```
print("Hello world")
```

```
Hello world
```

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

```
df=pd.read_csv("student.csv")
print(df.head())
```

```
   Unnamed: 0  Gender EthnicGroup          ParentEduc     LunchType TestPrep  \
0           0  female         NaN   bachelor's degree      standard     none
1           1  female     group C        some college      standard      NaN
2           2  female     group B     master's degree      standard     none
3           3    male     group A  associate's degree  free/reduced     none
4           4    male     group C        some college      standard     none

  ParentMaritalStatus PracticeSport IsFirstChild  NrSiblings TransportMeans  \
0             married     regularly          yes         3.0     school_bus
1             married     sometimes          yes         0.0            NaN
2              single     sometimes          yes         4.0     school_bus
3             married         never           no         1.0            NaN
4             married     sometimes          yes         0.0     school_bus

  WklyStudyHours  MathScore  ReadingScore  WritingScore
0            < 5         71            71            74
1        5 - 10         69            90            88
2            < 5         87            93            91
3        5 - 10         45            56            42
4        5 - 10         76            78            75
```

```
df.describe()
```

	Unnamed: 0	NrSiblings	MathScore	ReadingScore	WritingScore
count	30641.000000	29069.000000	30641.000000	30641.000000	30641.000000
mean	499.556607	2.145894	66.558402	69.377533	68.418622
std	288.747894	1.458242	15.361616	14.758952	15.443525
min	0.000000	0.000000	0.000000	10.000000	4.000000
25%	249.000000	1.000000	56.000000	59.000000	58.000000
50%	500.000000	2.000000	67.000000	70.000000	69.000000
75%	750.000000	3.000000	78.000000	80.000000	79.000000
max	999.000000	7.000000	100.000000	100.000000	100.000000

```
df.info()
```

```
df.isnull().sum()
```

Drop unnamed column

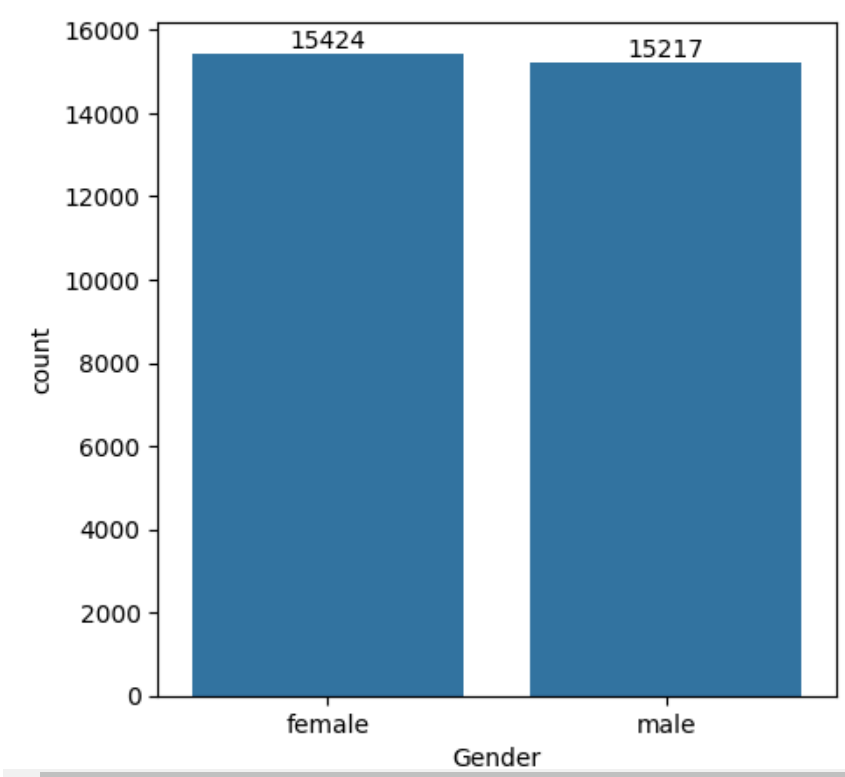
```
# Check if the column exists before attempting to drop it.
if "Unnamed: 0" in df.columns:
    df = df.drop("Unnamed: 0", axis=1)
    print(df.head())
else:
    print("Column 'Unnamed: 0' not found in DataFrame.")
print(df.head())
```

```
   Gender EthnicGroup          ParentEduc     LunchType TestPrep  \
0  female         NaN   bachelor's degree      standard     none
1  female     group C        some college      standard      NaN
2  female     group B     master's degree      standard     none
3    male     group A  associate's degree  free/reduced     none
4    male     group C        some college      standard     none

  ParentMaritalStatus PracticeSport IsFirstChild  NrSiblings TransportMeans  \
0             married     regularly          yes         3.0     school_bus
1             married     sometimes          yes         0.0            NaN
2              single     sometimes          yes         4.0     school_bus
3             married         never           no         1.0            NaN
4             married     sometimes          yes         0.0     school_bus

  WklyStudyHours  MathScore  ReadingScore  WritingScore
0            < 5         71            71            74
1         5 - 10         69            90            88
2            < 5         87            93            91
3         5 - 10         45            56            42
4         5 - 10         76            78            75
   Gender EthnicGroup          ParentEduc     LunchType TestPrep  \
0  female         NaN   bachelor's degree      standard     none
1  female     group C        some college      standard      NaN
2  female     group B     master's degree      standard     none
3    male     group A  associate's degree  free/reduced     none
4    male     group C        some college      standard     none

  ParentMaritalStatus PracticeSport IsFirstChild  NrSiblings TransportMeans  \
0             married     regularly          yes         3.0     school_bus
1             married     sometimes          yes         0.0            NaN
2              single     sometimes          yes         4.0     school_bus
3             married         never           no         1.0            NaN
4             married     sometimes          yes         0.0     school_bus

  WklyStudyHours  MathScore  ReadingScore  WritingScore
0            < 5         71            71            74
1         5 - 10         69            90            88
2            < 5         87            93            91
3         5 - 10         45            56            42
4         5 - 10         76            78            75
```

Start coding or generate with AI.

Gender Distribution

```
plt.figure(figsize=(5,5))
ax=sns.countplot(data=df,x="Gender")
ax.bar_label(ax.containers[0])
plt.show()
```



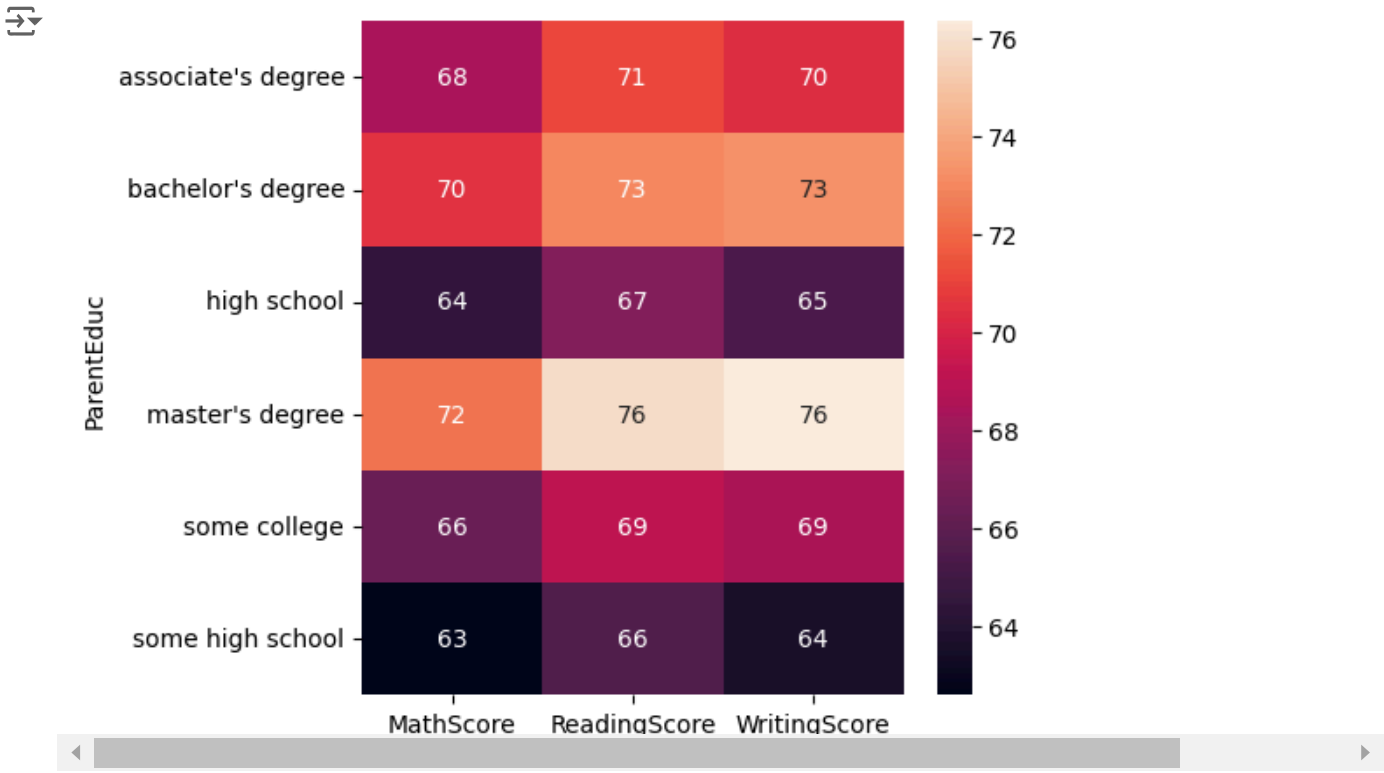
Double-click (or enter) to edit

From the above chart we have analyzed that no. of female is more than male

```
gb= df.groupby("ParentEduc").agg({"MathScore":"mean","ReadingScore":"mean","WritingScore":"mean"})
print(gb)
```

```
                    MathScore  ReadingScore  WritingScore
ParentEduc
associate's degree  68.365586     71.124324     70.299099
bachelor's degree   70.466627     73.062020     73.331069
high school         64.435731     67.213997     65.421136
master's degree     72.336134     75.832921     76.356896
some college        66.390472     69.179708     68.501432
some high school    62.584013     65.510785     63.632409
```

```
plt.figure(figsize=(5,5))
sns.heatmap(gb,annot= True)
plt.show()
```



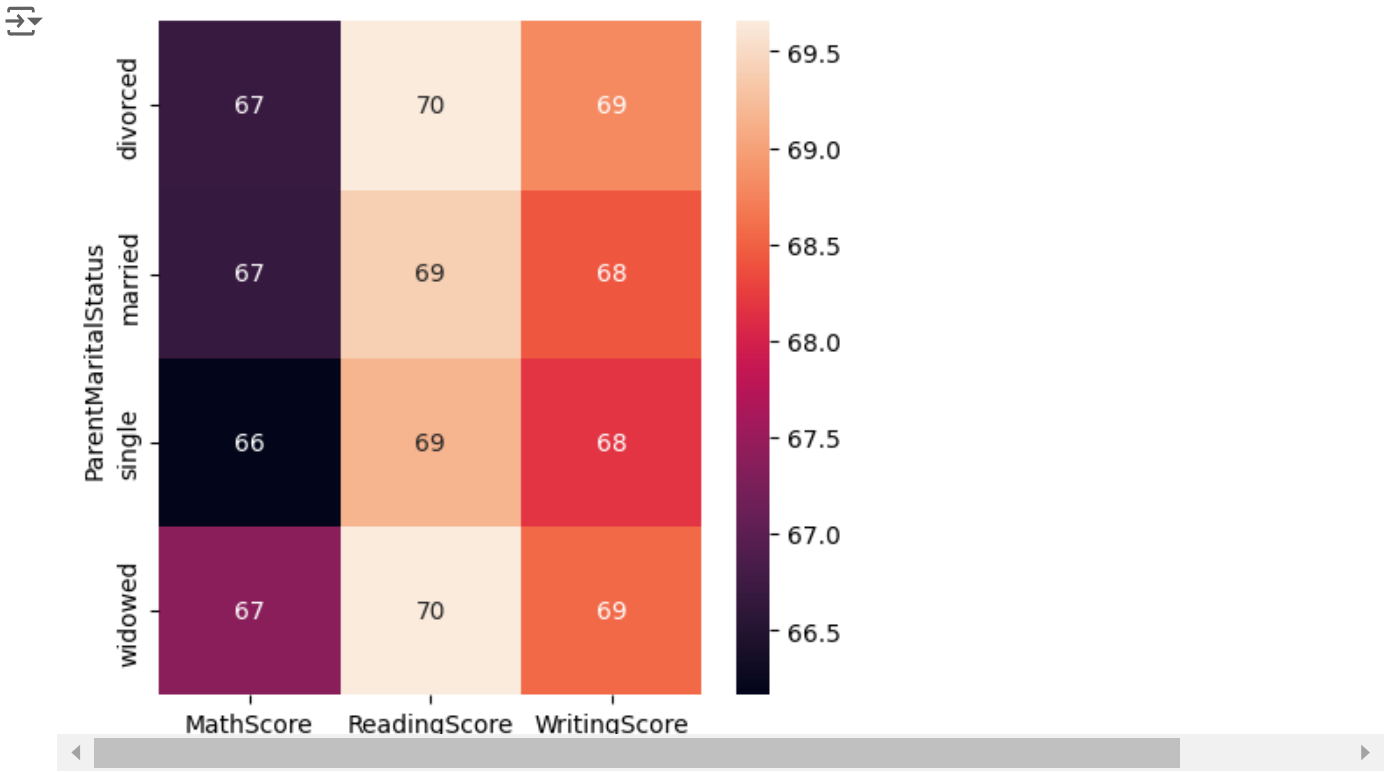
From the above chart we have seen that parent education have a good impact on student result

```
gb1= df.groupby("ParentMaritalStatus").agg({"MathScore":"mean","ReadingScore":"mean","WritingScore":"m
```

```
print(gb1)
```

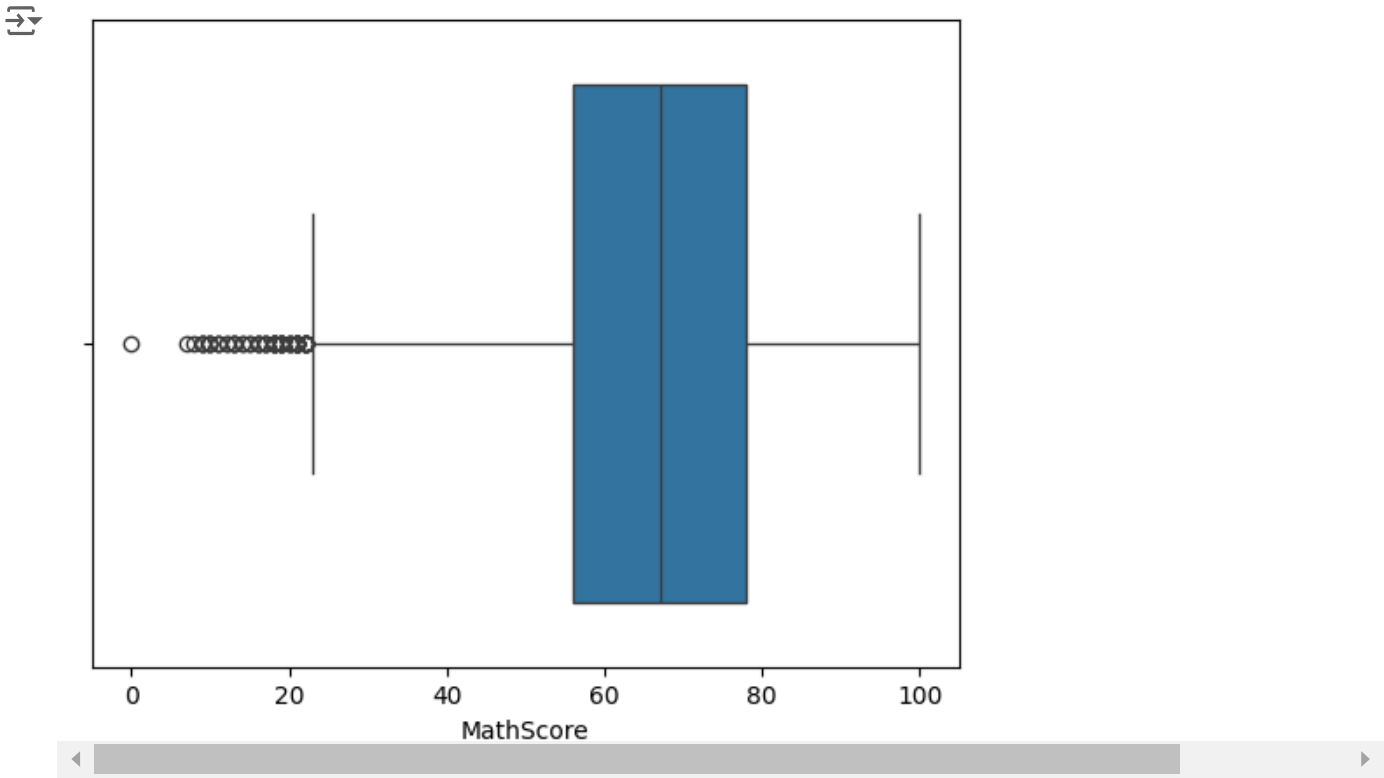
```
                    MathScore  ReadingScore  WritingScore
ParentMaritalStatus
divorced            66.691197     69.655011     68.799146
married             66.657326     69.389575     68.420981
single              66.165704     69.157250     68.174440
widowed             67.368866     69.651438     68.563452
```

```
plt.figure(figsize=(5,5))
sns.heatmap(gb1,annot= True)
plt.show()
```

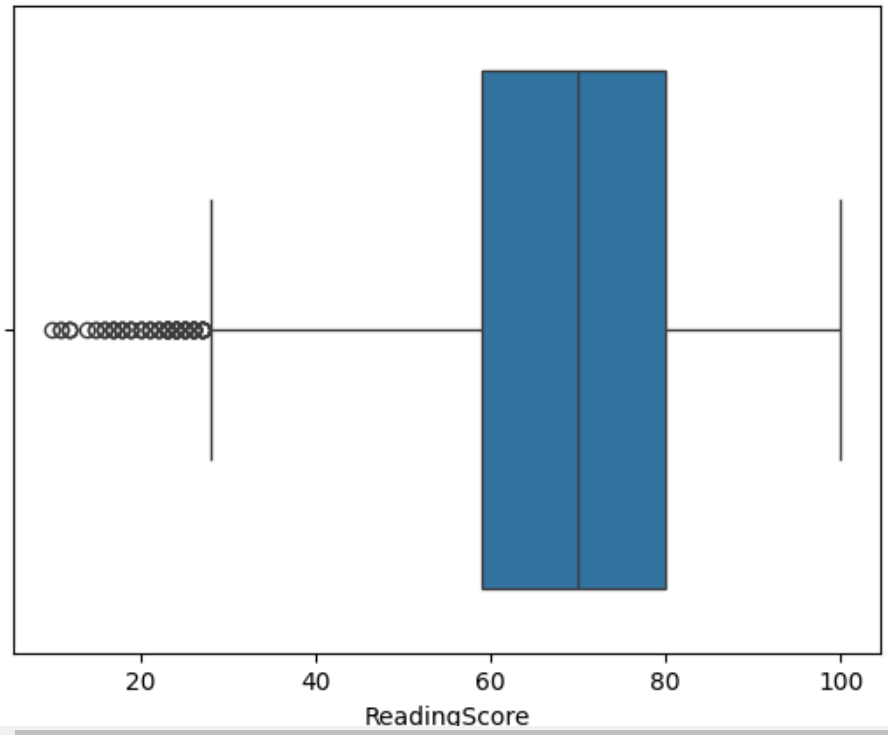


From the above chart we have seen parentmartialstatus don't have a huge impact (no impact/ negligible) on student's marks

```
sns.boxplot(data=df,x="MathScore")
plt.show()
```




```
sns.boxplot(data=df,x="ReadingScore")
plt.show()
```




```
print(df["EthnicGroup"].unique())
```

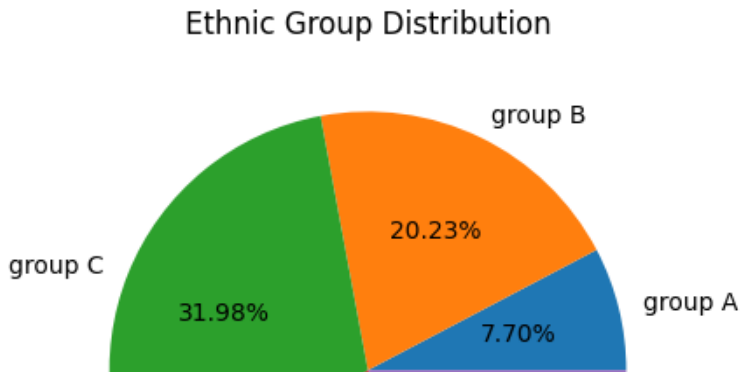
```
[nan 'group C' 'group B' 'group A' 'group D' 'group E']
```

Distibution of Ethnic groups

```
groupA=df.loc[(df['EthnicGroup']=="group A")].count()
print(groupA)
```

```
Gender                 2219
EthnicGroup            2219
ParentEduc             2078
LunchType              2219
TestPrep               2081
ParentMaritalStatus    2121
PracticeSport          2167
IsFirstChild           2168
NrSiblings             2096
TransportMeans         1999
WklyStudyHours         2146
MathScore              2219
ReadingScore           2219
WritingScore           2219
dtype: int64
```

```
groupA=df.loc[(df['EthnicGroup']=="group A")].count()
groupB=df.loc[(df['EthnicGroup']=="group B")].count()
groupC=df.loc[(df['EthnicGroup']=="group C")].count()
groupD=df.loc[(df['EthnicGroup']=="group D")].count()
groupE=df.loc[(df['EthnicGroup']=="group E")].count()
l=["group A","group B","group C","group D","group E"]
mlist=[groupA["EthnicGroup"],groupB["EthnicGroup"],groupC["EthnicGroup"],groupD["EthnicGroup"],groupE[
plt.pie(mlist,labels=l,autopct="%1.2f%%")
plt.title("Ethnic Group Distribution")
plt.show()
```


Ethnic Group Distribution
group B
20.23%
group C
31.98%
group A
7.70%