In [1]:
```
pip install pandas
```

```
Requirement already satisfied: pandas in c:\users\hp\anaconda3\lib\site-packages (0.2
5.1)
Requirement already satisfied: pytz>=2017.2 in c:\users\hp\anaconda3\lib\site-packages
(from pandas) (2019.3)
Requirement already satisfied: python-dateutil>=2.6.1 in c:\users\hp\anaconda3\lib\sit
e-packages (from pandas) (2.8.0)
Requirement already satisfied: numpy>=1.13.3 in c:\users\hp\anaconda3\lib\site-package
s (from pandas) (1.16.5)
Requirement already satisfied: six>=1.5 in c:\users\hp\anaconda3\lib\site-packages (fr
om python-dateutil>=2.6.1->pandas) (1.12.0)
Note: you may need to restart the kernel to use updated packages.
```

In [2]:
```
import pandas as pd
```

In [4]:
```
df = pd.read_csv(r"D:\College\TE\SEM-2\Practical\DSBDA\1\StudentsPerformance.csv")
```

In [5]:
```
print(df)
```

```
     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
..      ...            ...                         ...           ...
995  female        group E             master's degree      standard
996    male        group C                 high school  free/reduced
997  female        group C                 high school  free/reduced
998  female        group D                some college      standard
999  female        group D                some college  free/reduced

    test preparation course  math score  reading score  writing score
0                      none          72             72            NaN
1                 completed          69             90           88.0
2                      none          90             95           93.0
3                      none          47             57           44.0
4                      none          76             78           75.0
..                      ...         ...            ...           ...
995               completed          88             99           95.0
996                    none          62             55           55.0
997               completed          59             71           65.0
998               completed          68             78           77.0
999                    none          77             86           86.0

[1000 rows x 8 columns]
```

In [7]:
```
df.head(15)
```

Out[7]:

	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	NaN
1	female	group C	some college	standard	completed	69	90	88.0
2	female	group B	master's degree	standard	none	90	95	93.0
3	male	group A	associate's degree	free/reduced	none	47	57	44.0
4	male	group C	some college	standard	none	76	78	75.0
5	female	group B	associate's degree	standard	none	71	83	78.0
6	female	group B	some college	standard	completed	88	95	92.0
7	male	group B	some college	free/reduced	none	40	43	39.0
8	male	group D	high school	free/reduced	completed	64	64	67.0
9	female	group B	high school	free/reduced	none	38	60	50.0
10	male	group C	associate's degree	standard	none	58	54	52.0
11	male	group D	associate's degree	standard	none	40	52	43.0
12	female	group B	high school	standard	none	65	81	73.0
13	male	group A	some college	standard	completed	78	72	70.0
14	female	group A	master's degree	standard	none	50	53	58.0

In [8]:
```
df.isnull().sum()
```

Out[8]:
```
gender                       0
race/ethnicity               0
parental level of education  0
lunch                        0
test preparation course      0
math score                   0
reading score                0
writing score                1
dtype: int64
```

In [9]:
```
print(df.describe())
```

```
       math score  reading score  writing score
count  1000.00000    1000.000000     999.000000
mean     66.08900      69.169000      68.048048
std      15.16308      14.600192      15.202102
min       0.00000      17.000000      10.000000
25%      57.00000      59.000000      57.500000
50%      66.00000      70.000000      69.000000
75%      77.00000      79.000000      79.000000
max     100.00000     100.000000     100.000000
```

In [11]:
```
df.dtypes
```

Out[11]:
```
gender                         object
race/ethnicity                 object
parental level of education    object
lunch                          object
test preparation course        object
math score                      int64
reading score                   int64
writing score                 float64
dtype: object
```

In [12]:
```
df.dropna(axis=1)
```

Out[12]:

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

1000 rows × 7 columns

In [13]:
```
y = df.iloc[:, 0:1]
print(y)
```

```
     gender
0    female
1    female
2    female
3      male
4      male
..      ...
995  female
996    male
997  female
998  female
999  female

[1000 rows x 1 columns]
```

In [18]:
```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
y = le.fit_transform(y)
print(y)
```

```
[0 0 0 1 1 0 0 1 1 0 1 1 0 1 0 0 1 0 1 0 1 0 1 0 1 1 1 0 1 0 0 0 0 1 1 1 0
 0 0 1 1 0 0 1 0 1 0 0 0 1 1 1 1 1 0 0 0 1 1 0 1 1 1 0 0 1 1 0 1 0 0 1 0 1
 1 1 1 1 0 0 0 1 1 1 1 0 0 0 0 0 0 1 1 1 0 1 1 0 0 0 1 1 0 1 1 0 0 1 0 0 0
 1 1 0 0 1 0 0 0 0 0 1 0 1 1 0 1 1 1 0 1 1 1 0 1 1 1 1 0 1 0 0 0 1 1 0 1 1
 0 1 1 1 0 1 1 0 0 1 0 1 1 0 1 1 0 0 1 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1
 1 1 1 1 0 0 1 0 1 0 1 1 1 0 0 0 0 1 0 1 1 1 1 0 0 1 1 0 1 1 1 0 0 1 1 0 1
 0 1 0 0 0 1 1 0 1 1 0 1 1 1 1 0 1 1 1 0 0 1 1 1 1 0 0 1 1 0 0 1 1 0 0 1 0
 0 0 1 0 0 1 1 0 0 0 0 1 1 0 0 1 1 0 0 0 1 1 1 0 0 0 1 1 0 1 1 1 1 1 0 1 1
 1 1 1 1 1 1 0 1 0 1 1 1 0 0 0 1 1 0 0 1 0 1 1 0 0 0 0 0 0 0 1 1 1 0 1 1 1
 1 0 0 1 1 0 0 1 0 0 1 1 0 1 0 1 1 0 1 0 0 0 0 1 0 1 0 0 1 0 0 1 1 1 1 0 0
 1 0 1 0 0 1 0 0 0 1 0 1 1 0 0 0 0 0 0 0 1 1 0 1 1 0 1 0 0 1 1 0 1 0 0 0 0 1
 0 0 1 0 1 1 1 0 1 1 1 1 1 0 0 0 0 1 0 1 1 1 1 1 0 1 0 1 1 1 1 1 1 0 0 0 0
 1 0 1 1 1 1 0 0 0 1 0 1 0 1 0 1 1 1 0 0 1 0 0 1 0 1 0 0 0 0 0 0 0 1 1 0 1 1
 0 1 1 0 1 1 0 1 1 0 0 0 0 0 1 0 0 0 1 0 0 1 0 0 0 1 1 1 0 1 1 1 0 0 0 0 0
 0 0 1 0 1 1 1 1 1 0 0 0 0 0 1 0 1 0 1 0 1 1 1 1 0 0 0 1 0 1 0 1 1 1 0 1 1
 0 0 1 0 1 0 0 1 0 1 1 0 0 1 1 1 1 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1
 1 0 0 0 1 1 0 0 0 0 0 1 1 1 0 0 0 0 1 0 1 0 0 0 0 1 1 1 0 1 1 1 0 1 1 1 1
 0 1 1 0 0 1 1 0 0 1 0 1 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1 1 0 1 1 1 0 0 1 0
 0 0 1 1 0 1 0 0 0 0 0 0 1 1 0 1 1 0 1 0 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1
 0 0 1 1 1 1 0 1 0 0 1 0 0 1 0 0 1 0 1 0 1 1 1 0 1 0 1 0 1 0 1 0 1 1 0 1 1
 1 0 0 0 1 1 1 1 0 1 1 1 1 0 1 0 1 1 0 1 0 0 1 0 1 0 0 1 0 1 1 1 0 0 1 0 0
 0 0 1 0 0 0 0 1 0 0 0 1 0 0 0 1 1 0 0 1 0 1 0 1 1 0 0 0 1 0 0 1 1 1 1 0 1
 0 1 0 1 0 0 0 0 1 0 0 1 0 0 0 1 0 0 1 0 1 0 1 0 1 0 0 1 0 1 0 1 1 1 0 1 1
 0 0 1 1 0 1 0 1 1 0 0 1 0 1 1 1 1 1 1 1 0 1 1 0 1 1 1 0 0 1 0 0 1 0 0 0 1
 0 1 0 0 0 1 0 0 1 0 1 0 1 0 0 0 0 1 1 0 0 1 1 0 0 0 0 0 1 0 0 1 1 0 1 0 1
 1 1 0 1 0 1 1 1 1 1 1 1 0 1 1 1 0 1 1 0 0 1 0 1 0 1 0 0 1 0 1 1 0 0 1 0 0
 0 0 1 0 1 1 0 0 0 1 0 0 0 0 1 1 1 0 0 1 1 0 0 1 0 1 0 0 1 0 0 0 1 0 1 0 0
 0]


C:\Users\HP\Anaconda3\lib\site-packages\sklearn\preprocessing\label.py:235: DataConver
sionWarning: A column-vector y was passed when a 1d array was expected. Please change
the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

In [19]:
```
print(df['race/ethnicity'].value_counts())
```

```
group C    319
group D    262
group B    190
group E    140
group A     89
Name: race/ethnicity, dtype: int64
```

In [20]:
```
df_Lunch = pd.get_dummies(df['lunch'])
df_new = pd.concat([df, df_Lunch], axis=1)
print(df_new)
```

```
     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
..      ...            ...                         ...           ...
995  female        group E             master's degree      standard
996    male        group C                 high school  free/reduced
997  female        group C                 high school  free/reduced
998  female        group D                some college      standard
999  female        group D                some college  free/reduced

    test preparation course  math score  reading score  writing score  \
0                      none          72             72            NaN
1                 completed          69             90           88.0
2                      none          90             95           93.0
3                      none          47             57           44.0
4                      none          76             78           75.0
..                      ...         ...            ...            ...
995               completed          88             99           95.0
996                    none          62             55           55.0
997               completed          59             71           65.0
998               completed          68             78           77.0
999                    none          77             86           86.0

     free/reduced  standard
0               0         1
1               0         1
2               0         1
3               1         0
4               0         1
..            ...       ...
995             0         1
996             1         0
997             1         0
998             0         1
999             1         0

[1000 rows x 10 columns]
```

In [ ]: