

```
In [1]: pip install pandas matplotlib seaborn

Requirement already satisfied: pandas in c:\users\ninja\anaconda3\lib\site-packages (2.2.2)
Requirement already satisfied: matplotlib in c:\users\ninja\anaconda3\lib\site-packages (3.8.4)
Requirement already satisfied: seaborn in c:\users\ninja\anaconda3\lib\site-packages (0.13.2)
Requirement already satisfied: numpy<1.26.0 in c:\users\ninja\anaconda3\lib\site-packages (from pandas) (1.26.4)
Requirement already satisfied: python-dateutil<2.8.2 in c:\users\ninja\anaconda3\lib\site-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\ninja\anaconda3\lib\site-packages (from pandas) (2024.1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\ninja\anaconda3\lib\site-packages (from pandas) (2023.3)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\ninja\anaconda3\lib\site-packages (from matplotlib) (1.2.0)
Requirement already satisfied: cycler>=0.10 in c:\users\ninja\anaconda3\lib\site-packages (from matplotlib) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\ninja\anaconda3\lib\site-packages (from matplotlib) (4.51.0)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\ninja\anaconda3\lib\site-packages (from matplotlib) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\ninja\anaconda3\lib\site-packages (from matplotlib) (23.2)
Requirement already satisfied: pillow<=8 in c:\users\ninja\anaconda3\lib\site-packages (from matplotlib) (10.3.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\ninja\anaconda3\lib\site-packages (from matplotlib) (3.0.9)
Requirement already satisfied: six>=1.5 in c:\users\ninja\anaconda3\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

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

```
In [11]: pd.read_csv("C:\Users\ninja\Downloads\StudentPerformanceFactors.csv")

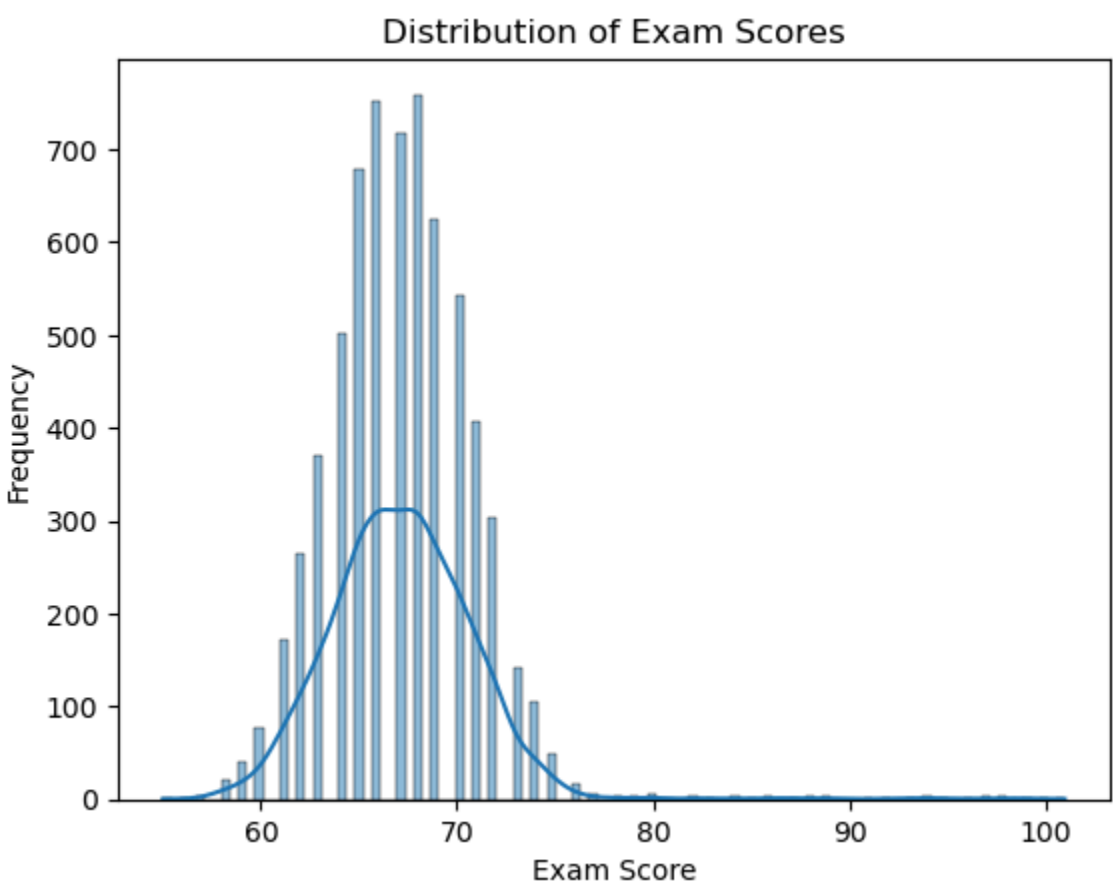
Out[11]:
```

	Hours_Studied	Attendance	Parental_Involvement	Access_to_Resources	Extracurricular_Activities	Sleep_Hours	Previous_Scores	Motivation_Level	Internet_Access	Tutoring_Sessions	Family_Income	Teacher_Quality	School_Type	Peer_Influence	Physical_Activity	Learning_Disabilities	Parental_Education_Level	Distance_from_Home	Gender	Exam_Score
0	23	84	Low	High	No	7	73	Low	Yes	0	Low	Medium	Public	Positive	3	No	High School	Near	Male	67
1	19	64	Low	Medium	No	8	59	Low	Yes	2	Medium	Medium	Public	Negative	4	No	College	Moderate	Female	61
2	24	98	Medium	Medium	Yes	7	91	Medium	Yes	2	Medium	Medium	Public	Neutral	4	No	Postgraduate	Near	Male	74
3	29	89	Low	Medium	Yes	8	98	Medium	Yes	1	Medium	Medium	Public	Negative	4	No	High School	Moderate	Male	71
4	19	92	Medium	Medium	Yes	6	65	Medium	Yes	3	Medium	High	Public	Neutral	4	No	College	Near	Female	70
...
6602	25	69	High	Medium	No	7	76	Medium	Yes	1	High	Medium	Public	Positive	2	No	High School	Near	Female	68
6603	23	76	High	Medium	No	8	81	Medium	Yes	3	Low	High	Public	Positive	2	No	High School	Near	Female	69
6604	20	90	Medium	Low	Yes	6	65	Low	Yes	3	Low	Medium	Public	Negative	2	No	Postgraduate	Near	Female	68
6605	10	86	High	High	Yes	6	91	High	Yes	2	Low	Medium	Private	Positive	3	No	High School	Far	Female	68
6606	15	67	Medium	Low	Yes	9	94	Medium	Yes	0	Medium	Medium	Public	Positive	4	No	Postgraduate	Near	Male	64
6607 rows × 20 columns																				

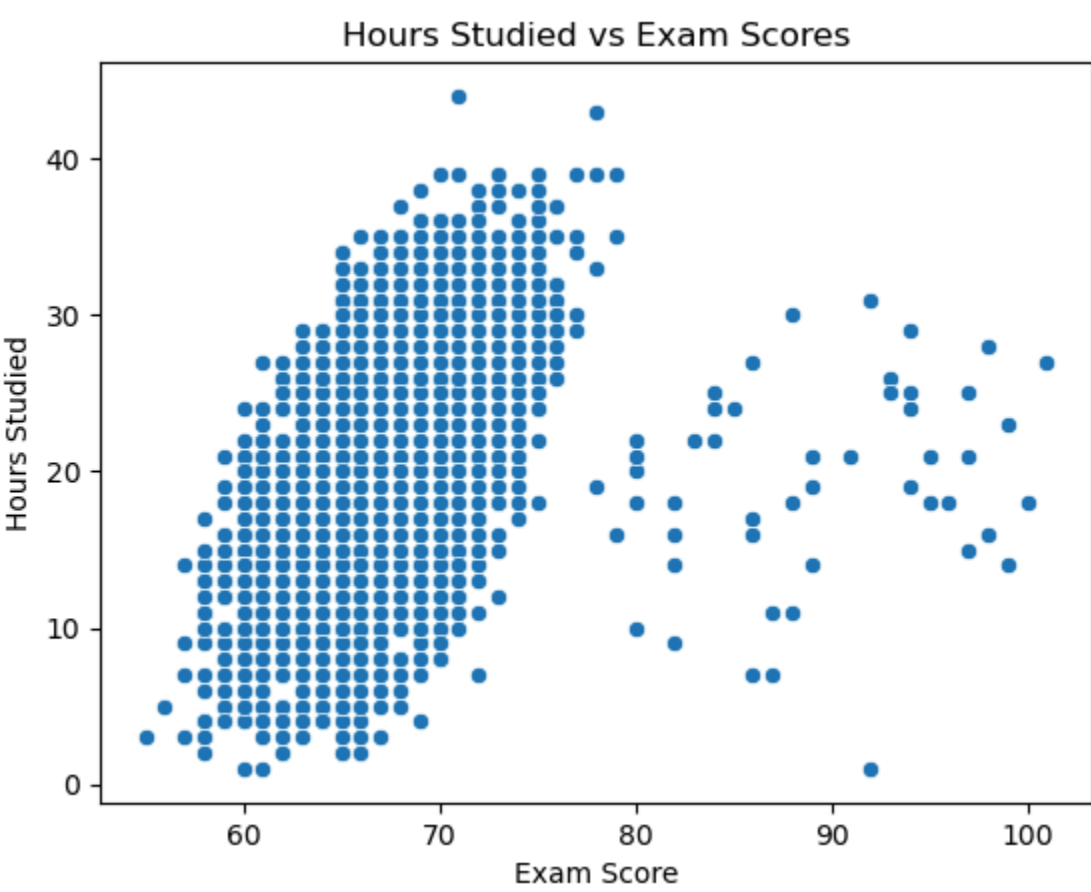
```
In [13]: df.head()
```

	Hours_Studied	Attendance	Parental_Involvement	Access_to_Resources	Extracurricular_Activities	Sleep_Hours	Previous_Scores	Motivation_Level	Internet_Access	Tutoring_Sessions	Family_Income	Teacher_Quality	School_Type	Peer_Influence	Physical_Activity	Learning_Disabilities	Parental_Education_Level	Distance_from_Home	Gender	Exam_Score
0	23	84	Low	High	No	7	73	Low	Yes	0	Low	Medium	Public	Positive	3	No	High School	Near	Male	67
1	19	64	Low	Medium	No	8	59	Low	Yes	2	Medium	Medium	Public	Negative	4	No	College	Moderate	Female	61
2	24	98	Medium	Medium	Yes	7	91	Medium	Yes	2	Medium	Medium	Public	Neutral	4	No	Postgraduate	Near	Male	74
3	29	89	Low	Medium	Yes	8	98	Medium	Yes	1	Medium	Medium	Public	Negative	4	No	High School	Moderate	Male	71
4	19	92	Medium	Medium	Yes	6	65	Medium	Yes	3	Medium	High	Public	Neutral	4	No	College	Near	Female	70

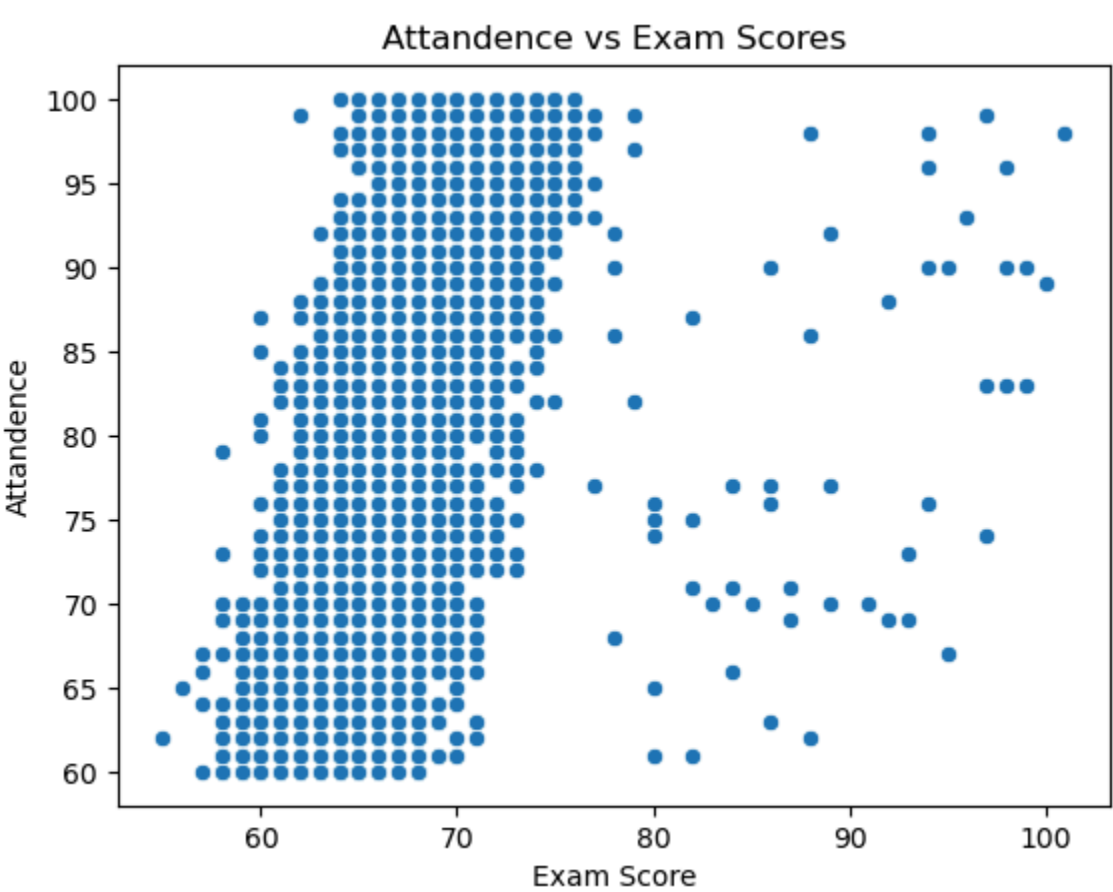
```
In [19]: sns.histplot(df['Exam_Score'], kde=True)
plt.title('Distribution of Exam Scores')
plt.xlabel('Exam Score')
plt.ylabel('Frequency')
plt.show()
```



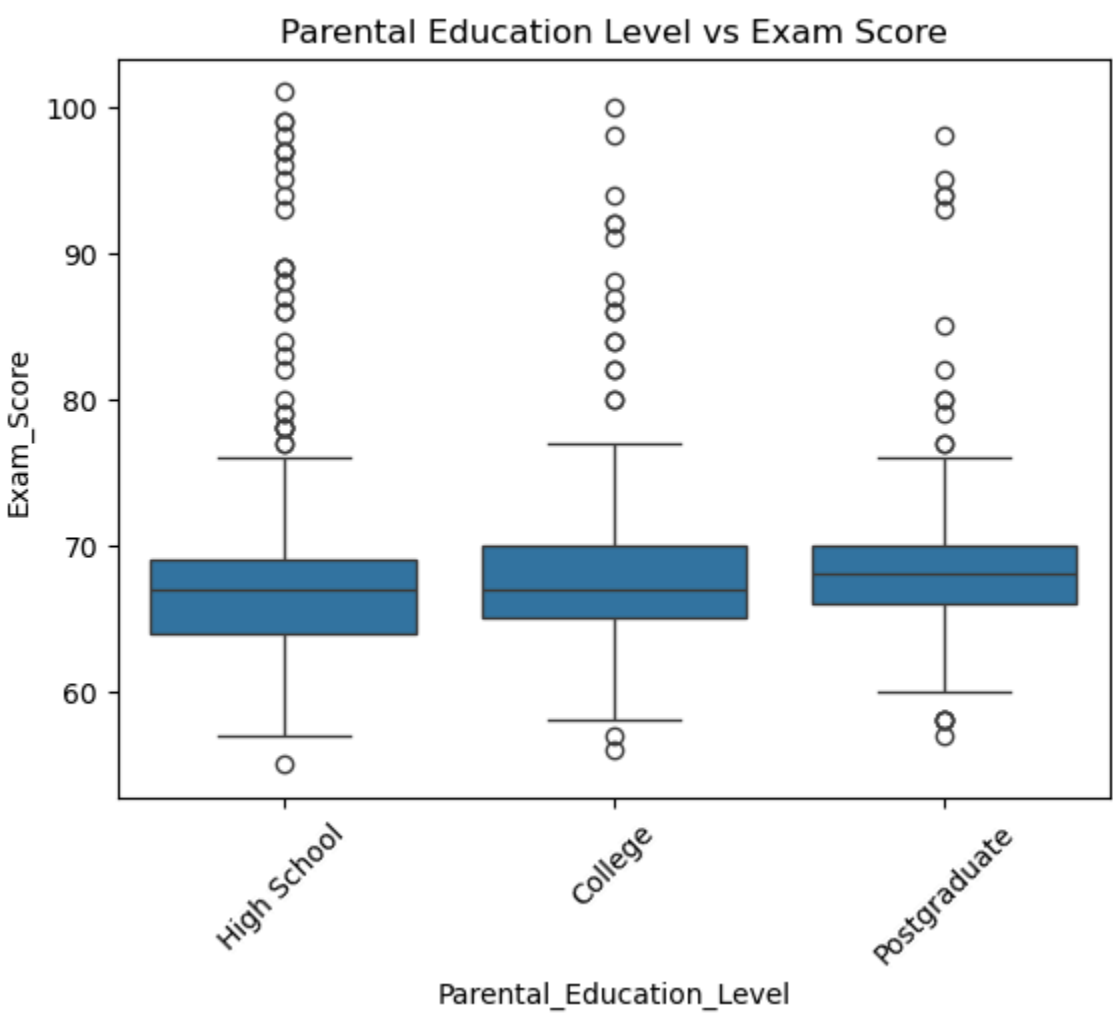
```
In [33]: sns.scatterplot(x='Exam_Score', y='Hours_Studied', data=df)
plt.title('Hours Studied vs Exam Scores')
plt.xlabel('Exam Score')
plt.ylabel('Hours Studied')
plt.show()
```



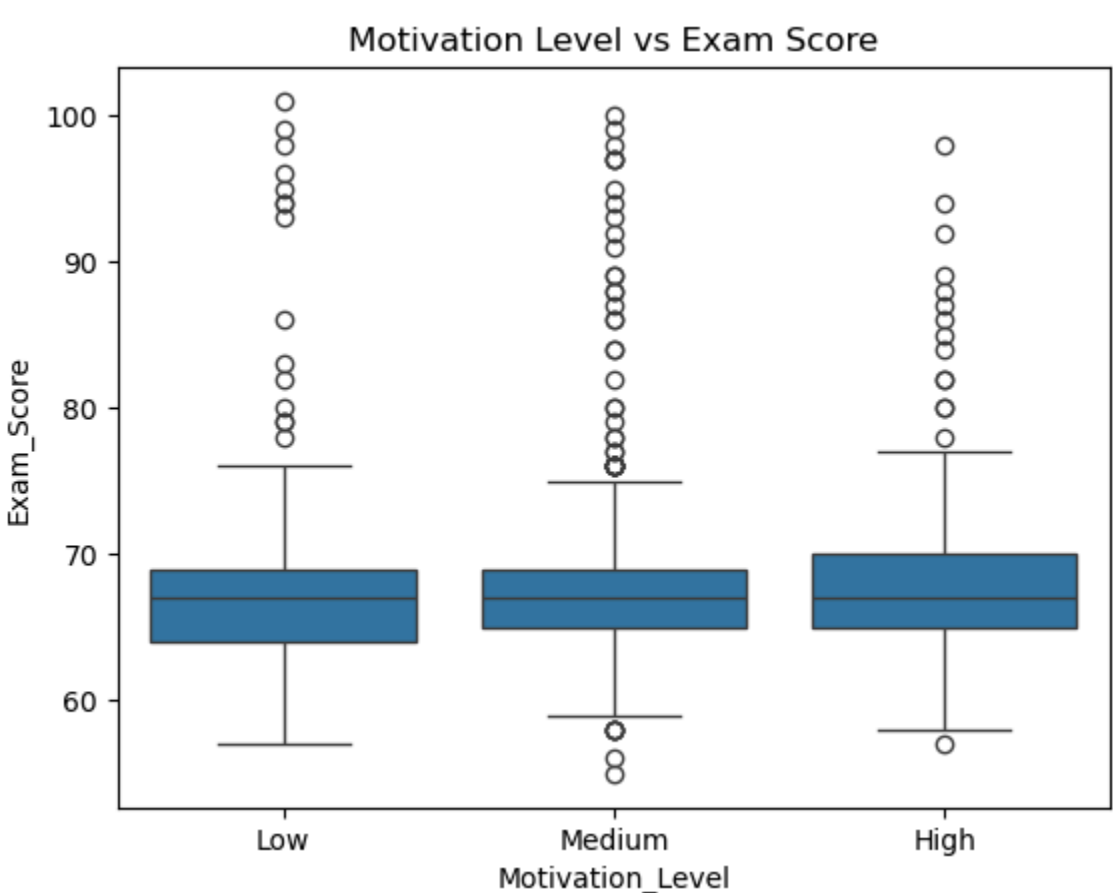
```
In [37]: sns.scatterplot(x='Exam_Score', y='Attendance', data=df)
plt.title('Attendance vs Exam Scores')
plt.xlabel('Exam Score')
plt.ylabel('Attendance')
plt.show()
```



```
In [39]: sns.boxplot(x='Parental_Education_Level', y='Exam_Score', data=df)
plt.title('Parental Education Level vs Exam Score')
plt.xticks(rotation=45)
plt.show()
```



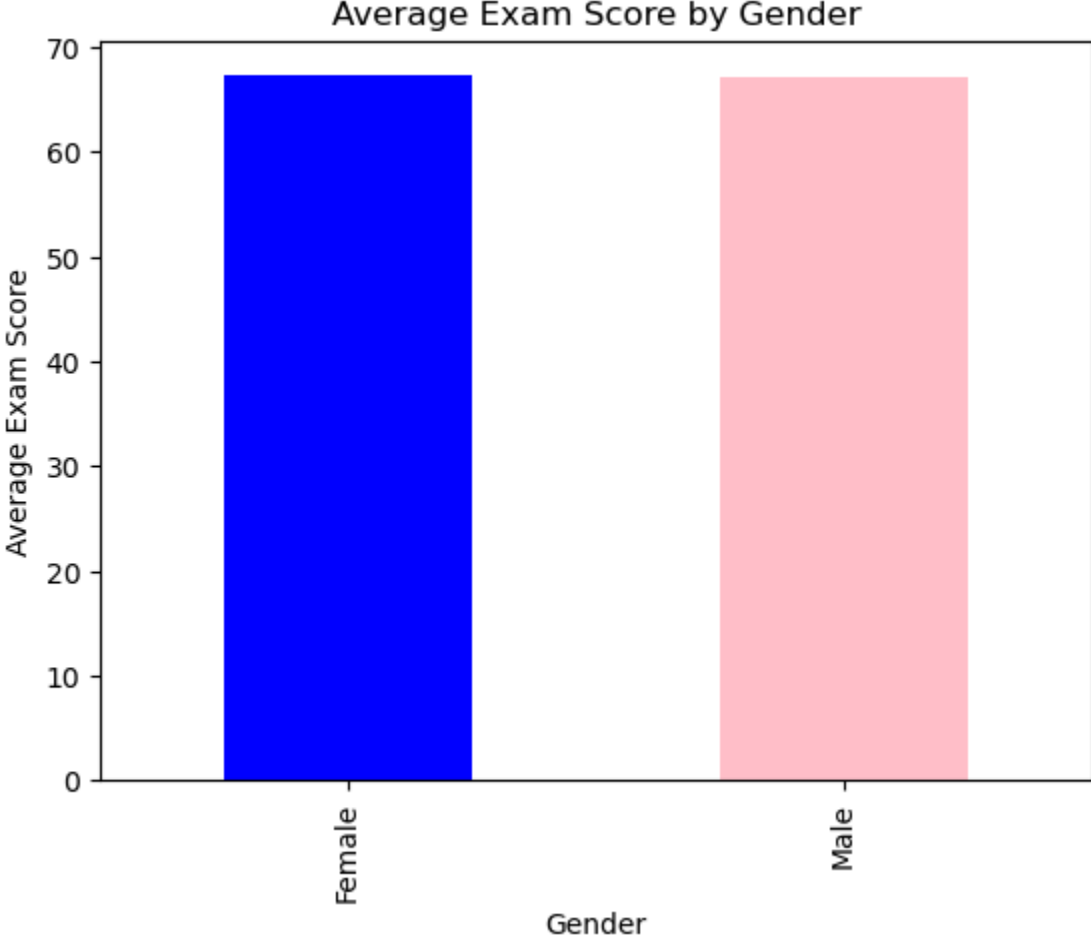
```
In [41]: sns.boxplot(x='Motivation_Level', y='Exam_Score', data=df)
plt.title('Motivation Level vs Exam Score')
plt.show()
```



```
In [47]: # Group by gender and calculate average exam score
gender_group = df.groupby('Gender')['Exam_Score'].mean()
print(gender_group)

# Plot the results
gender_group.plot(kind='bar', color=['blue', 'pink'])
plt.title('Average Exam Score by Gender')
plt.ylabel('Average Exam Score')
plt.show()

Gender
Female    67.244898
Male     67.228894
Name: Exam_Score, dtype: float64
```



```
In [49]: df['Attendance'].describe()
```

```
Out[49]: count    6607.000000
mean       79.977448
std        11.547475
min         60.000000
25%        70.000000
50%        80.000000
75%        90.000000
max        100.000000
Name: Attendance, dtype: float64
```

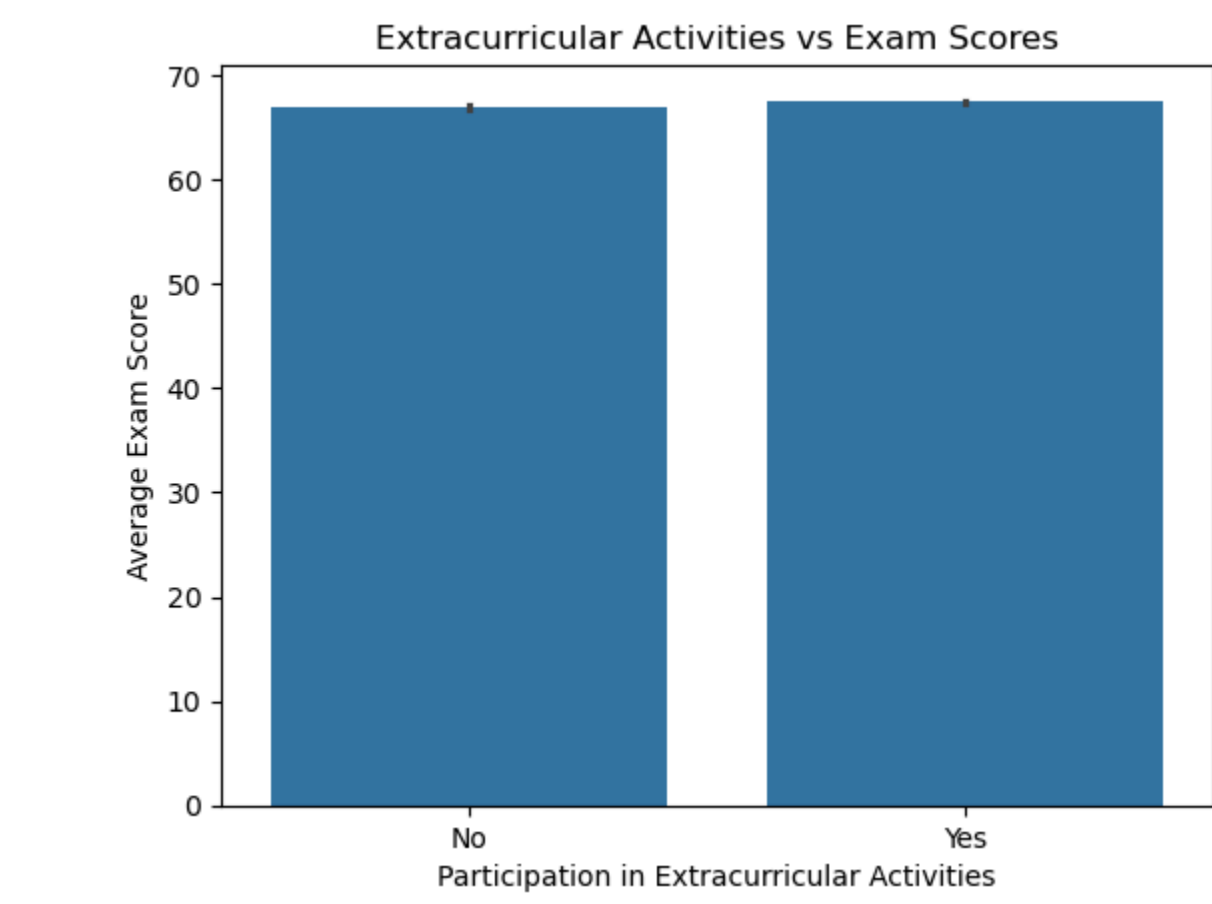
```
In [51]: df['Attendance'].unique()
```

```
Out[51]: array([ 84,  64,  98,  89,  92,  88,  78,  94,  80,  97,  83,  82,  68,
        60,  70,  75,  99,  74,  65,  62,  91,  90,  66,  69,  72,  63,
        61,  86,  77,  71,  67,  87,  73,  96, 100,  81,  95,  79,  85,
        76,  93], dtype=int64)
```

```
In [53]: sns.barplot(x='Extracurricular_Activities', y='Exam_Score', data=df)

# Add labels and title
plt.title('Extracurricular Activities vs Exam Scores')
plt.xlabel('Participation in Extracurricular Activities')
plt.ylabel('Average Exam Score')

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



In []: