Name: - L Prathyusha

Student data

In [3]:

```
import pandas as pd
import numpy as np
import seaborn as sns
```

In [4]:

```
data=pd.read_csv("C:/Users/Prathyu Lachireddy/Desktop/BP/Students.csv")
data
```

Out[4]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75
...	...	...	...	...	...	...	...	...
995	female	group E	master's degree	standard	completed	88	99	95
996	male	group C	high school	free/reduced	none	62	55	55
997	female	group C	high school	free/reduced	completed	59	71	65
998	female	group D	some college	standard	completed	68	78	77
999	female	group D	some college	free/reduced	none	77	86	86

1000 rows × 8 columns

In [5]:

```
data.head()
```

Out[5]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75

In [6]:

```
data.tail()
```

Out[6]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
995	female	group E	master's degree	standard	completed	88	99	95
996	male	group C	high school	free/reduced	none	62	55	55
997	female	group C	high school	free/reduced	completed	59	71	65
998	female	group D	some college	standard	completed	68	78	77
999	female	group D	some college	free/reduced	none	77	86	86

In [7]:

```
#columns and rows
data.shape
```

Out[7]:

```
(1000, 8)
```

In [8]:

```
#number of columns
data.columns
```

Out[8]:

```
Index(['gender', 'race/ethnicity', 'parental level of education', 'lunch',
       'test preparation course', 'math score', 'reading score',
       'writing score'],
      dtype='object')
```

In [9]:

```
#unit columns
data.nunique()
```

Out[9]:

```
gender                         2
race/ethnicity                 5
parental level of education    6
lunch                          2
test preparation course        2
math score                    81
reading score                 72
writing score                 77
dtype: int64
```

In [10]:

```
data.describe()
```

Out[10]:

	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

In [11]:

```
data['gender'].unique()
```

Out[11]:

```
array(['female', 'male'], dtype=object)
```

In [14]:

```
student=data.drop(['race/ethnicity','parental level of education'],axis=1)
student
```

Out[14]:

	gender	lunch	test preparation course	math score	reading score	writing score
0	female	standard	none	72	72	74
1	female	standard	completed	69	90	88
2	female	standard	none	90	95	93
3	male	free/reduced	none	47	57	44
4	male	standard	none	76	78	75
...	...	...	...	...	...	...
995	female	standard	completed	88	99	95
996	male	free/reduced	none	62	55	55
997	female	free/reduced	completed	59	71	65
998	female	standard	completed	68	78	77
999	female	free/reduced	none	77	86	86

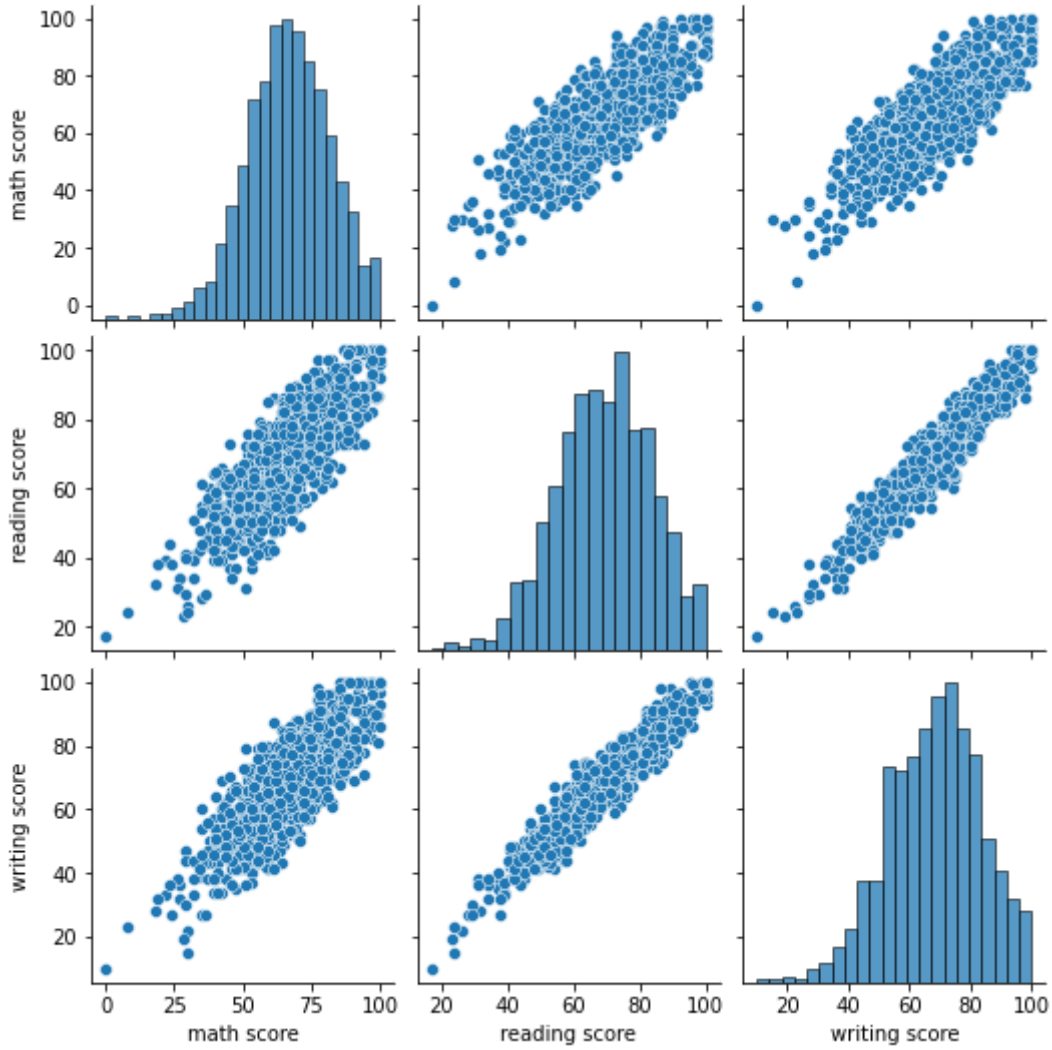
1000 rows × 6 columns

In [15]:

```
sns.pairplot(student)
```

Out[15]:

```
<seaborn.axisgrid.PairGrid at 0x20c5a681a90>
```

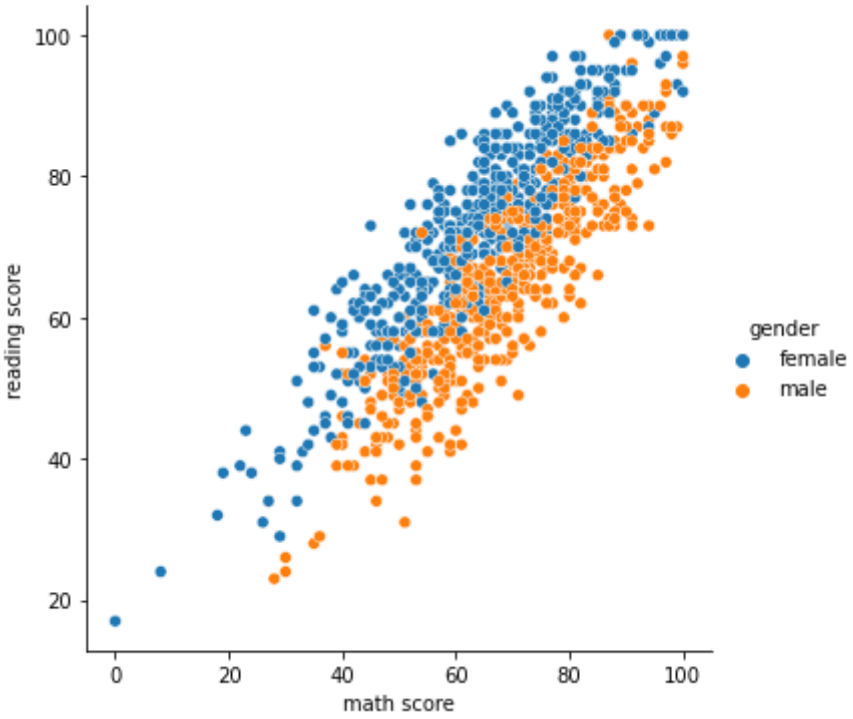


In [16]:

```
sns.relplot(x='math score',y='reading score',hue='gender',data=student)
```

Out[16]:

```
<seaborn.axisgrid.FacetGrid at 0x20c5af13b20>
```

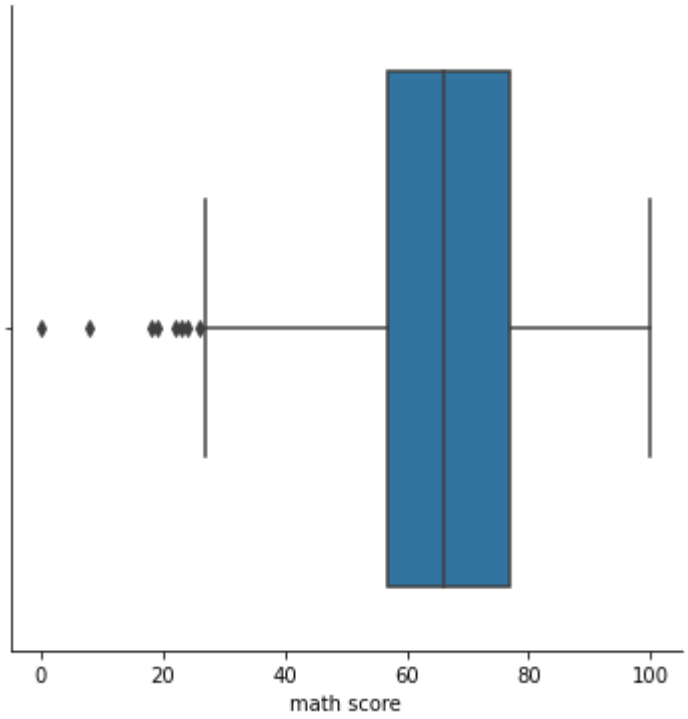


In [17]:

```
sns.catplot(x='math score', kind='box',data=student)
```

Out[17]:

```
<seaborn.axisgrid.FacetGrid at 0x20c5b1be2e0>
```



In [21]:

```
corelation=student.corr()
corelation
```

Out[21]:

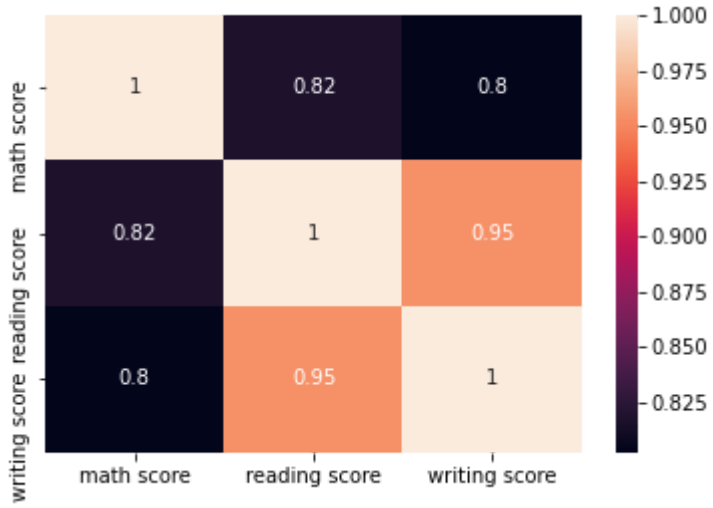
	math score	reading score	writing score
math score	1.000000	0.817580	0.802642
reading score	0.817580	1.000000	0.954598
writing score	0.802642	0.954598	1.000000

In [22]:

```
sns.heatmap(corelation,xticklabels=corelation.columns,yticklabels=corelation.columns,annot=
```

Out[22]:

<AxesSubplot:>



In [ ]: