In [11]:

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

In [2]:

```
#READING THE CSV FILE
df= pd.read_csv("Student_data.csv")
```

In [3]:

```
#DISPLAYS FIRST FEW ROWS
df.head()
```

Out[3]:

	StudentID	Age	Gender	Ethnicity	ParentalEducation	StudyTimeWeekly	Absences	Tutor
0	1001	17	1	0	2	19.833723	7	
1	1002	18	0	0	1	15.408756	0	
2	1003	15	0	2	3	4.210570	26	
3	1004	17	1	0	3	10.028829	14	
4	1005	17	1	0	2	4.672495	17	

In [4]:

```
#DISPLAYS LAST FEW ROWS
df.tail()
```

Out[4]:

	StudentID	Age	Gender	Ethnicity	ParentalEducation	StudyTimeWeekly	Absences	Tu
2387	3388	18	1	0	3	10.680555	2	
2388	3389	17	0	0	1	7.583217	4	
2389	3390	16	1	0	2	6.805500	20	
2390	3391	16	1	1	0	12.416653	17	
2391	3392	16	1	0	2	17.819907	13	

In [5]:

```
#INFO ABOUT THE DATASET
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2392 entries, 0 to 2391
Data columns (total 15 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   StudentID          2392 non-null   int64  
 1   Age                2392 non-null   int64  
 2   Gender             2392 non-null   int64  
 3   Ethnicity          2392 non-null   int64  
 4   ParentalEducation  2392 non-null   int64  
 5   StudyTimeWeekly    2392 non-null   float64
 6   Absences           2392 non-null   int64  
 7   Tutoring           2392 non-null   int64  
 8   ParentalSupport    2392 non-null   int64  
 9   Extracurricular    2392 non-null   int64  
 10  Sports             2392 non-null   int64  
 11  Music              2392 non-null   int64  
 12  Volunteering       2392 non-null   int64  
 13  GPA                2392 non-null   float64
 14  GradeClass         2392 non-null   float64
dtypes: float64(3), int64(12)
memory usage: 280.4 KB
```

In [6]:

```
df.head(5)
```

Out[6]:

	StudentID	Age	Gender	Ethnicity	ParentalEducation	StudyTimeWeekly	Absences	Tutor
0	1001	17	1	0	2	19.833723	7	
1	1002	18	0	0	1	15.408756	0	
2	1003	15	0	2	3	4.210570	26	
3	1004	17	1	0	3	10.028829	14	
4	1005	17	1	0	2	4.672495	17	

In [7]:

```
#DESCRIBES THE DATASET
df.describe()
```

Out[7]:

	StudentID	Age	Gender	Ethnicity	ParentalEducation	StudyTimeWe
count	2392.000000	2392.000000	2392.000000	2392.000000	2392.000000	2392.00
mean	2196.500000	16.468645	0.510870	0.877508	1.746237	9.77
std	690.655244	1.123798	0.499986	1.028476	1.000411	5.65
min	1001.000000	15.000000	0.000000	0.000000	0.000000	0.00
25%	1598.750000	15.000000	0.000000	0.000000	1.000000	5.04
50%	2196.500000	16.000000	1.000000	0.000000	2.000000	9.70
75%	2794.250000	17.000000	1.000000	2.000000	2.000000	14.40
max	3392.000000	18.000000	1.000000	3.000000	4.000000	19.97

In [9]:

```
# CHECKING FOR NULL VALUES
df.isnull().sum()
```

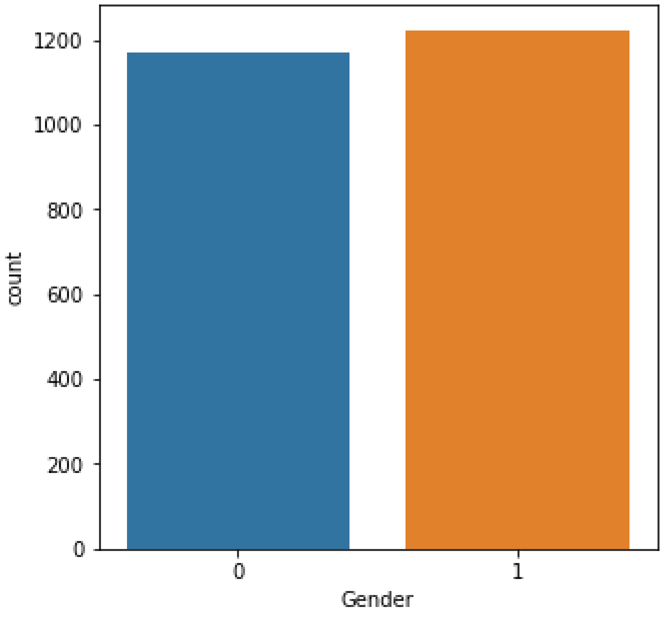
Out[9]:

```
StudentID            0
Age                  0
Gender               0
Ethnicity            0
ParentalEducation    0
StudyTimeWeekly      0
Absences             0
Tutoring             0
ParentalSupport      0
Extracurricular      0
Sports               0
Music                0
Volunteering         0
GPA                  0
GradeClass           0
dtype: int64
```

GENDER DISTRBUTION

In [19]:

```
plt.figure(figsize=(5,5))
sns.countplot(data=df,x="Gender")
plt.show()
```



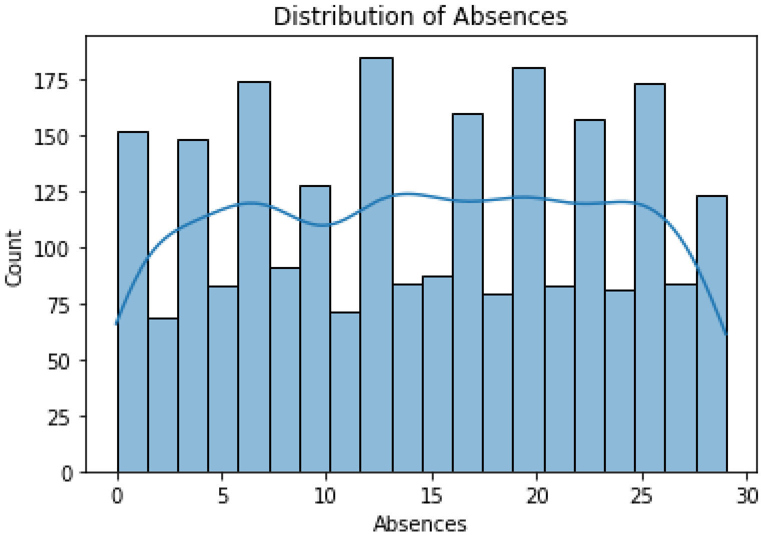
In [20]:

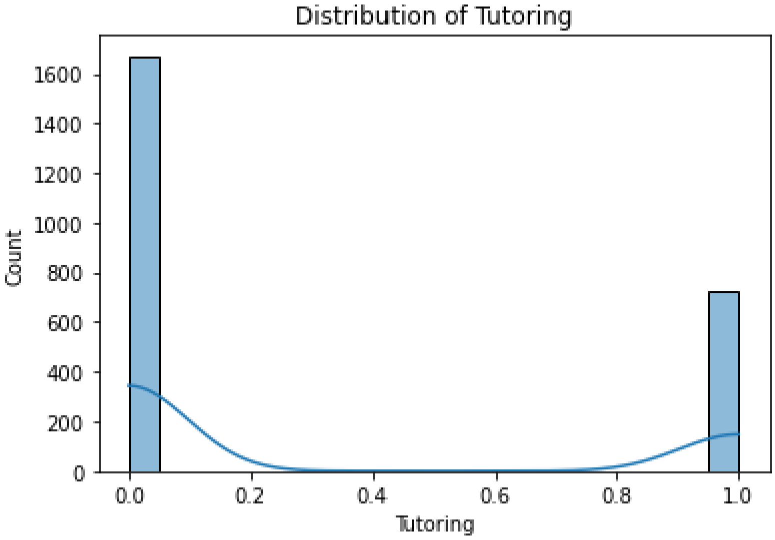
```
#IF WE ASSUME 0 TO BE MALES AND 1 TO BE FEMALES, FROM THE ABOVE CHART WE CA
```

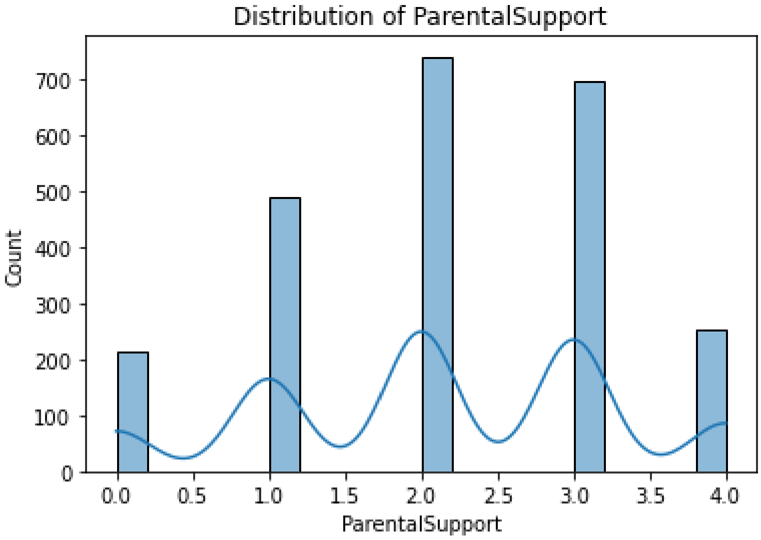
Data distribution - Numerical columns

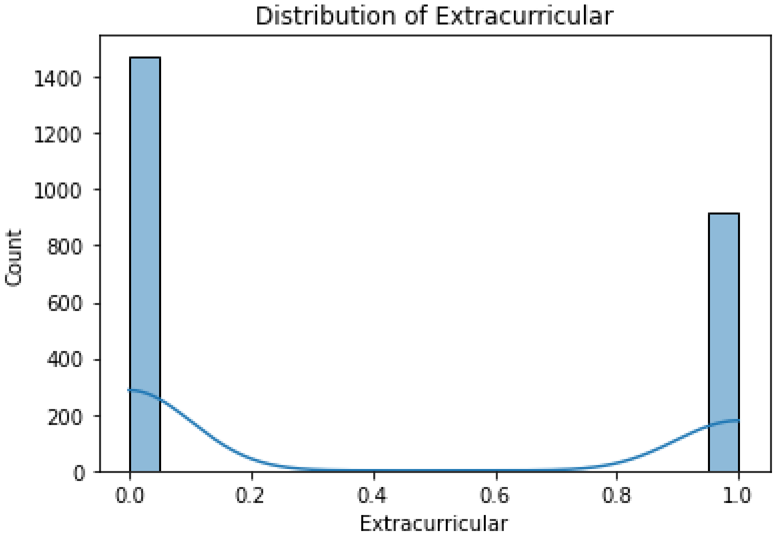
In [27]:

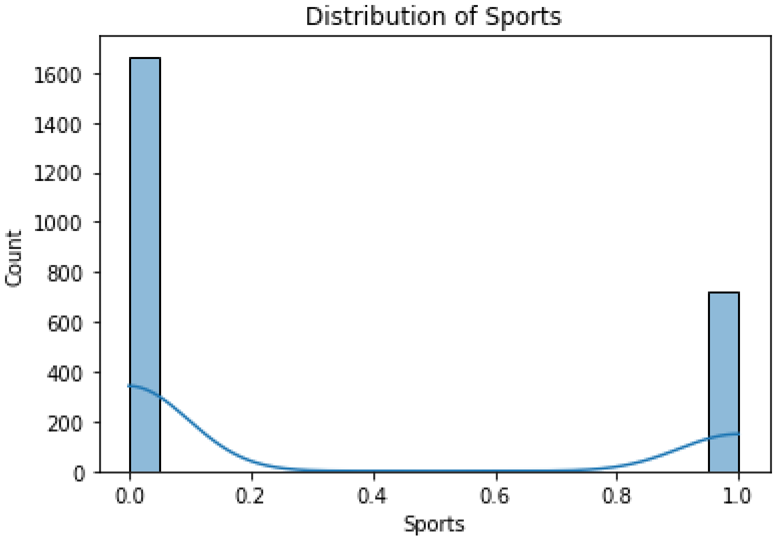
```
numerical_columns = ['Absences','Tutoring','ParentalSupport','Extracurricul
for col in numerical_columns:
    plt.figure()
    sns.histplot(df[col], bins=20, kde=True)
    plt.title(f'Distribution of {col}')
    plt.show()
```







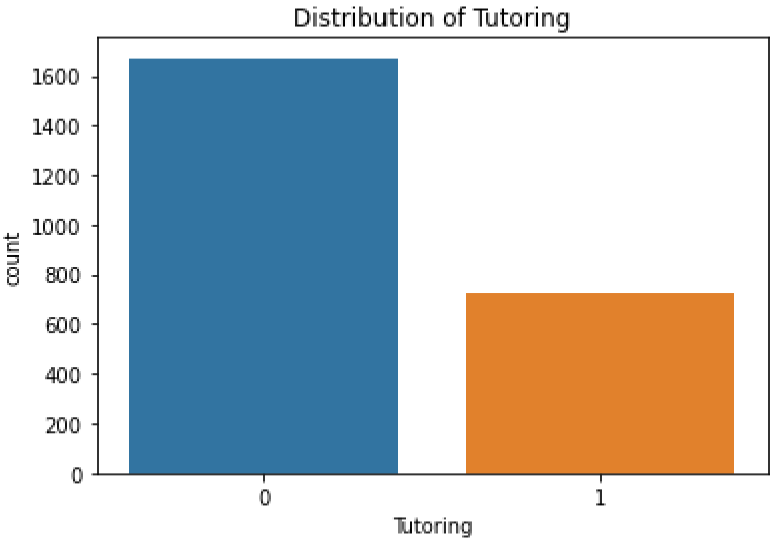


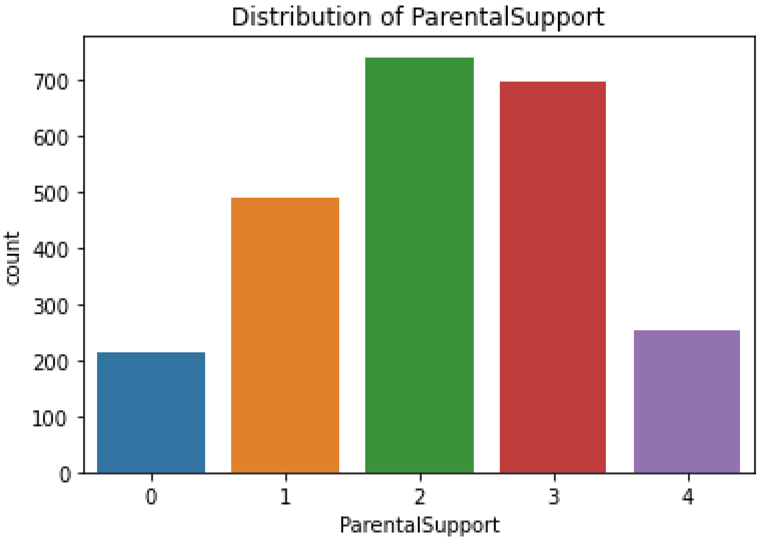


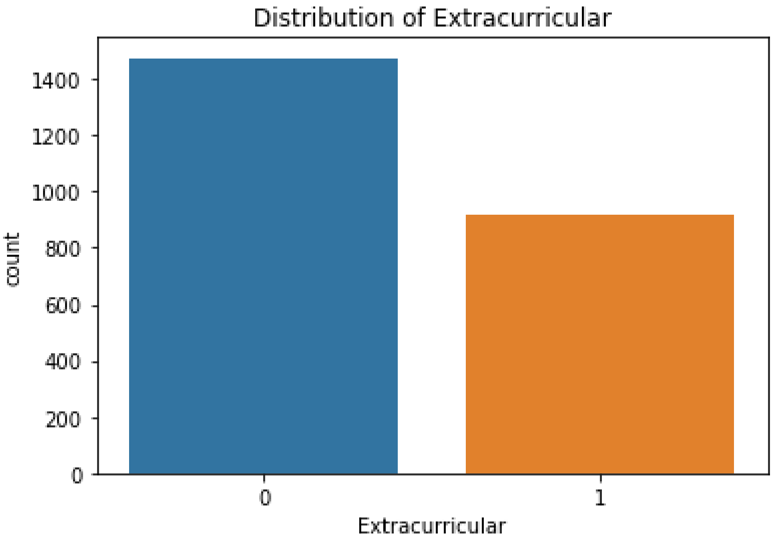
Categorical variables

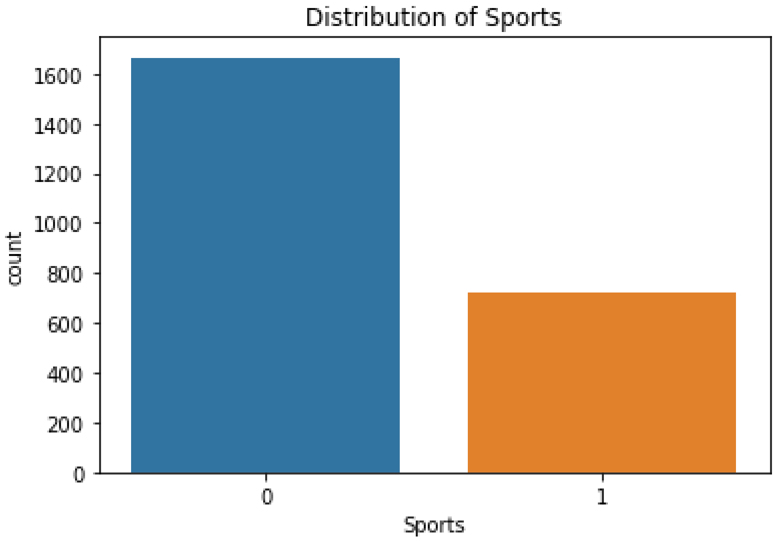
In [28]:

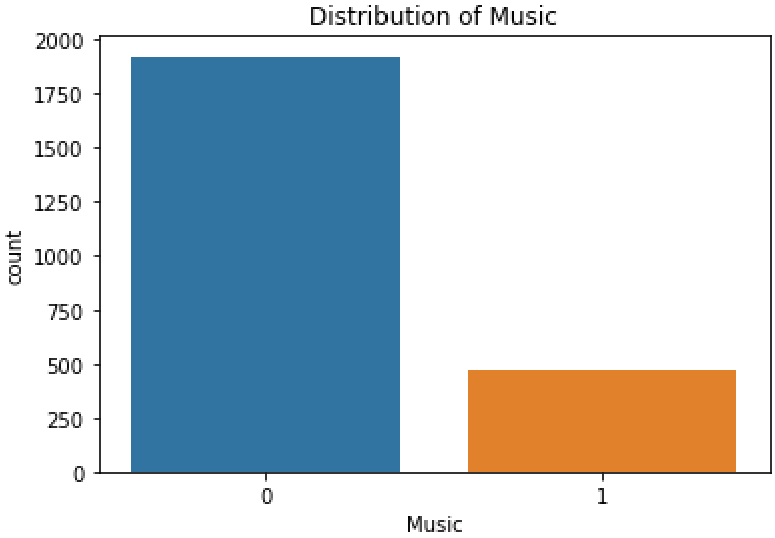
```
categorical_columns = ['Tutoring', 'ParentalSupport', 'Extracurricular', 'S
for col in categorical_columns:
    plt.figure()
    sns.countplot(x = df[col])
    plt.title(f'Distribution of {col}')
    plt.show()
```











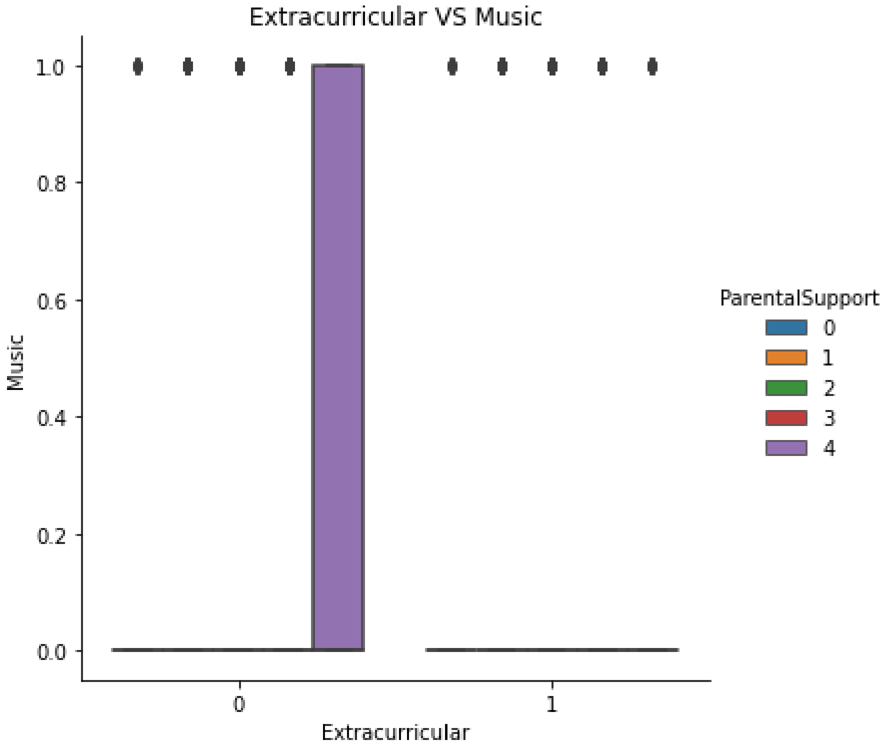
Looking at Extracurricular and ParentalSupport against Music

In [31]:

```
plt.figure(figsize=(20,20))
sns.catplot(x="Extracurricular", y="Music", hue="ParentalSupport", kind="bo
plt.title("Extracurricular VS Music")
plt.xlabel("Extracurricular")
plt.ylabel("Music")
```

Out[31]:

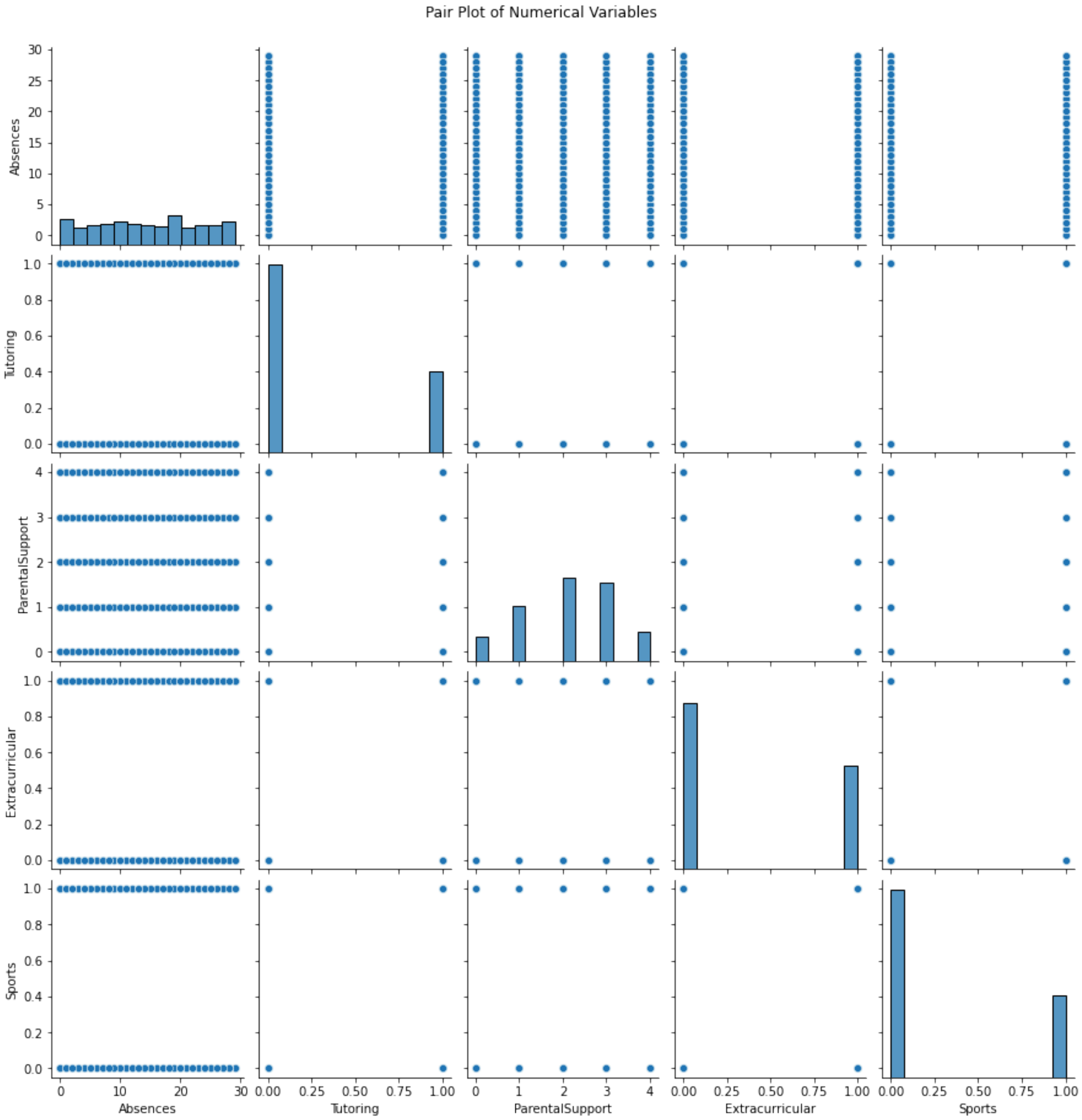
```
Text(15.062222222222218, 0.5, 'Music')

<Figure size 1440x1440 with 0 Axes>
```



Data Visualization - Pair Plot

In [32]:

```
sns.pairplot(df[numerical_columns])
plt.suptitle('Pair Plot of Numerical Variables', y=1.02)
plt.show()
```



In [33]:

```
df.corr()
```

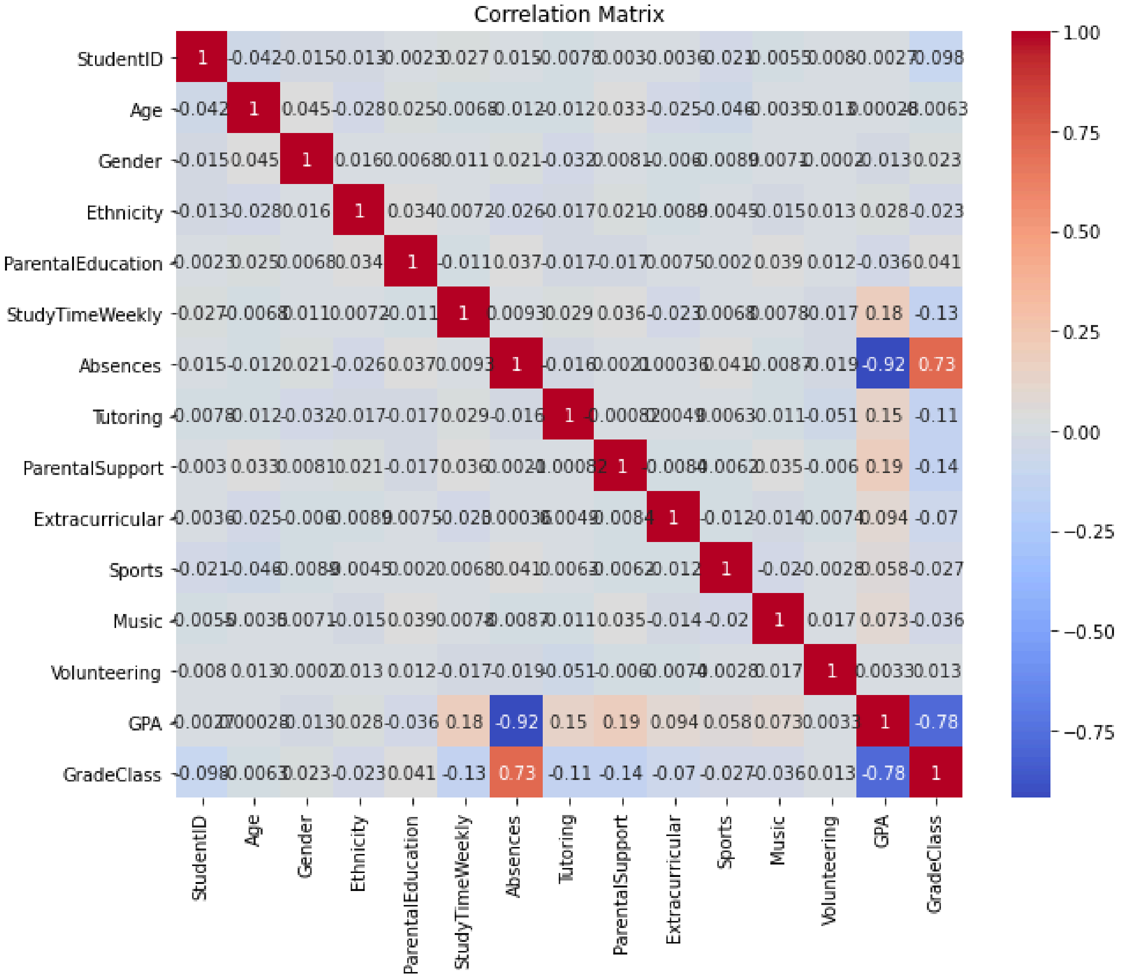
Out[33]:

	StudentID	Age	Gender	Ethnicity	ParentalEducation	StudyTi
StudentID	1.000000	-0.042255	-0.014625	-0.012990	-0.002307	
Age	-0.042255	1.000000	0.044895	-0.028473	0.025099	
Gender	-0.014625	0.044895	1.000000	0.016010	0.006771	
Ethnicity	-0.012990	-0.028473	0.016010	1.000000	0.033595	
ParentalEducation	-0.002307	0.025099	0.006771	0.033595	1.000000	
StudyTimeWeekly	0.026976	-0.006800	0.011469	0.007184	-0.011051	
Absences	0.014841	-0.011511	0.021479	-0.025712	0.036518	
Tutoring	-0.007834	-0.012076	-0.031597	-0.017440	-0.017340	
ParentalSupport	0.003016	0.033197	0.008065	0.020922	-0.017463	
Extracurricular	-0.003611	-0.025061	-0.005964	-0.008927	0.007479	
Sports	-0.020703	-0.046320	-0.008897	-0.004484	0.002029	

Correlation analysis

In [34]:

```
correlation_matrix = df.corr()
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```



In [ ]: