```
In [ ]:
```

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

Importing Most Frequently used Libraries for Data Analysis

```python
df= pd.read_csv(r"C:\Users\Appex\Desktop\Data Technical Proj\Expanded_data_with_more_features.csv")
```

```python
df
```

	Unnamed: 0	Gender	EthnicGroup	ParentEduc	LunchType	TestPrep	ParentMaritalStatus	PracticeSport	IsFirstChild	NrSiblings	TransportMeans	WklyStudyHours	MathScore	Rea
0	0	female	NaN	bachelor's degree	standard	none	married	regularly	yes	3.0	school_bus	< 5	71	
1	1	female	group C	some college	standard	NaN	married	sometimes	yes	0.0	NaN	5 - 10	69	
2	2	female	group B	master's degree	standard	none	single	sometimes	yes	4.0	school_bus	< 5	87	
3	3	male	group A	associate's degree	free/reduced	none	married	never	no	1.0	NaN	5 - 10	45	
4	4	male	group C	some college	standard	none	married	sometimes	yes	0.0	school_bus	5 - 10	76	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
30636	816	female	group D	high school	standard	none	single	sometimes	no	2.0	school_bus	5 - 10	59	
30637	890	male	group E	high school	standard	none	single	regularly	no	1.0	private	5 - 10	58	
30638	911	female	NaN	high school	free/reduced	completed	married	sometimes	no	1.0	private	5 - 10	61	
30639	934	female	group D	associate's degree	standard	completed	married	regularly	no	3.0	school_bus	5 - 10	82	
30640	960	male	group B	some college	standard	none	married	never	no	1.0	school_bus	5 - 10	64	

30641 rows × 15 columns

Before Analysis Summarize or Understand Data

```python
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
```

```python
df.columns
```

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

```python
df.describe() ## It gives the Central Tendency of Numerical Fields Present in Dataset
```

	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

```python
df.info() ## Gives the information of each column Data type and Not null count
```

```
<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
```

```python
## Counting Null Values in DataSet
df.isna().sum()
```

```
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
```

Drop unused Fields from DataFrame

```python
df=df.drop('Unnamed: 0',axis=1)
```

```python
df.head()
```

	Gender	EthnicGroup	ParentEduc	LunchType	TestPrep	ParentMaritalStatus	PracticeSport	IsFirstChild	NrSiblings	TransportMeans	WklyStudyHours	MathScore	ReadingScore	Writir
0	female	NaN	bachelor's degree	standard	none	married	regularly	yes	3.0	school_bus	< 5	71	71	
1	female	group C	some college	standard	NaN	married	sometimes	yes	0.0	NaN	5 - 10	69	90	
2	female	group B	master's degree	standard	none	single	sometimes	yes	4.0	school_bus	< 5	87	93	
3	male	group A	associate's degree	free/reduced	none	married	never	no	1.0	NaN	5 - 10	45	56	
4	male	group C	some college	standard	none	married	sometimes	yes	0.0	school_bus	5 - 10	76	78	

As minimum value in Maths subject is zero filtering out their Gender, ParentEduc, ParentMaritalStatus and PracticeSport

```python
df[df['MathScore']==0][['Gender', 'ParentEduc','ParentMaritalStatus', 'PracticeSport']]
```

	Gender	ParentEduc	ParentMaritalStatus	PracticeSport
55	female	some high school	single	regularly

Evaluate the Records of group A ,group B category from Ethnic Group and count the total records

```python
eg=df[df['EthnicGroup'].isin(['group A','group B'])]
eg
```

	Gender	EthnicGroup	ParentEduc	LunchType	TestPrep	ParentMaritalStatus	PracticeSport	IsFirstChild	NrSiblings	TransportMeans	WklyStudyHours	MathScore	ReadingScore
2	female	group B	master's degree	standard	none	single	sometimes	yes	4.0	school_bus	< 5	87	93
3	male	group A	associate's degree	free/reduced	none	married	never	no	1.0	NaN	5 - 10	45	56
5	female	group B	associate's degree	standard	none	married	regularly	yes	1.0	school_bus	5 - 10	73	84
6	female	group B	some college	standard	completed	widowed	never	no	1.0	private	5 - 10	85	93
7	male	group B	some college	free/reduced	none	married	sometimes	yes	1.0	private	> 10	41	43
...	...	...	...	...	...	...	...	...	...	...	...	...	...
30627	female	group A	high school	standard	completed	married	never	no	NaN	school_bus	> 10	58	77
30628	female	group B	NaN	free/reduced	none	single	sometimes	no	1.0	school_bus	5 - 10	55	63
30630	male	group B	associate's degree	free/reduced	none	married	sometimes	no	4.0	private	5 - 10	43	28
30634	male	group A	associate's degree	free/reduced	completed	NaN	sometimes	no	2.0	school_bus	5 - 10	65	60
30640	male	group B	some college	standard	none	married	never	no	1.0	school_bus	5 - 10	64	60

8045 rows × 14 columns

```python
## isin function can be used to filter rows based on multiple values
```

```python
eg.shape[0]
```

```
8045
```

Calculate the students records who have scored greater than 70 marks in each subject

```python
sr=df[(df['MathScore']>70) & (df['ReadingScore']>70) & (df['WritingScore']>70)]
```

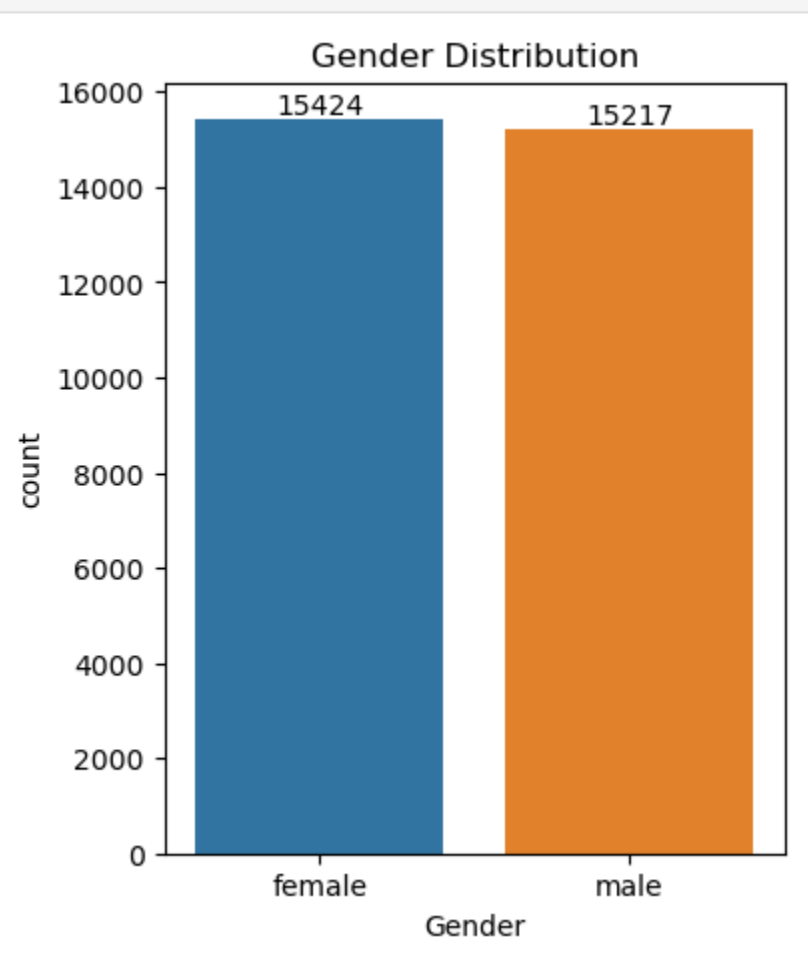
```python
sr
```

	Gender	EthnicGroup	ParentEduc	LunchType	TestPrep	ParentMaritalStatus	PracticeSport	IsFirstChild	NrSiblings	TransportMeans	WklyStudyHours	MathScore	ReadingScore
0	female	NaN	bachelor's degree	standard	none	married	regularly	yes	3.0	school_bus	< 5	71	71
2	female	group B	master's degree	standard	none	single	sometimes	yes	4.0	school_bus	< 5	87	93
4	male	group C	some college	standard	none	married	sometimes	yes	0.0	school_bus	5 - 10	76	78
5	female	group B	associate's degree	standard	none	married	regularly	yes	1.0	school_bus	5 - 10	73	84
6	female	group B	some college	standard	completed	widowed	never	no	1.0	private	5 - 10	85	93
...	...	...	...	...	...	...	...	...	...	...	...	...	...
30620	male	group C	bachelor's degree	standard	none	married	NaN	no	4.0	private	5 - 10	91	85
30623	female	group B	NaN	standard	completed	married	regularly	no	2.0	NaN	5 - 10	75	87
30632	female	group D	some college	standard	none	married	regularly	no	3.0	private	5 - 10	82	88
30633	female	group C	master's degree	standard	completed	married	never	no	2.0	school_bus	5 - 10	84	99
30639	female	group D	associate's degree	standard	completed	married	regularly	no	3.0	school_bus	5 - 10	82	90

9747 rows × 14 columns

Gender Distribution

```python
plt.figure(figsize=(4,5))
plt.title('Gender Distribution')
ax=sns.countplot(data=df, x='Gender')
ax.bar_label(ax.containers[0]) ## Adding Labels to know exact count
plt.show()
```



from the above chart we have analysed that female count is more than male count in the Dataset

Analysing What Parent's Education is Impacting on the Scores of the Students

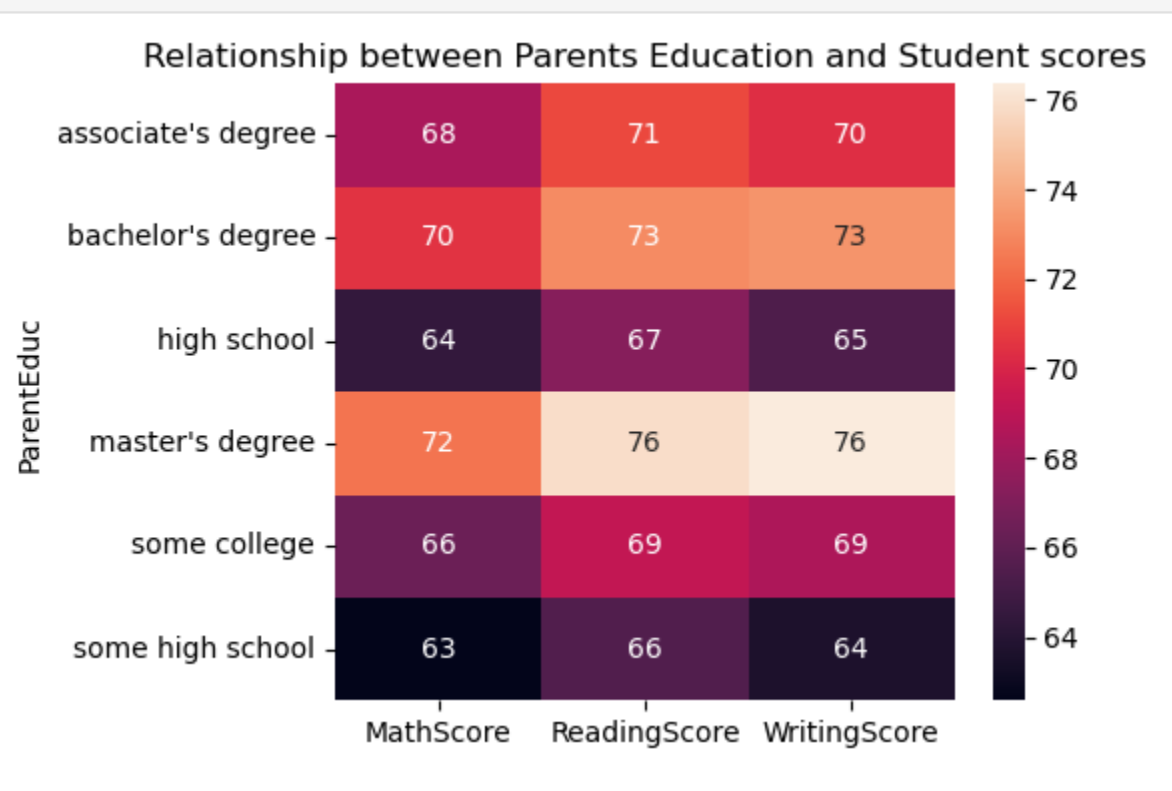
```python
pe=df.groupby('ParentEduc').agg({'MathScore':'mean','ReadingScore':'mean','WritingScore':'mean'})
```

```python
print(pe)
```

```
                    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
```

```python
## Visualizing the above Data with the Heatmap
```

```python
plt.figure(figsize=(5,4))
plt.title('Relationship between Parents Education and Student scores')
sns.heatmap(pe, annot=True) ## with the help of annot parameter we will get the values in to the heatmap
plt.show()
```



from the above chart we have concluded that the education of parent's have a good impact on student score's

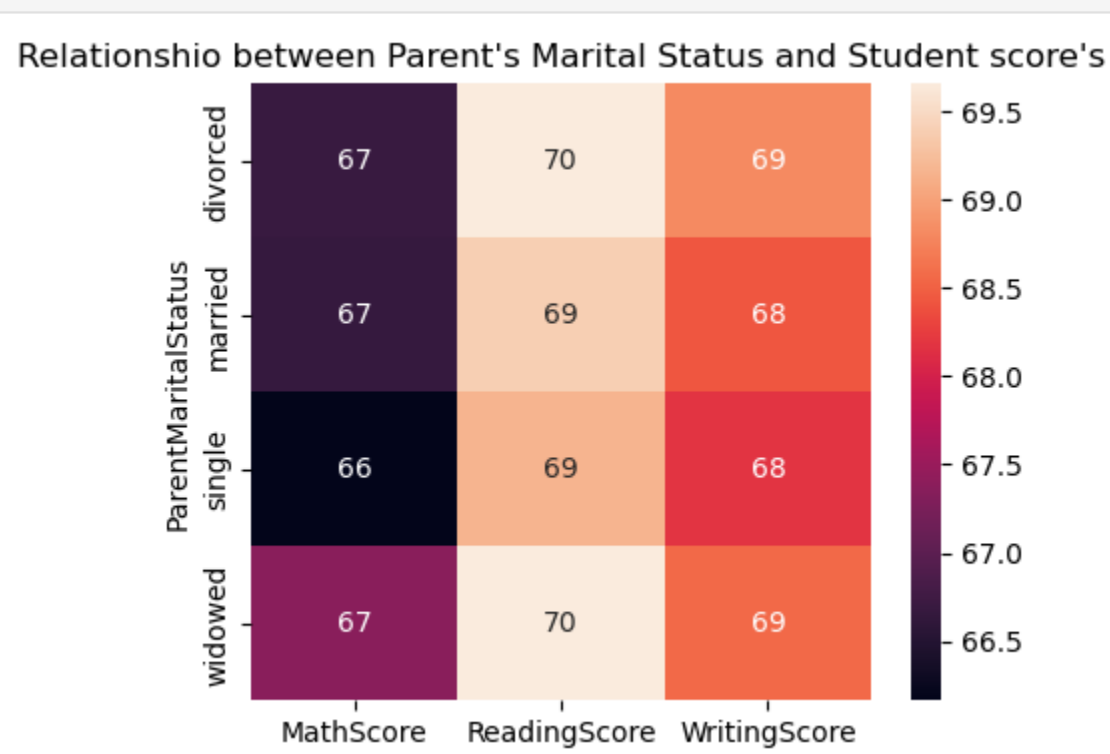
Analysing Does Parent's Marital Status Has Impact on student's score

```python
pm=df.groupby('ParentMaritalStatus').agg({'MathScore':'mean','ReadingScore':'mean','WritingScore':'mean'})
```

```python
pm
```

ParentMaritalStatus	MathScore	ReadingScore	WritingScore
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

```python
plt.figure(figsize=(5,4))
plt.title("Relationshio between Parent's Marital Status and Student score's")
sns.heatmap(pm, annot=True)
plt.show()
```



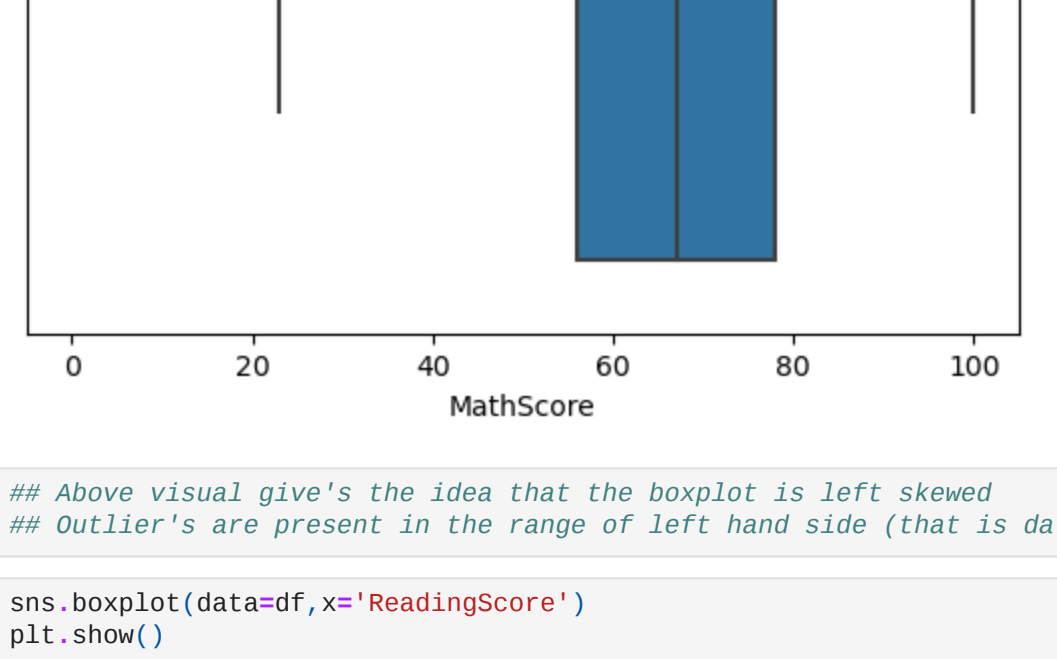

```python
## Score's of student from any category we can see that there's no large difference
```

from the above Analysis we can conclude that Marital Status has no impact on student score's

To Detect Outlier's in the Data

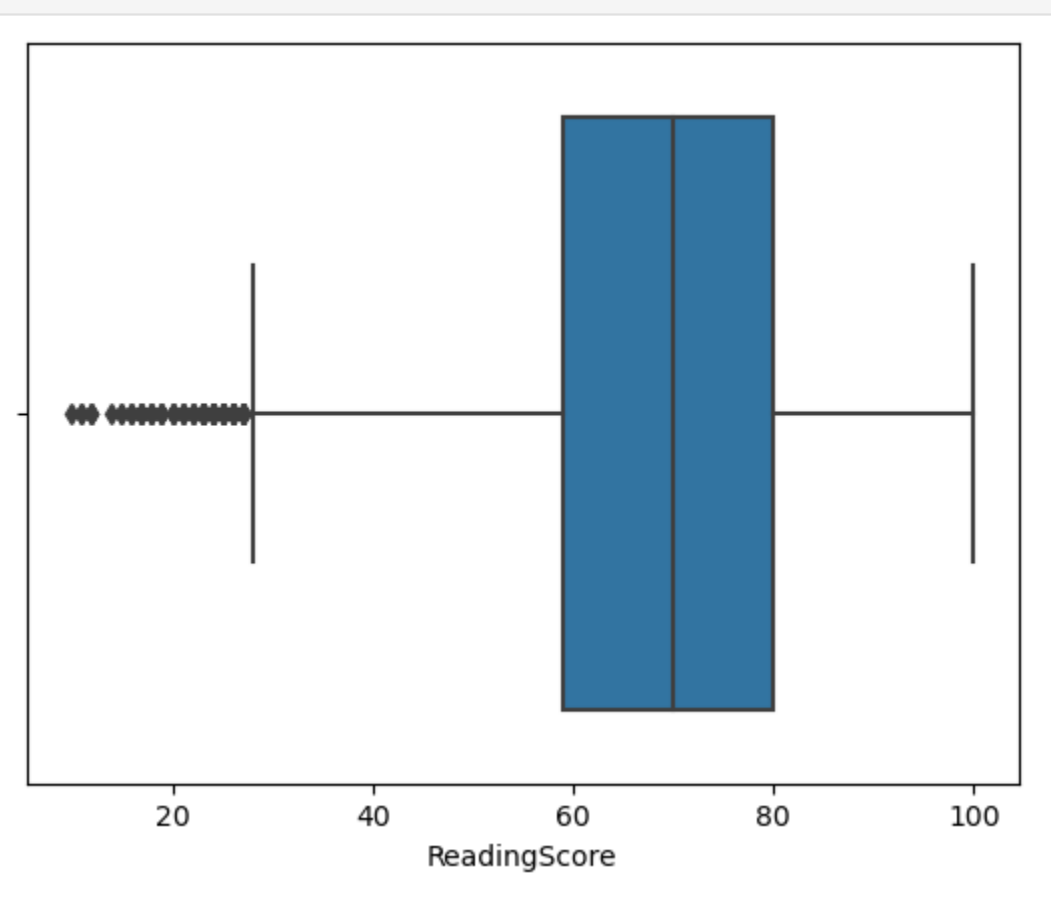
```python
## we can detect outlier's with the help of boxplot
```

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

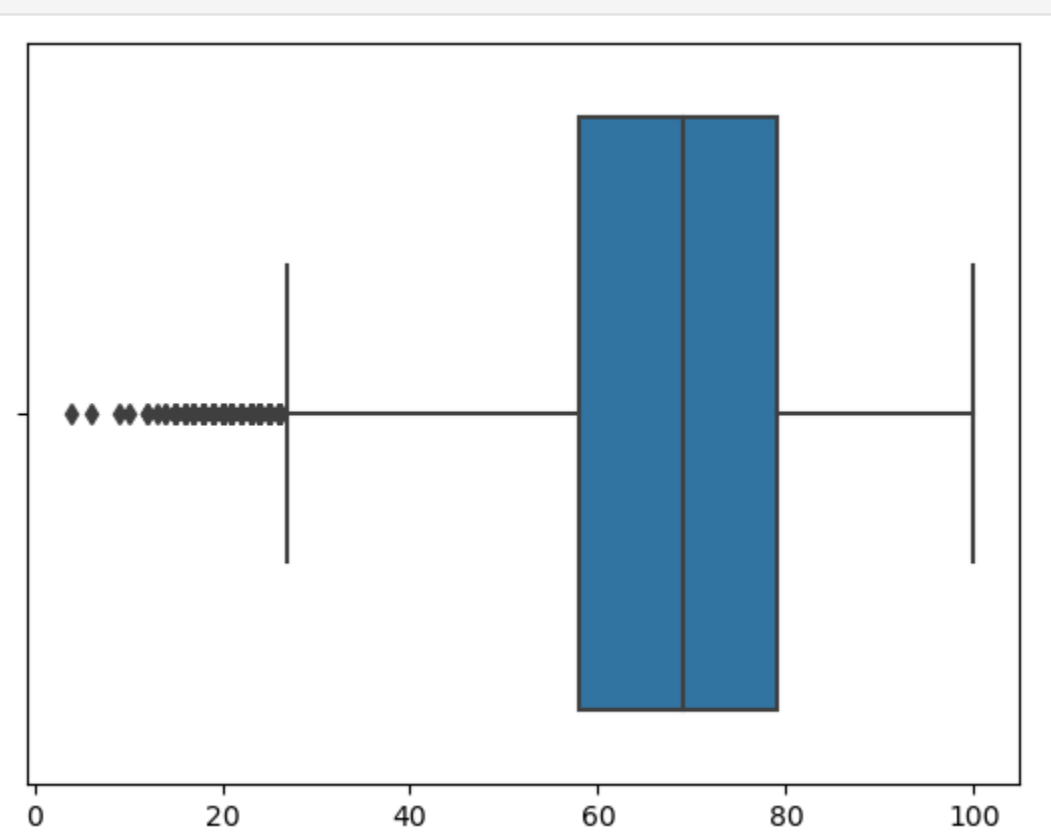



```python
## Above visual give's the idea that the boxplot is left skewed
## Outlier's are present in the range of left hand side (that is datapoints are lying in the range of 0-20 values)
```

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




```python
sns.boxplot(data=df,x='WritingScore')
plt.show()
```



From the above Visual we can conclude that one of the datapoint is lying on zero value in MathScore. Hence student's are facing difficulties to score mark's in Maths Subject

How many group's are present in EthnicGroup

```python
print(df['EthnicGroup'].unique())
```

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

```python
## In the above output we can see the group's
```

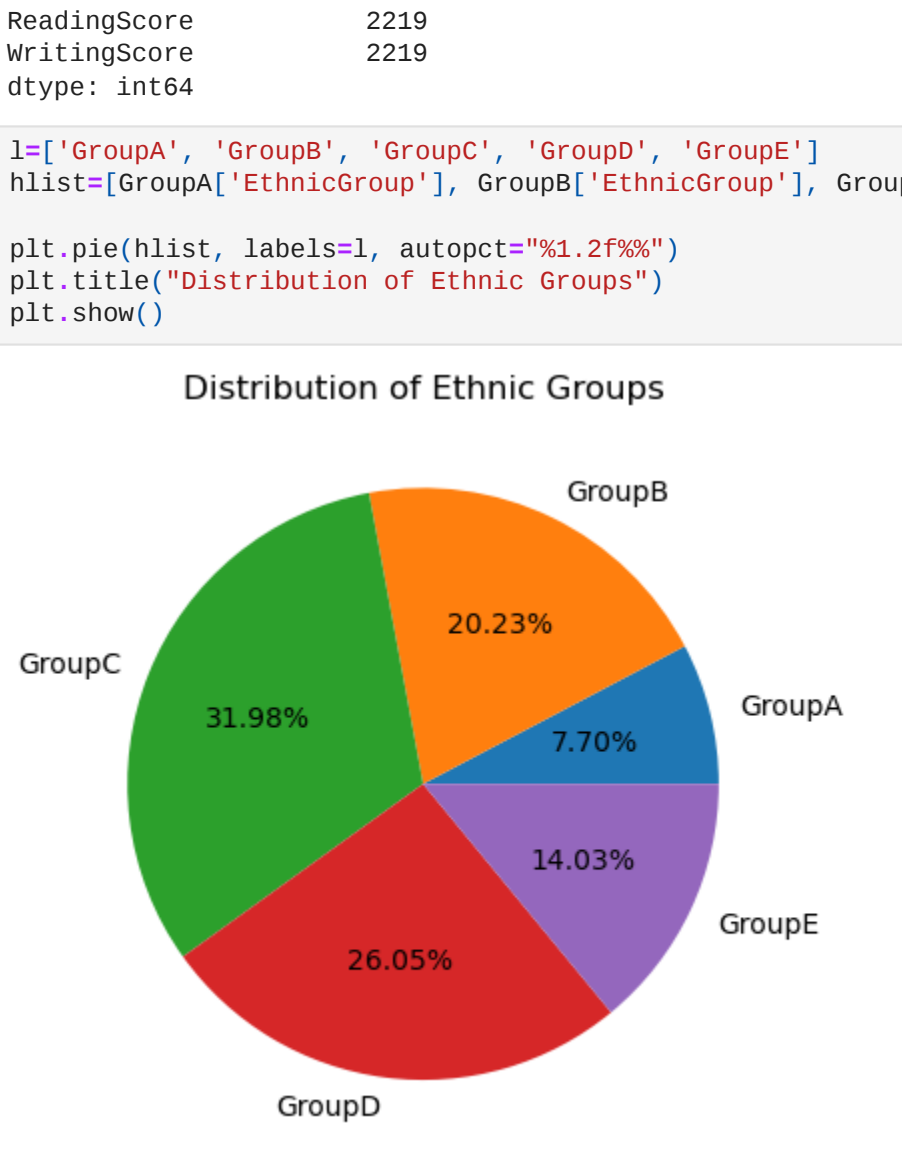
```python
GroupA=df[(df['EthnicGroup']=='group A')].count()
GroupB=df[(df['EthnicGroup']=='group B')].count()
GroupC=df[(df['EthnicGroup']=='group C')].count()
GroupD=df[(df['EthnicGroup']=='group D')].count()
GroupE=df[(df['EthnicGroup']=='group E')].count()
```

```python
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
```

```python
l=['GroupA', 'GroupB', 'GroupC', 'GroupD', 'GroupE']
hlist=[GroupA['EthnicGroup'], GroupB['EthnicGroup'], GroupC['EthnicGroup'] ,GroupD['EthnicGroup'] ,GroupE['EthnicGroup']]

plt.pie(hlist, labels=l, autopct="%1.2f%%")
plt.title("Distribution of Ethnic Groups")
plt.show()
```




```python
print(hlist)
```

```
[2219, 5826, 9212, 7503, 4041]
```

Cross checking the Ethnic Group Values

```python
ab=sns.countplot(data=df ,x="EthnicGroup")
ab.bar_label(ab.containers[0])
plt.show()
```