Student result analysis

September 14, 2023

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

[4]:
```
data = pd.read_csv("C:/Users/Vikas/Desktop/extended.csv")
```

[5]:
```
## i want to see the top 10 dataset
data.head(10)
```

[5]:

	Unnamed: 0	Gender	EthnicGroup	ParentEduc	LunchType	TestPrep	ParentMaritalStatus	PracticeSport	IsFirstChild	NrSiblings	TransportMeans	WklyStudyHours	MathScore	ReadingScore	WritingScore
0	0	female	NaN	bachelor's degree	standard	none	married	regularly	yes	3.0	school_bus	< 5	71	71	74
1	1	female	group C	some college	standard	NaN	married	sometimes	yes	0.0	NaN	5 - 10	69	90	88
2	2	female	group B	master's degree	standard	none	single	sometimes	yes	4.0	school_bus	< 5	87	93	91
3	3	male	group A	associate's degree	free/reduced	none	married	never	no	1.0					
4	4	male	group C	some college	standard	none	married	sometimes	yes	0.0					
5	5	female	group B	associate's degree	standard	none	married	regularly	yes	1.0					
6	6	female	group B	some college	standard	completed	widowed	never	no	1.0					
7	7	male	group B	some college	free/reduced	none	married	sometimes	yes	1.0					
8	8	male	group D	high school	free/reduced	completed	single	sometimes	no	3.0					
9	9	female	group B	high school	free/reduced	none	married	regularly	yes	NaN					

```
3           NaN         5 - 10         45            56            42
4     school_bus         5 - 10         76            78            75
5     school_bus         5 - 10         73            84            79
6        private         5 - 10         85            93            89
7        private           > 10         41            43            39
8        private           > 10         65            64            68
9        private            < 5         37            59            50
```

[45]:
```
## i want to see the bottom 5 rows
data.tail()
```

[45]:
```
       Gender EthnicGroup          ParentEduc     LunchType   TestPrep  \
30636  female     group D         high school      standard       none
30637    male     group E         high school      standard       none
30638  female         NaN         high school  free/reduced  completed
30639  female     group D  associate's degree      standard  completed
30640    male     group B        some college      standard       none

      ParentMaritalStatus PracticeSport IsFirstChild  NrSiblings  \
30636              single     sometimes           no         2.0
30637              single     regularly           no         1.0
30638             married     sometimes           no         1.0
30639             married     regularly           no         3.0
30640             married         never           no         1.0

      TransportMeans WklyStudyHours  MathScore  ReadingScore  WritingScore
30636     school_bus         5 - 10         59            61            65
30637        private         5 - 10         58            53            51
30638        private         5 - 10         61            70            67
30639     school_bus         5 - 10         82            90            93
30640     school_bus         5 - 10         64            60            58
```

[6]:
```
## i want to see the numerical col min max count 25% 50%
data.describe()
```

[6]:
```
         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
```

[7]:
```
## i want to see the which type of data i have and i want to see null values of␣
 ↪each col
```

```
data.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
```

[8]:
```
## i wan to see null value of each col
data.isnull().sum()
```

[8]:
```
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
```

[15]:
```
## i want to delet the unnamed col
data = data.drop("Unnamed: 0", axis=1)
data.head()
```

[15]:
```
   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
```

[47]:
```
data.columns
```

[47]:
```
Index(['Gender', 'EthnicGroup', 'ParentEduc', 'LunchType', 'TestPrep',
       'ParentMaritalStatus', 'PracticeSport', 'IsFirstChild', 'NrSiblings',
       'TransportMeans', 'WklyStudyHours', 'MathScore', 'ReadingScore',
       'WritingScore'],
      dtype='object')
```

[ ]:
```
## change the weekly study hour col
data['WklyStudyHours'] = data['WklyStudyHours'].str.replace("05-Oct",5-10)
data.head()
```

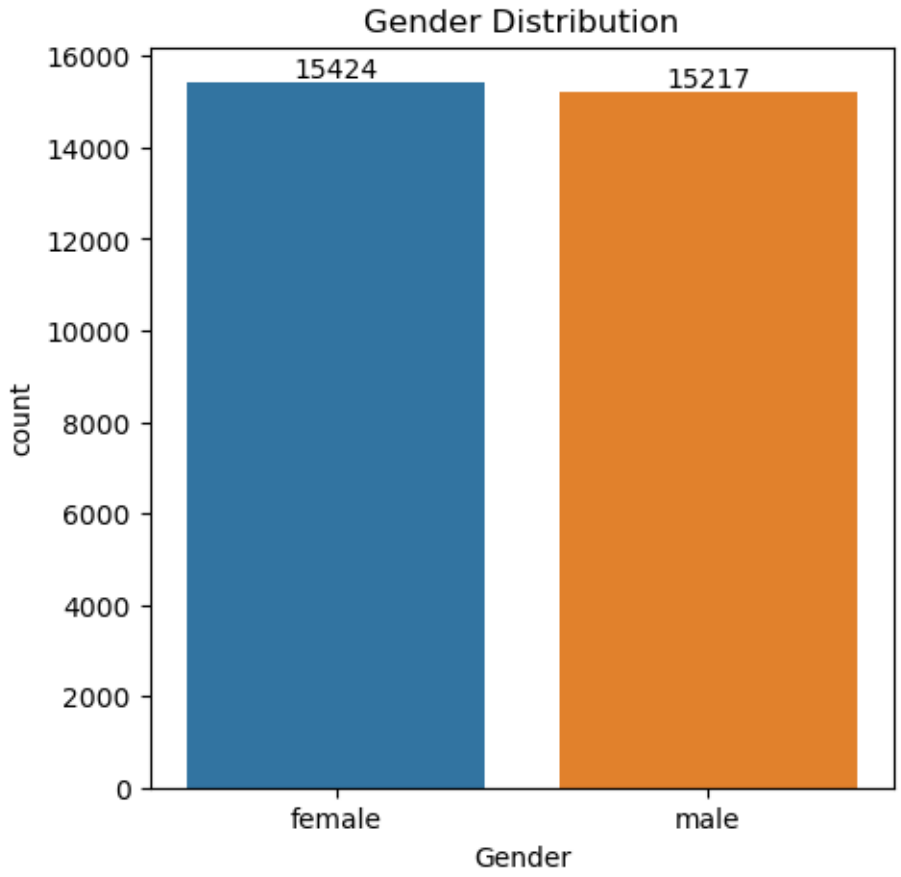
[18]:
```
data.head()
```

[18]:
```
   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
```

[ ]:

[31]:
```
plt.figure(figsize=(5,5))
ax=sns.countplot(data=data,x='Gender')
ax.bar_label(ax.containers[0])
plt.title("Gender Distribution")
plt.show()
```



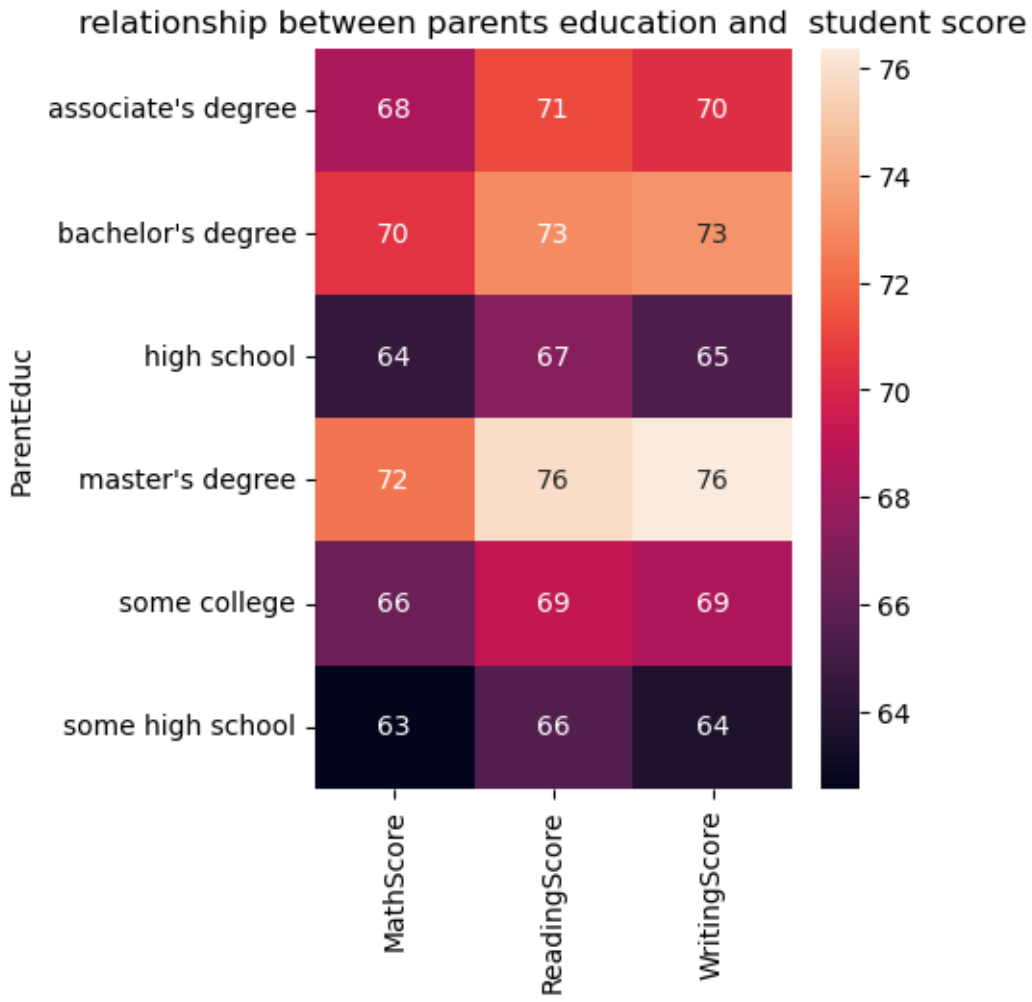
0.1 from the above chart wehave analysed that the number of female in the data is the more then the number of males

[22]:
```
gb = data.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
```

[32]:
```
plt.figure(figsize=(4,5))
sns.heatmap(gb,annot=True)
plt.title("relationship between parents education and  student score")
plt.show
```

[32]:
```
<function matplotlib.pyplot.show(close=None, block=None)>
```



0.2 from the above chart we have concludedthat the education of the parents we have a good impact

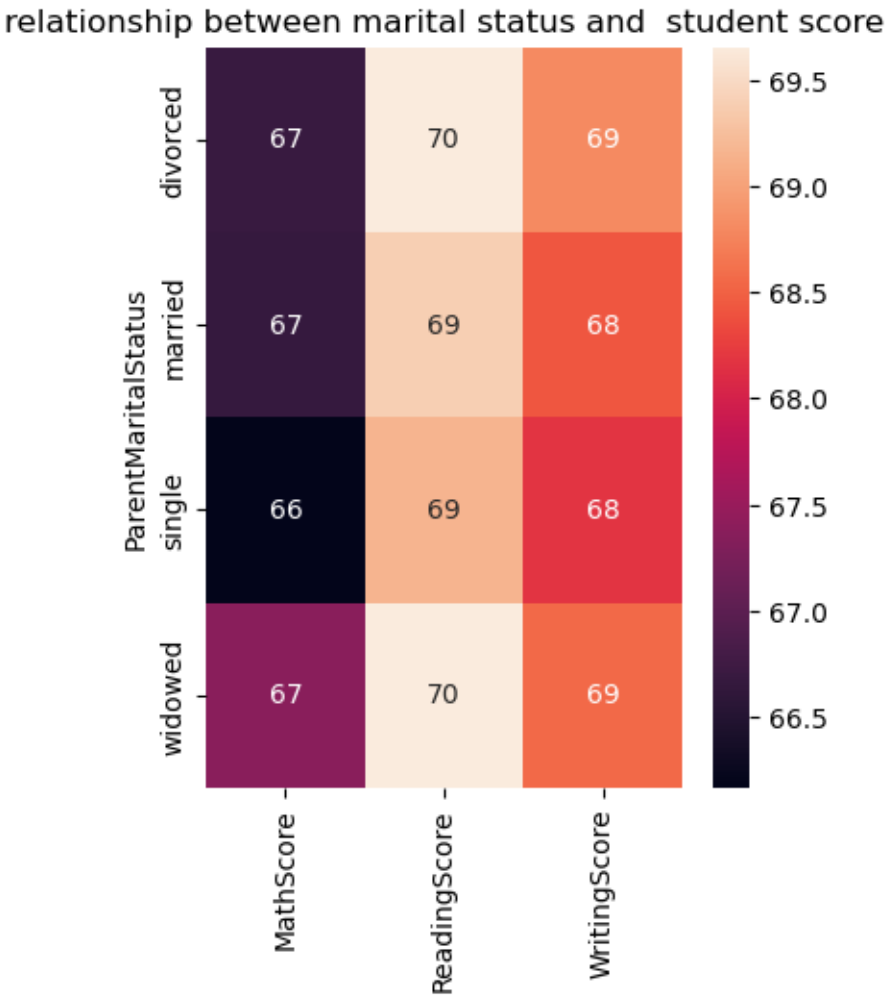
[27]:
```
gb1 = data.groupby("ParentMaritalStatus").agg({"MathScore":
  ↪'mean',"ReadingScore":'mean',"WritingScore":'mean'})
print(gb1)
```

```
                    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
```

[33]:
```
plt.figure(figsize=(4,5))
sns.heatmap(gb1,annot=True)
```

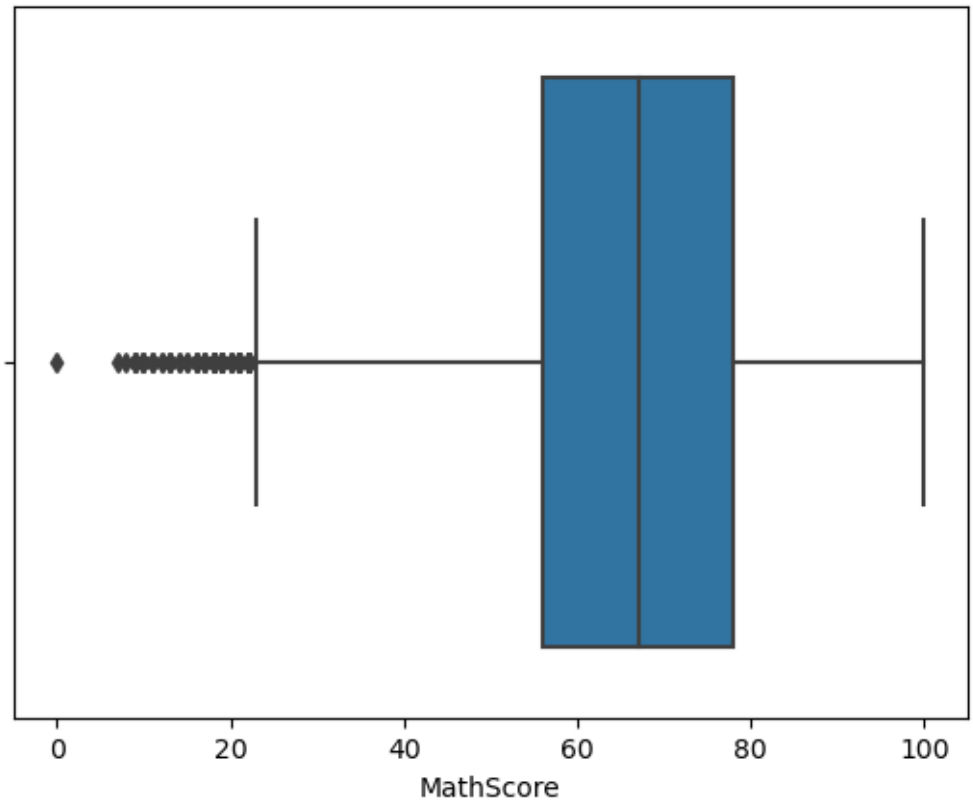
```
plt.title("relationship between marital status and  student score")
plt.show
```

[33]: <function matplotlib.pyplot.show(close=None, block=None)>



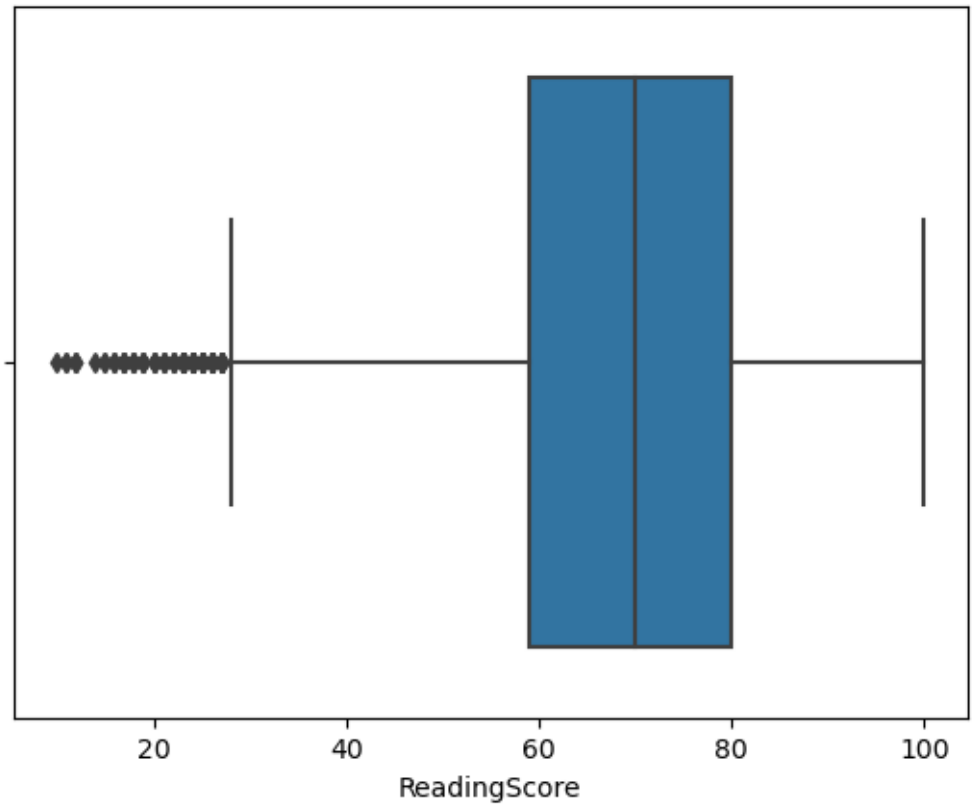
0.3 from the above chart we have concluded that there is no/negligible impact on the student score due to parent marital status

[43]:
```
sns.boxplot(data=data,x="MathScore")
plt.show()
```

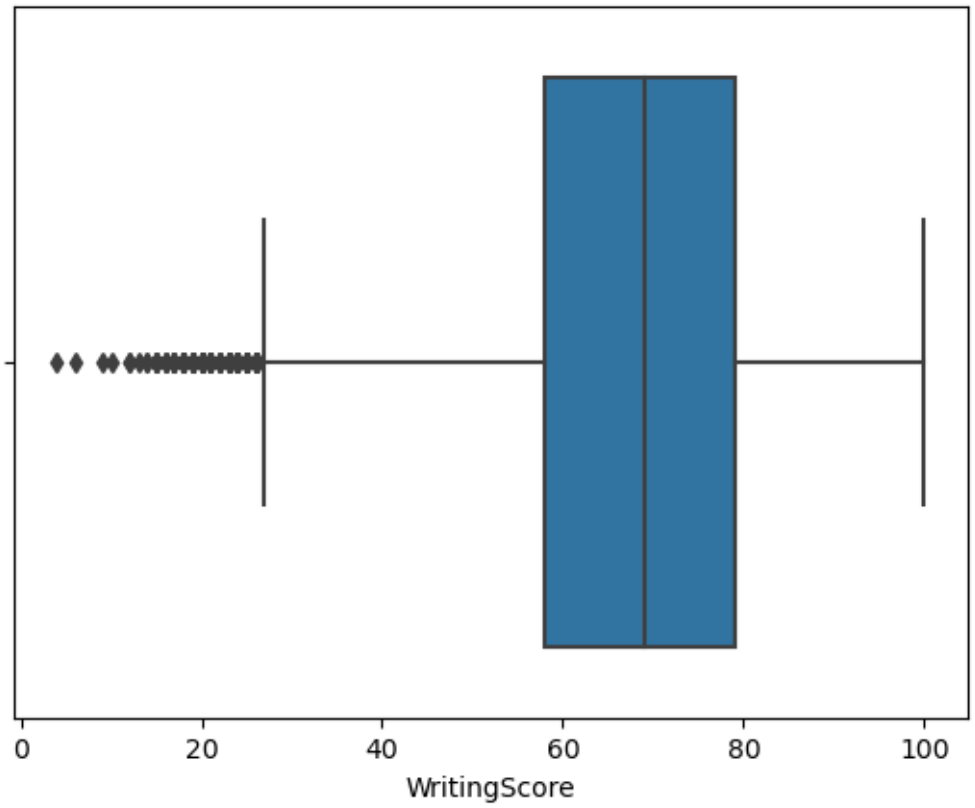


[44]:
```
sns.boxplot(data=data,x="ReadingScore")
plt.show
```

[44]:
```
<function matplotlib.pyplot.show(close=None, block=None)>
```



[48]:
```
sns.boxplot(data=data,x="WritingScore")
plt.show()
```



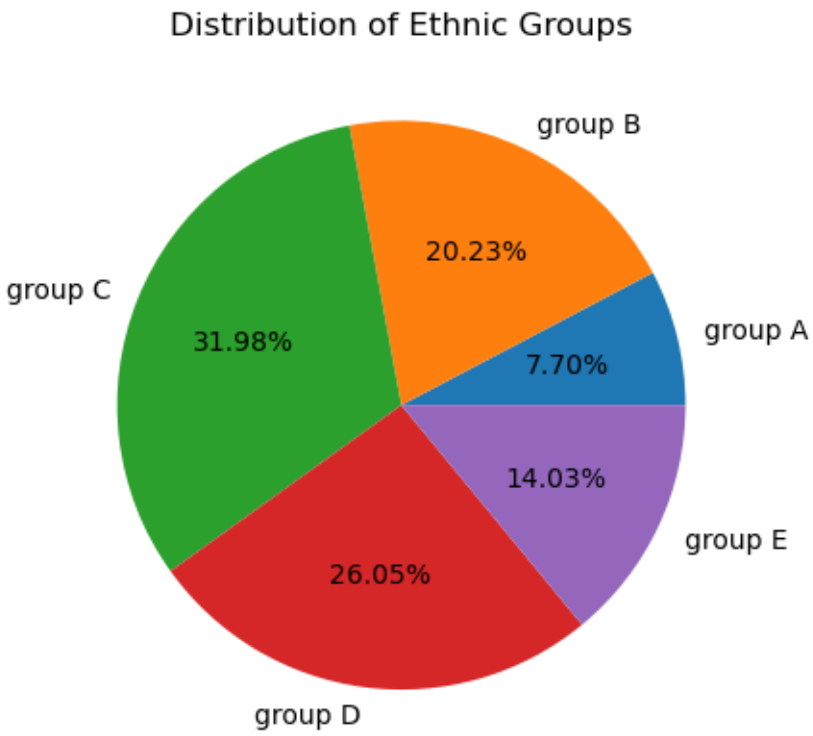
[49]:
```
print(data['EthnicGroup'].unique())
```

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

1 Distribution of ethnic group

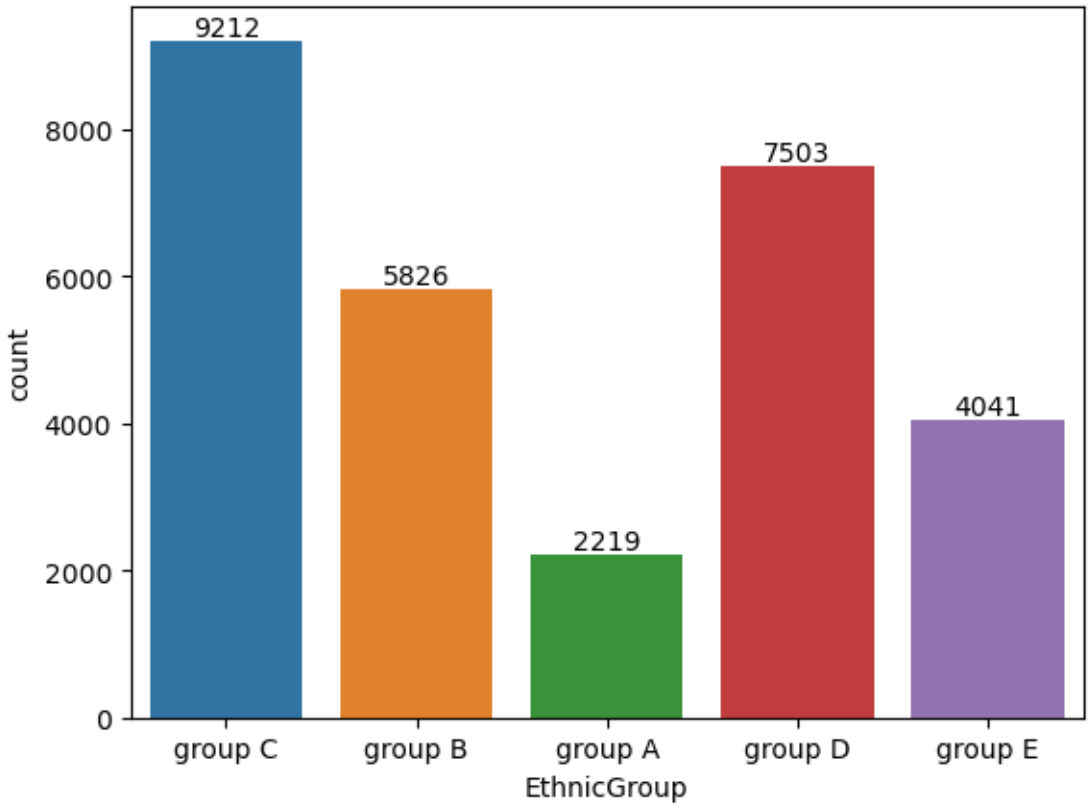
[64]:
```
groupA = data.loc[(data['EthnicGroup']=="group A")].count()
groupB = data.loc[(data['EthnicGroup']=="group B")].count()
groupC = data.loc[(data['EthnicGroup']=="group C")].count()
groupD = data.loc[(data['EthnicGroup']=="group D")].count()
groupE = data.loc[(data['EthnicGroup']=="group E")].count()
l = ["group A","group B","group C","group D","group E"]
mlist =␣
 ↪[groupA['EthnicGroup'],groupB['EthnicGroup'],groupC['EthnicGroup'],groupD['EthnicGroup'],gr
print(mlist)
plt.pie(mlist,labels = l,autopct ="%1.2f%%")
plt.title("Distribution of Ethnic Groups")
plt.show()
```

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



[66]:
```
ax = sns.countplot(data=data,x="EthnicGroup")
ax.bar_label(ax.containers[0])
```

[66]:
```
[Text(0, 0, '9212'),
 Text(0, 0, '5826'),
 Text(0, 0, '2219'),
 Text(0, 0, '7503'),
 Text(0, 0, '4041')]
```


9212
7503
5826
4041
2219
8000
6000
4000
2000
0
count
group C
group B
group A
group D
group E
EthnicGroup

[ ]: