

Python & it's Libraries

Project Name - Student Scores Analysis

By Utkarsh Srivastava

Importing Libraries

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [2]: df = pd.read_csv(r"C:\Users\Dell\Downloads\Student Score.csv")
```

```
In [3]: df
```

```
Out[3]:
```

	Unnamed: 0	Gender	EthnicGroup	ParentEduc	LunchType	TestPrep	ParentMaritalStatus	PracticeSport	IsFirstChild	NrSiblings	TransportMeans	WklyStu
0	0	female	NaN	bachelor's degree	standard	none	married	regularly	yes	3.0	school_bus	
1	1	female	group C	some college	standard	NaN	married	sometimes	yes	0.0	NaN	
2	2	female	group B	master's degree	standard	none	single	sometimes	yes	4.0	school_bus	
3	3	male	group A	associate's degree	free/reduced	none	married	never	no	1.0	NaN	
4	4	male	group C	some college	standard	none	married	sometimes	yes	0.0	school_bus	
...
30636	816	female	group D	high school	standard	none	single	sometimes	no	2.0	school_bus	
30637	890	male	group E	high school	standard	none	single	regularly	no	1.0	private	
30638	911	female	NaN	high school	free/reduced	completed	married	sometimes	no	1.0	private	

30639	934	female	group D	associate's degree	standard	completed		married	regularly	no	3.0	school_bus
30640	960	male	group B	some college	standard	none		married	never	no	1.0	school_bus

30641 rows × 15 columns

Get Top five Data

In [4]: df.head()

Out[4]:

	Unnamed: 0	Gender	EthnicGroup	ParentEduc	LunchType	TestPrep	ParentMaritalStatus	PracticeSport	IsFirstChild	NrSiblings	TransportMeans	WklyStudyHours
0	0	female	NaN	bachelor's degree	standard	none	married	regularly	yes	3.0	school_bus	
1	1	female	group C	some college	standard	NaN	married	sometimes	yes	0.0	NaN	05-12
2	2	female	group B	master's degree	standard	none	single	sometimes	yes	4.0	school_bus	
3	3	male	group A	associate's degree	free/reduced	none	married	never	no	1.0	NaN	05-12
4	4	male	group C	some college	standard	none	married	sometimes	yes	0.0	school_bus	05-12

Get Bottom five Data

In [5]: df.tail()

Out[5]:

	Unnamed: 0	Gender	EthnicGroup	ParentEduc	LunchType	TestPrep	ParentMaritalStatus	PracticeSport	IsFirstChild	NrSiblings	TransportMeans	WklyStudyHours
30636	816	female	group D	high school	standard	none	single	sometimes	no	2.0	school_bus	
30637	890	male	group E	high school	standard	none	single	regularly	no	1.0	private	
30638	911	female	NaN	high school	free/reduced	completed	married	sometimes	no	1.0	private	
30639	934	female	group D	associate's degree	standard	completed	married	regularly	no	3.0	school_bus	
30640	960	male	group B	some college	standard	none	married	never	no	1.0	school_bus	

DataFrame Information

```
In [6]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30641 entries, 0 to 30640
Data columns (total 15 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Unnamed: 0            30641 non-null  int64
1   Gender                30641 non-null  object
2   EthnicGroup           28801 non-null  object
3   ParentEduc            28796 non-null  object
4   LunchType             30641 non-null  object
5   TestPrep              28811 non-null  object
6   ParentMaritalStatus   29451 non-null  object
7   PracticeSport         30010 non-null  object
8   IsFirstChild          29737 non-null  object
9   NrSiblings            29069 non-null  float64
10  TransportMeans        27507 non-null  object
11  WklyStudyHours        29686 non-null  object
12  MathScore             30641 non-null  int64
13  ReadingScore          30641 non-null  int64
14  WritingScore          30641 non-null  int64
dtypes: float64(1), int64(4), object(10)
memory usage: 3.5+ MB
```

Statistics Summary

```
In [7]: df.describe()
```

Out[7]:

	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

Shape of Data

```
In [8]: df.shape
```

```
Out[8]: (30641, 15)
```

DataFrame Correlation

```
In [9]: df.corr
```

```
Out[9]: <bound method DataFrame.corr of      Unnamed: 0  Gender  EthnicGroup      ParentEduc      LunchType  \
0              0  female      NaN  bachelor's degree      standard
1              1  female  group C      some college      standard
2              2  female  group B  master's degree      standard
3              3   male  group A  associate's degree  free/reduced
4              4   male  group C      some college      standard
...          ...    ...      ...          ...          ...
30636         816  female  group D      high school      standard
30637         890   male  group E      high school      standard
30638         911  female      NaN      high school  free/reduced
30639         934  female  group D  associate's degree      standard
30640         960   male  group B      some college      standard
```

```
      TestPrep  ParentMaritalStatus  PracticeSport  IsFirstChild  NrSiblings  \
0           none           married      regularly           yes           3.0
1           NaN           married      sometimes           yes           0.0
2           none           single      sometimes           yes           4.0
3           none           married           never           no           1.0
4           none           married      sometimes           yes           0.0
...          ...          ...          ...          ...          ...
30636         none           single      sometimes           no           2.0
30637         none           single      regularly           no           1.0
30638  completed           married      sometimes           no           1.0
30639  completed           married      regularly           no           3.0
30640         none           married           never           no           1.0
```

```
      TransportMeans  WklyStudyHours  MathScore  ReadingScore  WritingScore
0      school_bus      < 5           71           71           74
1              NaN      05-Oct           69           90           88
2      school_bus      < 5           87           93           91
3              NaN      05-Oct           45           56           42
4      school_bus      05-Oct           76           78           75
...          ...          ...          ...          ...          ...
30636  school_bus      05-Oct           59           61           65
30637     private      05-Oct           58           53           51
```

30638	private	05-Oct	61	70	67
30639	school_bus	05-Oct	82	90	93
30640	school_bus	05-Oct	64	60	58

[30641 rows x 15 columns]>

DataFrame Columns

In [10]: `df.columns`

Out[10]: Index(['Unnamed: 0', 'Gender', 'EthnicGroup', 'ParentEduc', 'LunchType', 'TestPrep', 'ParentMaritalStatus', 'PracticeSport', 'IsFirstChild', 'NrSiblings', 'TransportMeans', 'WklyStudyHours', 'MathScore', 'ReadingScore', 'WritingScore'], dtype='object')

Missing Values

In [11]: `df.isnull().sum()`

Out[11]:

Unnamed: 0	0
Gender	0
EthnicGroup	1840
ParentEduc	1845
LunchType	0
TestPrep	1830
ParentMaritalStatus	1190
PracticeSport	631
IsFirstChild	904
NrSiblings	1572
TransportMeans	3134
WklyStudyHours	955
MathScore	0
ReadingScore	0
WritingScore	0
dtype:	int64

Dropping Columns

In [12]: `df.drop("Unnamed: 0", axis=1, inplace=True)`

In [13]: `df`

Out[13]:

	Gender	EthnicGroup	ParentEduc	LunchType	TestPrep	ParentMaritalStatus	PracticeSport	IsFirstChild	NrSiblings	TransportMeans	WklyStudyHours	MathScore	ReadingScore	WritingScore
--	--------	-------------	------------	-----------	----------	---------------------	---------------	--------------	------------	----------------	----------------	-----------	--------------	--------------

0	female	NaN	bachelor's degree	standard	none	married	regularly	yes	3.0	school_bus	< 5
1	female	group C	some college	standard	NaN	married	sometimes	yes	0.0	NaN	05-Oct
2	female	group B	master's degree	standard	none	single	sometimes	yes	4.0	school_bus	< 5
3	male	group A	associate's degree	free/reduced	none	married	never	no	1.0	NaN	05-Oct
4	male	group C	some college	standard	none	married	sometimes	yes	0.0	school_bus	05-Oct
...
30636	female	group D	high school	standard	none	single	sometimes	no	2.0	school_bus	05-Oct
30637	male	group E	high school	standard	none	single	regularly	no	1.0	private	05-Oct
30638	female	NaN	high school	free/reduced	completed	married	sometimes	no	1.0	private	05-Oct
30639	female	group D	associate's degree	standard	completed	married	regularly	no	3.0	school_bus	05-Oct
30640	male	group B	some college	standard	none	married	never	no	1.0	school_bus	05-Oct

30641 rows × 14 columns

Replace Values

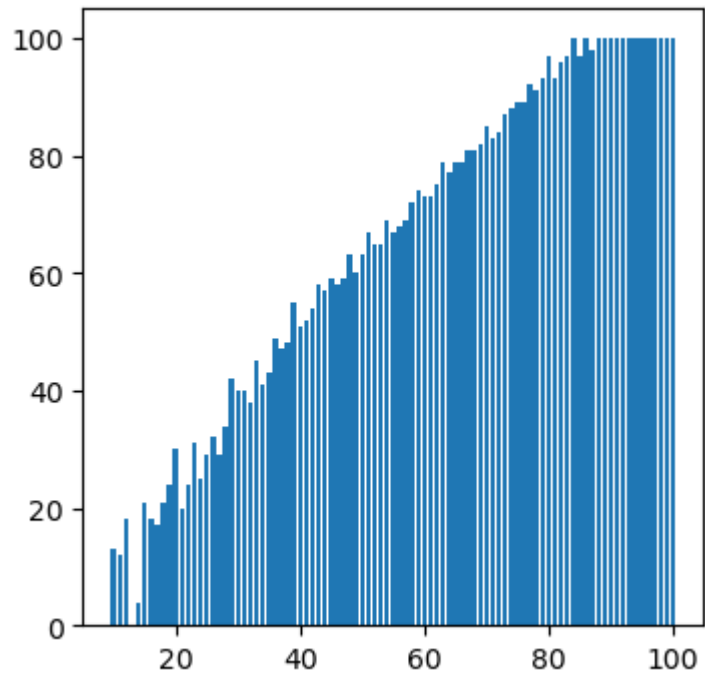
```
In [14]: df['WklyStudyHours']=df['WklyStudyHours'].str.replace('05-Oct','5-10')
```

```
In [15]: df.head(5)
```

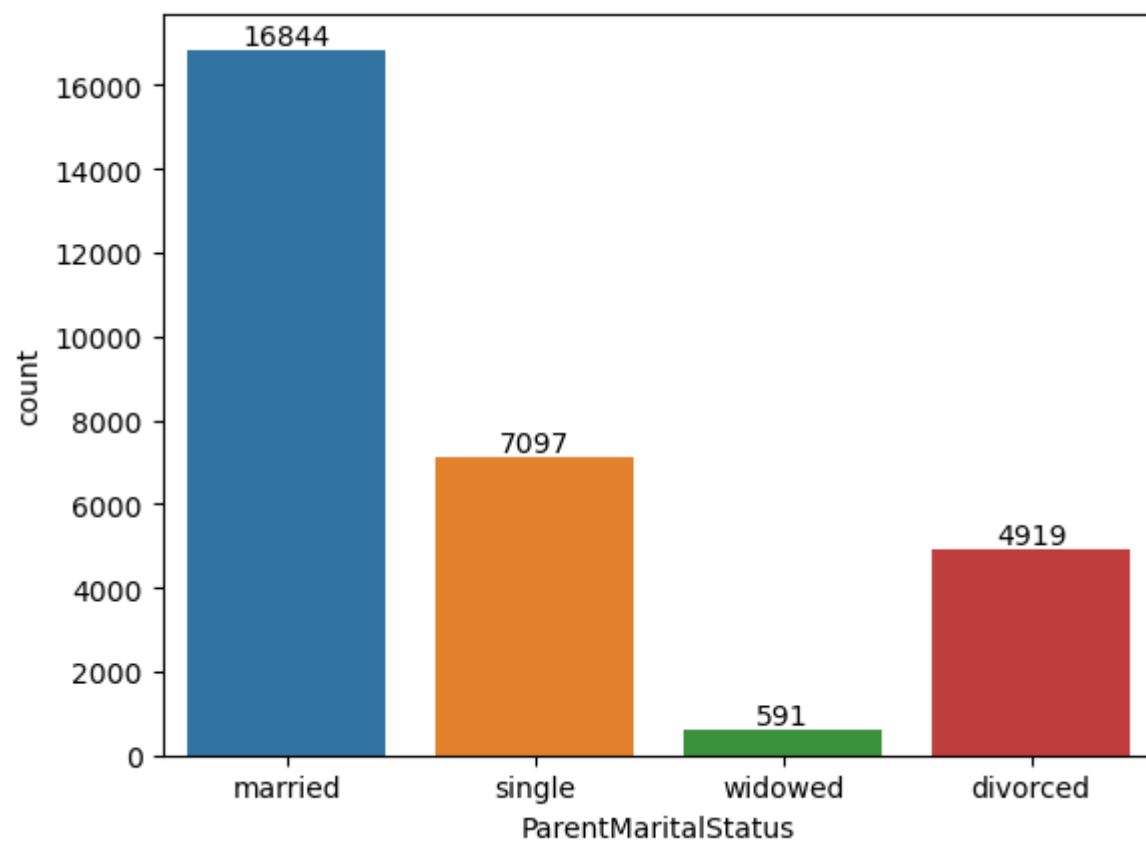
	Gender	EthnicGroup	ParentEduc	LunchType	TestPrep	ParentMaritalStatus	PracticeSport	IsFirstChild	NrSiblings	TransportMeans	WklyStudyHours	MathSc
0	female	NaN	bachelor's degree	standard	none	married	regularly	yes	3.0	school_bus	< 5	
1	female	group C	some college	standard	NaN	married	sometimes	yes	0.0	NaN	5-10	
2	female	group B	master's degree	standard	none	single	sometimes	yes	4.0	school_bus	< 5	
3	male	group A	associate's degree	free/reduced	none	married	never	no	1.0	NaN	5-10	

Exploratory Data Analysis

```
In [16]: x = df['ReadingScore']  
y = df['WritingScore']  
  
plt.figure(figsize=(4,4))  
plt.bar(x,y, data=df)  
plt.show()
```



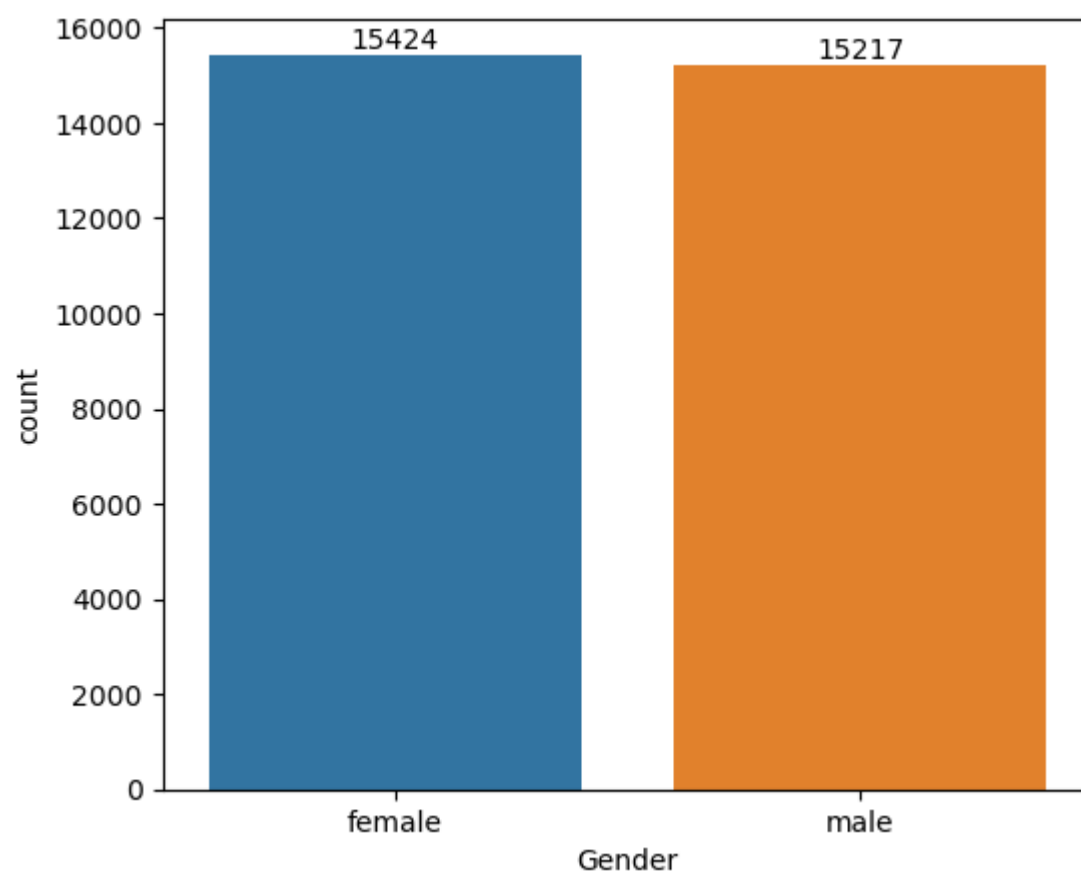
```
In [17]: ax=sns.countplot(data=df, x='ParentMaritalStatus')  
ax.bar_label(ax.containers[0])  
plt.figure(figsize=(5,4))  
plt.show()
```



<Figure size 500x400 with 0 Axes>

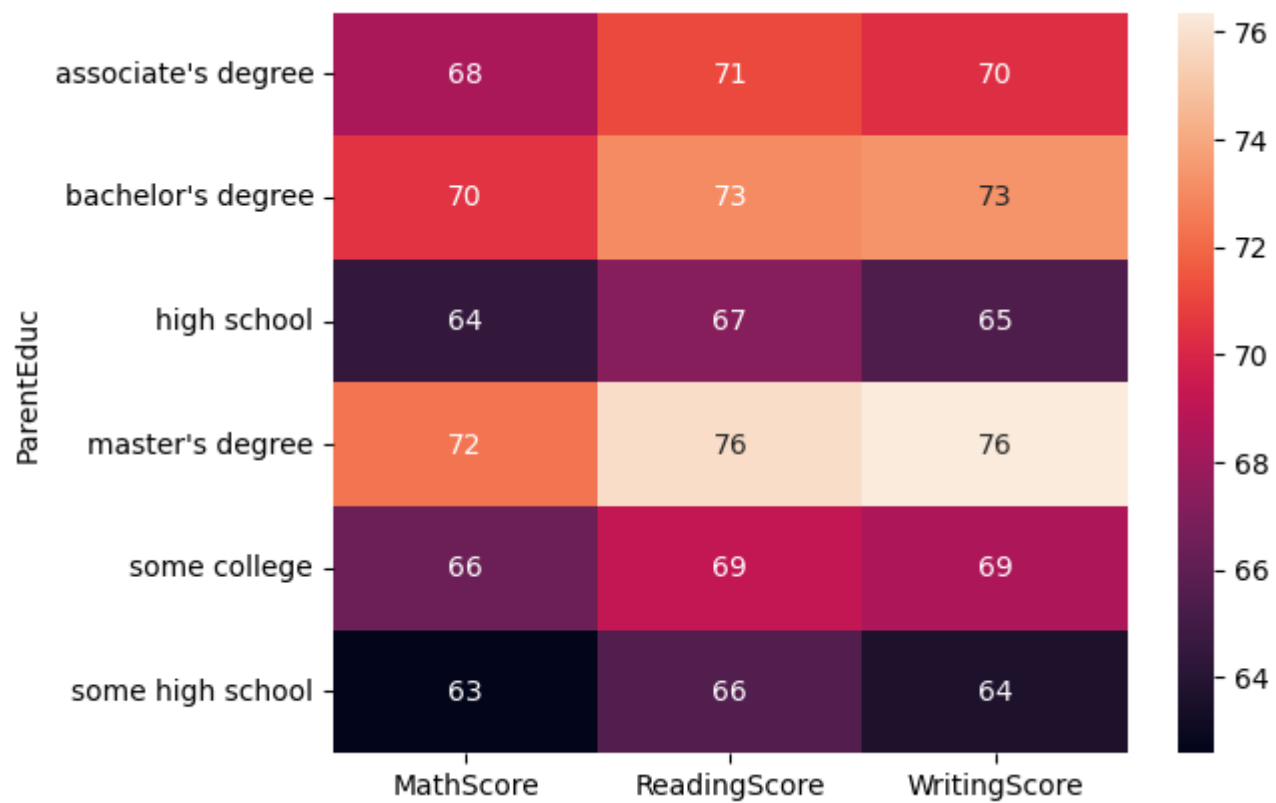
Gender Distribution

```
In [18]: plt.figure(figsize=(6,5))
ax=sns.countplot(data=df, x='Gender')
ax.bar_label(ax.containers[0])
plt.show()
```

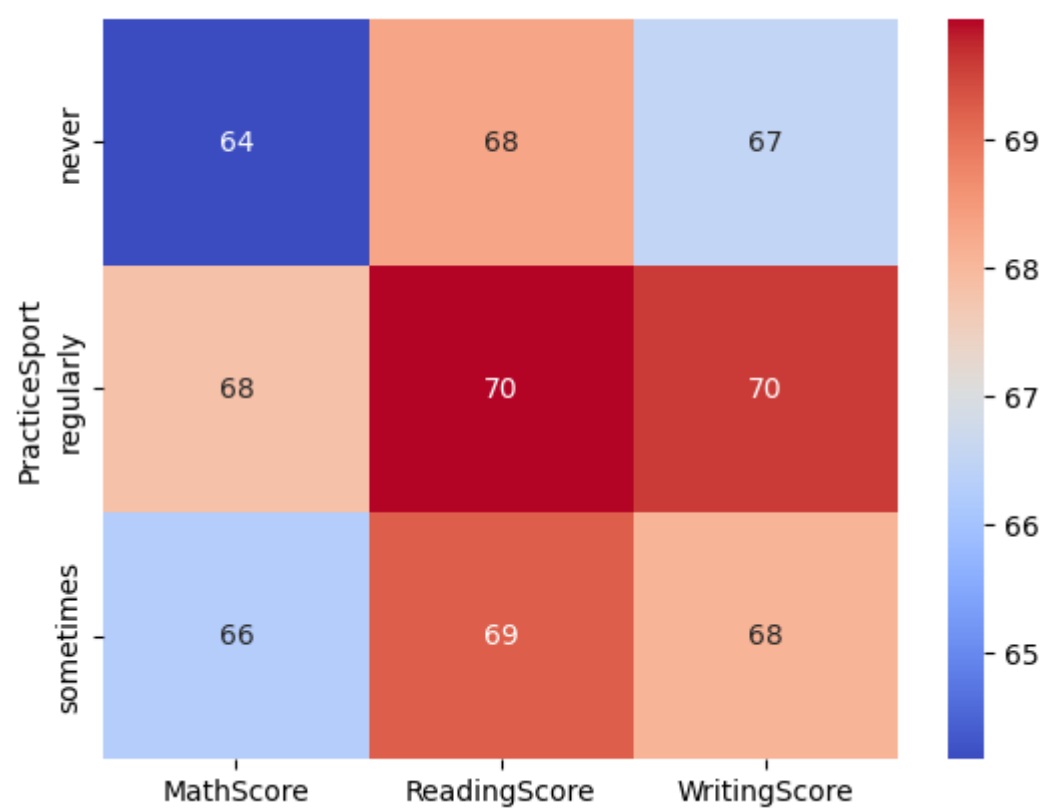
```
In [19]: gb = df.groupby('ParentEduc').agg({'MathScore':'mean', 'ReadingScore':'mean', 'WritingScore':'mean'})  
plt.show()
```

```
In [20]: sns.heatmap(data=gb, annot=True)  
plt.show()
```

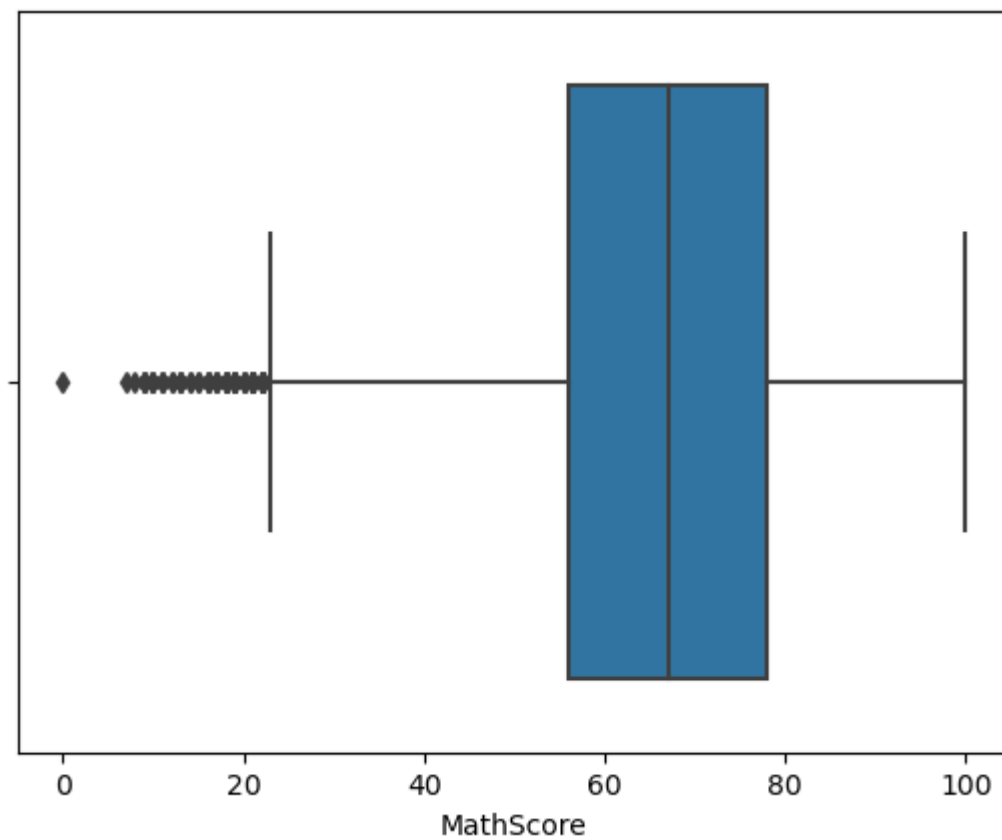


```
In [21]: gb1 = df.groupby('PracticeSport').agg({'MathScore':'mean', 'ReadingScore':'mean', 'WritingScore':'mean'})  
plt.show()
```

```
In [22]: sns.heatmap(data=gb1, annot=True, cmap='coolwarm')  
plt.show()
```



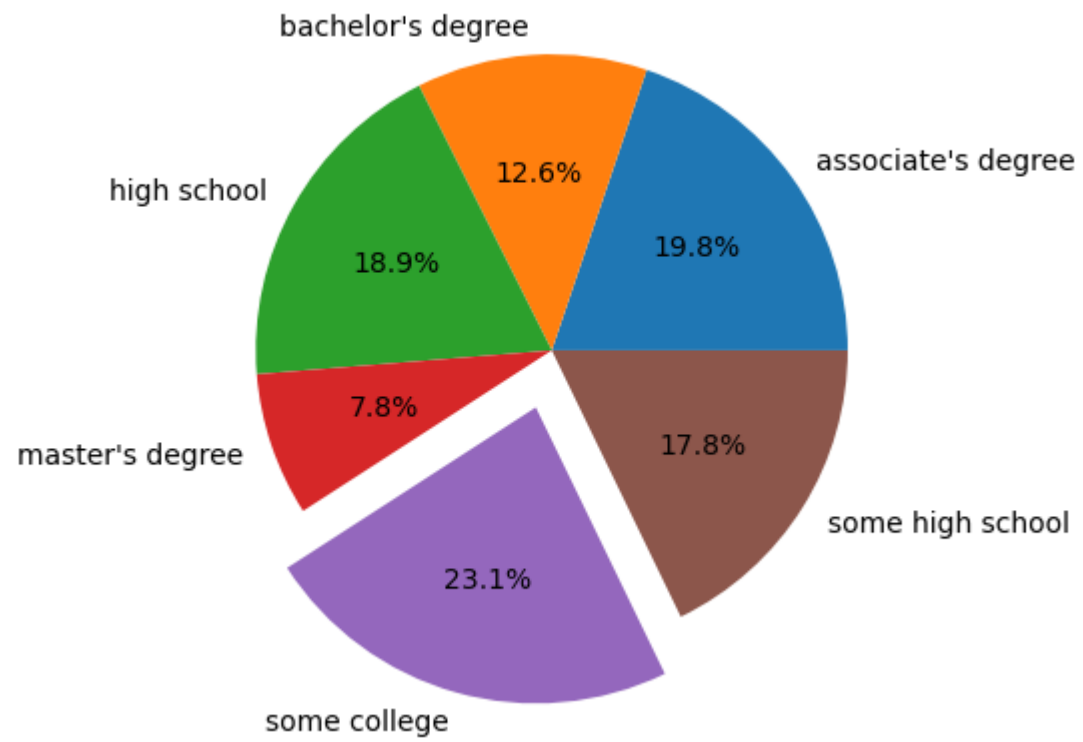
```
In [23]: sns.boxplot(data=df, x='MathScore')  
plt.show()
```



Pie Chart Distribution

```
In [24]: grouped_by = df.groupby("ParentEduc")["WritingScore"].sum()
ex = [0.0,0.0,0.0,0.0,0.2,0.0]
print(grouped_by)
plt.pie(grouped_by.values, labels = grouped_by.index, autopct = "%1.1f%", explode=ex)
plt.show()
```

```
ParentEduc
associate's degree    390160
bachelor's degree    248299
high school          372050
master's degree      154470
some college         454370
some high school     351060
Name: WritingScore, dtype: int64
```



Thank You