Reading the .csv file and creating a dataframe using pandas library

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import scipy
from scipy import stats

df = pd.read_csv("43.csv")
df.head()
```

```
   gender     race parental level of education         lunch  \
0  female  group B           bachelor's degree      standard
1  female  group C            some high school  free/reduced
2    male  group C             master's degree  free/reduced
3  female  group C                  high school  free/reduced
4  female  group C          associate's degree  free/reduced

  test preparation course  math score  reading score  writing score
0                    none        99.0           85.0           88.0
1                    none        96.0          103.0          102.0
2                    none       117.0           29.0          107.0
3               completed        74.0           70.0           58.0
4                    none       103.0           98.0           89.0
```

Data cleaning

handling impossible values (replacing with nan) --solution to first half of introductory question 2

```
#checking how many values greater than 100
(df["math score"]>100).sum()##----309 VALUES

#checking how many values less than 0
(df["math score"]<0).sum() ##----0 VALUES


df['math score']=df['math score'].apply(lambda x: np.nan if (x > 100
or x < 0) else x)

#do the same for reading score and writing score
df['reading score']=df['reading score'].apply(lambda x: np.nan if (x >
100 or x < 0) else x)

df['writing score']=df['writing score'].apply(lambda x: np.nan if (x >
100 or x < 0) else x)
```

replacing Nan values in marks with respective means

```
mathscoremean = df['math score'].mean()
readingscoremean = df['reading score'].mean()
writingscoremean = df['writing score'].mean()

df['math score'].fillna(mathscoremean,inplace=True)
df['reading score'].fillna(readingscoremean, inplace=True)
df['writing score'].fillna(writingscoremean, inplace=True)
df['math score']
```

```
0      99.000000
1      96.000000
2      83.587464
3      74.000000
4      83.587464
         ...
995    83.587464
996    89.000000
997    86.000000
998    95.000000
999    83.587464
Name: math score, Length: 1000, dtype: float64
```

calculating and adding percentages column

--solution to introductory question 1 -- done by calculating sum of marks columns and dividing by 3

```
df['percentage'] =df.iloc[:,-3:-1].sum(axis=1)/3
```

Removing any row with missing categorical data --solution to second half of introductory question 2

```
df.dropna(subset=['gender', 'race', 'parental level of education',
'lunch',
                  'test preparation course'],inplace=True)

df['reading score'].min()
```

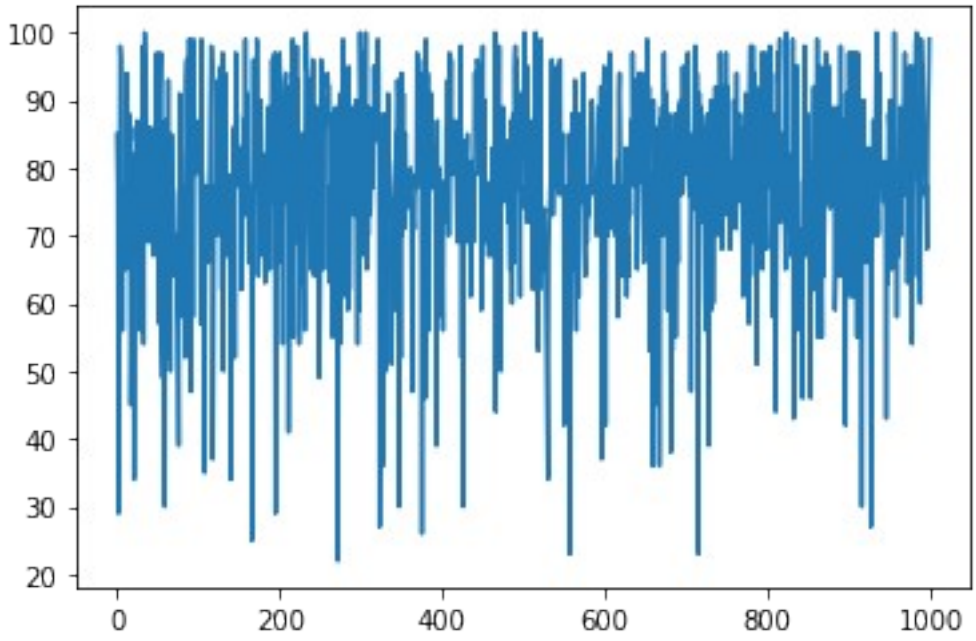
```
22.0
```

Plotting the reading score

plot of the marks scored by 100 students

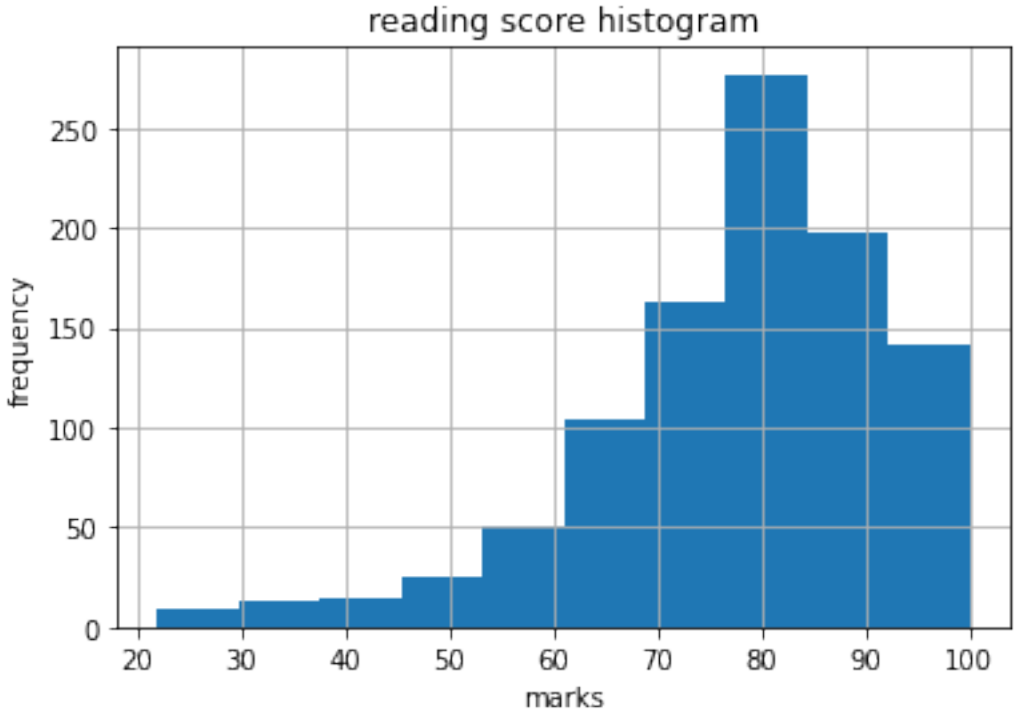
```
readingscores = df['reading score']
readingscores.plot()
```

```
<AxesSubplot:>
```




```
readingscores.hist()
plt.ylabel("frequency")
plt.xlabel("marks")
plt.title('reading score histogram')
```

```
Text(0.5, 1.0, 'reading score histogram')
```



```
scipy.stats.probplot(readingscores.values, dist="norm", plot=plt)
```

```
((array([-3.19614407e+00, -2.93082153e+00, -2.78275298e+00, -
2.67831789e+00,
         -2.59683872e+00, -2.52964120e+00, -2.47223199e+00, -
2.42197180e+00,
         -2.37717389e+00, -2.33669258e+00, -2.29971295e+00, -
2.26563401e+00,
         -2.23399937e+00, -2.20445406e+00, -2.17671631e+00, -
2.15055863e+00,
         -2.12579459e+00, -2.10226944e+00, -2.07985327e+00, -
2.05843592e+00,
         -2.03792314e+00, -2.01823365e+00, -1.99929684e+00, -
1.98105096e+00,
         -1.96344170e+00, -1.94642103e+00, -1.92994625e+00, -
1.91397921e+00,
         -1.89848574e+00, -1.88343502e+00, -1.86879924e+00, -
1.85455314e+00,
         -1.84067379e+00, -1.82714021e+00, -1.81393324e+00, -
1.80103530e+00,
         -1.78843018e+00, -1.77610299e+00, -1.76403993e+00, -
1.75222824e+00,
         -1.74065607e+00, -1.72931241e+00, -1.71818701e+00, -
1.70727031e+00,
         -1.69655335e+00, -1.68602778e+00, -1.67568575e+00, -
1.66551991e+00,
         -1.65552333e+00, -1.64568950e+00, -1.63601229e+00, -
1.62648591e+00,
         -1.61710489e+00, -1.60786406e+00, -1.59875853e+00, -
1.58978365e+00,
         -1.58093503e+00, -1.57220849e+00, -1.56360006e+00, -
1.55510597e+00,
         -1.54672262e+00, -1.53844658e+00, -1.53027460e+00, -
1.52220355e+00,
         -1.51423046e+00, -1.50635248e+00, -1.49856690e+00, -
1.49087111e+00,
         -1.48326261e+00, -1.47573903e+00, -1.46829806e+00, -
1.46093751e+00,
         -1.45365527e+00, -1.44644932e+00, -1.43931770e+00, -
1.43225854e+00,
         -1.42527004e+00, -1.41835046e+00, -1.41149814e+00, -
1.40471146e+00,
         -1.39798887e+00, -1.39132887e+00, -1.38473002e+00, -
1.37819092e+00,
         -1.37171023e+00, -1.36528664e+00, -1.35891890e+00, -
1.35260579e+00,
         -1.34634613e+00, -1.34013879e+00, -1.33398265e+00, -
1.32787667e+00,
         -1.32181979e+00, -1.31581102e+00, -1.30984938e+00, -
1.30393394e+00,
```

```
         -1.29806378e+00, -1.29223801e+00, -1.28645577e+00, -
1.28071622e+00,
         -1.27501856e+00, -1.26936199e+00, -1.26374575e+00, -
1.25816909e+00,
         -1.25263128e+00, -1.24713163e+00, -1.24166943e+00, -
1.23624404e+00,
         -1.23085479e+00, -1.22550106e+00, -1.22018222e+00, -
1.21489768e+00,
         -1.20964685e+00, -1.20442917e+00, -1.19924407e+00, -
1.19409101e+00,
         -1.18896946e+00, -1.18387892e+00, -1.17881887e+00, -
1.17378882e+00,
         -1.16878830e+00, -1.16381683e+00, -1.15887397e+00, -
1.15395925e+00,
         -1.14907225e+00, -1.14421254e+00, -1.13937971e+00, -
1.13457334e+00,
         -1.12979304e+00, -1.12503842e+00, -1.12030909e+00, -
1.11560470e+00,
         -1.11092486e+00, -1.10626923e+00, -1.10163745e+00, -
1.09702919e+00,
         -1.09244410e+00, -1.08788187e+00, -1.08334217e+00, -
1.07882469e+00,
         -1.07432912e+00, -1.06985515e+00, -1.06540250e+00, -
1.06097087e+00,
         -1.05655998e+00, -1.05216955e+00, -1.04779931e+00, -
1.04344900e+00,
         -1.03911833e+00, -1.03480708e+00, -1.03051496e+00, -
1.02624175e+00,
         -1.02198720e+00, -1.01775107e+00, -1.01353312e+00, -
1.00933313e+00,
         -1.00515087e+00, -1.00098611e+00, -9.96838646e-01, -
9.92708257e-01,
         -9.88594736e-01, -9.84497874e-01, -9.80417471e-01, -
9.76353326e-01,
         -9.72305244e-01, -9.68273034e-01, -9.64256504e-01, -
9.60255471e-01,
         -9.56269751e-01, -9.52299165e-01, -9.48343535e-01, -
9.44402689e-01,
         -9.40476456e-01, -9.36564667e-01, -9.32667157e-01, -
9.28783764e-01,
         -9.24914327e-01, -9.21058689e-01, -9.17216695e-01, -
9.13388193e-01,
         -9.09573032e-01, -9.05771065e-01, -9.01982145e-01, -
8.98206130e-01,
         -8.94442879e-01, -8.90692253e-01, -8.86954114e-01, -
8.83228329e-01,
         -8.79514764e-01, -8.75813289e-01, -8.72123774e-01, -
8.68446093e-01,
         -8.64780121e-01, -8.61125734e-01, -8.57482811e-01, -
8.53851232e-01,
```

```
         -8.50230879e-01, -8.46621637e-01, -8.43023389e-01, -
8.39436023e-01,
         -8.35859428e-01, -8.32293493e-01, -8.28738110e-01, -
8.25193172e-01,
         -8.21658574e-01, -8.18134212e-01, -8.14619983e-01, -
8.11115785e-01,
         -8.07621519e-01, -8.04137087e-01, -8.00662390e-01, -
7.97197334e-01,
         -7.93741822e-01, -7.90295763e-01, -7.86859063e-01, -
7.83431632e-01,
         -7.80013379e-01, -7.76604216e-01, -7.73204055e-01, -
7.69812810e-01,
         -7.66430395e-01, -7.63056726e-01, -7.59691720e-01, -
7.56335294e-01,
         -7.52987367e-01, -7.49647859e-01, -7.46316690e-01, -
7.42993782e-01,
         -7.39679058e-01, -7.36372441e-01, -7.33073856e-01, -
7.29783229e-01,
         -7.26500484e-01, -7.23225550e-01, -7.19958354e-01, -
7.16698826e-01,
         -7.13446895e-01, -7.10202490e-01, -7.06965545e-01, -
7.03735990e-01,
         -7.00513758e-01, -6.97298783e-01, -6.94091000e-01, -
6.90890342e-01,
         -6.87696747e-01, -6.84510150e-01, -6.81330489e-01, -
6.78157702e-01,
         -6.74991726e-01, -6.71832502e-01, -6.68679969e-01, -
6.65534068e-01,
         -6.62394739e-01, -6.59261926e-01, -6.56135569e-01, -
6.53015612e-01,
         -6.49901999e-01, -6.46794674e-01, -6.43693581e-01, -
6.40598666e-01,
         -6.37509875e-01, -6.34427155e-01, -6.31350451e-01, -
6.28279713e-01,
         -6.25214887e-01, -6.22155924e-01, -6.19102770e-01, -
6.16055377e-01,
         -6.13013694e-01, -6.09977673e-01, -6.06947263e-01, -
6.03922417e-01,
         -6.00903086e-01, -5.97889224e-01, -5.94880783e-01, -
5.91877716e-01,
         -5.88879977e-01, -5.85887522e-01, -5.82900303e-01, -
5.79918277e-01,
         -5.76941399e-01, -5.73969625e-01, -5.71002912e-01, -
5.68041215e-01,
         -5.65084493e-01, -5.62132703e-01, -5.59185802e-01, -
5.56243750e-01,
         -5.53306504e-01, -5.50374024e-01, -5.47446270e-01, -
5.44523200e-01,
         -5.41604776e-01, -5.38690958e-01, -5.35781706e-01, -
5.32876981e-01,
```

```
         -5.29976746e-01, -5.27080962e-01, -5.24189591e-01, -
5.21302596e-01,
         -5.18419939e-01, -5.15541583e-01, -5.12667493e-01, -
5.09797631e-01,
         -5.06931961e-01, -5.04070449e-01, -5.01213058e-01, -
4.98359754e-01,
         -4.95510501e-01, -4.92665265e-01, -4.89824012e-01, -
4.86986707e-01,
         -4.84153318e-01, -4.81323810e-01, -4.78498150e-01, -
4.75676306e-01,
         -4.72858244e-01, -4.70043933e-01, -4.67233339e-01, -
4.64426432e-01,
         -4.61623179e-01, -4.58823549e-01, -4.56027510e-01, -
4.53235032e-01,
         -4.50446084e-01, -4.47660635e-01, -4.44878655e-01, -
4.42100114e-01,
         -4.39324982e-01, -4.36553229e-01, -4.33784826e-01, -
4.31019744e-01,
         -4.28257953e-01, -4.25499425e-01, -4.22744131e-01, -
4.19992042e-01,
         -4.17243131e-01, -4.14497369e-01, -4.11754729e-01, -
4.09015182e-01,
         -4.06278702e-01, -4.03545260e-01, -4.00814830e-01, -
3.98087386e-01,
         -3.95362899e-01, -3.92641344e-01, -3.89922694e-01, -
3.87206923e-01,
         -3.84494005e-01, -3.81783914e-01, -3.79076624e-01, -
3.76372109e-01,
         -3.73670345e-01, -3.70971306e-01, -3.68274966e-01, -
3.65581301e-01,
         -3.62890286e-01, -3.60201896e-01, -3.57516108e-01, -
3.54832895e-01,
         -3.52152235e-01, -3.49474103e-01, -3.46798475e-01, -
3.44125328e-01,
         -3.41454638e-01, -3.38786380e-01, -3.36120533e-01, -
3.33457072e-01,
         -3.30795975e-01, -3.28137218e-01, -3.25480779e-01, -
3.22826634e-01,
         -3.20174762e-01, -3.17525139e-01, -3.14877744e-01, -
3.12232554e-01,
         -3.09589546e-01, -3.06948700e-01, -3.04309992e-01, -
3.01673402e-01,
         -2.99038907e-01, -2.96406486e-01, -2.93776117e-01, -
2.91147780e-01,
         -2.88521452e-01, -2.85897113e-01, -2.83274741e-01, -
2.80654316e-01,
         -2.78035816e-01, -2.75419222e-01, -2.72804512e-01, -
2.70191665e-01,
         -2.67580662e-01, -2.64971482e-01, -2.62364105e-01, -
2.59758510e-01,
```

```
         -2.57154677e-01, -2.54552587e-01, -2.51952219e-01, -
2.49353553e-01,
         -2.46756571e-01, -2.44161251e-01, -2.41567575e-01, -
2.38975523e-01,
         -2.36385076e-01, -2.33796214e-01, -2.31208918e-01, -
2.28623169e-01,
         -2.26038948e-01, -2.23456235e-01, -2.20875012e-01, -
2.18295260e-01,
         -2.15716959e-01, -2.13140092e-01, -2.10564640e-01, -
2.07990583e-01,
         -2.05417904e-01, -2.02846583e-01, -2.00276603e-01, -
1.97707945e-01,
         -1.95140591e-01, -1.92574523e-01, -1.90009722e-01, -
1.87446170e-01,
         -1.84883849e-01, -1.82322742e-01, -1.79762830e-01, -
1.77204096e-01,
         -1.74646521e-01, -1.72090089e-01, -1.69534780e-01, -
1.66980578e-01,
         -1.64427465e-01, -1.61875423e-01, -1.59324435e-01, -
1.56774483e-01,
         -1.54225550e-01, -1.51677619e-01, -1.49130673e-01, -
1.46584693e-01,
         -1.44039663e-01, -1.41495566e-01, -1.38952384e-01, -
1.36410101e-01,
         -1.33868699e-01, -1.31328161e-01, -1.28788471e-01, -
1.26249611e-01,
         -1.23711565e-01, -1.21174315e-01, -1.18637845e-01, -
1.16102138e-01,
         -1.13567178e-01, -1.11032947e-01, -1.08499428e-01, -
1.05966607e-01,
         -1.03434464e-01, -1.00902985e-01, -9.83721525e-02, -
9.58419497e-02,
         -9.33123603e-02, -9.07833679e-02, -8.82549559e-02, -
8.57271081e-02,
         -8.31998079e-02, -8.06730390e-02, -7.81467851e-02, -
7.56210298e-02,
         -7.30957569e-02, -7.05709499e-02, -6.80465928e-02, -
6.55226692e-02,
         -6.29991630e-02, -6.04760579e-02, -5.79533377e-02, -
5.54309862e-02,
         -5.29089874e-02, -5.03873251e-02, -4.78659831e-02, -
4.53449454e-02,
         -4.28241959e-02, -4.03037184e-02, -3.77834970e-02, -
3.52635155e-02,
         -3.27437579e-02, -3.02242082e-02, -2.77048504e-02, -
2.51856683e-02,
         -2.26666462e-02, -2.01477678e-02, -1.76290173e-02, -
1.51103786e-02,
         -1.25918357e-02, -1.00733728e-02, -7.55497366e-03, -
5.03662249e-03,
```

```
         -2.51830326e-03,  0.00000000e+00,  2.51830326e-03,
5.03662249e-03,
          7.55497366e-03,  1.00733728e-02,  1.25918357e-02,
1.51103786e-02,
          1.76290173e-02,  2.01477678e-02,  2.26666462e-02,
2.51856683e-02,
          2.77048504e-02,  3.02242082e-02,  3.27437579e-02,
3.52635155e-02,
          3.77834970e-02,  4.03037184e-02,  4.28241959e-02,
4.53449454e-02,
          4.78659831e-02,  5.03873251e-02,  5.29089874e-02,
5.54309862e-02,
          5.79533377e-02,  6.04760579e-02,  6.29991630e-02,
6.55226692e-02,
          6.80465928e-02,  7.05709499e-02,  7.30957569e-02,
7.56210298e-02,
          7.81467851e-02,  8.06730390e-02,  8.31998079e-02,
8.57271081e-02,
          8.82549559e-02,  9.07833679e-02,  9.33123603e-02,
9.58419497e-02,
          9.83721525e-02,  1.00902985e-01,  1.03434464e-01,
1.05966607e-01,
          1.08499428e-01,  1.11032947e-01,  1.13567178e-01,
1.16102138e-01,
          1.18637845e-01,  1.21174315e-01,  1.23711565e-01,
1.26249611e-01,
          1.28788471e-01,  1.31328161e-01,  1.33868699e-01,
1.36410101e-01,
          1.38952384e-01,  1.41495566e-01,  1.44039663e-01,
1.46584693e-01,
          1.49130673e-01,  1.51677619e-01,  1.54225550e-01,
1.56774483e-01,
          1.59324435e-01,  1.61875423e-01,  1.64427465e-01,
1.66980578e-01,
          1.69534780e-01,  1.72090089e-01,  1.74646521e-01,
1.77204096e-01,
          1.79762830e-01,  1.82322742e-01,  1.84883849e-01,
1.87446170e-01,
          1.90009722e-01,  1.92574523e-01,  1.95140591e-01,
1.97707945e-01,
          2.00276603e-01,  2.02846583e-01,  2.05417904e-01,
2.07990583e-01,
          2.10564640e-01,  2.13140092e-01,  2.15716959e-01,
2.18295260e-01,
          2.20875012e-01,  2.23456235e-01,  2.26038948e-01,
2.28623169e-01,
          2.31208918e-01,  2.33796214e-01,  2.36385076e-01,
2.38975523e-01,
          2.41567575e-01,  2.44161251e-01,  2.46756571e-01,
2.49353553e-01,
```

```
          2.51952219e-01,  2.54552587e-01,  2.57154677e-01,
2.59758510e-01,
          2.62364105e-01,  2.64971482e-01,  2.67580662e-01,
2.70191665e-01,
          2.72804512e-01,  2.75419222e-01,  2.78035816e-01,
2.80654316e-01,
          2.83274741e-01,  2.85897113e-01,  2.88521452e-01,
2.91147780e-01,
          2.93776117e-01,  2.96406486e-01,  2.99038907e-01,
3.01673402e-01,
          3.04309992e-01,  3.06948700e-01,  3.09589546e-01,
3.12232554e-01,
          3.14877744e-01,  3.17525139e-01,  3.20174762e-01,
3.22826634e-01,
          3.25480779e-01,  3.28137218e-01,  3.30795975e-01,
3.33457072e-01,
          3.36120533e-01,  3.38786380e-01,  3.41454638e-01,
3.44125328e-01,
          3.46798475e-01,  3.49474103e-01,  3.52152235e-01,
3.54832895e-01,
          3.57516108e-01,  3.60201896e-01,  3.62890286e-01,
3.65581301e-01,
          3.68274966e-01,  3.70971306e-01,  3.73670345e-01,
3.76372109e-01,
          3.79076624e-01,  3.81783914e-01,  3.84494005e-01,
3.87206923e-01,
          3.89922694e-01,  3.92641344e-01,  3.95362899e-01,
3.98087386e-01,
          4.00814830e-01,  4.03545260e-01,  4.06278702e-01,
4.09015182e-01,
          4.11754729e-01,  4.14497369e-01,  4.17243131e-01,
4.19992042e-01,
          4.22744131e-01,  4.25499425e-01,  4.28257953e-01,
4.31019744e-01,
          4.33784826e-01,  4.36553229e-01,  4.39324982e-01,
4.42100114e-01,
          4.44878655e-01,  4.47660635e-01,  4.50446084e-01,
4.53235032e-01,
          4.56027510e-01,  4.58823549e-01,  4.61623179e-01,
4.64426432e-01,
          4.67233339e-01,  4.70043933e-01,  4.72858244e-01,
4.75676306e-01,
          4.78498150e-01,  4.81323810e-01,  4.84153318e-01,
4.86986707e-01,
          4.89824012e-01,  4.92665265e-01,  4.95510501e-01,
4.98359754e-01,
          5.01213058e-01,  5.04070449e-01,  5.06931961e-01,
5.09797631e-01,
          5.12667493e-01,  5.15541583e-01,  5.18419939e-01,
5.21302596e-01,
```

```
          5.24189591e-01,  5.27080962e-01,  5.29976746e-01,
5.32876981e-01,
          5.35781706e-01,  5.38690958e-01,  5.41604776e-01,
5.44523200e-01,
          5.47446270e-01,  5.50374024e-01,  5.53306504e-01,
5.56243750e-01,
          5.59185802e-01,  5.62132703e-01,  5.65084493e-01,
5.68041215e-01,
          5.71002912e-01,  5.73969625e-01,  5.76941399e-01,
5.79918277e-01,
          5.82900303e-01,  5.85887522e-01,  5.88879977e-01,
5.91877716e-01,
          5.94880783e-01,  5.97889224e-01,  6.00903086e-01,
6.03922417e-01,
          6.06947263e-01,  6.09977673e-01,  6.13013694e-01,
6.16055377e-01,
          6.19102770e-01,  6.22155924e-01,  6.25214887e-01,
6.28279713e-01,
          6.31350451e-01,  6.34427155e-01,  6.37509875e-01,
6.40598666e-01,
          6.43693581e-01,  6.46794674e-01,  6.49901999e-01,
6.53015612e-01,
          6.56135569e-01,  6.59261926e-01,  6.62394739e-01,
6.65534068e-01,
          6.68679969e-01,  6.71832502e-01,  6.74991726e-01,
6.78157702e-01,
          6.81330489e-01,  6.84510150e-01,  6.87696747e-01,
6.90890342e-01,
          6.94091000e-01,  6.97298783e-01,  7.00513758e-01,
7.03735990e-01,
          7.06965545e-01,  7.10202490e-01,  7.13446895e-01,
7.16698826e-01,
          7.19958354e-01,  7.23225550e-01,  7.26500484e-01,
7.29783229e-01,
          7.33073856e-01,  7.36372441e-01,  7.39679058e-01,
7.42993782e-01,
          7.46316690e-01,  7.49647859e-01,  7.52987367e-01,
7.56335294e-01,
          7.59691720e-01,  7.63056726e-01,  7.66430395e-01,
7.69812810e-01,
          7.73204055e-01,  7.76604216e-01,  7.80013379e-01,
7.83431632e-01,
          7.86859063e-01,  7.90295763e-01,  7.93741822e-01,
7.97197334e-01,
          8.00662390e-01,  8.04137087e-01,  8.07621519e-01,
8.11115785e-01,
          8.14619983e-01,  8.18134212e-01,  8.21658574e-01,
8.25193172e-01,
          8.28738110e-01,  8.32293493e-01,  8.35859428e-01,
8.39436023e-01,
```

```
          8.43023389e-01,  8.46621637e-01,  8.50230879e-01,
8.53851232e-01,
          8.57482811e-01,  8.61125734e-01,  8.64780121e-01,
8.68446093e-01,
          8.72123774e-01,  8.75813289e-01,  8.79514764e-01,
8.83228329e-01,
          8.86954114e-01,  8.90692253e-01,  8.94442879e-01,
8.98206130e-01,
          9.01982145e-01,  9.05771065e-01,  9.09573032e-01,
9.13388193e-01,
          9.17216695e-01,  9.21058689e-01,  9.24914327e-01,
9.28783764e-01,
          9.32667157e-01,  9.36564667e-01,  9.40476456e-01,
9.44402689e-01,
          9.48343535e-01,  9.52299165e-01,  9.56269751e-01,
9.60255471e-01,
          9.64256504e-01,  9.68273034e-01,  9.72305244e-01,
9.76353326e-01,
          9.80417471e-01,  9.84497874e-01,  9.88594736e-01,
9.92708257e-01,
          9.96838646e-01,  1.00098611e+00,  1.00515087e+00,
1.00933313e+00,
          1.01353312e+00,  1.01775107e+00,  1.02198720e+00,
1.02624175e+00,
          1.03051496e+00,  1.03480708e+00,  1.03911833e+00,
1.04344900e+00,
          1.04779931e+00,  1.05216955e+00,  1.05655998e+00,
1.06097087e+00,
          1.06540250e+00,  1.06985515e+00,  1.07432912e+00,
1.07882469e+00,
          1.08334217e+00,  1.08788187e+00,  1.09244410e+00,
1.09702919e+00,
          1.10163745e+00,  1.10626923e+00,  1.11092486e+00,
1.11560470e+00,
          1.12030909e+00,  1.12503842e+00,  1.12979304e+00,
1.13457334e+00,
          1.13937971e+00,  1.14421254e+00,  1.14907225e+00,
1.15395925e+00,
          1.15887397e+00,  1.16381683e+00,  1.16878830e+00,
1.17378882e+00,
          1.17881887e+00,  1.18387892e+00,  1.18896946e+00,
1.19409101e+00,
          1.19924407e+00,  1.20442917e+00,  1.20964685e+00,
1.21489768e+00,
          1.22018222e+00,  1.22550106e+00,  1.23085479e+00,
1.23624404e+00,
          1.24166943e+00,  1.24713163e+00,  1.25263128e+00,
1.25816909e+00,
          1.26374575e+00,  1.26936199e+00,  1.27501856e+00,
1.28071622e+00,
```

```
          1.28645577e+00,  1.29223801e+00,  1.29806378e+00,
1.30393394e+00,
          1.30984938e+00,  1.31581102e+00,  1.32181979e+00,
1.32787667e+00,
          1.33398265e+00,  1.34013879e+00,  1.34634613e+00,
1.35260579e+00,
          1.35891890e+00,  1.36528664e+00,  1.37171023e+00,
1.37819092e+00,
          1.38473002e+00,  1.39132887e+00,  1.39798887e+00,
1.40471146e+00,
          1.41149814e+00,  1.41835046e+00,  1.42527004e+00,
1.43225854e+00,
          1.43931770e+00,  1.44644932e+00,  1.45365527e+00,
1.46093751e+00,
          1.46829806e+00,  1.47573903e+00,  1.48326261e+00,
1.49087111e+00,
          1.49856690e+00,  1.50635248e+00,  1.51423046e+00,
1.52220355e+00,
          1.53027460e+00,  1.53844658e+00,  1.54672262e+00,
1.55510597e+00,
          1.56360006e+00,  1.57220849e+00,  1.58093503e+00,
1.58978365e+00,
          1.59875853e+00,  1.60786406e+00,  1.61710489e+00,
1.62648591e+00,
          1.63601229e+00,  1.64568950e+00,  1.65552333e+00,
1.66551991e+00,
          1.67568575e+00,  1.68602778e+00,  1.69655335e+00,
1.70727031e+00,
          1.71818701e+00,  1.72931241e+00,  1.74065607e+00,
1.75222824e+00,
          1.76403993e+00,  1.77610299e+00,  1.78843018e+00,
1.80103530e+00,
          1.81393324e+00,  1.82714021e+00,  1.84067379e+00,
1.85455314e+00,
          1.86879924e+00,  1.88343502e+00,  1.89848574e+00,
1.91397921e+00,
          1.92994625e+00,  1.94642103e+00,  1.96344170e+00,
1.98105096e+00,
          1.99929684e+00,  2.01823365e+00,  2.03792314e+00,
2.05843592e+00,
          2.07985327e+00,  2.10226944e+00,  2.12579459e+00,
2.15055863e+00,
          2.17671631e+00,  2.20445406e+00,  2.23399937e+00,
2.26563401e+00,
          2.29971295e+00,  2.33669258e+00,  2.37717389e+00,
2.42197180e+00,
          2.47223199e+00,  2.52964120e+00,  2.59683872e+00,
2.67831789e+00,
          2.78275298e+00,  2.93082153e+00,  3.19614407e+00]),
  array([ 22.        ,  23.        ,  23.        ,  25.        ,
```

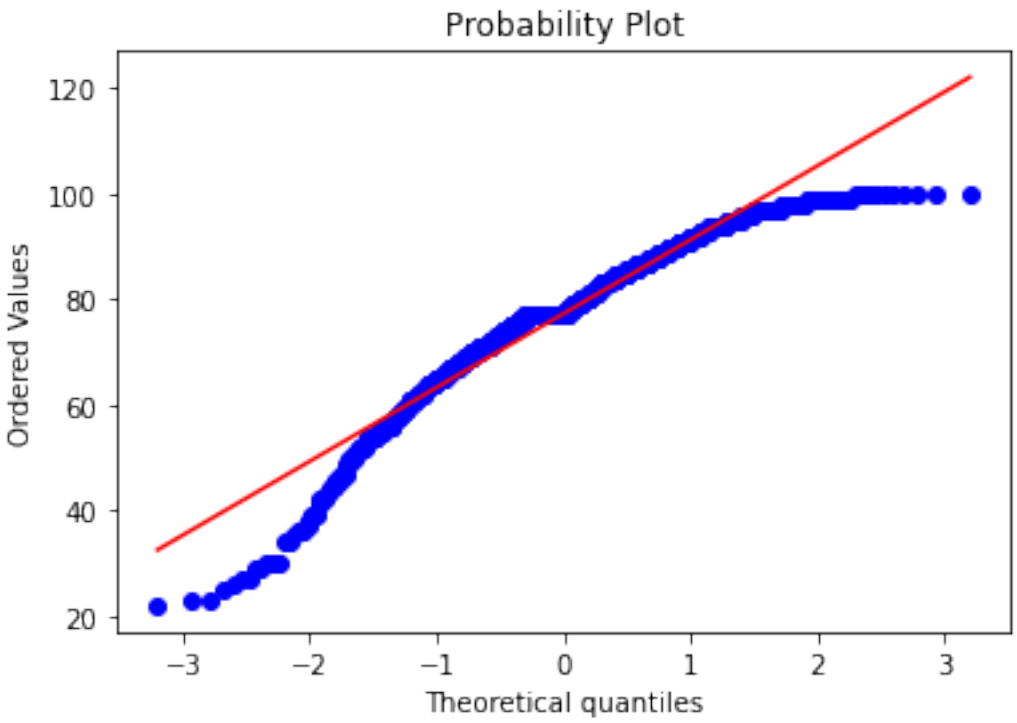
26. , 27. , 27. , 29. ,
29. , 30. , 30. , 30. ,
30. , 34. , 34. , 34. ,
35. , 36. , 36. , 36. ,
37. , 37. , 38. , 39. ,
39. , 39. , 41. , 42. ,
42. , 42. , 43. , 43. ,
44. , 44. , 45. , 45. ,
46. , 46. , 46. , 47. ,
47. , 47. , 47. , 49. ,
49. , 50. , 50. , 50. ,
50. , 51. , 51. , 51. ,
52. , 52. , 52. , 52. ,
52. , 52. , 53. , 53. ,
53. , 54. , 54. , 54. ,
54. , 54. , 54. , 54. ,
54. , 54. , 55. , 55. ,
55. , 55. , 55. , 55. ,
55. , 56. , 56. , 56. ,
56. , 56. , 56. , 56. ,
56. , 56. , 56. , 56. ,
57. , 57. , 57. , 57. ,
58. , 58. , 58. , 58. ,
58. , 58. , 58. , 59. ,
59. , 59. , 59. , 59. ,
59. , 59. , 59. , 60. ,
60. , 60. , 60. , 60. ,
61. , 61. , 61. , 61. ,
61. , 61. , 61. , 61. ,
61. , 61. , 61. , 62. ,
62. , 62. , 62. , 62. ,
62. , 62. , 62. , 62. ,
62. , 62. , 63. , 63. ,
63. , 63. , 63. , 63. ,
63. , 64. , 64. , 64. ,
64. , 64. , 64. , 64. ,
64. , 64. , 64. , 64. ,
64. , 64. , 64. , 65. ,
65. , 65. , 65. , 65. ,
65. , 65. , 65. , 65. ,
65. , 65. , 65. , 65. ,
65. , 65. , 65. , 65. ,
66. , 66. , 66. , 66. ,
66. , 66. , 66. , 66. ,
66. , 66. , 66. , 67. ,
67. , 67. , 67. , 67. ,
67. , 67. , 67. , 67. ,
67. , 67. , 67. , 67. ,
67. , 67. , 67. , 67. ,
67. , 68. , 68. , 68. ,

68. , 68. , 68. , 68. ,
68. , 68. , 68. , 68. ,
68. , 68. , 68. , 68. ,
69. , 69. , 69. , 69. ,
69. , 69. , 69. , 69. ,
69. , 69. , 69. , 69. ,
69. , 69. , 69. , 69. ,
70. , 70. , 70. , 70. ,
70. , 70. , 70. , 70. ,
70. , 70. , 70. , 70. ,
70. , 70. , 70. , 70. ,
70. , 70. , 70. , 71. ,
71. , 71. , 71. , 71. ,
71. , 71. , 71. , 71. ,
71. , 71. , 71. , 71. ,
71. , 71. , 71. , 71. ,
71. , 71. , 71. , 71. ,
71. , 71. , 71. , 71. ,
71. , 71. , 71. , 72. ,
72. , 72. , 72. , 72. ,
72. , 72. , 72. , 72. ,
72. , 72. , 72. , 72. ,
72. , 72. , 72. , 72. ,
73. , 73. , 73. , 73. ,
73. , 73. , 73. , 73. ,
73. , 73. , 73. , 73. ,
73. , 73. , 73. , 73. ,
73. , 73. , 73. , 73. ,
74. , 74. , 74. , 74. ,
74. , 74. , 74. , 74. ,
74. , 74. , 74. , 74. ,
74. , 74. , 74. , 74. ,
74. , 74. , 74. , 74. ,
74. , 75. , 75. , 75. ,
75. , 75. , 75. , 75. ,
75. , 75. , 75. , 75. ,
75. , 75. , 75. , 75. ,
75. , 75. , 75. , 75. ,
75. , 75. , 75. , 75. ,
75. , 76. , 76. , 76. ,
76. , 76. , 76. , 76. ,
76. , 76. , 76. , 76. ,
76. , 76. , 76. , 76. ,
76. , 76. , 76. , 77. ,
77. , 77. , 77. , 77. ,
77. , 77. , 77. , 77. ,
77. , 77. , 77. , 77. ,
77. , 77. , 77. , 77. ,
77. , 77. , 77. , 77. ,
77. , 77. , 77. , 77. ,

```
77.        ,  77.        ,  77.        ,  77.        ,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
77.35077951,  77.35077951,  77.35077951,  77.35077951,
78.        ,  78.        ,  78.        ,  78.        ,
78.        ,  78.        ,  78.        ,  78.        ,
78.        ,  78.        ,  78.        ,  78.        ,
78.        ,  78.        ,  78.        ,  78.        ,
78.        ,  78.        ,  79.        ,  79.        ,
79.        ,  79.        ,  79.        ,  79.        ,
79.        ,  79.        ,  79.        ,  79.        ,
79.        ,  79.        ,  79.        ,  79.        ,
79.        ,  79.        ,  79.        ,  79.        ,
79.        ,  79.        ,  79.        ,  79.        ,
79.        ,  79.        ,  79.        ,  80.        ,
80.        ,  80.        ,  80.        ,  80.        ,
80.        ,  80.        ,  80.        ,  80.        ,
80.        ,  80.        ,  80.        ,  80.        ,
80.        ,  80.        ,  80.        ,  80.        ,
80.        ,  80.        ,  80.        ,  80.        ,
80.        ,  80.        ,  80.        ,  80.        ,
80.        ,  81.        ,  81.        ,  81.        ,
81.        ,  81.        ,  81.        ,  81.        ,
81.        ,  81.        ,  81.        ,  81.        ,
81.        ,  81.        ,  81.        ,  81.        ,
81.        ,  81.        ,  81.        ,  81.        ,
81.        ,  81.        ,  81.        ,  82.        ,
82.        ,  82.        ,  82.        ,  82.        ,
```

82. , 82. , 82. , 82. ,
82. , 82. , 82. , 82. ,
82. , 82. , 83. , 83. ,
83. , 83. , 83. , 83. ,
83. , 83. , 83. , 83. ,
83. , 83. , 83. , 83. ,
83. , 83. , 83. , 83. ,
83. , 83. , 83. , 83. ,
83. , 83. , 84. , 84. ,
84. , 84. , 84. , 84. ,
84. , 84. , 84. , 84. ,
84. , 84. , 84. , 84. ,
84. , 84. , 84. , 84. ,
85. , 85. , 85. , 85. ,
85. , 85. , 85. , 85. ,
85. , 85. , 85. , 85. ,
85. , 85. , 85. , 85. ,
85. , 85. , 85. , 85. ,
85. , 85. , 85. , 85. ,
85. , 85. , 85. , 85. ,
85. , 85. , 85. , 86. ,
86. , 86. , 86. , 86. ,
86. , 86. , 86. , 86. ,
86. , 86. , 86. , 86. ,
86. , 86. , 86. , 86. ,
86. , 86. , 86. , 86. ,
86. , 86. , 86. , 86. ,
86. , 86. , 87. , 87. ,
87. , 87. , 87. , 87. ,
87. , 87. , 87. , 87. ,
87. , 87. , 87. , 87. ,
87. , 87. , 87. , 87. ,
87. , 87. , 87. , 87. ,
87. , 87. , 87. , 87. ,
87. , 88. , 88. , 88. ,
88. , 88. , 88. , 88. ,
88. , 88. , 88. , 88. ,
88. , 88. , 88. , 88. ,
88. , 88. , 88. , 88. ,
88. , 88. , 88. , 88. ,
89. , 89. , 89. , 89. ,
89. , 89. , 89. , 89. ,
89. , 89. , 89. , 89. ,
89. , 89. , 89. , 89. ,
89. , 89. , 89. , 89. ,
89. , 89. , 90. , 90. ,
90. , 90. , 90. , 90. ,
90. , 90. , 90. , 90. ,
90. , 90. , 90. , 90. ,

```
        90.        ,  90.        ,  90.        ,  90.        ,
        90.        ,  90.        ,  90.        ,  90.        ,
        91.        ,  91.        ,  91.        ,  91.        ,
        91.        ,  91.        ,  91.        ,  91.        ,
        91.        ,  91.        ,  91.        ,  91.        ,
        91.        ,  91.        ,  91.        ,  91.        ,
        91.        ,  91.        ,  91.        ,  91.        ,
        91.        ,  91.        ,  91.        ,  92.        ,
        92.        ,  92.        ,  92.        ,  92.        ,
        92.        ,  92.        ,  92.        ,  92.        ,
        92.        ,  92.        ,  92.        ,  92.        ,
        92.        ,  92.        ,  92.        ,  92.        ,
        92.        ,  92.        ,  93.        ,  93.        ,
        93.        ,  93.        ,  93.        ,  93.        ,
        93.        ,  93.        ,  93.        ,  93.        ,
        93.        ,  93.        ,  93.        ,  93.        ,
        93.        ,  94.        ,  94.        ,  94.        ,
        94.        ,  94.        ,  94.        ,  94.        ,
        94.        ,  94.        ,  94.        ,  94.        ,
        94.        ,  94.        ,  94.        ,  94.        ,
        94.        ,  94.        ,  94.        ,  94.        ,
        94.        ,  94.        ,  94.        ,  94.        ,
        94.        ,  94.        ,  94.        ,  95.        ,
        95.        ,  95.        ,  95.        ,  95.        ,
        95.        ,  95.        ,  95.        ,  95.        ,
        95.        ,  95.        ,  95.        ,  95.        ,
        95.        ,  95.        ,  95.        ,  95.        ,
        95.        ,  95.        ,  96.        ,  96.        ,
        96.        ,  96.        ,  96.        ,  96.        ,
        96.        ,  96.        ,  96.        ,  96.        ,
        96.        ,  96.        ,  96.        ,  97.        ,
        97.        ,  97.        ,  97.        ,  97.        ,
        97.        ,  97.        ,  97.        ,  97.        ,
        97.        ,  97.        ,  97.        ,  97.        ,
        97.        ,  97.        ,  97.        ,  97.        ,
        97.        ,  97.        ,  97.        ,  97.        ,
        97.        ,  97.        ,  97.        ,  97.        ,
        98.        ,  98.        ,  98.        ,  98.        ,
        98.        ,  98.        ,  98.        ,  98.        ,
        98.        ,  98.        ,  98.        ,  98.        ,
        98.        ,  98.        ,  98.        ,  99.        ,
        99.        ,  99.        ,  99.        ,  99.        ,
        99.        ,  99.        ,  99.        ,  99.        ,
        99.        ,  99.        ,  99.        ,  99.        ,
        99.        ,  99.        ,  99.        ,  99.        ,
       100.        , 100.        , 100.        , 100.        ,
       100.        , 100.        , 100.        , 100.        ,
       100.        , 100.        , 100.        ]),
(14.01228436393028, 77.32972658392184, 0.9703407383070618))
```



Excluding Outliers the distribution is normal f(x) =1/(sd*root(*2pi)*e^(-1/2*(x-mean)/sd)^2) mean medain mode are equal distribution can be described by the mean and median

generate grades of students

```
def grade(x):
    if(x>90 and x<100):
        return 'S'
    elif(x>80):
        return 'A'
    elif(x>70):
        return 'B'
    elif(x>60):
        return 'C'
    elif(x>40):
        return 'D'
    else:
        return 'F'


df['grade']=df['percentage'].apply(lambda x:grade(x))
df['grade'].head()
```

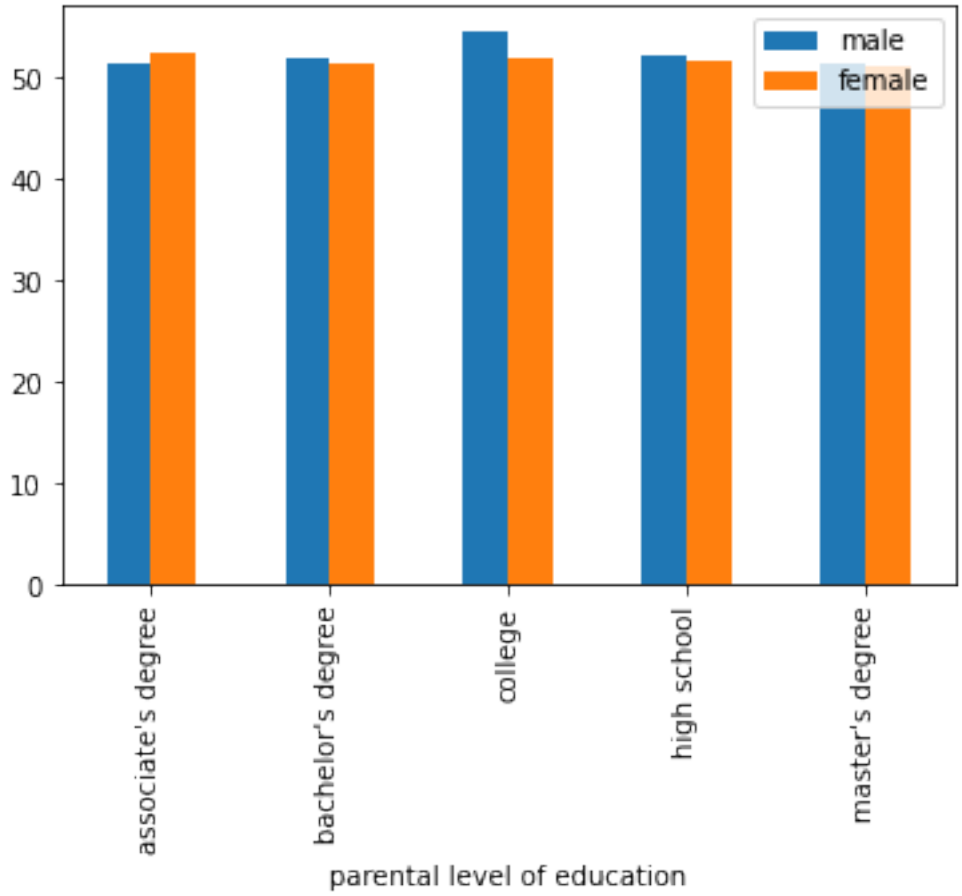
```
0    D
1    D
2    F
3    D
```

```
4    C
Name: grade, dtype: object
```

```
#replacing some high school with high school and some college with
college
df['parental level of education'].replace('some high school','high
school',inplace=True)
df['parental level of education'].replace(
    'some college', 'college', inplace=True)
parent_percentage_relation = df[['parental level of education',
'percentage','gender']]
parent_percentage_relation.dropna(subset=['gender'],inplace=True)
parent_percentage_relation['gender'] =
parent_percentage_relation['gender'].str.lower()
parent_percentage_relation_male =
parent_percentage_relation[parent_percentage_relation['gender']=='male
']
parent_percentage_relation_female =
parent_percentage_relation[parent_percentage_relation['gender'] ==
'female']
parent_percentage_relation_male =
parent_percentage_relation_male.groupby(
    ['parental level of education']).mean()
parent_percentage_relation_female =
parent_percentage_relation_female.groupby(
    ['parental level of education']).mean()

ppgr = pd.DataFrame()
# ppgr['parental level of education']
=parent_percentage_relation_female['parental level of
education'].values
ppgr['male'] = parent_percentage_relation_male['percentage'].values
ppgr['female']=parent_percentage_relation_female['percentage'].values

ppgr['parental level of
education']=(parent_percentage_relation_male.index)
ppgr.plot(x='parental level of education',y=["male", "female"],
kind="bar")
```

```
C:\Users\prach\AppData\Local\Temp/ipykernel_10724/1648020781.py:7:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  parent_percentage_relation['gender'] =
parent_percentage_relation['gender'].str.lower()
```

```
<AxesSubplot:xlabel='parental level of education'>
```



Task questions

```
set(df['race'].values) # checking if there are impossible values
df['race'].isnull().sum() # no nan values

simrstudents = df.sample(n=100)
stratstudents =df.groupby('race', group_keys=False).apply(lambda x:
x.sample(frac = 0.0997))
simmathmean = simrstudents['math score'].mean()
stratmathmean = stratstudents['math score'].mean()

simerror = mathscoremean - simmathmean
straterror = stratmathmean - mathscoremean
```

Sampling Error is lower in case of stratified random sampling

```
df
```

```
     gender     race parental level of education         lunch  \
0    female  group B           bachelor's degree      standard
```

```
1    female  group C                high school  free/reduced
2      male  group C             master's degree  free/reduced
3    female  group C                high school  free/reduced
4    female  group C          associate's degree  free/reduced
..      ...     ...                         ...           ...
995  female  group A                high school      standard
996    male  group B           bachelor's degree      standard
997  female  group C          associate's degree      standard
998  female  group D           bachelor's degree  free/reduced
999  female  group C             master's degree  free/reduced

    test preparation course  math score  reading score  writing score
\
0                      none   99.000000       85.00000      88.000000

1                      none   96.000000       77.35078      77.903955

2                      none   83.587464       29.00000      77.903955

3                 completed   74.000000       70.00000      58.000000

4                      none   83.587464       98.00000      89.000000

..                      ...         ...            ...           ...

995                    none   83.587464       77.35078      77.903955

996                    none   89.000000       68.00000      69.000000

997                    none   86.000000       84.00000      79.000000

998                    none   95.000000       91.00000      91.000000

999               completed   83.587464       99.00000     100.000000


     percentage grades grade
0     57.666667      D     D
1     51.751578      D     D
2     35.634652      F     F
3     42.666667      D     D
4     62.333333      C     C
..          ...    ...   ...
995   51.751578      D     D
996   45.666667      D     D
997   54.333333      D     D
998   60.666667      C     C
999   66.333333      C     C
```

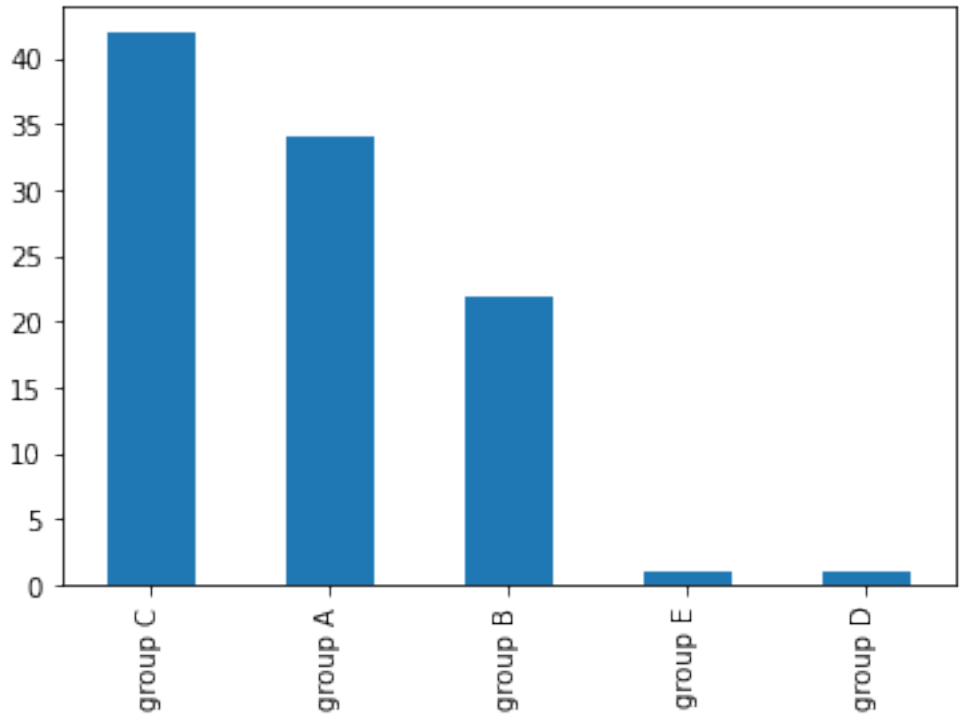
```
[995 rows x 11 columns]
```

Plotting bar graph of race distibution in samples

Simple random sampling (Bar graph with respect to race)

```
simrstudents['race'].value_counts().plot(kind='bar')
```

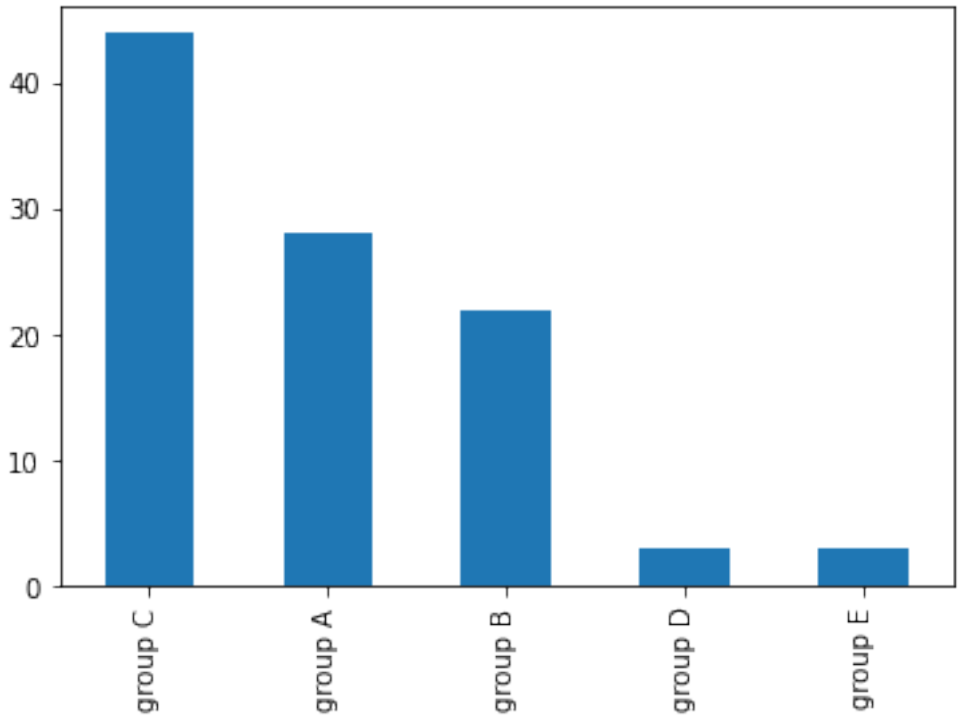
```
<AxesSubplot:>
```



Stratified random sampling (Bar graph with respect to race)

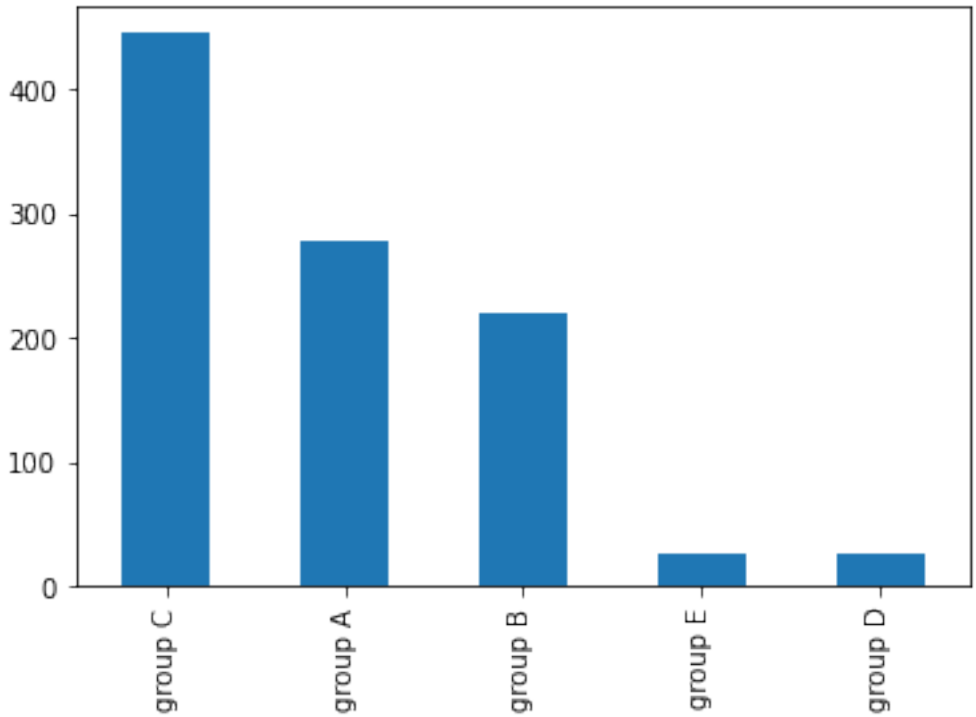
```
stratstudents['race'].value_counts().plot(kind='bar')
```

```
<AxesSubplot:>
```




```
df['race'].value_counts().plot(kind='bar')
```

```
<AxesSubplot:>
```

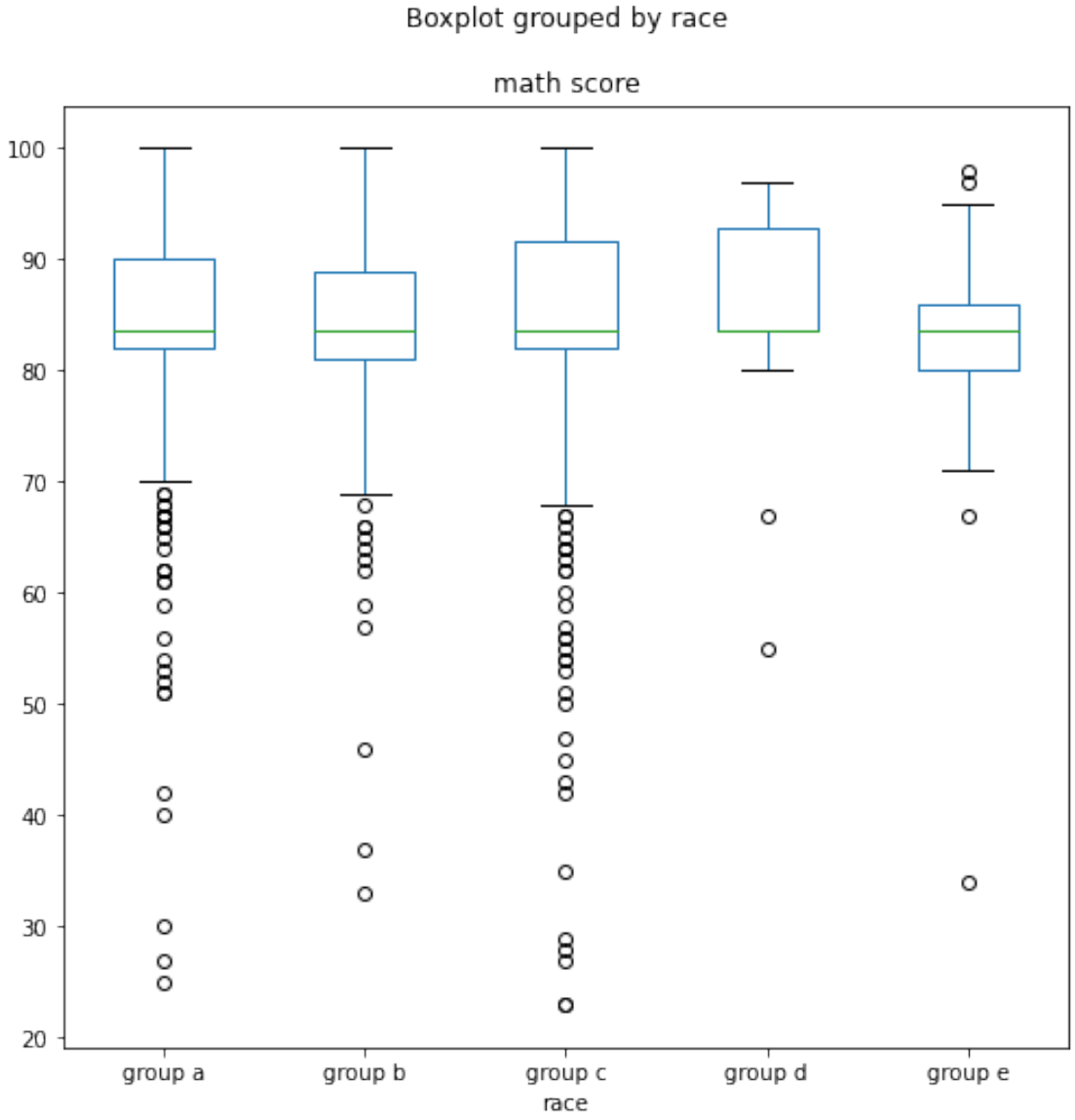


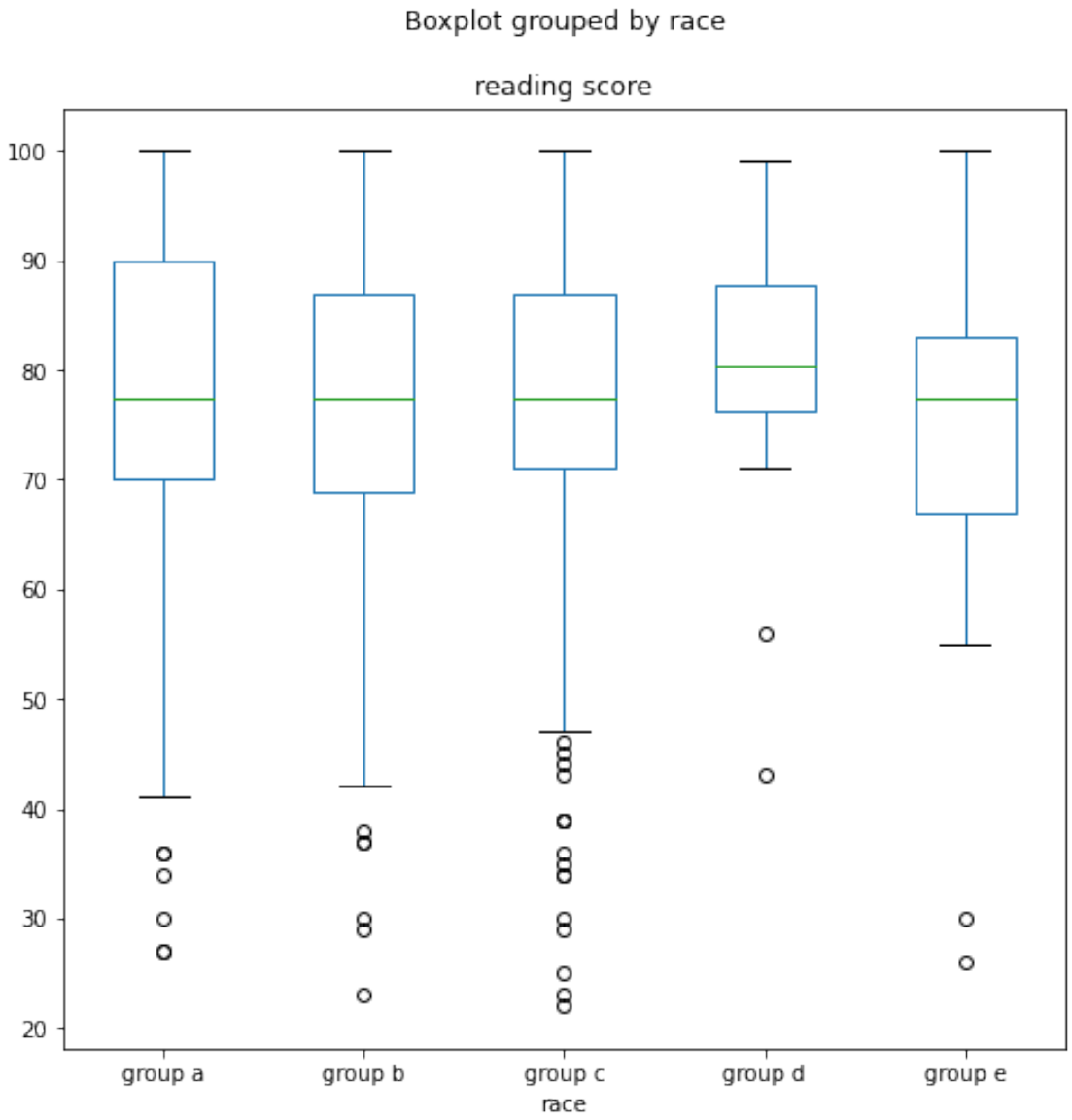
The graphs for simple and stratified random sampling are very similar to that of the Population However the stratified random sample has better representation of the distribution of race in the actual population

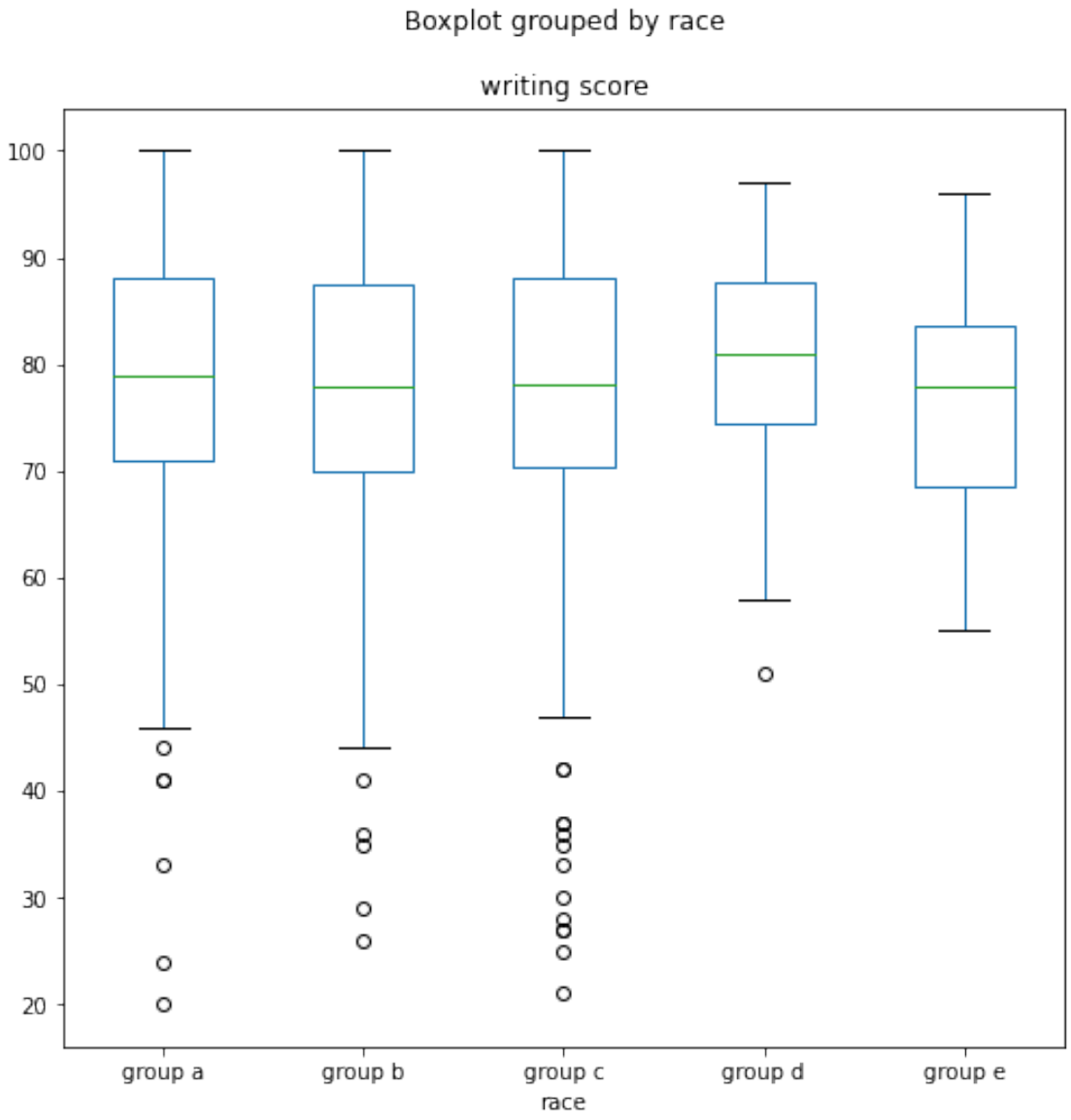
```
df['race'] = df['race'].str.lower()
df['race'].dropna()
```

```
df.boxplot(by='race',column=['math score'],grid=False,figsize=(8,8))
df.boxplot(by='race', column=['reading score'], grid=False,
figsize=(8, 8))
df.boxplot(by='race',column=['writing
score'],grid=False,figsize=(8,8))
```

```
<AxesSubplot:title={'center':'writing score'}, xlabel='race'>
```




Boxplot grouped by race
reading score
100
90
80
70
60
50
40
30
20
group a
group b
group c
group d
group e
race


Boxplot grouped by race
writing score
100
90
80
70
60
50
40
30
20
group a
group b
group c
group d
group e
race