PYTHON PROJECT

Analysis of Student Scores Dataset

INTRODUCTION

Hello my name is Zonab Zahra and here we will analyse students scores data using Python in Jupyter Notebook.

Purpose:

The purpose of this project is to analyze a dataset containing student scores to uncover insights about the performance of students based on various demographic and socio-economic factors. This analysis aims to provide a better understanding of how different variables like gender, parental education, and ethnicity impact student academic performance.

Objectives:

1. Examine the Distribution of Student Scores: Analyze the distribution of scores in Math, Reading, and Writing.
2. Gender Analysis: Investigate the distribution of scores across different genders to identify any disparities.
3. Parental Education Influence: Determine how the level of parental education affects student performance.
4. Parental Marital Status Impact: Explore the relationship between parental marital status and student scores.
5. Ethnic Group Analysis: Analyze the distribution and performance of students from different ethnic groups.
6. Data Cleaning and Preprocessing: Ensure the data is clean and ready for analysis by handling missing values and irrelevant columns.

Importing Libraries

- pandas: For data manipulation and analysis.
- numpy: For numerical operations.
- seaborn: For statistical data visualization.
- matplotlib: For creating static, animated, and interactive visualizations

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

Loading and Viewing Data

```
df = pd.read_csv("D:\\My Data Practise\\Dataset\\archive\\student_score.csv")
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
```

Data Overview

Descriptive Statistics and Data Overview

Understanding the dataset's basic characteristics is crucial before diving into deeper analysis. In this step, we use various pandas functions to get an overview of the dataset, including summary statistics, data types, and missing values

Descriptive Statistics:

```
1 df.describe()
```

	Unnamed: 0	NrSiblings	MathScore	ReadingScore	WritingScore
ount	30641.000000	29069.000000	30641.000000	30641.000000	30641.000000
nean	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

Data Information:

```
1 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
```

Missing Values Check:

```
1 df.isnull().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
```

Data Cleaning

- Dropping unnecessary columns:

Drop unnamed column

```
1  df = df.drop("Unnamed: 0",axis = 1)
2  print(df.head())
```

Reason for Dropping the Column:

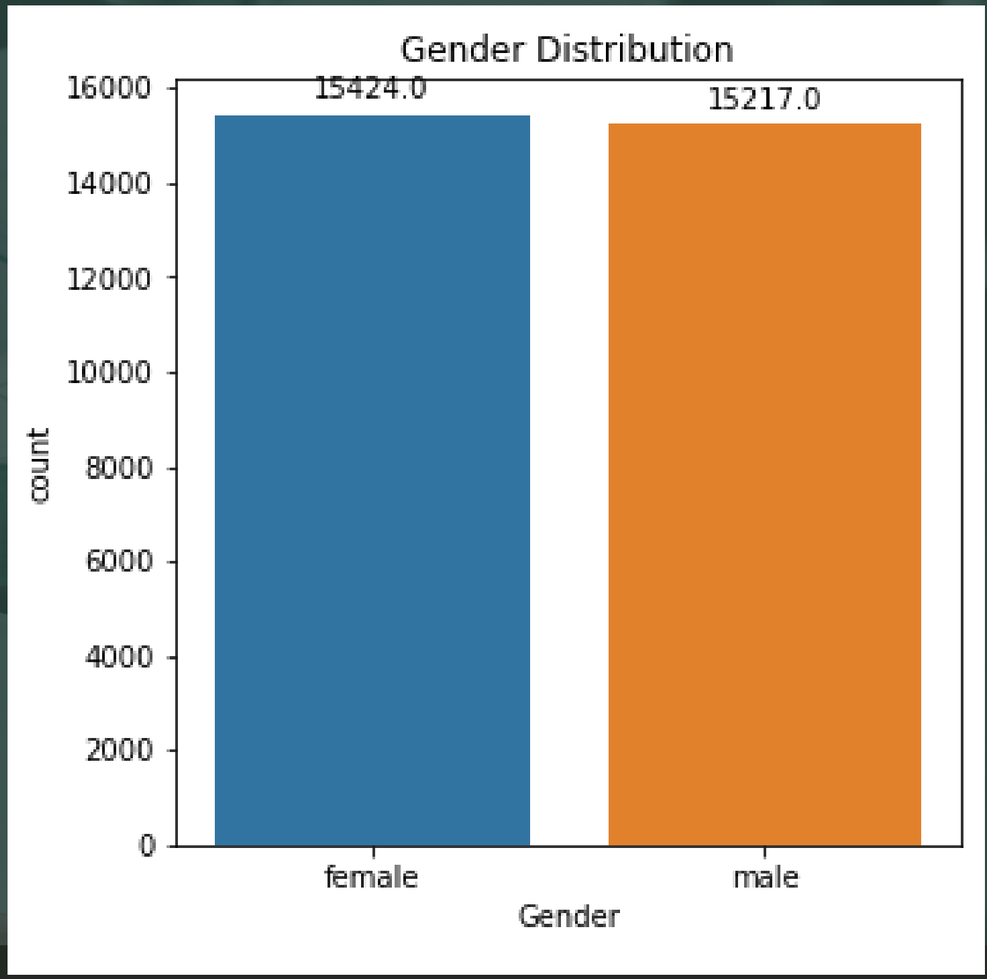
The dataset initially contains a column named "Unnamed: 0". This column typically appears when a CSV file is saved with an index in pandas, and when the CSV is reloaded, it results in an extra column representing the old index. This column is usually redundant because it doesn't provide any meaningful information for the analysis.

Dropping the "Unnamed: 0" column ensures that the dataset only contains relevant features that contribute to the analysis. This step helps in maintaining a clean and concise dataset, making it easier to perform accurate data analysis and visualization.

Gender Distribution

- Visualization of gender distribution using seaborn:

```
plt.figure(figsize = (6,6))
ax = sns.countplot(data = df,x = "Gender")
for p in ax.patches:
    ax.annotate(format(p.get_height(), '.1f'),
                (p.get_x() + p.get_width() / 2., p.get_height()),
                ha = 'center', va = 'center',
                xytext = (0, 9),  # 9 points vertical offset
                textcoords = 'offset points')
plt.title("Gender Distribution")
plt.show()
```



From this chart we have analysed that the number of females in the data is more than the number of males

Parental Education and Scores

- Grouping by parental education and calculating mean scores:

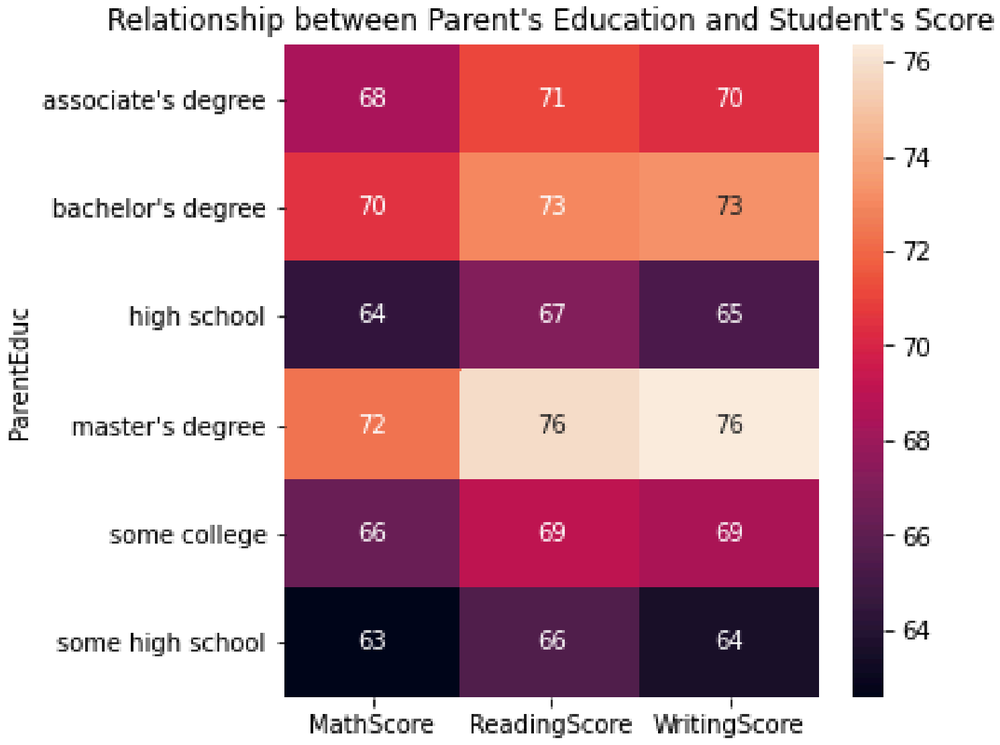
```
gb = df.groupby("ParentEduc").agg({"MathScore":'mean',"ReadingScore":'mean',"WritingScore":'mean'})
print(gb)
```

ParentEduc	MathScore	ReadingScore	WritingScore
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

from this chart we have concluded that the Education of Parent's have a good impact on their Scores

Heatmap visualization:

```
plt.figure(figsize = (5,5))
sns.heatmap(gb, annot = True)
plt.title("Relationship between Parent's Education and Student's Score")
plt.show()
```



Parental Marital Status and Scores

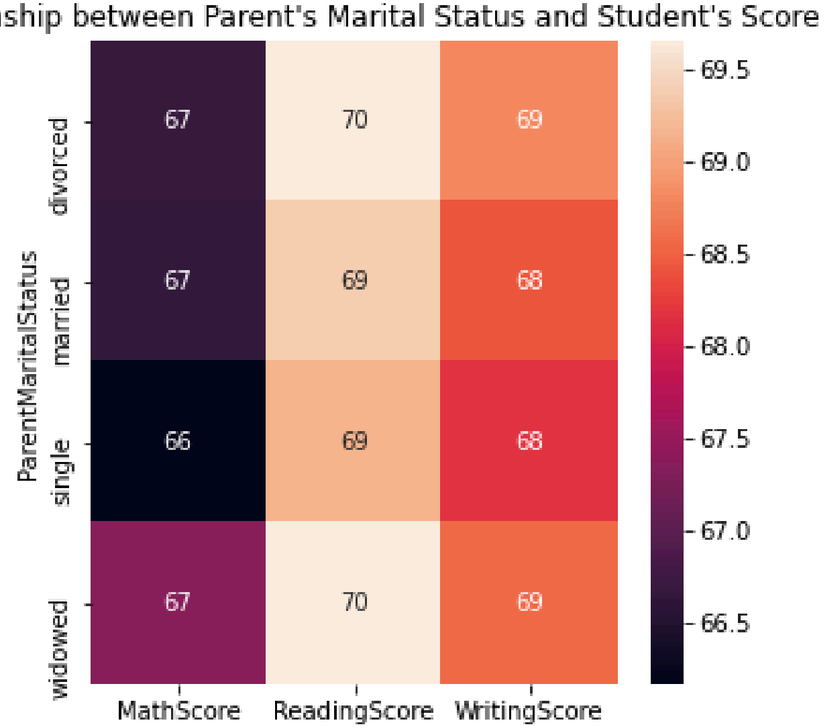
Grouping by parental marital status and calculating mean scores:

```
gb1 = df.groupby("ParentMaritalStatus").agg({"MathScore":'mean',"ReadingScore":'mean',"WritingScore":'mean'})
print(gb1)
```

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

Heatmap visualization:

```
plt.figure(figsize = (5,5))
sns.heatmap(gb1, annot = True)
plt.title("Relationship between Parent's Marital Status and Student's Score")
plt.show()
```

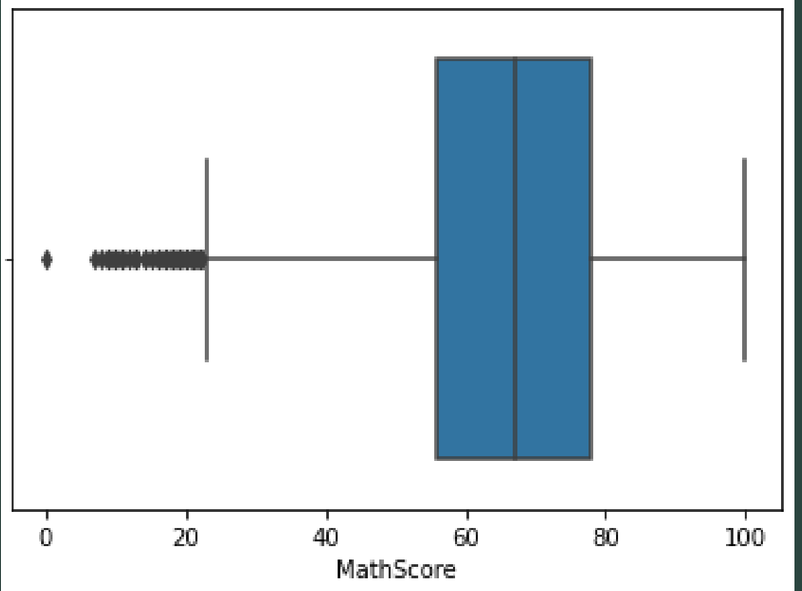


From this chart we have concluded that there is no/negligible impact on the student's score due to their parent's marital status

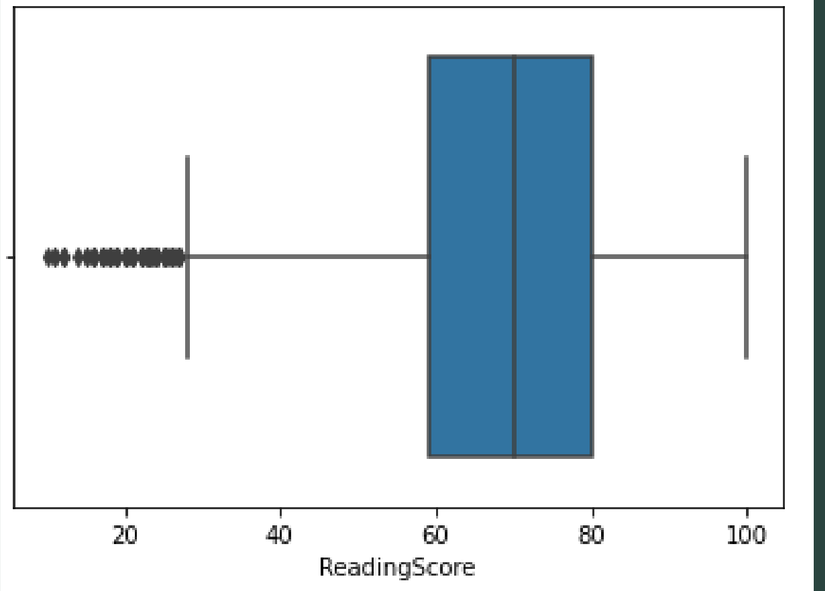
Boxplots of Scores

Visualize distribution of scores using boxplots:

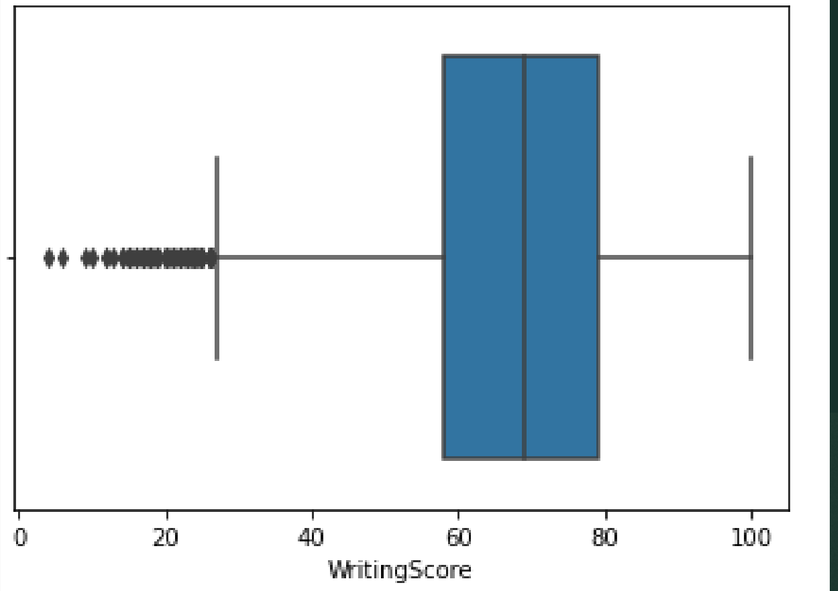
```
1 sns.boxplot(data = df, x = "MathScore")
2 plt.show()
```




```
1 sns.boxplot(data = df, x = "ReadingScore")
2 plt.show()
```




```
1 sns.boxplot(data = df, x = "WritingScore")
2 plt.show()
```



Ethnic Group Distribution

- Unique ethnic groups in the dataset:

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

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

- Count of students in each ethnic group:

```
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"]
mlist = [groupA["EthnicGroup"],groupB["EthnicGroup"],groupC["EthnicGroup"],groupD["EthnicGroup"],groupE["EthnicGroup"]]

print(mlist)

```

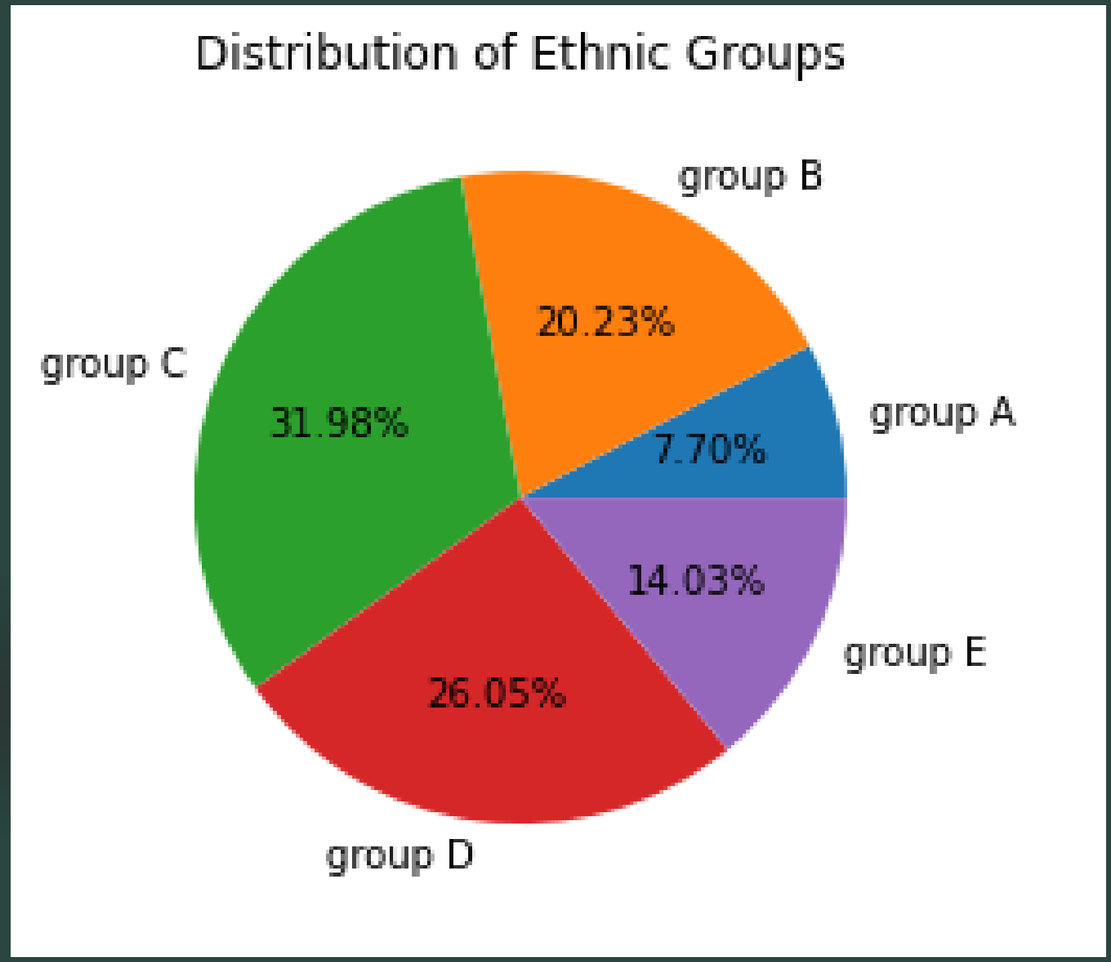
```
[2219, 5826, 9212, 7503, 4041]
```

Ethnic Group Distribution Pie Chart

- Pie chart visualization:

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

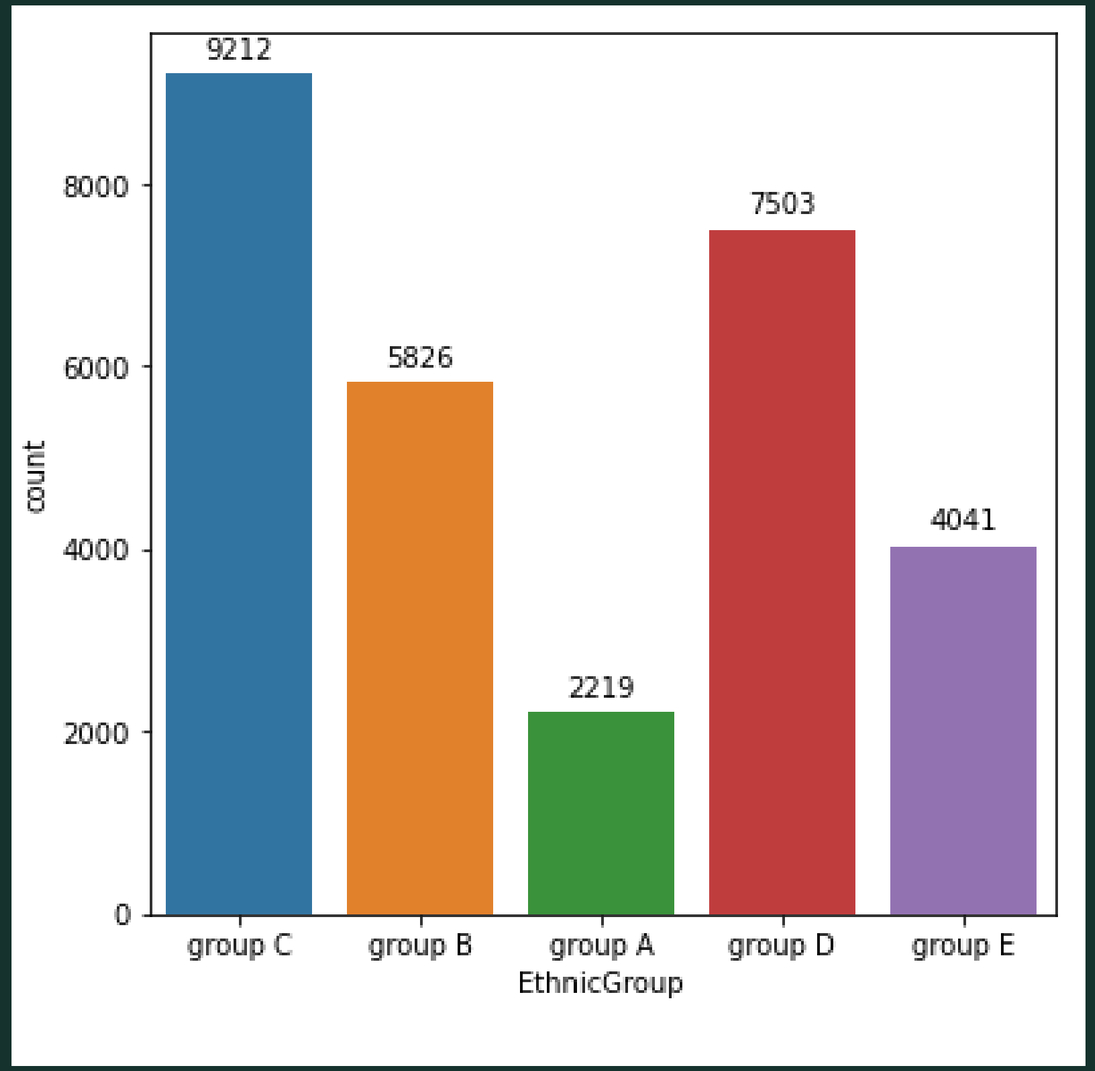
The pie chart shows the proportion of each ethnic group in the dataset, providing a clear visual representation of the ethnic diversity among students.



Ethnic Group Count Plot

- **Count plot visualization:**

```
plt.figure(figsize = (6,6))
ax = sns.countplot(data = df,x = "EthnicGroup")
for p in ax.patches:
    ax.annotate(format(p.get_height(), '.0f'),
                (p.get_x() + p.get_width() / 2., p.get_height()),
                ha = 'center', va = 'center',
                xytext = (0, 9),
                textcoords = 'offset points')
```



- **Explanation: This code creates a count plot using Seaborn to show the number of students in each ethnic group. Annotations are added to display the exact count above each bar.**
- **Visualization: The count plot provides a detailed view of the student count in each ethnic group, complementing the pie chart by showing the actual numbers.**

Summary of Key Findings

- Gender Distribution
- Parental Education and Scores
- Parental Marital Status and Scores
- Score Distributions
- Ethnic Group Distribution
- Ethnic Group and Performance

Insights and Observations

- **Gender Analysis:**

There may be slight differences in academic performance based on gender, but the dataset needs to be carefully interpreted to avoid generalizations.

- **Parental Influence:**

Both parental education and marital status have noticeable impacts on student performance. This underscores the importance of a supportive home environment in educational outcomes.

- **Ethnic Diversity:**

The dataset reflects a diverse student population. Ensuring equitable educational opportunities for all ethnic groups remains crucial.

Conclusion

This project successfully utilized data analysis and visualization techniques to uncover insights into student performance based on various demographic and socio-economic factors. The findings highlight the critical role of parental education and family structure in academic success, as well as the importance of understanding and addressing diversity within the student population. By continuing this analysis and exploring additional factors, we can contribute to creating more equitable and effective educational strategies.