Importing Libraries

In [2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
```

In [4]:
```
pd.set_option("display.max_columns",None)
```

In [7]:
```
# reading dataset
df=pd.read_csv("C:\\Users\\DILEEP  V\\OneDrive\\Desktop\\Data_Science_Projec
```

In [8]:
```
# first five rows of dataset
```

In [16]:
```
df.head()
```

Out[16]:

	student_id	name	gender	age	grade_level	math_score	reading_score	writing_score	atten
0	S1	Student_1	Other	17	10	74	61	90	
1	S2	Student_2	Male	17	12	99	70	91	
2	S3	Student_3	Other	17	9	59	60	99	
3	S4	Student_4	Other	17	12	70	88	69	
4	S5	Student_5	Male	15	9	85	77	94	

In [12]:
```
# last five records of dataset
```

In [17]:
```
1 df.tail()
```

Out[17]:

	student_id	name	gender	age	grade_level	math_score	reading_score	writing_score
995	S996	Student_996	Female	15	10	76	75	55
996	S997	Student_997	Female	17	12	83	68	98
997	S998	Student_998	Other	16	10	60	77	92
998	S999	Student_999	Other	17	9	94	66	97
999	S1000	Student_1000	Male	17	9	96	92	93

In [18]:
```
1 # dataset information
```

In [19]:
```
1 df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 15 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   student_id        1000 non-null   object 
 1   name              1000 non-null   object 
 2   gender            1000 non-null   object 
 3   age               1000 non-null   int64  
 4   grade_level       1000 non-null   int64  
 5   math_score        1000 non-null   int64  
 6   reading_score     1000 non-null   int64  
 7   writing_score     1000 non-null   int64  
 8   attendance_rate   1000 non-null   float64
 9   parent_education  1000 non-null   object 
 10  study_hours       1000 non-null   float64
 11  internet_access   1000 non-null   object 
 12  lunch_type        1000 non-null   object 
 13  extra_activities  1000 non-null   object 
 14  final_result      1000 non-null   object 
dtypes: float64(2), int64(5), object(8)
memory usage: 117.3+ KB
```

In [20]:
```
1 # dataset desciption
```

In [21]:
```
1 df.describe()
```

Out[21]:

	age	grade_level	math_score	reading_score	writing_score	attendance_rate	study_
count	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000	1000.(
mean	15.999000	10.473000	75.165000	74.293000	75.150000	89.879179	2.9
std	0.817109	1.136029	14.304349	14.312652	14.395094	5.725007	1.1
min	15.000000	9.000000	50.000000	50.000000	50.000000	80.000614	1.(
25%	15.000000	9.000000	63.000000	62.000000	63.000000	84.971772	1.9
50%	16.000000	10.000000	75.000000	74.000000	75.000000	89.980889	2.9
75%	17.000000	12.000000	88.000000	86.000000	88.000000	94.629778	3.9
max	17.000000	12.000000	99.000000	99.000000	99.000000	99.954988	4.9

In [22]:
```
1 # checking for null values
```

In [24]:
```
1 df.isnull().sum()
```

Out[24]:
```
student_id          0
name                0
gender              0
age                 0
grade_level         0
math_score          0
reading_score       0
writing_score       0
attendance_rate     0
parent_education    0
study_hours         0
internet_access     0
lunch_type          0
extra_activities    0
final_result        0
dtype: int64
```

In [25]:
```
1 # checking for duplicated rows
```

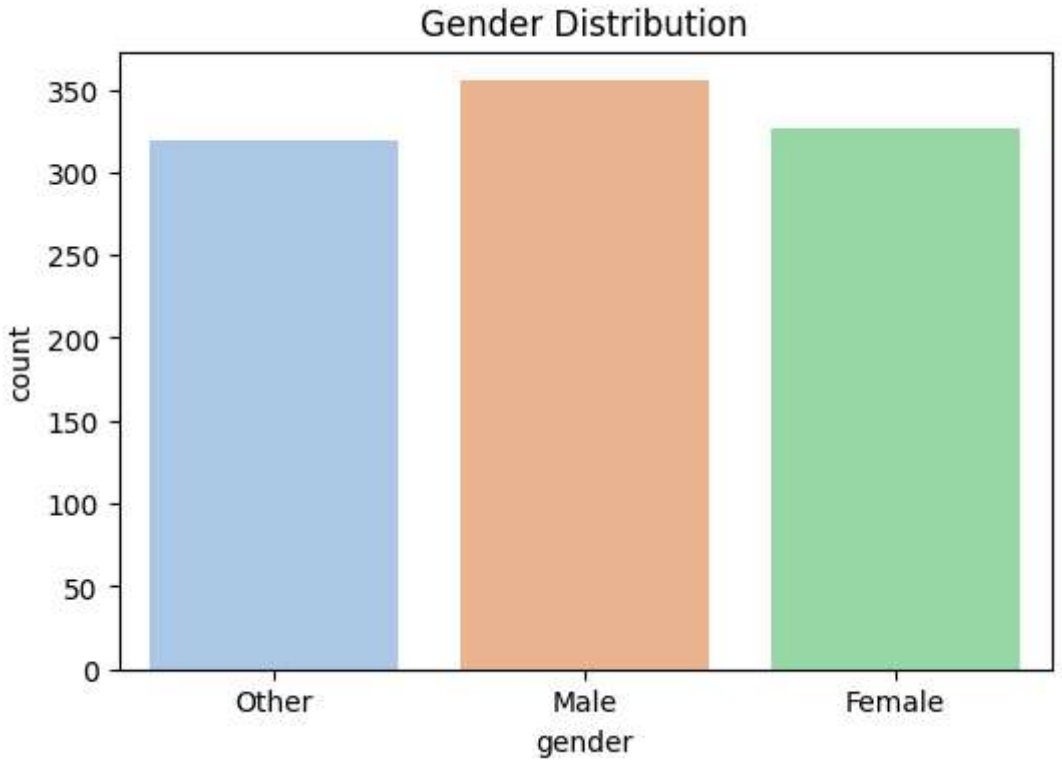
In [31]:
```
1 df.duplicated().sum()
```

Out[31]:
```
0
```

In [32]:
```
1 # gender distribution
```

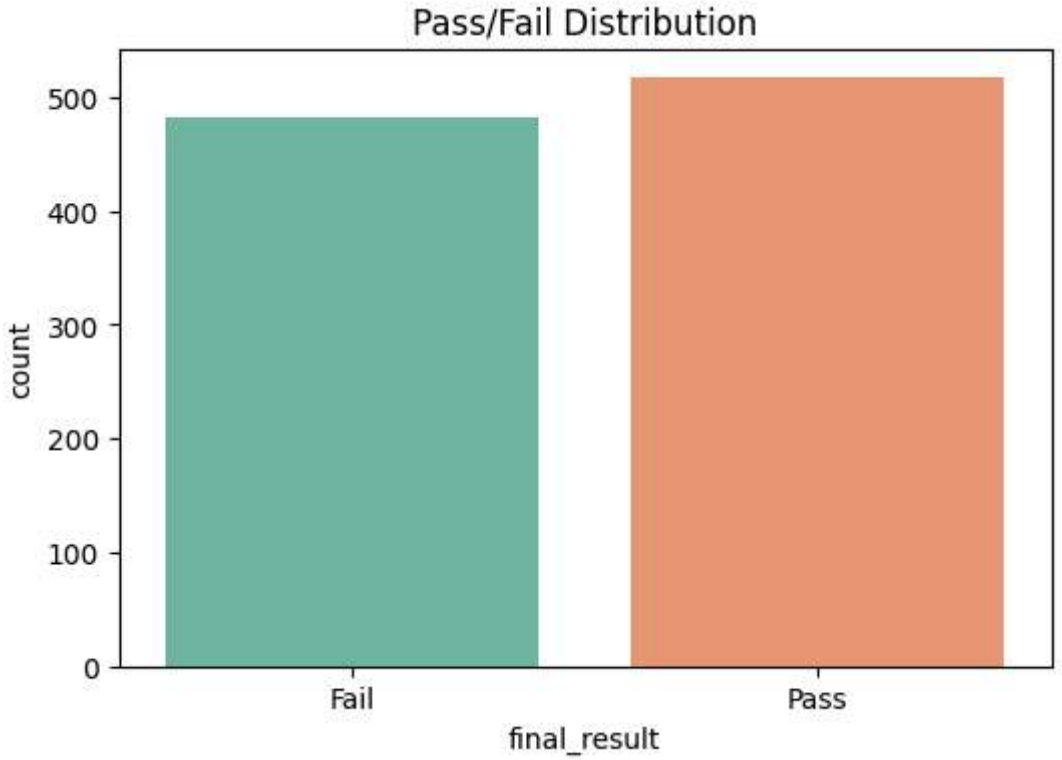
In [35]:

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

plt.figure(figsize=(6, 4))
sns.countplot(data=df, x="gender", hue="gender", palette="pastel", legend=Fa
plt.title("Gender Distribution")
plt.show()

```

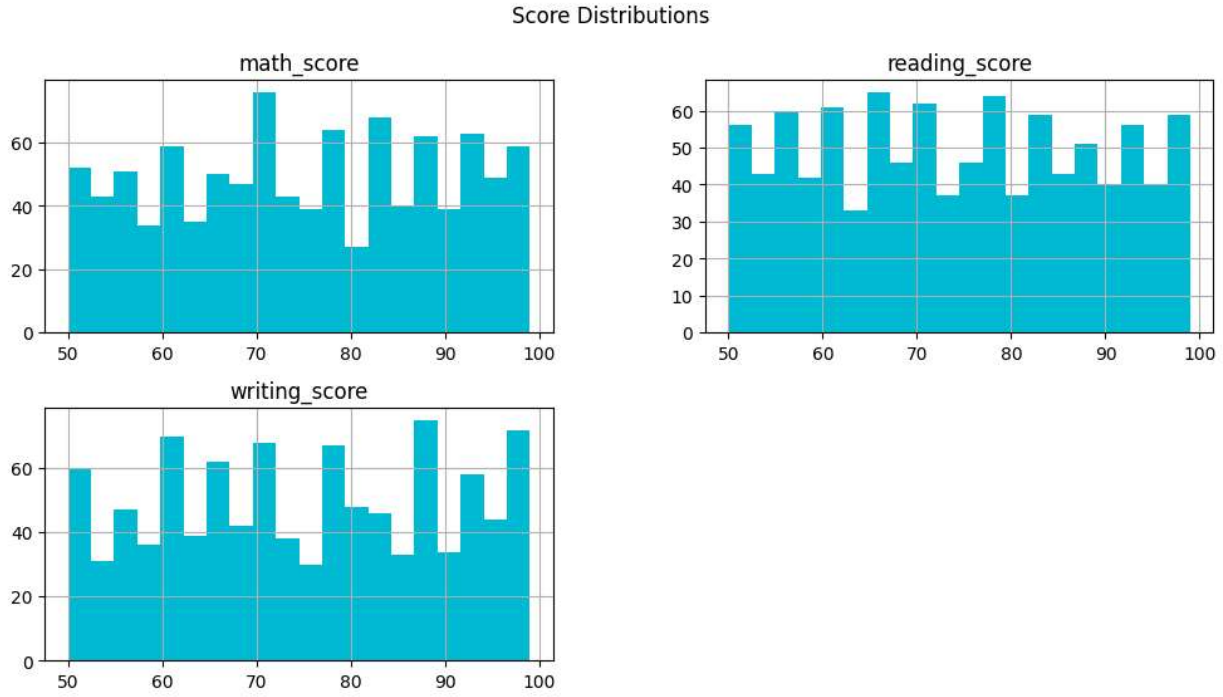


In [37]:

```
# Final Result Distribution
import matplotlib.pyplot as plt
import seaborn as sns

plt.figure(figsize=(6, 4))
sns.countplot(data=df, x="final_result", hue="final_result", palette="Set2",
plt.title("Pass/Fail Distribution")
plt.show()
```

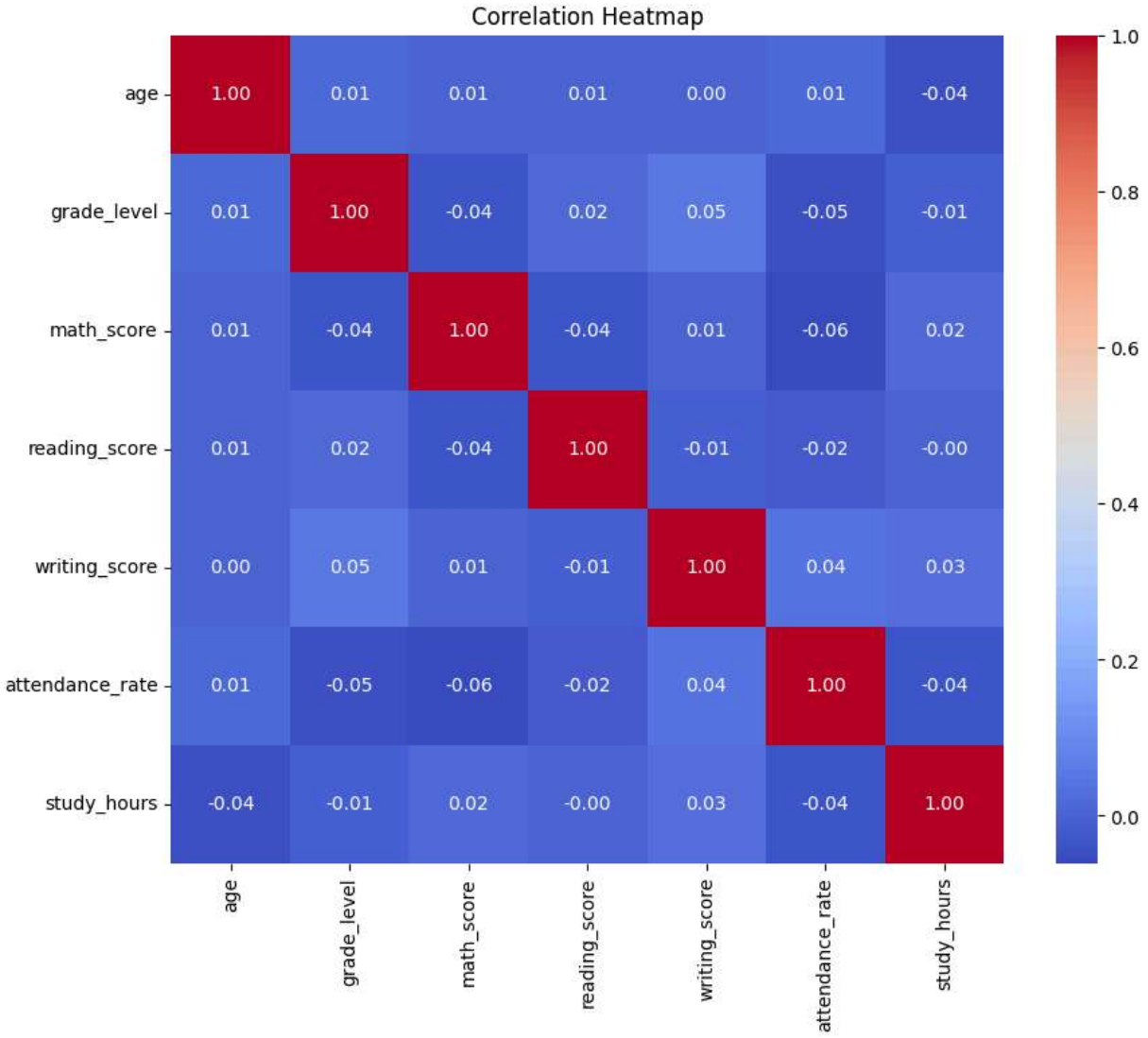


In [38]:
```
# Score Distributions
score_cols = ["math_score", "reading_score", "writing_score"]
df[score_cols].hist(bins=20, figsize=(12, 6), color="#00bcd4")
plt.suptitle("Score Distributions")
plt.show()
```



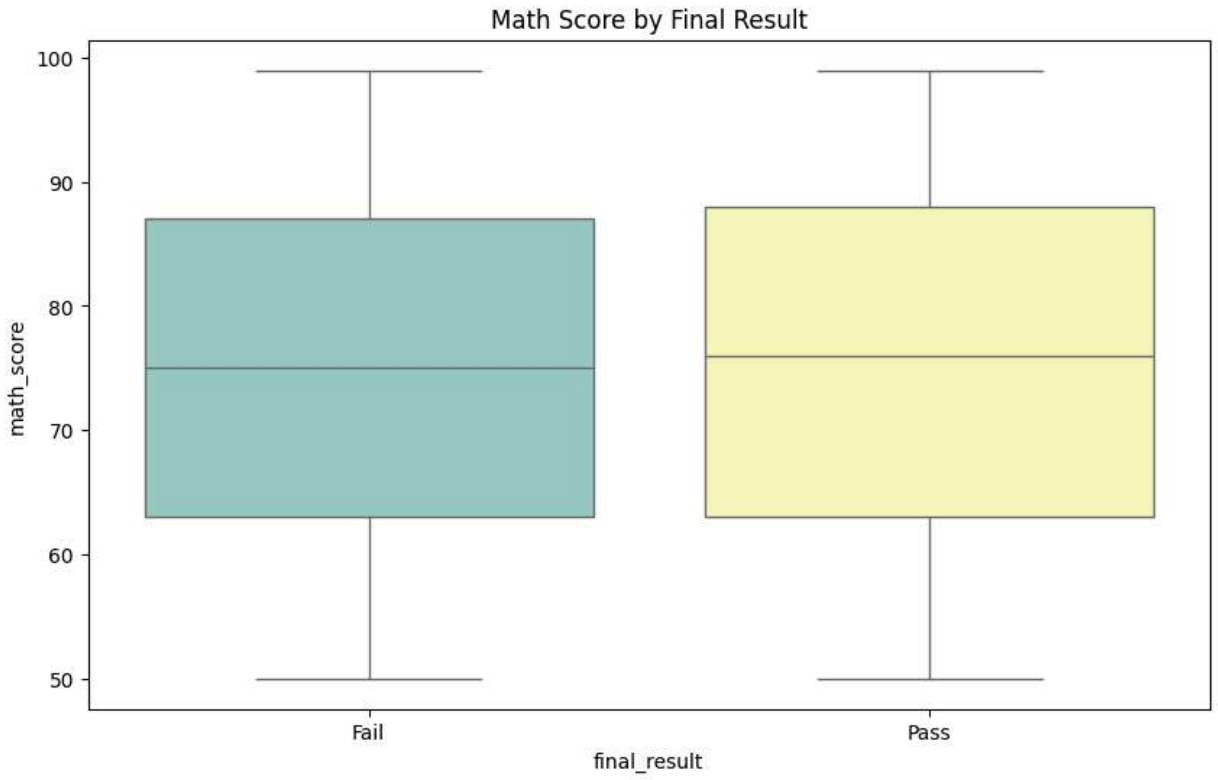
In [39]:

```
# Correlation Heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(df.select_dtypes(include=["float64", "int64"]).corr(), annot=Tru
plt.title("Correlation Heatmap")
plt.show()
```

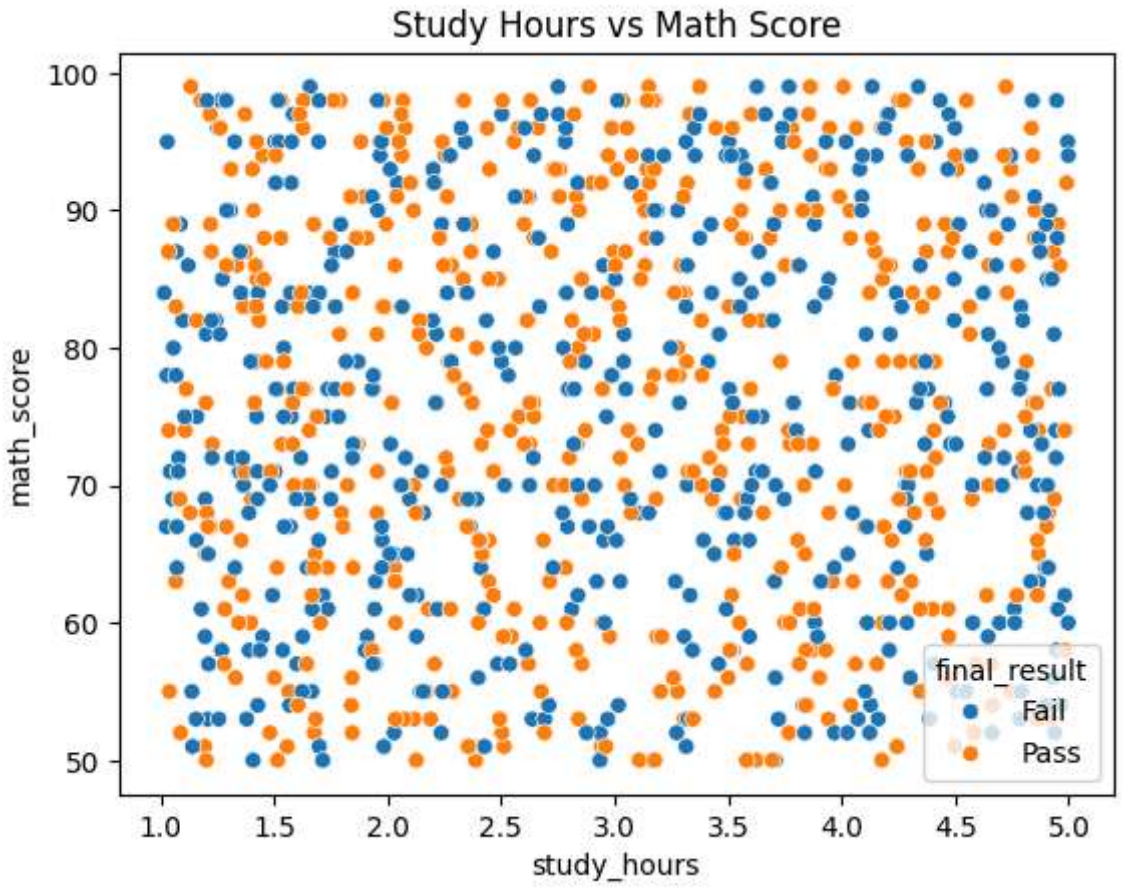


In [41]:

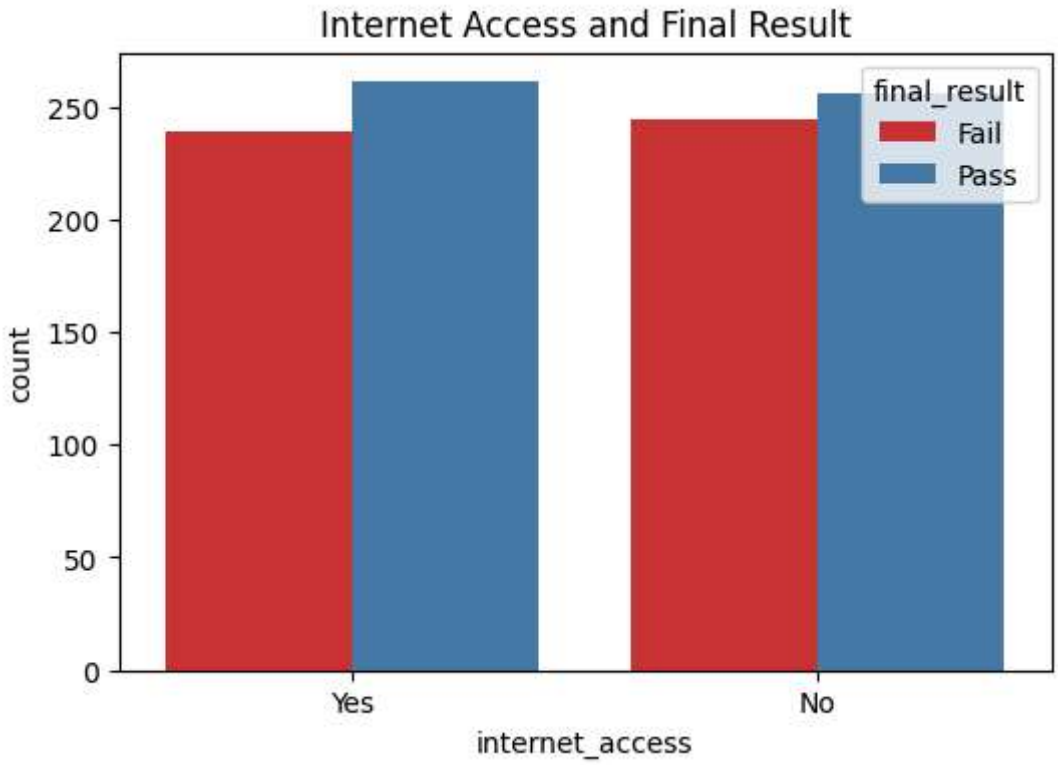
```
# Scores by Final Result
plt.figure(figsize=(10, 6))
sns.boxplot(data=df, x="final_result", y="math_score", hue="final_result", p
plt.title("Math Score by Final Result")
plt.show()
```



In [42]:
```
# Study Hours vs Scores
sns.scatterplot(data=df, x="study_hours", y="math_score", hue="final_result"
plt.title("Study Hours vs Math Score")
plt.show()
```



In [43]:
```
# Internet Access vs Final Result
plt.figure(figsize=(6,4))
sns.countplot(data=df, x="internet_access", hue="final_result", palette="Set
plt.title("Internet Access and Final Result")
plt.show()
```



In [45]:
```
#parrent education analysis
plt.figure(figsize=(10, 5))
sns.boxplot(data=df, x="parent_education", y="reading_score",
            hue="parent_education", palette="cool", legend=False)
plt.xticks(rotation=45)
plt.title("Reading Score vs Parent Education Level")
plt.show()
```