Install and Import pandas, numpy, matplotlib, seaborn:

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

Load CSV File:

In [116…
```
df= pd.read_csv("Student_scores.csv")
df
```

Information of Dataset:

In [11]:
```
df.describe()
```

Out[11]:

	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

Checking Data type:

In [12]:
```
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
```

Checking NULL Values:

In [14]:
```
df.isnull().sum()
```

Out[14]:
```
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 Unnamed Column:

In [17]:
```
df = df.drop("Unnamed: 0", axis=1)
df
```

Out[17]:

	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	71	71	74
1	female	group C	some college	standard	NaN	married	sometimes	yes	0.0	NaN	5 - 10	69	90	88
2	female	group B	master's degree	standard	none	single	sometimes	yes	4.0	school_bus	< 5	87	93	91
3	male	group A	associate's degree	free/reduced	none	married	never	no	1.0	NaN	5 - 10	45	56	42
4	male	group C	some college	standard	none	married	sometimes	yes	0.0	school_bus	5 - 10	76	78	75
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
30636	female	group D	high school	standard	none	single	sometimes	no	2.0	school_bus	5 - 10	59	61	65
30637	male	group E	high school	standard	none	single	regularly	no	1.0	private	5 - 10	58	53	51
30638	female	NaN	high school	free/reduced	completed	married	sometimes	no	1.0	private	5 - 10	61	70	67
30639	female	group D	associate's degree	standard	completed	married	regularly	no	3.0	school_bus	5 - 10	82	90	93
30640	male	group B	some college	standard	none	married	never	no	1.0	school_bus	5 - 10	64	60	58

30641 rows × 14 columns

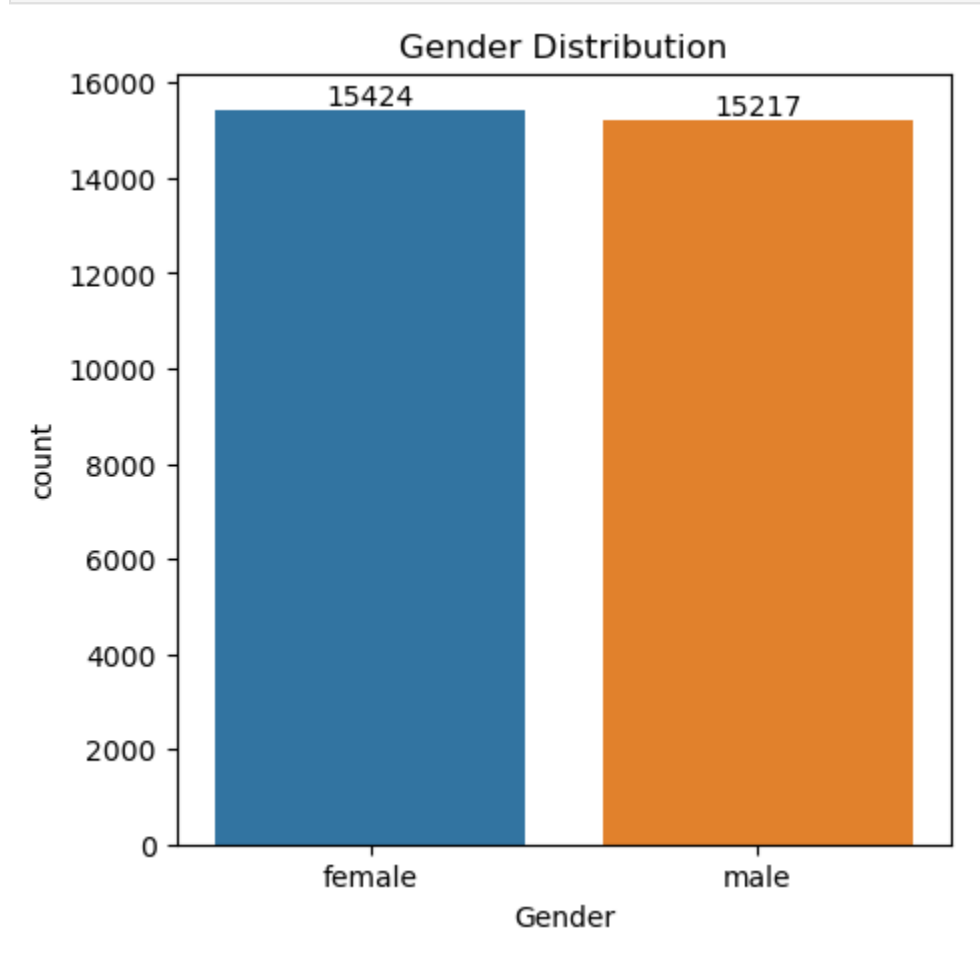
Change Weekly Hours Column Data:

In [ ]:
```
#df["WklyStudyHours"] = df["WklyStudyHours"].str.replace("where you want to replace")
```

ANALYSIS OF DATA

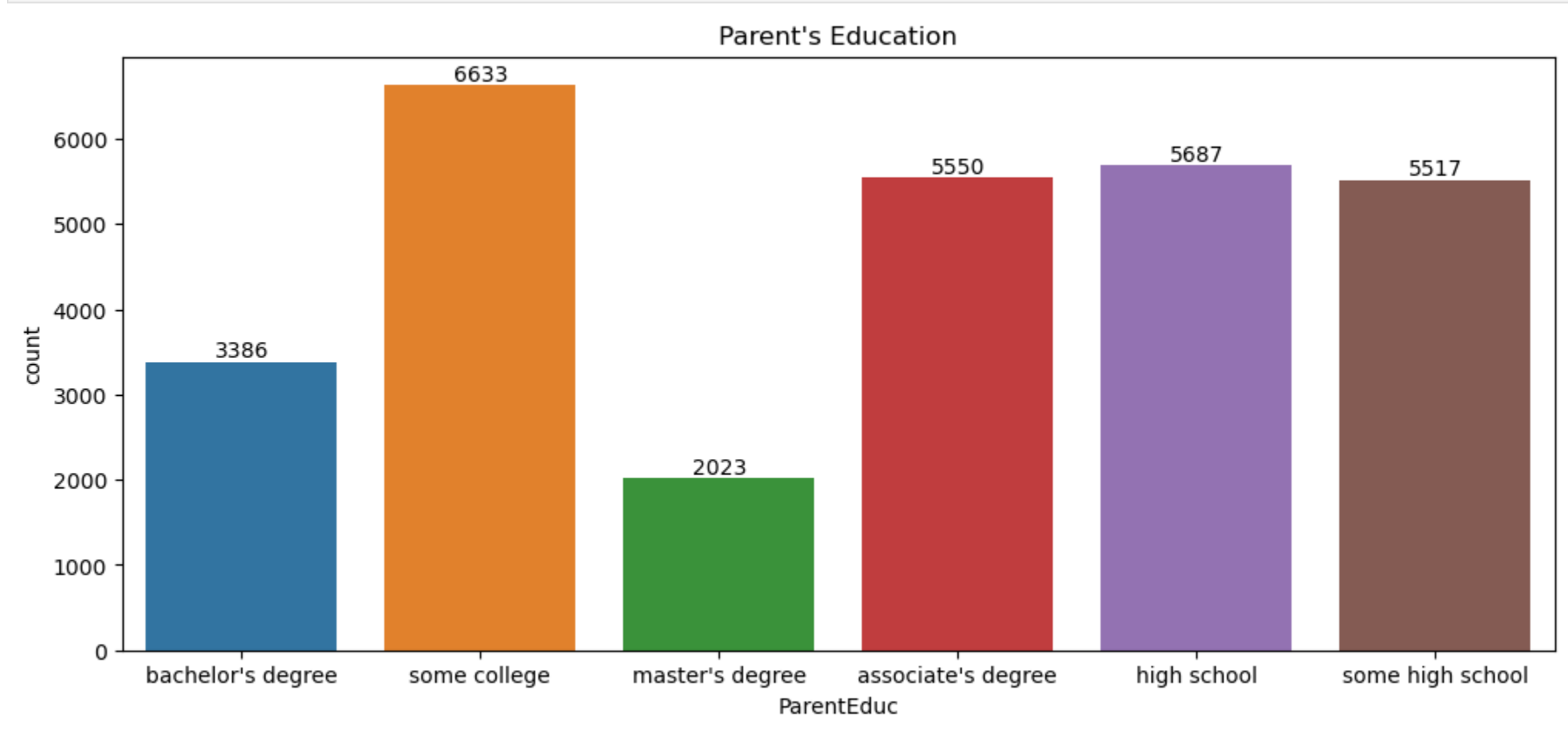
Gender Distribution:

In [79]:
```
plt.figure(figsize = (5,5))
ax = sns.countplot(data = df, x = "Gender")
ax.bar_label(ax.containers[0])
plt.title("Gender Distribution")
plt.show()
```



From the above chart, we have analysed that there are more females in the school as compare to the males.

In [78]:
```
plt.figure(figsize=(12,5))
ax0 = sns.countplot(data = df, x= "ParentEduc")
ax0.bar_label(ax0.containers[0])
plt.title("Parent's Education")
plt.show()
```

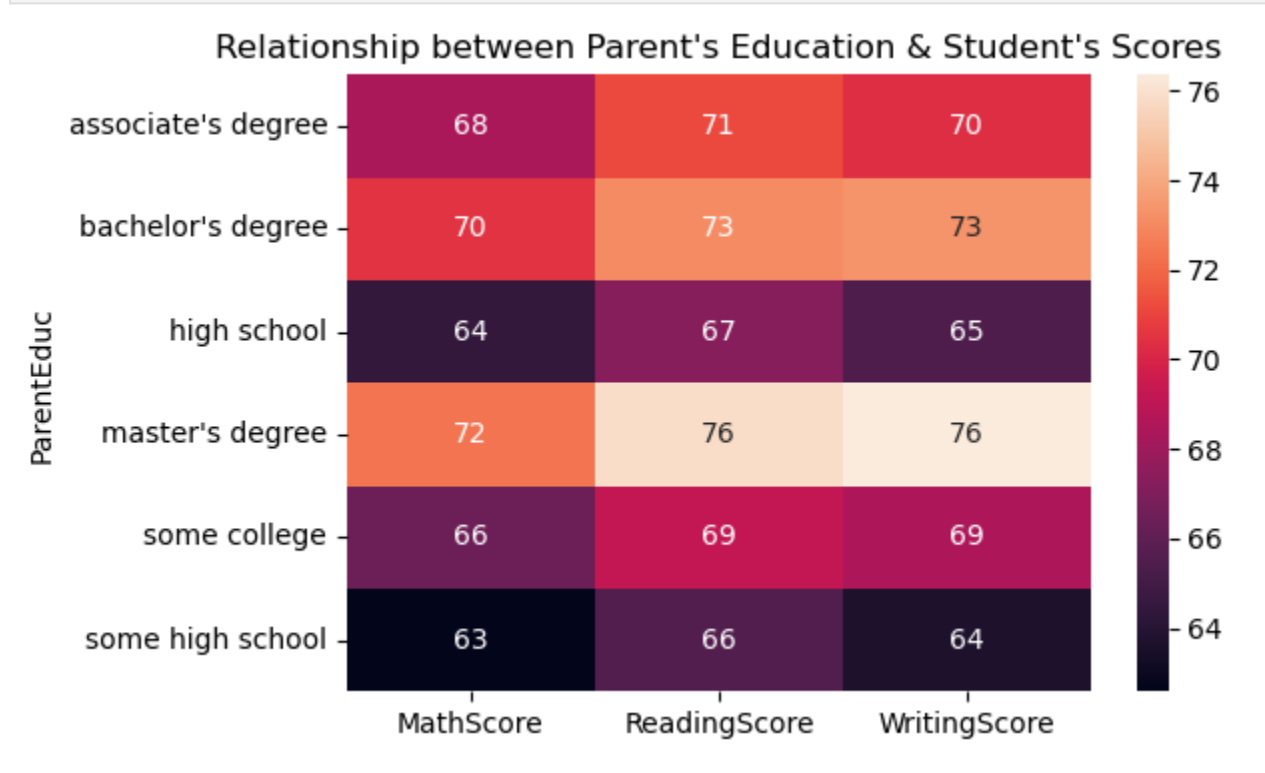


Impact of Parent's Education on Student's Scores:

In [54]:
```
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
```

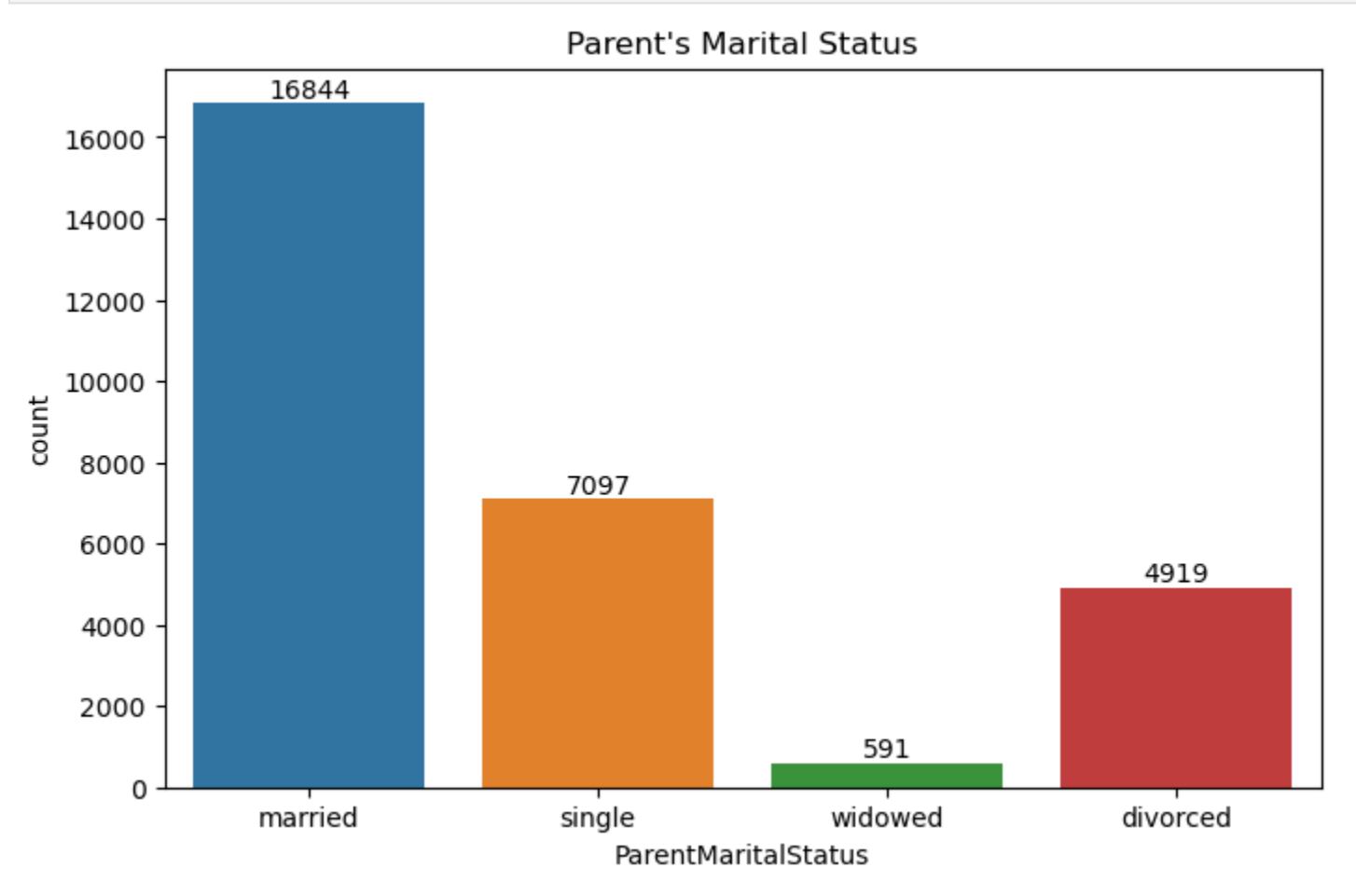
In [77]:
```
plt.figure(figsize=(6,4))
sns.heatmap(gb, annot= True)
plt.title("Relationship between Parent's Education & Student's Scores")
plt.show()
```



From the above chart, we conculded that the educaton of parents have a great impact on their scores.

Marital Status:

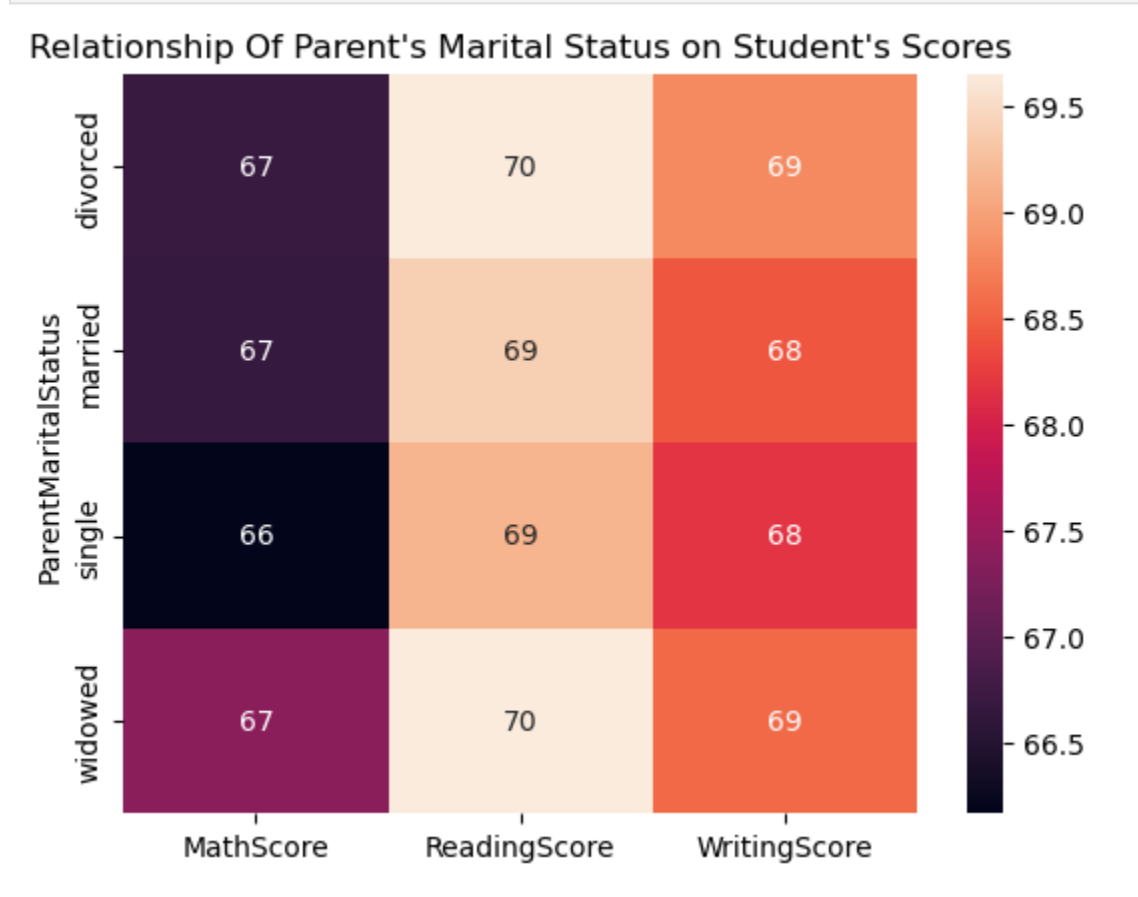
In [76]:
```
plt.figure(figsize = (8,5))
ax1 = sns.countplot(data = df, x = "ParentMaritalStatus" )
ax1.bar_label(ax1.containers[0])
plt.title("Parent's Marital Status")
plt.show()
```



In [65]:
```
gb1 = df.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
```

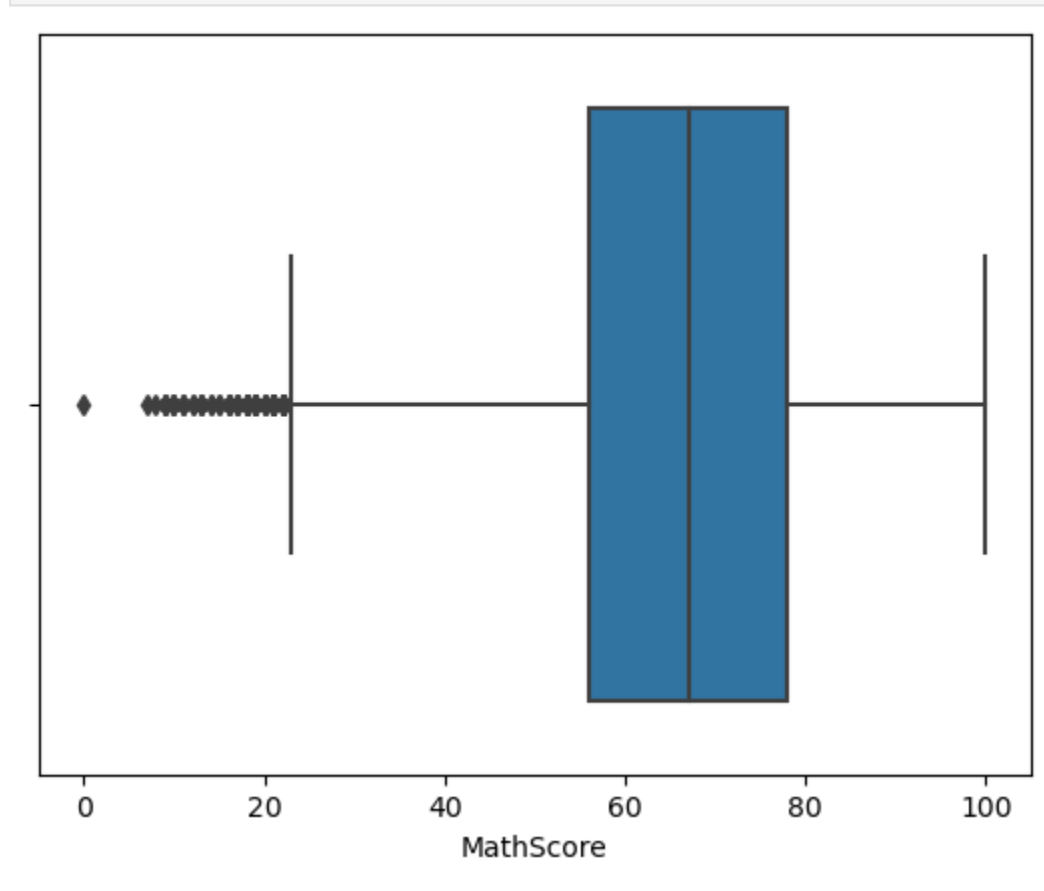
In [81]:
```
sns.heatmap(gb1, annot = True)
plt.title("Relationship Of Parent's Marital Status on Student's Scores")
plt.show()
```



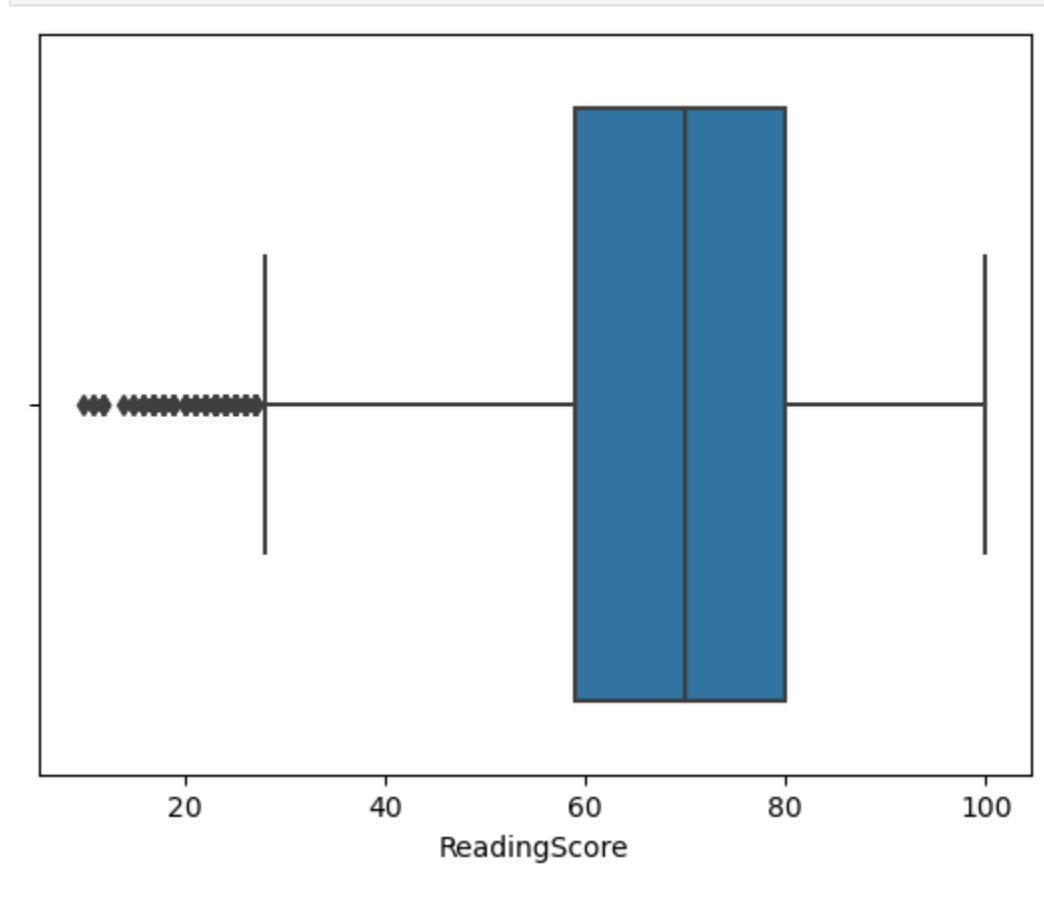
From the above chart, we have concluded that there is no impact on the score on the basis of Parents Marital Status.

Subject Scores:

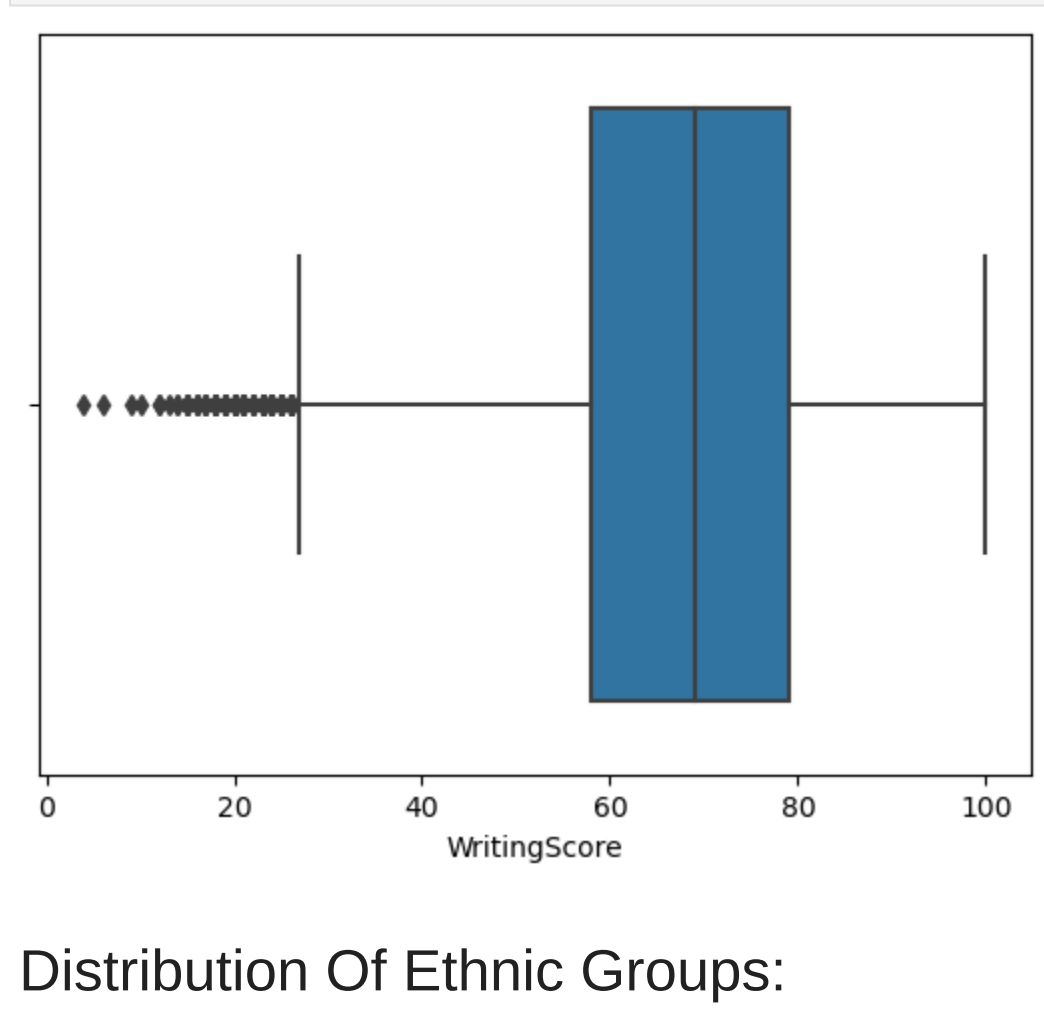
In [87]:
```
sns.boxplot(data=df, x= "MathScore")
plt.show()
```



In [88]:
```
sns.boxplot(data= df, x= "ReadingScore")
plt.show()
```



In [89]:
```
sns.boxplot(data=df, x="WritingScore")
plt.show()
```



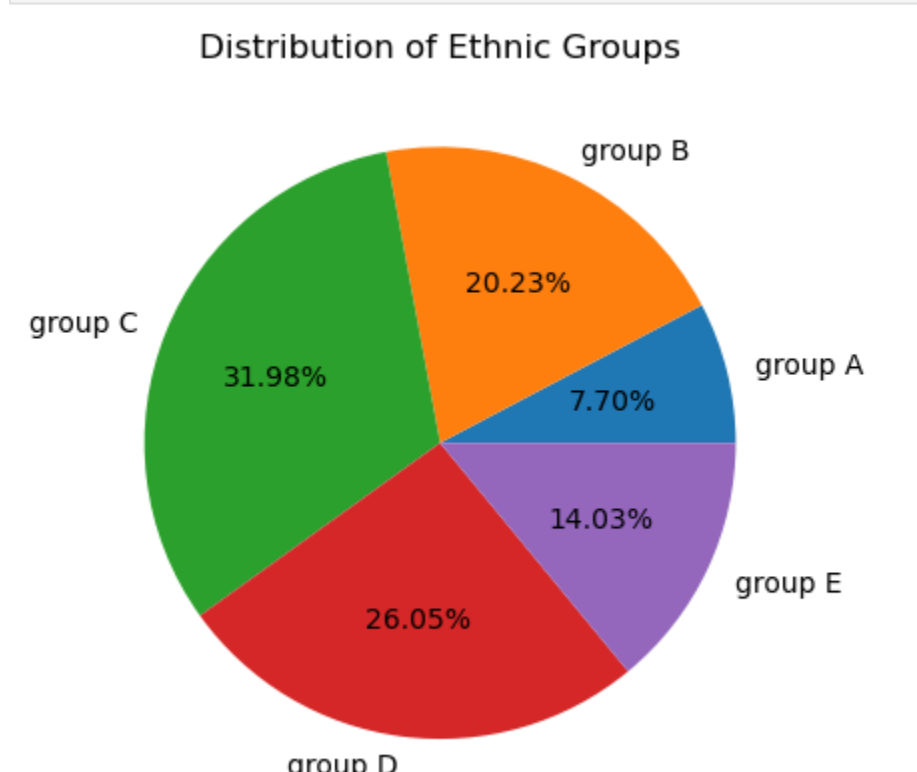
Distribution Of Ethnic Groups:

In [90]:
```
print(df["EthnicGroup"].unique())
```

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

In [110…
```
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"]
nlist = [groupA["EthnicGroup"], groupB["EthnicGroup"], groupC["EthnicGroup"], groupD["EthnicGroup"], groupE["EthnicGroup"]]
plt.pie(nlist, labels = L, autopct = "%1.2f%%")
plt.title("Distribution of Ethnic Groups")
plt.show()
```



From the above chart, we analysed that major Ethnic Group is Group C (31.98%), and then Group D (26.05%), Group B (20.23%), Group E (14.03%), Group A (7.70%)

CONCLUSION

From the above analysis, we concluded that there are more female students as compare to the male students. Although, the parent's education impacted the student;s score. Most of the students get low scores in Math as compare to the reading and writing scores. However, there is no impact on the scores on the basis of a parent's marital status.

LinkedIn: https://www.linkedin.com/in/haroonariff/

Thankyou