

```
In [1]: import pandas as pd
```

```
In [2]: import numpy as np
```

```
In [3]: df = pd.read_csv('/Users/Yash Bahekar/Downloads/Practicals/ds/Final/StudentsPerform
```

```
In [4]: df
```

```
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 [9]: from sklearn import preprocessing
```

```
In [10]: label_encoder = preprocessing.LabelEncoder()
```

```
In [14]: df['gender'] = label_encoder.fit_transform(df['gender']);
```

```
In [16]: df['gender'].unique()
```

```
Out[16]: array([0, 1])
```

```
In [17]: df
```

```
Out[17]:
```

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

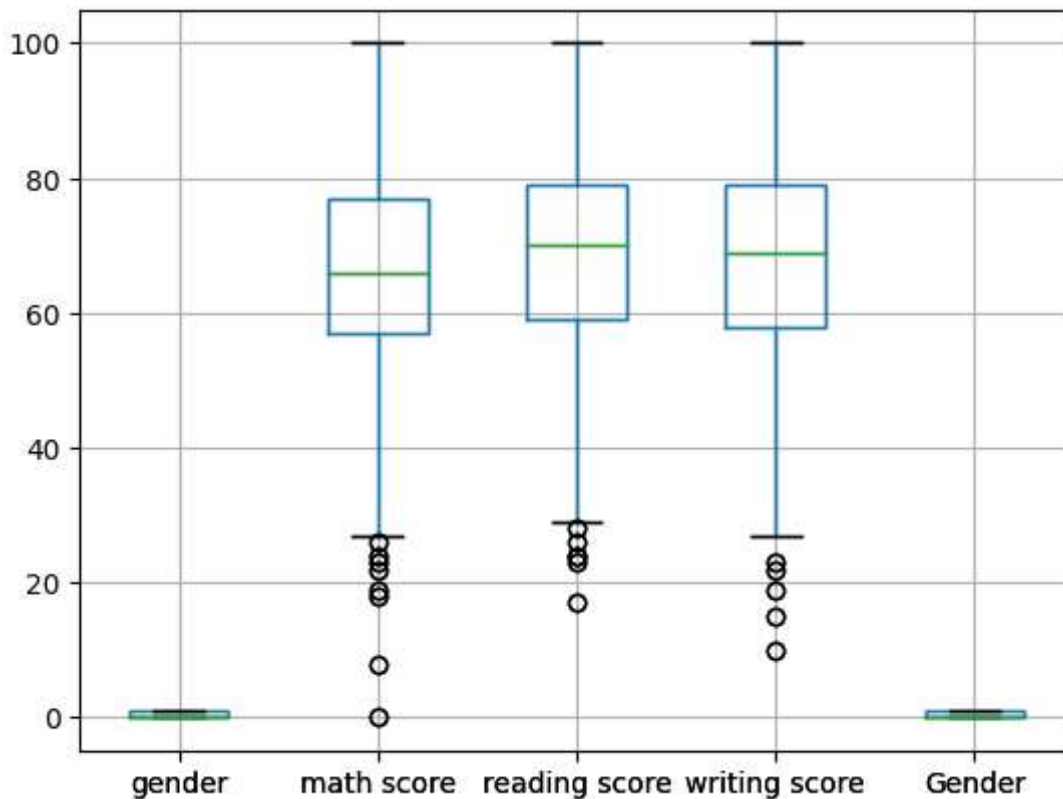
1000 rows × 9 columns



```
In [20]: import matplotlib.pyplot as plt
```

```
In [25]: boxplot = df.boxplot()
```

```
In [26]: plt.show()
```



```
In [40]: mean = df['math score'].mean()
         mean
```

```
Out[40]: 66.089
```

```
In [41]: df['math score'].median()
```

```
Out[41]: 66.0
```

```
In [50]: std = df['math score'].std()
         std
```

```
Out[50]: 15.16308009600945
```

```
In [51]: df['math score'].mode()
```

```
Out[51]: 0    65
         Name: math score, dtype: int64
```

```
In [52]: import scipy.stats as stats
```

```
In [53]: zscores = stats.zscore(df['math score'])
```

```
In [54]: zscores
```

```
Out[54]: 0      0.390024
         1      0.192076
         2      1.577711
         3     -1.259543
         4      0.653954
         ...
        995     1.445746
        996    -0.269803
        997    -0.467751
        998     0.126093
        999     0.719937
        Name: math score, Length: 1000, dtype: float64
```

```
In [55]: threshold = 0
```

```
In [56]: outlier = []
```

```
In [57]: for i in df['math score']:
         z = (i-mean)/std
         if z > threshold:
             outlier.append(i)
         print('outlier in dataset is', outlier)
```

```
outlier in dataset is [72, 69, 90, 76, 71, 88, 78, 69, 88, 69, 74, 73, 69, 67, 70, 6
9, 97, 81, 74, 75, 82, 77, 88, 71, 82, 79, 69, 67, 80, 72, 73, 76, 71, 73, 71, 79, 7
8, 79, 68, 85, 98, 87, 70, 77, 99, 84, 75, 78, 79, 91, 88, 83, 87, 72, 82, 89, 87, 7
5, 74, 70, 71, 76, 88, 88, 73, 68, 100, 77, 70, 82, 75, 81, 96, 68, 67, 72, 94, 79,
81, 71, 97, 80, 77, 76, 77, 69, 78, 67, 69, 74, 82, 81, 74, 80, 87, 84, 83, 87, 86,
72, 68, 88, 76, 67, 92, 83, 80, 84, 73, 80, 75, 85, 89, 68, 71, 80, 78, 70, 79, 99,
76, 69, 88, 71, 69, 88, 83, 85, 73, 73, 70, 81, 97, 67, 88, 77, 76, 86, 78, 67, 71,
90, 81, 67, 80, 74, 69, 99, 73, 67, 68, 71, 77, 83, 67, 75, 71, 82, 82, 71, 90, 83,
76, 69, 67, 79, 72, 77, 75, 87, 80, 85, 69, 73, 84, 74, 82, 80, 85, 71, 80, 87, 79,
67, 73, 73, 77, 76, 85, 74, 88, 82, 87, 70, 84, 71, 84, 71, 74, 68, 82, 86, 69, 68,
73, 75, 75, 70, 89, 67, 78, 73, 79, 67, 69, 86, 81, 100, 79, 100, 72, 71, 77, 75, 8
4, 72, 77, 91, 83, 68, 90, 71, 76, 80, 76, 73, 68, 70, 79, 83, 81, 68, 76, 74, 94, 9
5, 82, 68, 79, 86, 76, 77, 76, 74, 67, 71, 91, 69, 68, 68, 79, 73, 85, 97, 75, 79, 8
1, 82, 78, 92, 72, 79, 79, 87, 77, 73, 74, 96, 92, 68, 71, 91, 74, 81, 81, 77, 73, 6
9, 71, 68, 74, 92, 69, 76, 84, 75, 85, 69, 94, 82, 88, 95, 100, 100, 69, 67, 79, 75,
84, 71, 67, 80, 86, 76, 74, 72, 74, 70, 69, 68, 85, 73, 77, 69, 90, 74, 73, 80, 77,
73, 81, 69, 69, 73, 70, 81, 67, 94, 85, 77, 93, 73, 77, 79, 75, 79, 87, 89, 93, 80,
98, 81, 76, 73, 96, 76, 91, 74, 81, 68, 73, 68, 87, 67, 92, 81, 80, 81, 72, 69, 69,
87, 71, 68, 79, 77, 84, 70, 69, 78, 74, 76, 68, 75, 72, 94, 77, 76, 91, 72, 68, 89,
70, 70, 69, 67, 76, 87, 82, 73, 75, 90, 87, 72, 94, 77, 85, 78, 71, 69, 68, 74, 77,
73, 75, 74, 90, 91, 80, 71, 87, 82, 97, 75, 87, 81, 71, 97, 82, 78, 70, 82, 90, 80,
81, 71, 93, 69, 86, 85, 88, 67, 85, 73, 93, 67, 88, 79, 67, 70, 69, 68, 100, 91, 69,
72, 74, 74, 67, 70, 98, 70, 67, 85, 77, 72, 78, 81, 82, 94, 75, 74, 72, 84, 92, 75,
100, 72, 68, 75, 89, 78, 91, 81, 79, 78, 74, 81, 67, 86, 88, 68, 77]
```

```
In [ ]:
```