Experiment - 1

Exercise -1

```
import numpy as np

df = pd.read_csv('/content/student_performance.csv')

final_scores = df['Final_Score'].to_numpy()

print(final_scores)

mean_score = np.mean(final_scores)
median_score = np.median(final_scores)
std_score = np.std(final_scores)

print("Mean:", mean_score)
print("Median:", median_score)
print("Standard Deviation:", std_score)

min_score = np.min(final_scores)
max_score = np.max(final_scores)

normalized_scores = (final_scores - min_score) / (max_score - min_score)

normalized_scores
```

Output -

```
[52 57 60 64 68 71 74 77 79 83 63 70 75 56 69 73 80 58 72 78]
Mean: 68.95
Median: 70.5
Standard Deviation: 8.71478628538876
array([0.        , 0.16129032, 0.25806452, 0.38709677, 0.51612903,
       0.61290323, 0.70967742, 0.80645161, 0.87096774, 1.        ,
       0.35483871, 0.58064516, 0.74193548, 0.12903226, 0.5483871 ,
       0.67741935, 0.90322581, 0.19354839, 0.64516129, 0.83870968])
```

Exercise - 2

```python
import pandas as pd
df = pd.read_csv('/content/student_performance.csv')
df.head()
print("Shape:", df.shape)

print("Columns:", df.columns)

print("\nMissing Values:\n", df.isnull().sum())
def performance_label(score):
    if score >= 75:
        return "Excellent"
    elif score >= 60:
        return "Good"
    else:
        return "Average"

df['Performance'] = df['Final_Score'].apply(performance_label)

df.head()
```

Output -

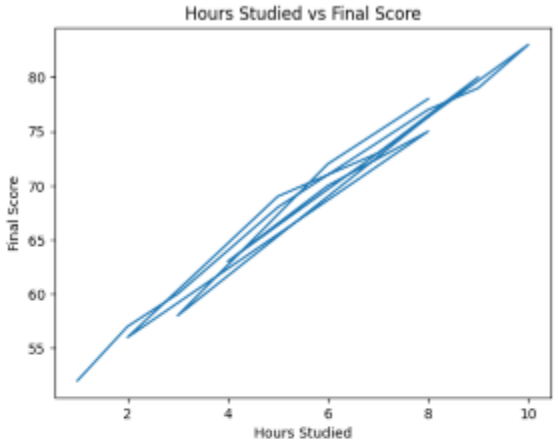
```
Shape: (20, 5)
Columns: Index(['Hours_Studied', 'Attendance', 'Assignment_Score', 'Midterm_Score',
       'Final_Score'],
      dtype='object')

Missing Values:
 Hours_Studied       0
Attendance          0
Assignment_Score    0
Midterm_Score       0
Final_Score         0
dtype: int64
```

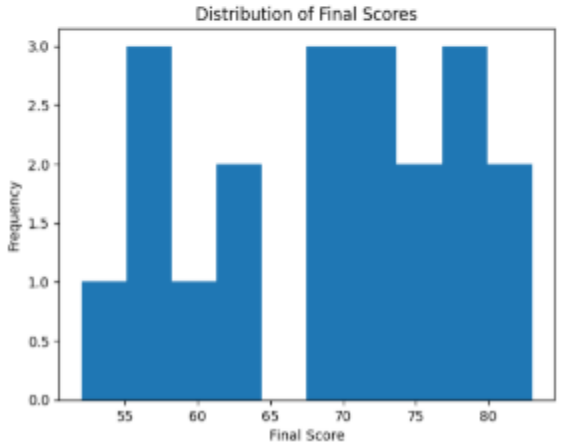
	Hours_Studied	Attendance	Assignment_Score	Midterm_Score	Final_Score	Performance
0	1	60	55	50	52	Average
1	2	65	58	55	57	Average
2	3	70	60	58	60	Good
3	4	75	65	62	64	Good
4	5	80	68	65	68	Good

Exercise 3 -

```
import matplotlib.pyplot as plt
plt.figure()
plt.plot(df['Hours_Studied'], df['Final_Score'])
plt.xlabel('Hours Studied')
plt.ylabel('Final Score')
plt.title('Hours Studied vs Final Score')
plt.show()

plt.figure()
plt.hist(df['Final_Score'])
plt.xlabel('Final Score')
plt.ylabel('Frequency')
plt.title('Distribution of Final Scores')
plt.show()
```

Output -

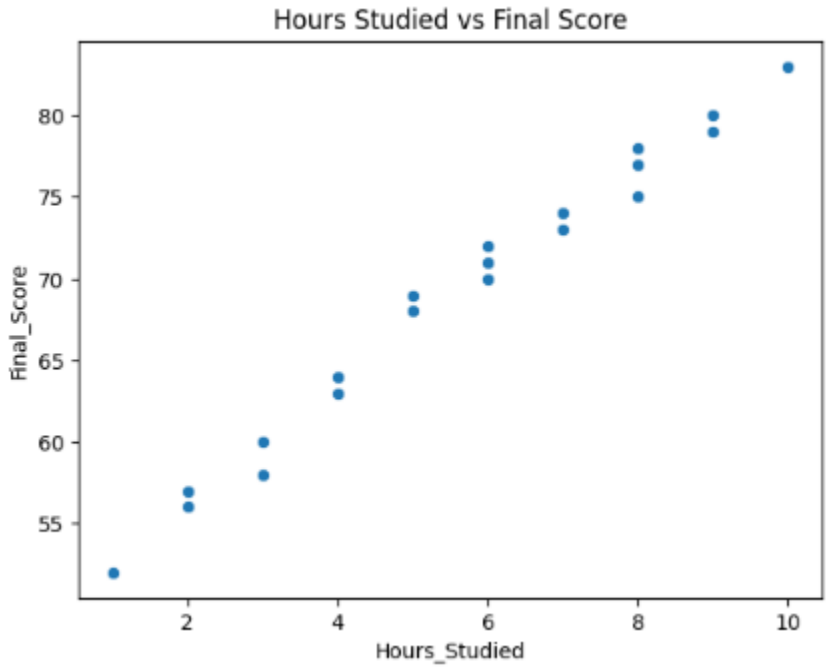


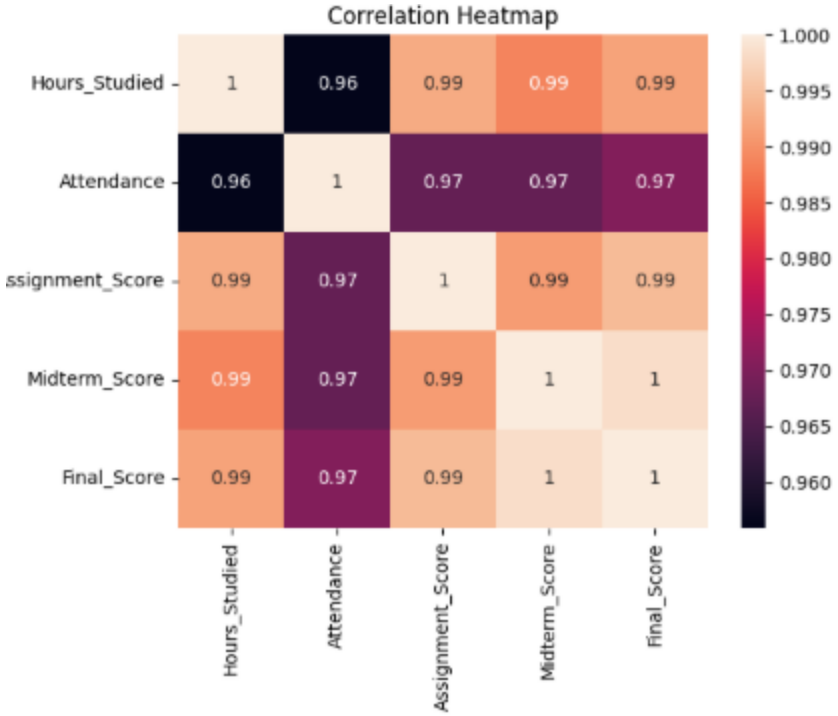


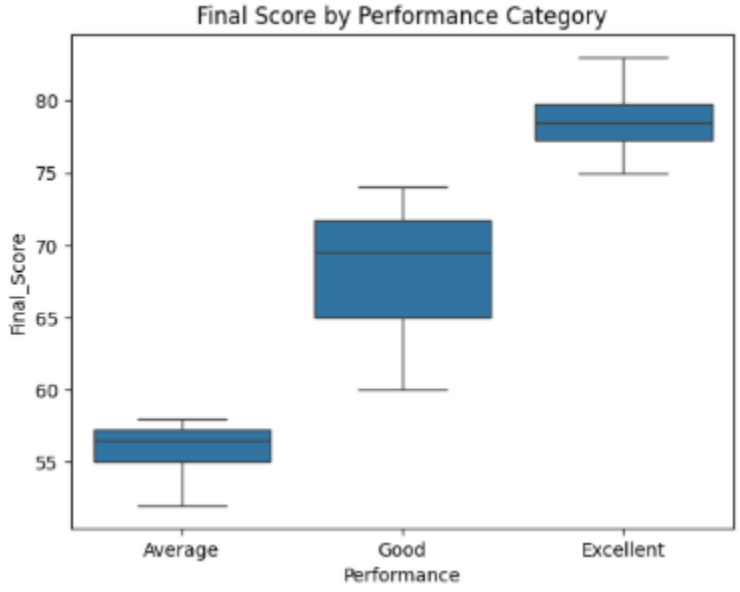
Exercise - 4

```
import seaborn as sns

sns.scatterplot(
    x='Hours_Studied',
    y='Final_Score',
    data=df
)
plt.title('Hours Studied vs Final Score')
plt.show()

correlation_matrix = df.corr(numeric_only=True)

sns.heatmap(
    correlation_matrix,
    annot=True
)
plt.title('Correlation Heatmap')
plt.show()

sns.boxplot(
    x='Performance',
    y='Final_Score',
    data=df
)
plt.title('Final Score by Performance Category')
plt.show()
```

Output -




Correlation Heatmap
Hours_Studied
Attendance
ssignment_Score
Midterm_Score
Final_Score
1
0.96
0.99
0.99
0.99
0.96
1
0.97
0.97
0.97
0.99
0.97
1
0.99
0.99
0.99
0.97
0.99
1
1
0.99
0.97
0.99
1
1
Hours_Studied
Attendance
Assignment_Score
Midterm_Score
Final_Score
1.000
0.995
0.990
0.985
0.980
0.975
0.970
0.965
0.960


Final Score by Performance Category
Final_Score
80
75
70
65
60
55
Average
Good
Excellent
Performance