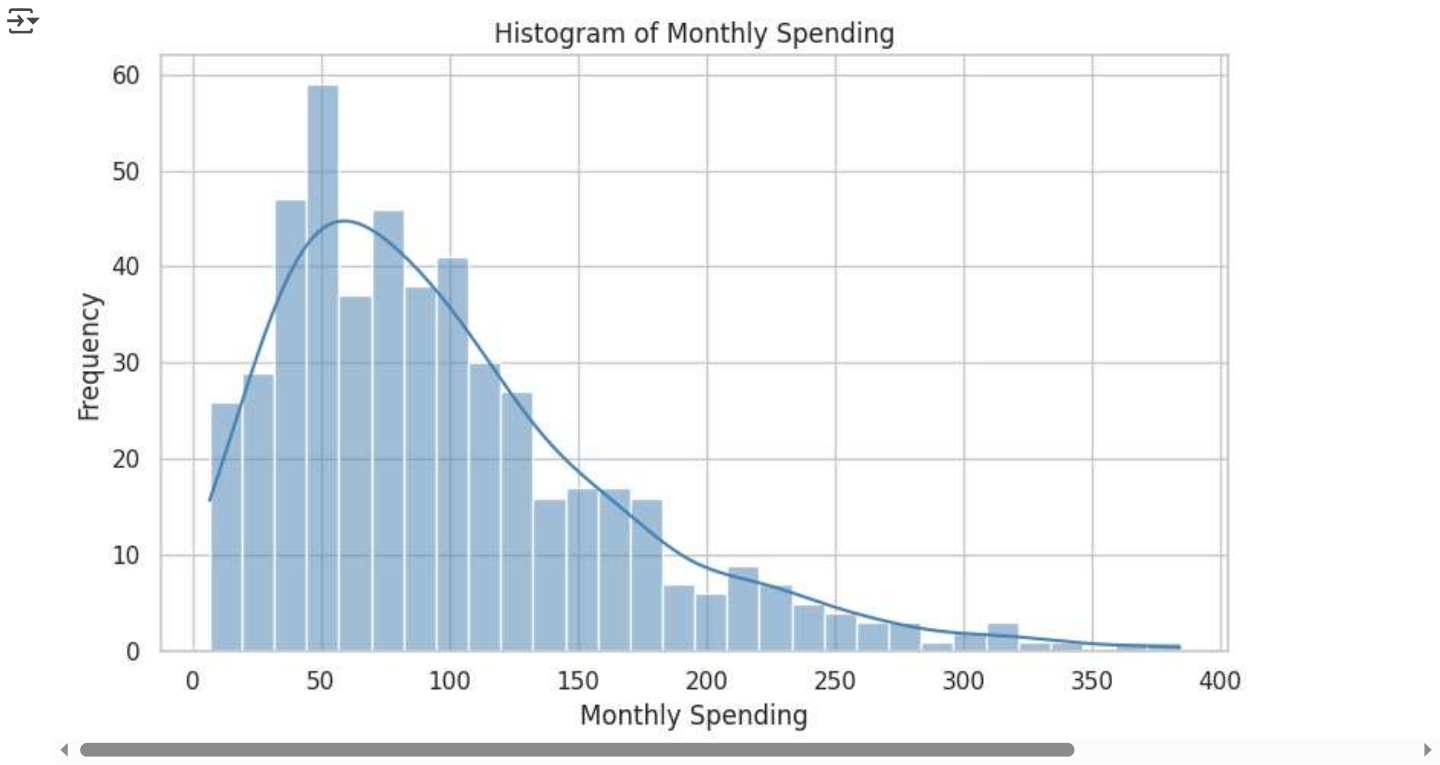
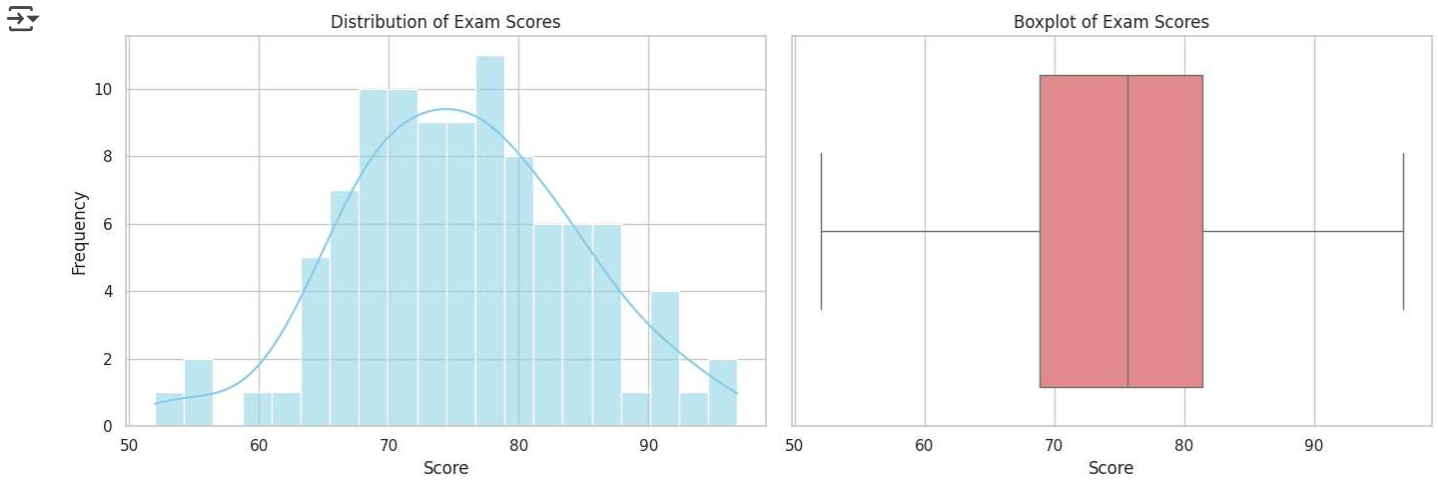
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

# Simulate some example monthly spending data
# Replace this with your real data: df = pd.read_csv("your_data.csv")
np.random.seed(42)
monthly_spending = np.random.gamma(shape=2, scale=50, size=500)
df = pd.DataFrame({'Monthly_Spending': monthly_spending})

# Set Seaborn style for better aesthetics


# Create histogram with KDE
plt.figure(figsize=(8, 5))
sns.histplot(df['Monthly_Spending'], bins=30, kde=True, color='steelblue')

# Add titles and labels
plt.title('Histogram of Monthly Spending')
plt.xlabel('Monthly Spending')
plt.ylabel('Frequency')

# Show plot
plt.tight_layout()
plt.show()
```




```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

# Simulate exam scores for 100 students (you can replace this with your actual data)
np.random.seed(1)
exam_scores = np.random.normal(loc=75, scale=10, size=100)  # mean=75, std=10
# Clip scores to a realistic range (e.g., 0-100)
exam_scores = np.clip(exam_scores, 0, 100)
```

```
df = pd.DataFrame({'Exam_Score': exam_scores})

# Set Seaborn style
sns.set(style="whitegrid")

# Create figure with two subplots
fig, axes = plt.subplots(1, 2, figsize=(14, 5))

# Histogram with KDE
sns.histplot(df['Exam_Score'], bins=20, kde=True, color='skyblue', ax=axes[0])
axes[0].set_title('Distribution of Exam Scores')
axes[0].set_xlabel('Score')
axes[0].set_ylabel('Frequency')

# Boxplot to identify outliers
sns.boxplot(x=df['Exam_Score'], color='lightcoral', ax=axes[1])
axes[1].set_title('Boxplot of Exam Scores')
axes[1].set_xlabel('Score')

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



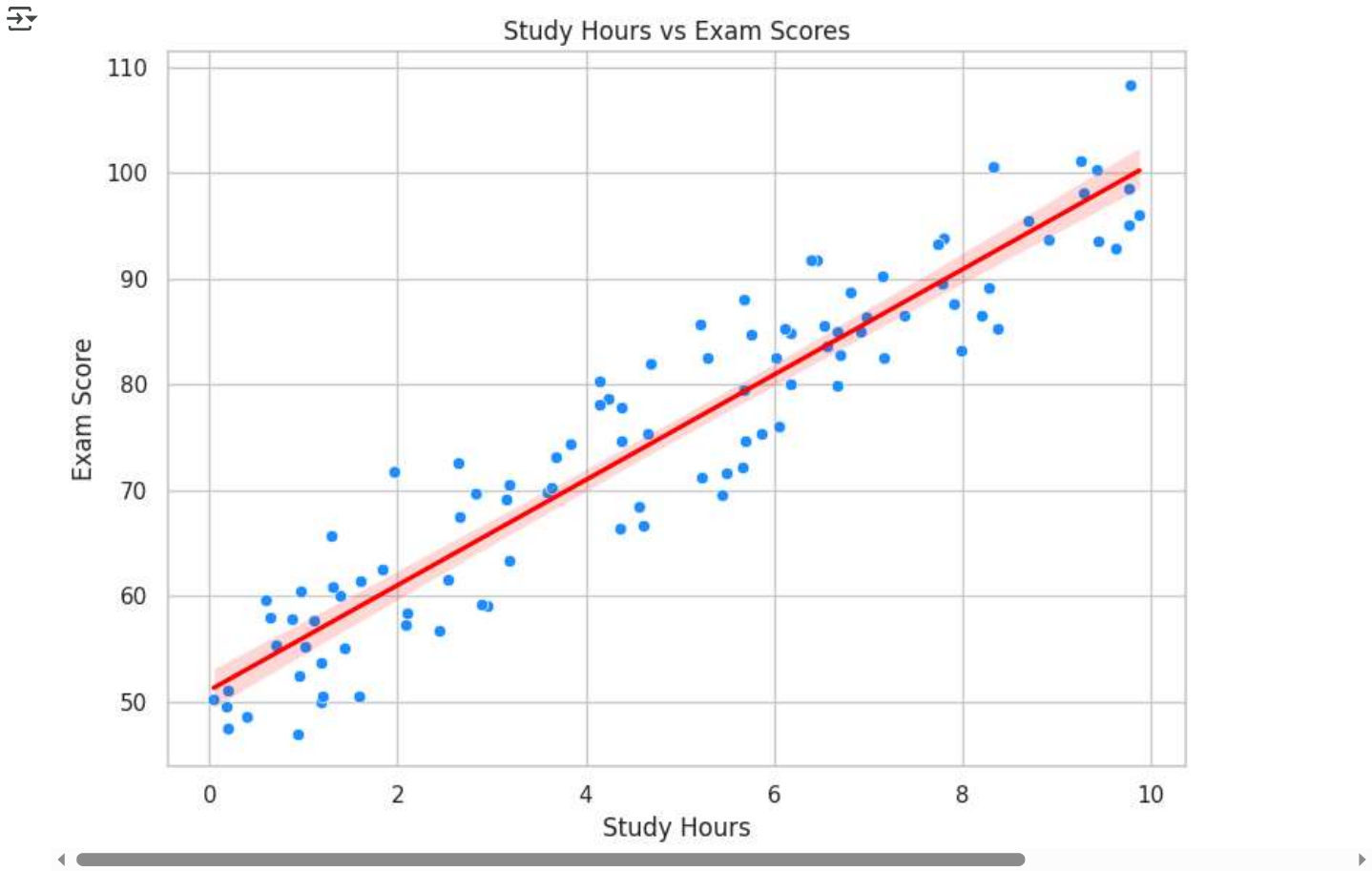

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

# Simulate example rating data
np.random.seed(42)
ratings = np.random.choice([1, 2, 3, 4, 5], size=300, p=[0.05, 0.1, 0.15, 0.3, 0.4])
df = pd.DataFrame({'Rating': ratings})

# Set Seaborn style
sns.set(style="whitegrid")

# Countplot (bar plot of frequencies)
plt.figure(figsize=(8, 5))
sns.countplot(x='Rating', data=df, palette='viridis', order=[1, 2, 3, 4, 5])

# Add labels and title
plt.title('Distribution of Product Review Ratings')
plt.xlabel('Rating (Stars)')
plt.ylabel('Number of Reviews')
```

```
plt.xticks([0, 1, 2, 3, 4], ['1 Star', '2 Stars', '3 Stars', '4 Stars', '5 Stars'])

# Show the plot
plt.tight_layout()
plt.show()
```

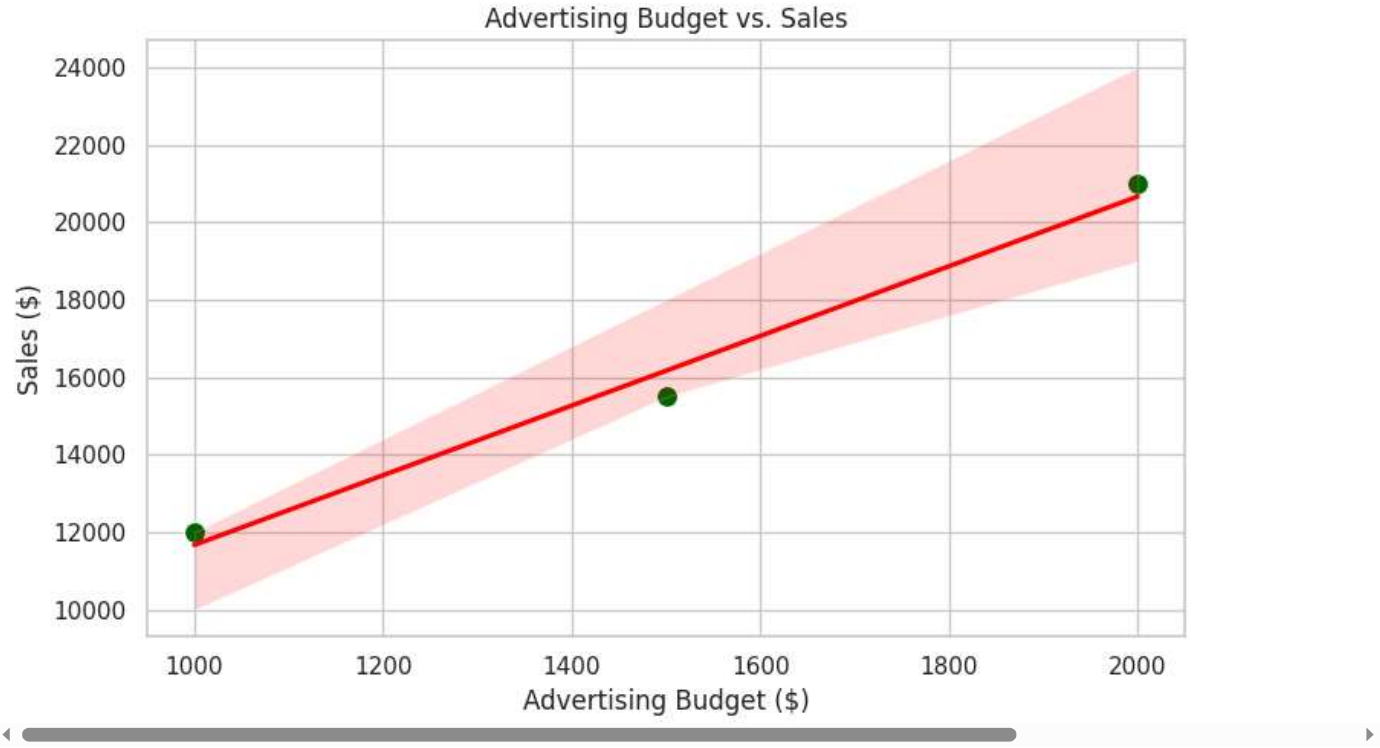
```
<ipython-input-9-eb30a1118e2d>:16: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hu

  sns.countplot(x='Rating', data=df, palette='viridis', order=[1, 2, 3, 4, 5])
```


```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

# Simulate some sample data (replace with your real data if available)
np.random.seed(0)
study_hours = np.random.uniform(0, 10, 100)
exam_scores = 50 + 5 * study_hours + np.random.normal(0, 5, 100)  # Adding noise

df = pd.DataFrame({
    'Study_Hours': study_hours,
    'Exam_Score': exam_scores
})

# Set Seaborn style
sns.set(style="whitegrid")

# Create scatter plot with regression line
plt.figure(figsize=(8, 6))
sns.scatterplot(x='Study_Hours', y='Exam_Score', data=df, color='dodgerblue')
sns.regplot(x='Study_Hours', y='Exam_Score', data=df, scatter=False, color='red')  # Optional trend line

# Titles and labels
plt.title('Study Hours vs Exam Scores')
plt.xlabel('Study Hours')
plt.ylabel('Exam Score')
```

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



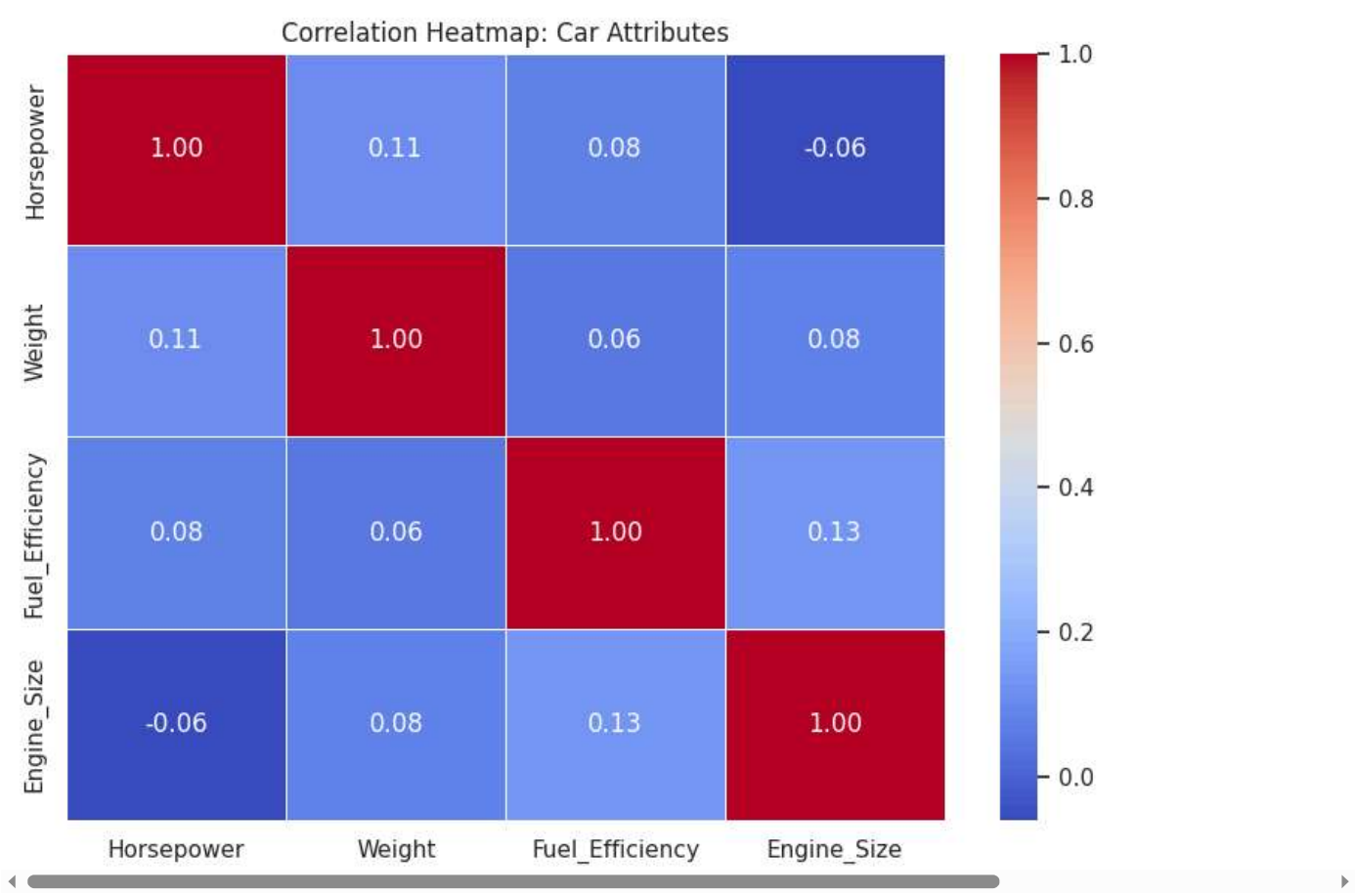

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Example data: replace with your real monthly values
data = {
    'Month': ['Jan', 'Feb', 'Mar'],
    'Ad_Budget': [1000, 1500, 2000],  # in dollars
    'Sales': [12000, 15500, 21000]    # in dollars
}
df = pd.DataFrame(data)

# Set style
sns.set(style="whitegrid")

# Scatter plot
plt.figure(figsize=(8, 5))
sns.scatterplot(x='Ad_Budget', y='Sales', data=df, s=100, color='darkgreen')

# Optional: add a regression line
sns.regplot(x='Ad_Budget', y='Sales', data=df, scatter=False, color='red')

# Labeling
plt.title('Advertising Budget vs. Sales')
plt.xlabel('Advertising Budget ($)')
plt.ylabel('Sales ($)')

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



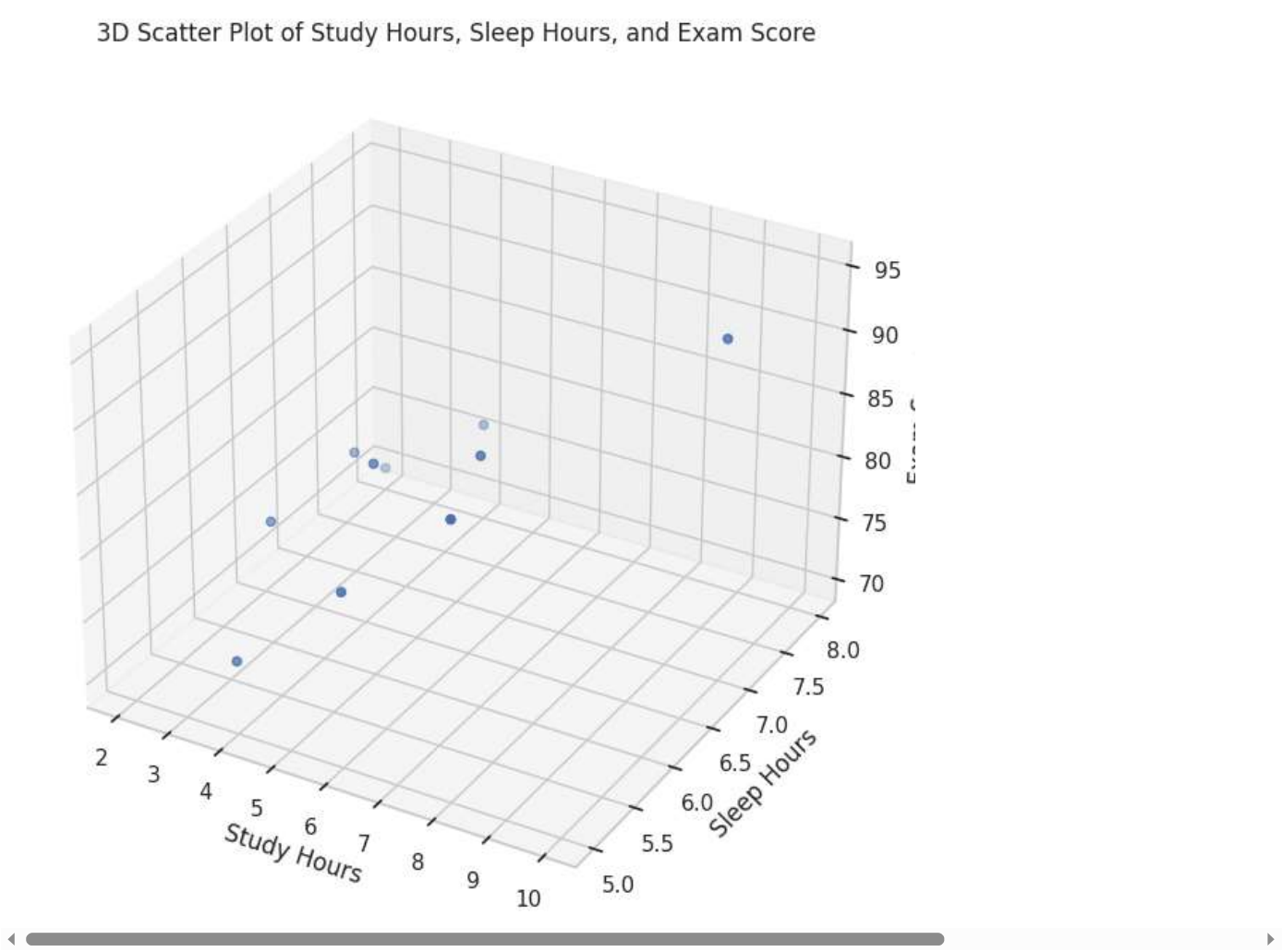

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

# Simulate sample data (replace this with your real dataset)
np.random.seed(0)
age = np.random.randint(20, 80, size=100)
blood_sugar = 70 + (age * 0.5) + np.random.normal(0, 10, size=100)  # Simulate a slight upward trend with noise
df = pd.DataFrame({'Age': age, 'Blood_Sugar': blood_sugar})

# Set style
sns.set(style="whitegrid")

# Create scatter plot
plt.figure(figsize=(8, 6))
sns.scatterplot(x='Age', y='Blood_Sugar', data=df, color='mediumblue')

# Optional: regression/trend line
sns.regplot(x='Age', y='Blood_Sugar', data=df, scatter=False, color='red')

# Titles and labels
plt.title('Age vs. Blood Sugar Level')
plt.xlabel('Age (years)')
plt.ylabel('Blood Sugar Level (mg/dL)')

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


```
from mpl_toolkits.mplot3d import Axes3D

# Set up 3D plot
fig = plt.figure(figsize=(10, 6))
ax = fig.add_subplot(111, projection='3d')

# Scatter plot
ax.scatter(df['Size_sqft'], df['Bedrooms'], df['Price'], c=df['Price'], cmap='viridis', s=50)

# Labels
ax.set_title('3D Scatter: Size vs Bedrooms vs Price')
ax.set_xlabel('Size (sqft)')
ax.set_ylabel('Bedrooms')
ax.set_zlabel('Price ($)')

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

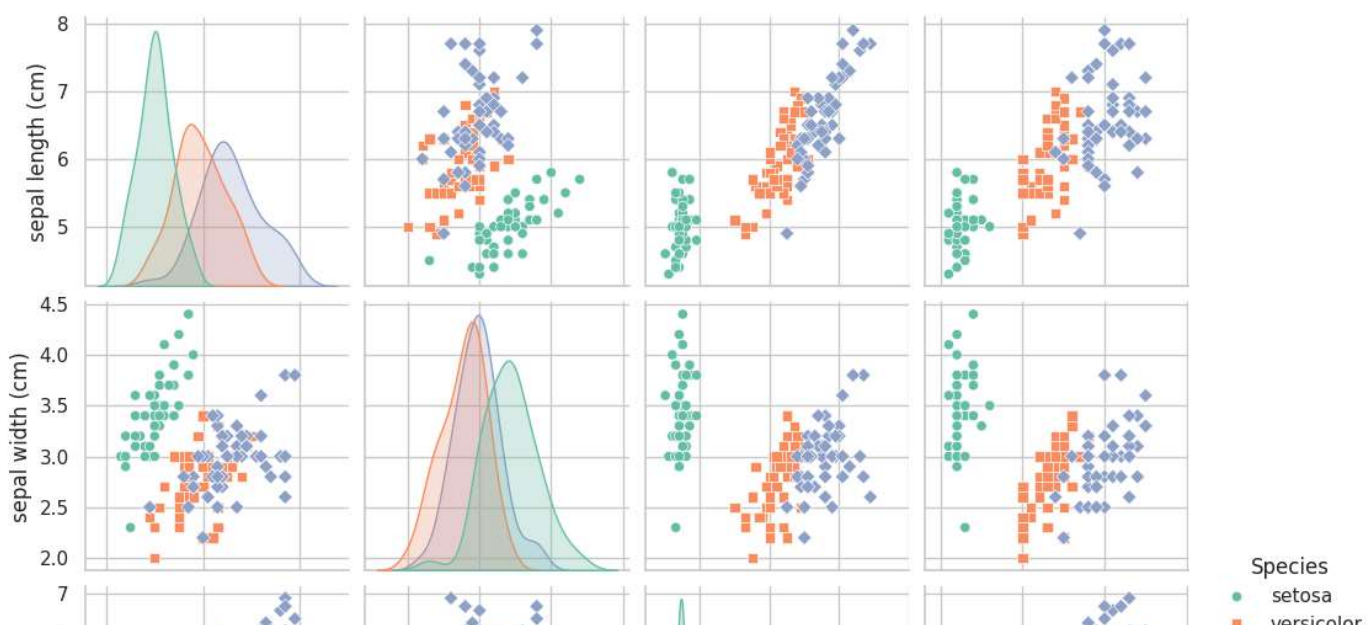

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

# Simulate car dataset
np.random.seed(0)
df = pd.DataFrame({
    'Horsepower': np.random.normal(150, 30, 100),
    'Weight': np.random.normal(3000, 400, 100),
    'Fuel_Efficiency': np.random.normal(25, 5, 100),
    'Engine_Size': np.random.normal(2.5, 0.5, 100)
})

# Calculate correlation matrix
corr_matrix = df.corr()

# Create heatmap
plt.figure(figsize=(8, 6))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', linewidths=0.5, fmt=".2f")
plt.title('Correlation Heatmap: Car Attributes')
plt.tight_layout()
plt.show()
```




```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import numpy as np
import pandas as pd

# Sample Data
data = {
    'Study Hours': [5, 3, 8, 4, 7, 6, 10, 2, 3, 4],
    'Sleep Hours': [6, 7, 5, 8, 6, 5, 7, 8, 6, 5],
    'Exam Score': [85, 78, 90, 76, 88, 82, 95, 70, 78, 74]
}

# Create a DataFrame
df = pd.DataFrame(data)

# Create a 3D scatter plot
fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(111, projection='3d')

# Scatter plot: Study Hours, Sleep Hours, Exam Score
ax.scatter(df['Study Hours'], df['Sleep Hours'], df['Exam Score'], color='b', marker='o')

# Labels and Title
ax.set_xlabel('Study Hours')
ax.set_ylabel('Sleep Hours')
ax.set_zlabel('Exam Score')
ax.set_title('3D Scatter Plot of Study Hours, Sleep Hours, and Exam Score')

plt.show()
```




```
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.datasets import load_iris
import pandas as pd

# Load the Iris dataset from sklearn
iris = load_iris()

# Convert it into a pandas DataFrame
df = pd.DataFrame(data=iris.data, columns=iris.feature_names)

# Add the target (species) as a new column
df['Species'] = iris.target
df['Species'] = df['Species'].map({0: 'setosa', 1: 'versicolor', 2: 'virginica'})

# Create a pairplot
sns.pairplot(df, hue='Species', markers=["o", "s", "D"], palette='Set2')

# Display the plot
plt.show()
```


sepal length (cm)
sepal width (cm)
8
7
6
5
4.5
4.0
3.5
3.0
2.5
2.0
Species
setosa