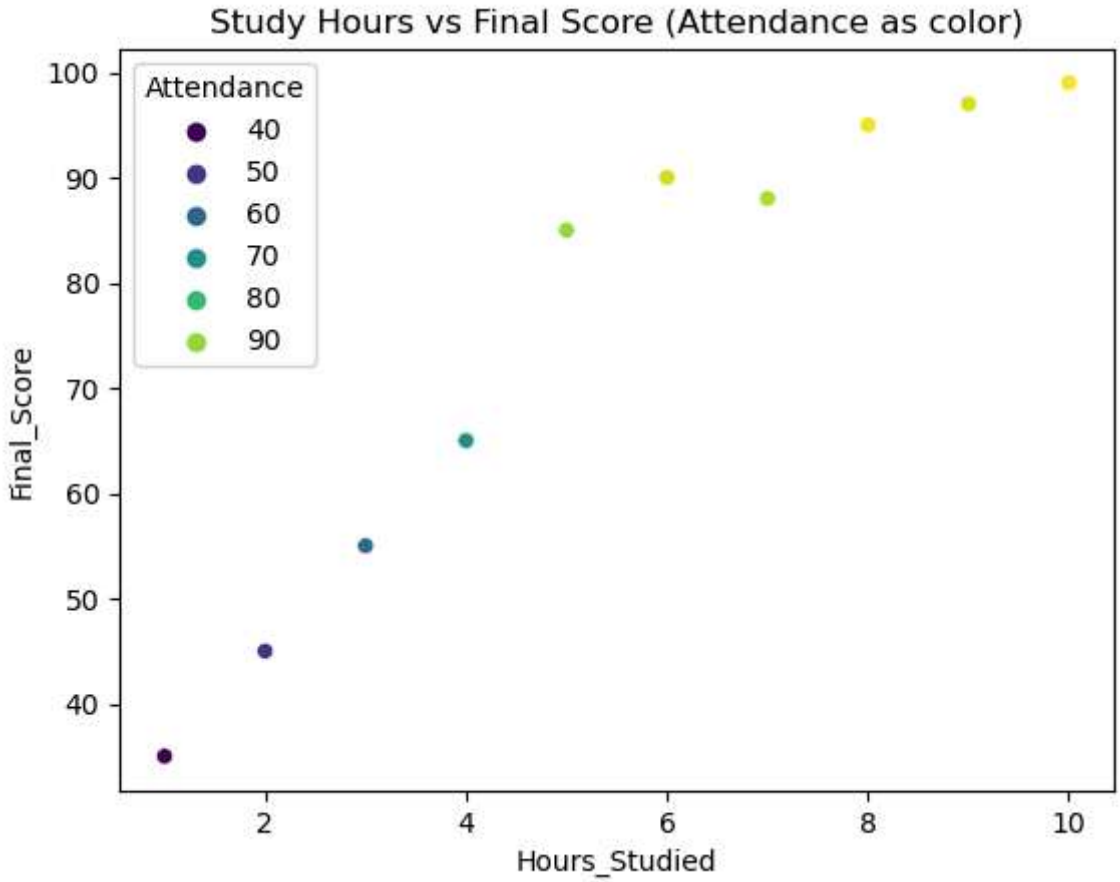
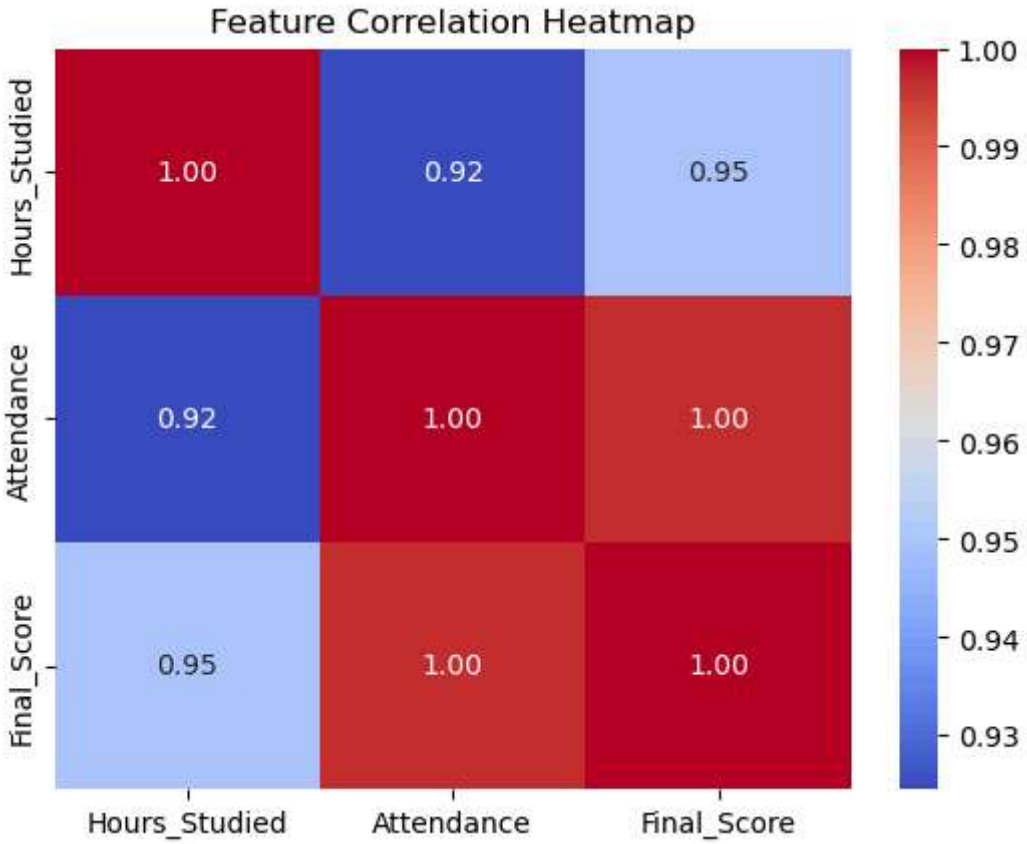
In [69]:
```
import os
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Make sure output folder exists
outdir = "outputs"
os.makedirs(outdir, exist_ok=True)

# Load data (either from CSV or manual sample)
df = pd.read_csv(r"C:\Users\mdasf\Downloads\student-score-prediction-github\s

# Scatter plot
plt.figure()
sns.scatterplot(
    x="Hours_Studied",
    y="Final_Score",
    hue="Attendance",
    data=df
)
plt.title("Study Hours vs Final Score (Attendance as color)")
plt.savefig(os.path.join(outdir, "scatter_hours_vs_score.png"), bbox_inches='
plt.close()
visualize(df)
model, r2, mae = train_and_evaluate(df)
prediction = predict(model, hours=4, attendance=80)

print("=== Model Performance ===")
print(f"R² Score: {r2:.4f}")
print(f"Mean Absolute Error: {mae:.4f}")
print("\n=== Example Prediction ===")
print(f"Predicted score for 4 study hours & 80% attendance: {prediction}")

if __name__ == "__main__":
    main()
```

```
=== Model Performance ===
R² Score: 0.9938
Mean Absolute Error: 1.2496

=== Example Prediction ===
Predicted score for 4 study hours & 80% attendance: 74.22

C:\Users\mdasf\anaconda3\Lib\site-packages\sklearn\base.py:464: UserWarnin
g: X does not have valid feature names, but LinearRegression was fitted wit
h feature names
  warnings.warn(
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[69], line 35
     32 print(f"Predicted score for 4 study hours & 80% attendance: {predic
tion}")
     34 if __name__ == "__main__":
---> 35     main()
     36 plt.show()

Cell In[56], line 61, in main()
     59 def main():
     60     df = load_data()
---> 61     df = clean_data(df)
     63     visualize(df)
     65     model, r2, mae = train_and_evaluate(df)

Cell In[55], line 20, in clean_data(df)
     19 def clean_data(df):
---> 20     df = df.dropna(subset=["Hours_Studied", "Attendance", "Final_Sc
ore"])
     21     df = df[(df["Hours_Studied"] >= 0) &
     22             (df["Attendance"].between(0, 100)) &
     23             (df["Final_Score"].between(0, 100))]
     24     return df.reset_index(drop=True)

AttributeError: 'NoneType' object has no attribute 'dropna'
```

In [72]:
```
import os
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Make sure output folder exists
outdir = "outputs"
os.makedirs(outdir, exist_ok=True)

# Sample dataset (replace with your CSV if you have one)
df = pd.read_csv(r"C:\Users\mdasf\Downloads\student-score-prediction-github\s

# --- Scatter Plot ---
plt.figure()
sns.scatterplot(
    x="Hours_Studied",
    y="Final_Score",
    hue="Attendance",
    data=df,
    palette="viridis"
)
plt.title("Study Hours vs Final Score (Attendance as color)")
plt.savefig(os.path.join(outdir, "scatter_hours_vs_score.png"), bbox_inches='
plt.show()

# --- Correlation Heatmap ---
plt.figure()
corr = df[["Hours_Studied", "Attendance", "Final_Score"]].corr()
sns.heatmap(corr, annot=True, cmap="coolwarm", fmt=".2f")
plt.title("Feature Correlation Heatmap")
plt.savefig(os.path.join(outdir, "correlation_heatmap.png"), bbox_inches="tig
plt.show()
```


Study Hours vs Final Score (Attendance as color)
Attendance
40
50
60
70
80
90
Final_Score
100
90
80
70
60
50
40
2
4
6
8
10
Hours_Studied


Feature Correlation Heatmap
Hours_Studied
Attendance
Final_Score
1.00
0.92
0.95
0.92
1.00
1.00
0.95
1.00
1.00
Hours_Studied
Attendance
Final_Score
1.00
0.99
0.98
0.97
0.96
0.95
0.94
0.93