

Test001

August 1, 2025

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

```
[2]: student1 = pd.read_csv("../..../datasets/Students.csv")
```

```
[3]: student1.info()
```

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 1000 entries, 0 to 999

Data columns (total 8 columns):

#	Column	Non-Null Count	Dtype
0	gender	1000 non-null	object
1	race/ethnicity	1000 non-null	object
2	parental level of education	1000 non-null	object
3	lunch	1000 non-null	object
4	test preparation course	1000 non-null	object
5	math score	1000 non-null	int64
6	reading score	1000 non-null	int64
7	writing score	1000 non-null	int64

dtypes: int64(3), object(5)

memory usage: 62.6+ KB

```
[4]: student1.head()
```

```
[4]:  gender race/ethnicity parental level of education      lunch \
0  female      group B      bachelor's degree      standard
1  female      group C      some college      standard
2  female      group B      master's degree      standard
3   male      group A      associate's degree  free/reduced
4   male      group C      some college      standard

test preparation course  math score  reading score  writing score
0              none          72           72           74
1        completed          69           90           88
2              none          90           95           93
3              none          47           57           44
```

4	none	76	78	75
---	------	----	----	----

```
[5]: studenti.describe()
```

```
[5]:
```

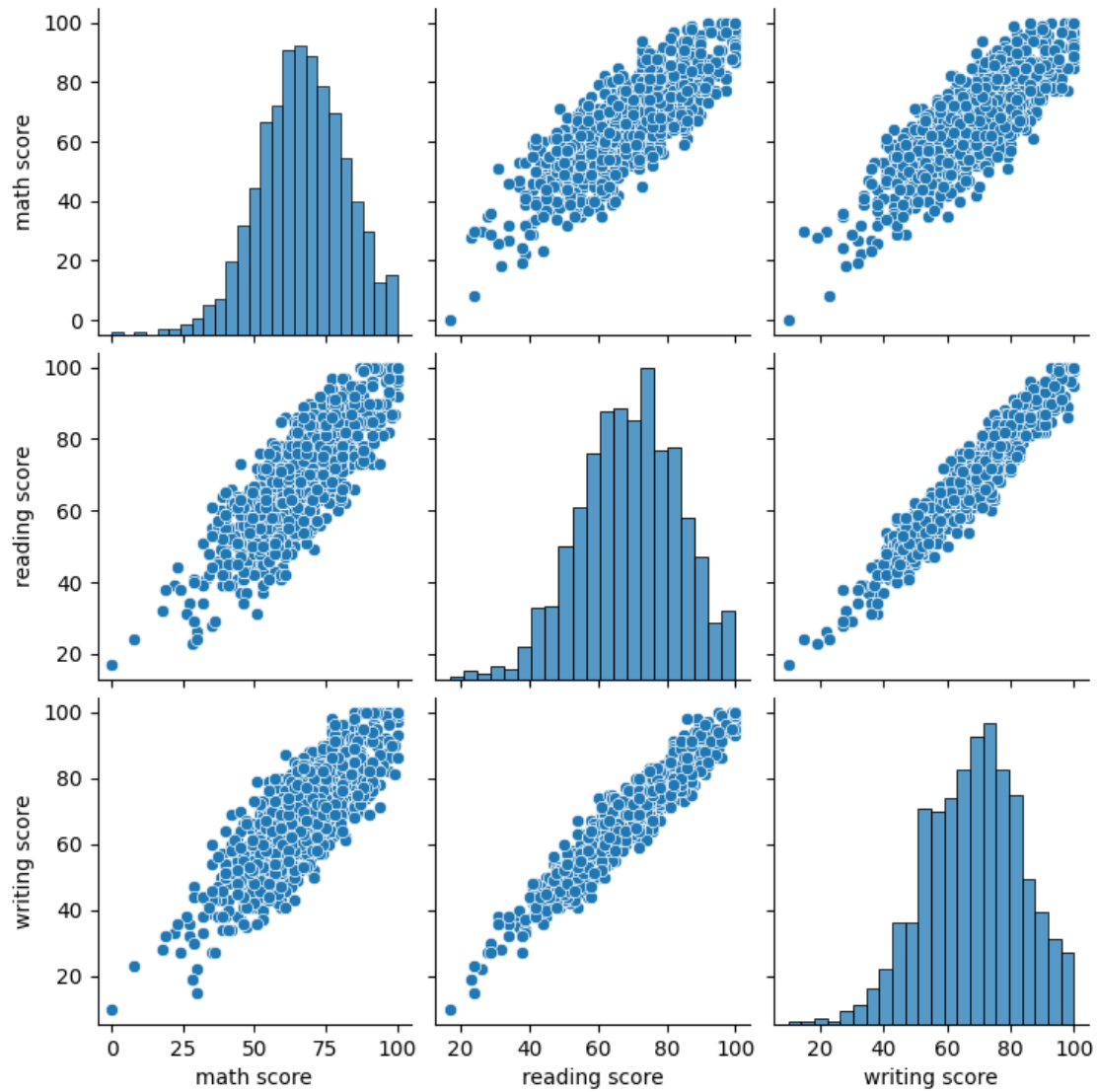
	math score	reading score	writing score
count	1000.00000	1000.000000	1000.000000
mean	66.08900	69.169000	68.054000
std	15.16308	14.600192	15.195657
min	0.00000	17.000000	10.000000
25%	57.00000	59.000000	57.750000
50%	66.00000	70.000000	69.000000
75%	77.00000	79.000000	79.000000
max	100.00000	100.000000	100.000000

```
[6]: studenti.corr(numeric_only=True)
```

```
[6]:
```

	math score	reading score	writing score
math score	1.000000	0.817580	0.802642
reading score	0.817580	1.000000	0.954598
writing score	0.802642	0.954598	1.000000

```
[7]: a = sns.pairplot(studenti)
```



```
[8]: sns.barplot(studenti, x = "race/ethnicity", y = studenti["race/ethnicity"].
      ↪value_counts().iloc[0], estimator="sum", hue="gender")
```

```
[8]: <Axes: xlabel='race/ethnicity'>
```

