EDA on Student Score Data Set

Using Python

By VISHAL KURHADE

Name: Vishal Kurhade

Email-id: kurhadev772@gmail.com

Contact No: +91 9834495309

https://github.com/VishalKurhade?tab=repositories

https://www.linkedin.com/in/vishal-kurhade-183a591b0/

Contents

1. Introduction
2. Problem Statement
3. Importing libraries & data
4. Dropping Unnamed: 0 Column
5. Data Info
6. Describe
7. Gender Distribution
8. Relation between Parent education and student score
9. Impact of Parental Marital Status on Student Scores
10. Distribution of students by Ethnic group.
11. Relation between the number of siblings and student score
12. Effect of test preparation on student’s score
13. Conclusion

1. Introduction

Hello there! My name is Vishal Kurhade.

I would like to share with you about a project I worked on, where I performed Exploratory Data Analysis on a Student Score dataset using Python. I utilized popular libraries like Pandas, NumPy, Seaborn and Matplotlib in order to extract valuable insights. By cleaning the data and analyzing it, I was able to build visually appealing graphs and charts to uncover patterns and trends.

2. Problem Statement

This exploratory data analysis (EDA) report aims to investigate the factors that have the most significant influence on students' test scores in a given dataset. By leveraging Python programming language and appropriate statistical techniques, I seek to identify and understand the key features that contribute to variations in student’s academic performance.

Through comprehensive data exploration and analysis, I aim to uncover patterns, correlations, and insights that can inform educational stakeholders about the factors that play a pivotal role in determining test scores. Ultimately, this analysis will empower decision-makers in education to better understand the nuances of student performance and tailor interventions or strategies to improve academic outcomes effectively.

EDA on Student Score Dataset by VISHAL KURHADE

In [3]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
```

In [4]:
```
df=pd.read_csv('student_dataset.csv')
```

In [22]:
```
df.head()
```

Out[22]:

	Unnamed: 0	Gender	EthnicGroup	ParentEduc	LunchType	TestPrep	ParentMaritalStatus	PracticeSport	IsFirstChild	NrSiblings	TransportMeans
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

EDA on Student Score Dataset by VISHAL KURHADE

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

]:
```
df=pd.read_csv('student_dataset.csv')
```

]:
```
df.head()
```

]:

duc	LunchType	TestPrep	ParentMaritalStatus	PracticeSport	IsFirstChild	NrSiblings	TransportMeans	WklyStudyHours	MathScore	ReadingScore	WritingScore
or's ree	standard	none	married	regularly	yes	3.0	school_bus	< 5	71	71	74
me ege	standard	NaN	married	sometimes	yes	0.0	NaN	5 - 10	69	90	88
er's ree	standard	none	single	sometimes	yes	4.0	school_bus	< 5	87	93	91
te's ree	free/reduced	none	married	never	no	1.0	NaN	5 - 10	45	56	42
me ege	standard	none	married	sometimes	yes	0.0	school_bus	5 - 10	76	78	75

4. Dropping Unnamed: 0 Column

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

```
Out[8]: Index(['Unnamed: 0', 'Gender', 'EthnicGroup', 'ParentEduc', 'LunchType',
               'TestPrep', 'ParentMaritalStatus', 'PracticeSport', 'IsFirstChild',
               'NrSiblings', 'TransportMeans', 'WklyStudyHours', 'MathScore',
               'ReadingScore', 'WritingScore'],
              dtype='object')
```

```
In [14]: df.drop(['Unnamed: 0'],axis=1, inplace=True)
```

In [16]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30641 entries, 0 to 30640
Data columns (total 14 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   Gender                30641 non-null  object 
 1   EthnicGroup           28801 non-null  object 
 2   ParentEduc            28796 non-null  object 
 3   LunchType             30641 non-null  object 
 4   TestPrep              28811 non-null  object 
 5   ParentMaritalStatus   29451 non-null  object 
 6   PracticeSport         30010 non-null  object 
 7   IsFirstChild          29737 non-null  object 
 8   NrSiblings            29069 non-null  float64
 9   TransportMeans        27507 non-null  object 
 10  WklyStudyHours        29686 non-null  object 
 11  MathScore             30641 non-null  int64  
 12  ReadingScore          30641 non-null  int64  
 13  WritingScore          30641 non-null  int64  
dtypes: float64(1), int64(3), object(10)
memory usage: 3.3+ MB
```

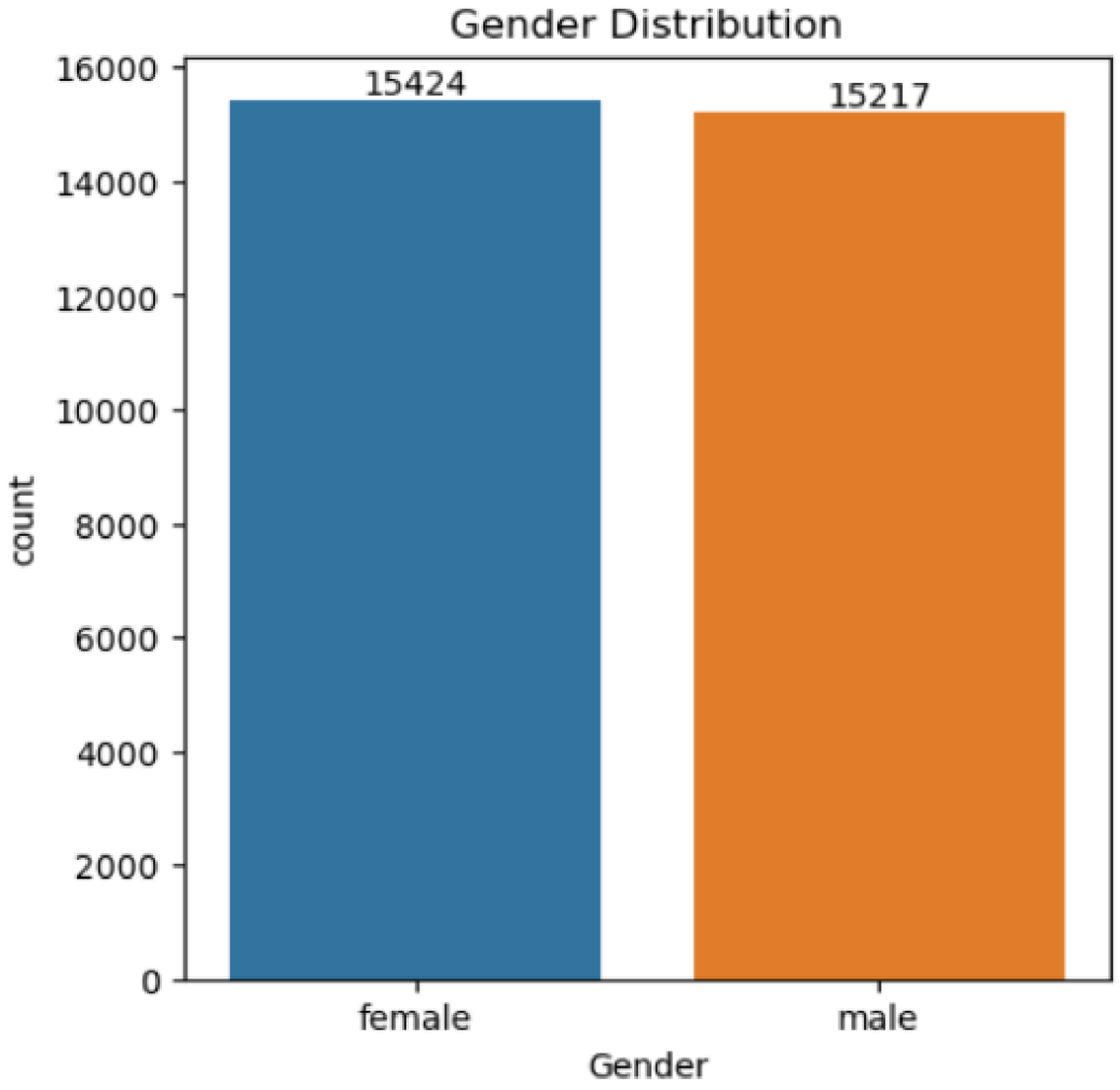
```
In [23]: df.describe()
```

Out[23]:

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

7. Gender Distribution

```
In [33]: plt.figure(figsize=(5,5))
vx=sns.countplot(data=df, x="Gender")
plt.title("Gender Distribution")
vx.bar_label(vx.containers[0])
plt.show()
```



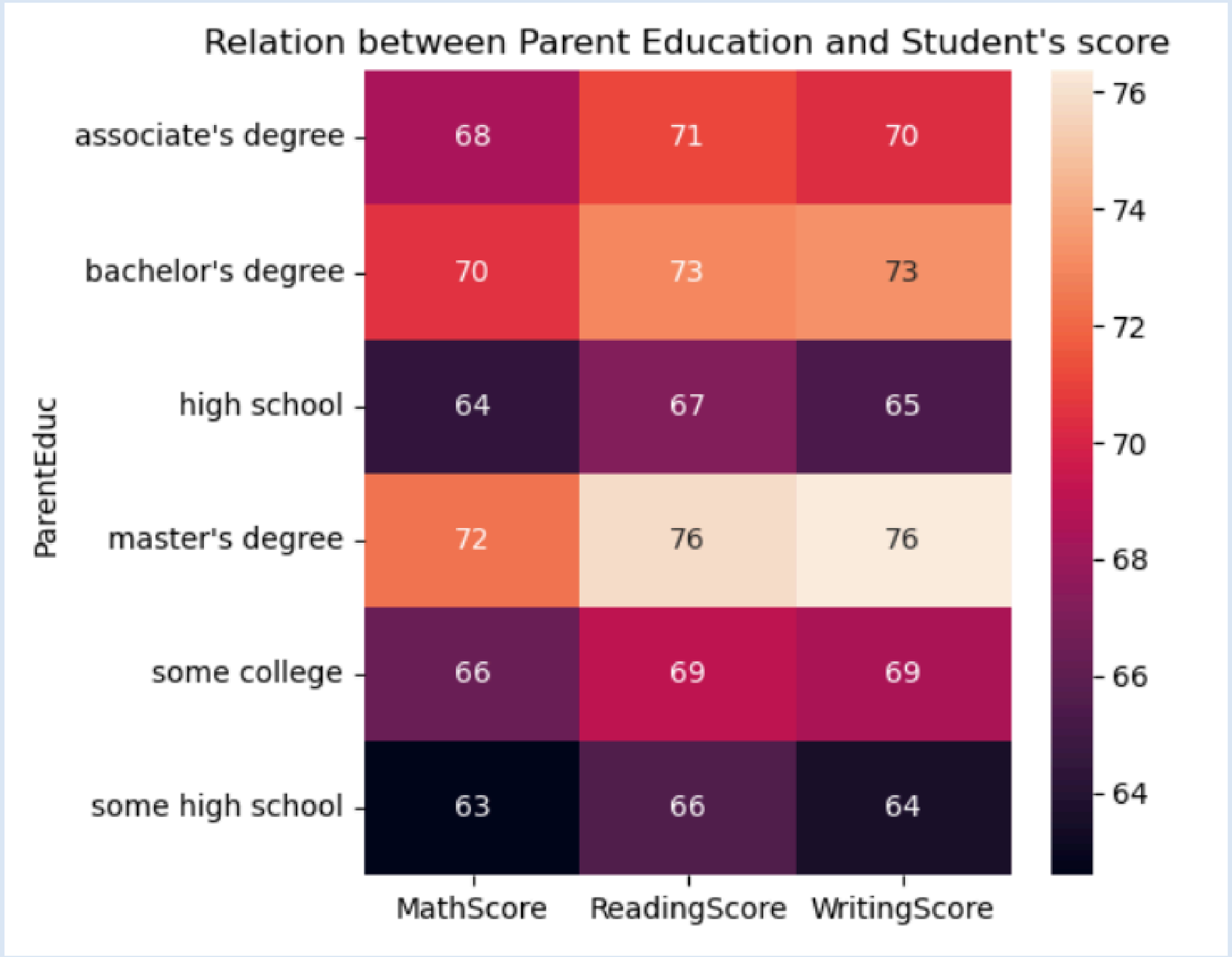
From this, we can conclude that the number of females is more than the number of males.

8. Relation between Parent education and student score

In [26]:
```
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 [32]:
```
plt.figure(figsize=(5,5))
plt.title("Relation between Parent Education and Student's score")
sns.heatmap(gb, annot=True)
plt.show()
```



From this heatmap, it is visible that the education of parents is directly proportional to student score.
This means the higher the parent education the higher the student's score will be.

9. Impact of Parental Marital Status on Student Scores

```
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
```

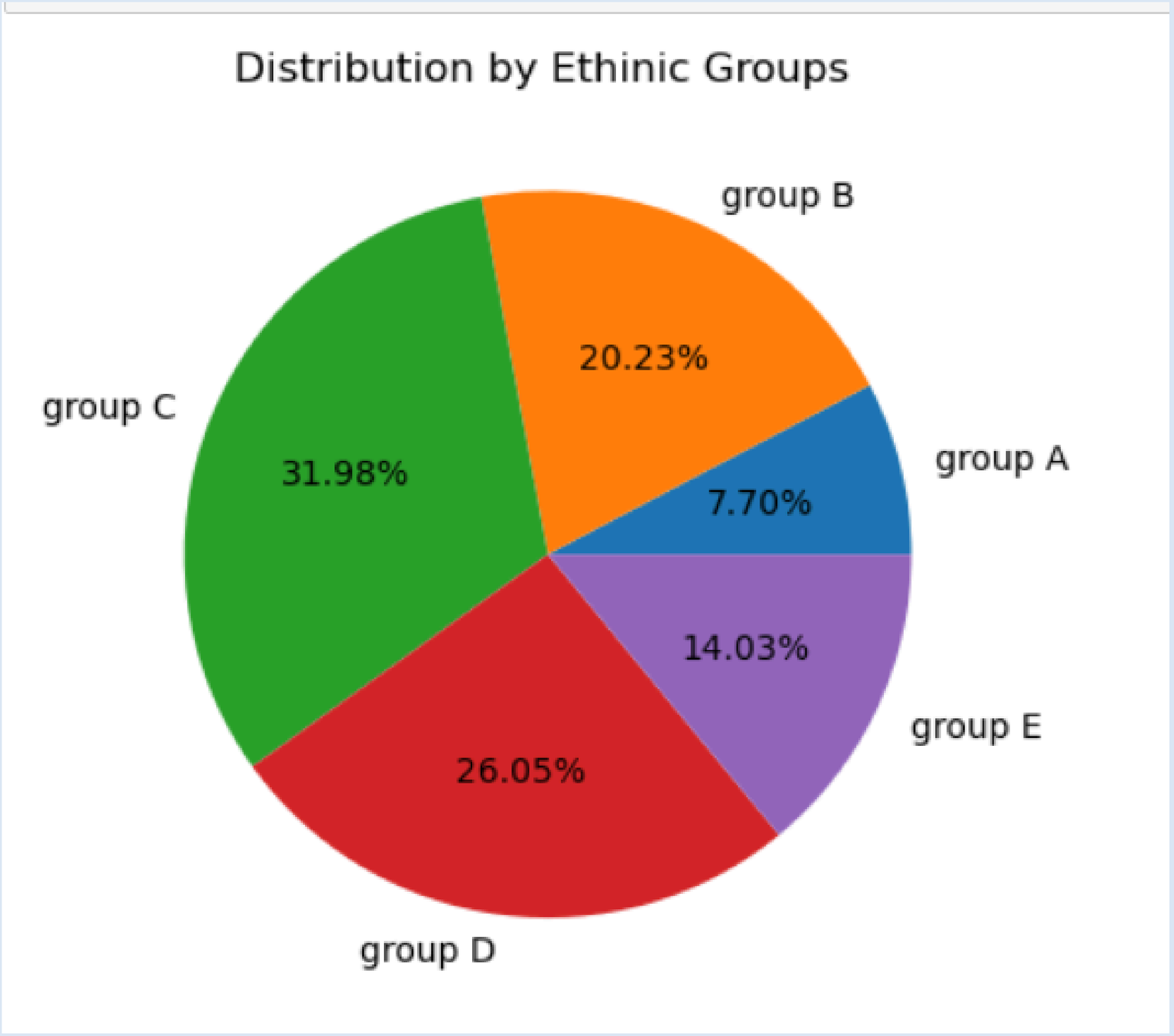
Based on the above, we can conclude that the marital status of the parent does not affect the student's score.

10. Distribution of students by Ethnic group.

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

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

```
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"]
list=[groupA["EthnicGroup"],groupB["EthnicGroup"],groupC["EthnicGroup"],groupD["EthnicGroup"],groupE["EthnicGroup"]]
plt.title("Distribution by Ethinic Groups ")
plt.pie(list,labels=l, autopct="%1.2f%%" )
plt.show()
```

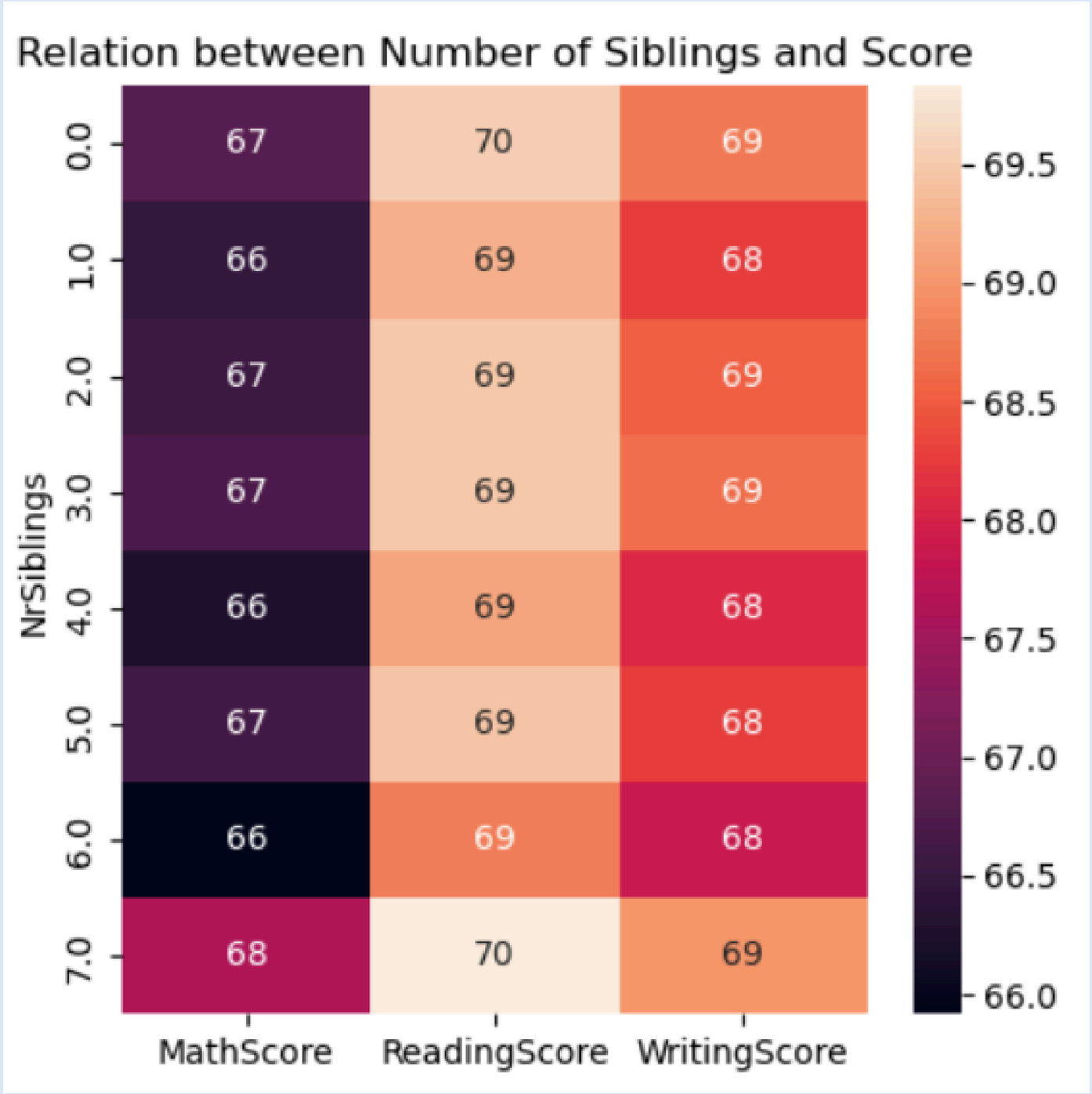


Based on the pie chart, it can be inferred that Ethnic Group C is the predominant group in terms of ethnicity comparison.

11. Relation between the number of siblings and student score

```
gb2=df.groupby("NrSiblings").agg({"MathScore":'mean',"ReadingScore":'mean',"WritingScore":'mean'})
print(gb2)
plt.figure(figsize=(5,5))
plt.title("Relation between Number of Siblings and Score")
sns.heatmap(gb2, annot=True)
plt.show()
```

NrSiblings	MathScore	ReadingScore	WritingScore
0.0	66.819449	69.547812	68.746515
1.0	66.473896	69.259097	68.245345
2.0	66.554934	69.472018	68.522533
3.0	66.719092	69.488159	68.650498
4.0	66.245495	69.144169	68.073444
5.0	66.630303	69.453788	68.282576
6.0	65.917219	68.801325	67.860927
7.0	67.615120	69.828179	68.986254



Based on the heatmap analysis, it can be inferred that the number of siblings has no impact on a student's academic performance.

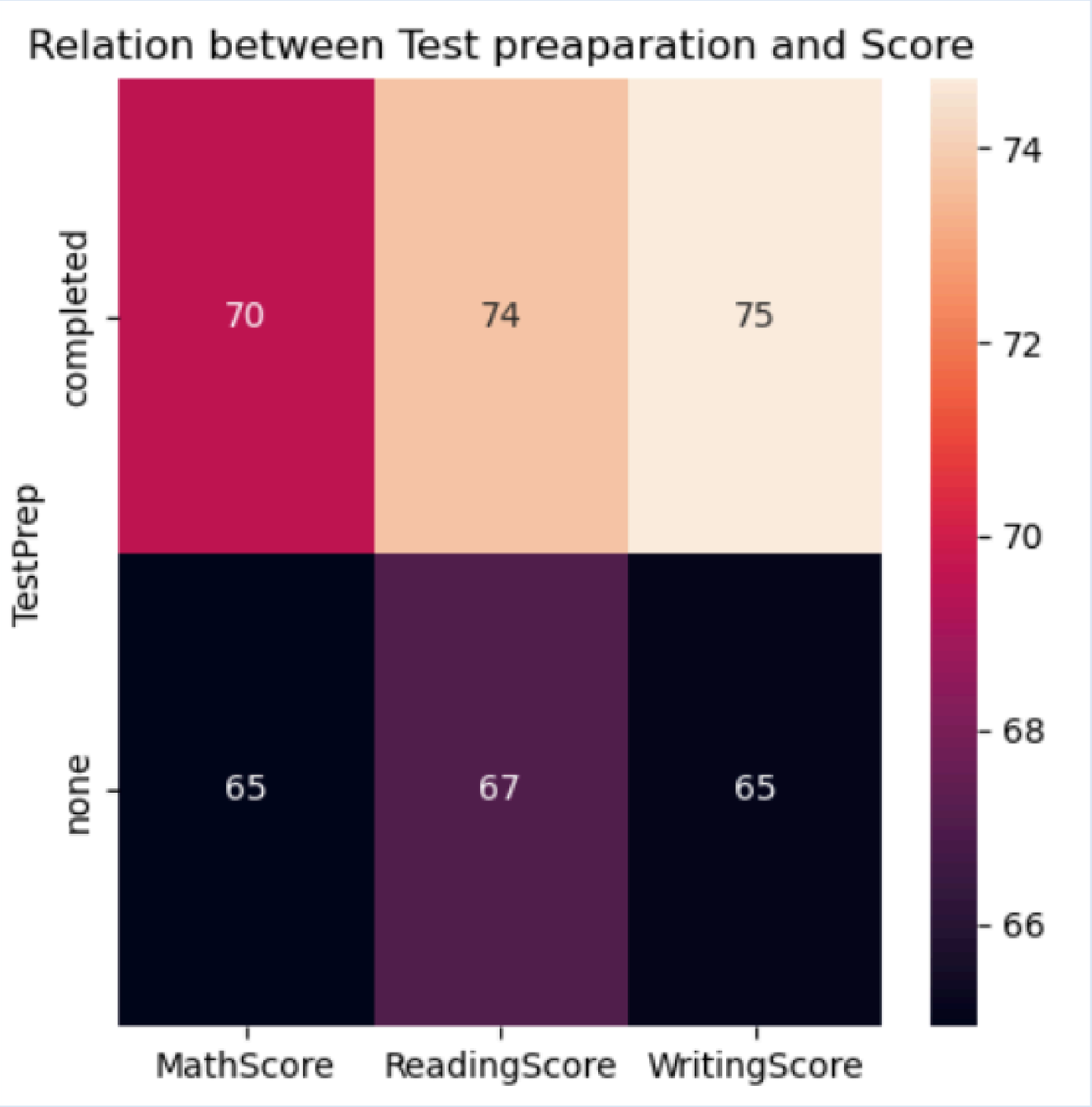
12. Effect of test preparation on student's score

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

```
['none' nan 'completed']
```

```
gb3=df.groupby("TestPrep").agg({"MathScore":'mean',"ReadingScore":'mean',"WritingScore":'mean'})
print(gb3)
plt.figure(figsize=(5,5))
plt.title("Relation between Test preaparation and Score")
sns.heatmap(gb3, annot=True)
plt.show()
```

```
           MathScore  ReadingScore  WritingScore
TestPrep
completed   69.54666     73.732998     74.703265
none        64.94877     67.051071     65.092756
```



Based on the heatmap, it is evident that students who completed test preparation scored higher than those who did not.

In conclusion, students whose parents have higher education levels and those who have undergone test preparation tend to score higher than other students.