Data Wrangling II Create an “Academic performance” dataset of students and perform the following operations using Python.

1. Scan all variables for missing values and inconsistencies. If there are missing values and/or inconsistencies, use any of the suitable techniques to deal with them.
2. Scan all numeric variables for outliers. If there are outliers, use any of the suitable techniques to deal with them.
3. Apply data transformations on at least one of the variables. The purpose of this transformation should be one of the following reasons: to change the scale for better understanding of the variable, to convert a non-linear relation into a linear one, or to decrease the skewness and convert the distribution into a normal distribution.

```
import pandas as pd
import numpy as np
```

```
df = pd.read_csv("StudentsPerformance.csv")
df
```

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	female	72	72	74.0	78.0	1	Pune
1	female	69	90	88.0	NaN	2	na
2	female	90	95	93.0	74.0	2	Nashik
3	male	47	57	NaN	78.0	1	Na
4	male	na	78	75.0	81.0	3	Pune
5	female	71	Na	78.0	70.0	4	na
6	male	12	44	52.0	12.0	2	Nashik
7	[illegible]	NaN	65	67.0	49.0	1	Pune
[illegible]	[illegible]	[illegible]	77	89.0	55.0	0	NaN

Saved successfully! ×

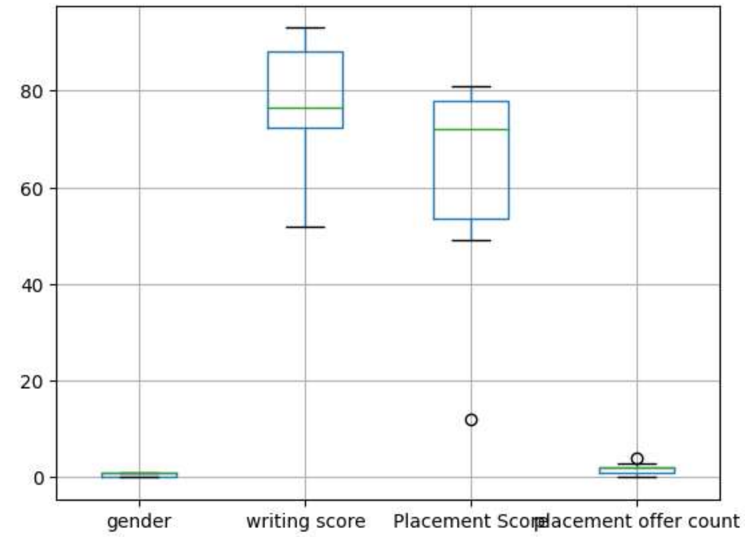
```
df.isnull()
```

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	False	False	False	False	False	False	False
1	False	False	False	False	True	False	False
2	False	False	False	False	False	False	False
3	False	False	False	True	False	False	False
4	False	False	False	False	False	False	False
5	False	False	False	False	False	False	False
6	False	False	False	False	False	False	False
7	False	True	False	False	False	False	False
8	False	False	False	False	False	False	True

```
from sklearn import preprocessing
x = preprocessing.LabelEncoder()
df['gender'] = x.fit_transform(df['gender'])
df
```

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	0	72	72	74.0	78.0	1	Pune

```
import matplotlib.pyplot as plt
boxplot = df.boxplot()
plt.show()
```



```
meanv = df['writing score'].mean()
df['writing score'].fillna(value = meanv, inplace = True)
df
```

Saved successfully!

			score	writing score	Placement Score	placement offer count	Region
0	0	72	72	74.0	78.0	1	Pune
1	0	69	90	88.0	NaN	2	na
2	0	90	95	93.0	74.0	2	Nashik
3	1	47	57	77.0	78.0	1	Na
4	1	na	78	75.0	81.0	3	Pune
5	0	71	Na	78.0	70.0	4	na
6	1	12	44	52.0	12.0	2	Nashik
7	1	NaN	65	67.0	49.0	1	Pune
8	1	5	77	89.0	55.0	0	NaN

```
import scipy.stats as stats
mean = df['writing score'].mean()
std = df['writing score'].std()
zscores = stats.zscore(df['writing score'])
zscores
```

```
0   -0.253546
1    0.929670
2    1.352247
3    0.000000
4   -0.169031
5    0.084515
6   -2.112886
7   -0.845154
8    1.014185
Name: writing score, dtype: float64
```

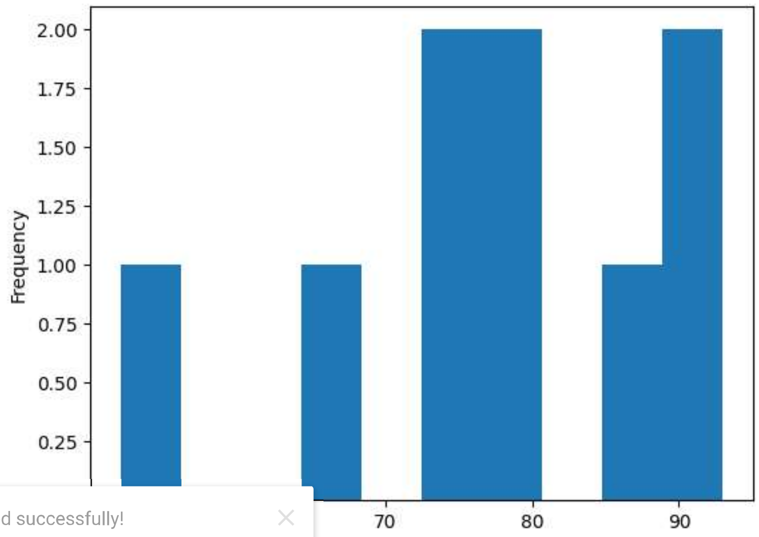
```
threshold = 0
mean = df['writing score'].mean()
std = df['writing score'].std()
outlier=[]
for i in df['writing score']:
   z=(i-mean)/std
   if z>threshold:

    outlier.append(i)
print('outlier is ',outlier)
```

```
outlier is  [88.0, 93.0, 78.0, 89.0]
```

```
import matplotlib.pyplot as plt
df['writing score'].plot(kind='hist')
```

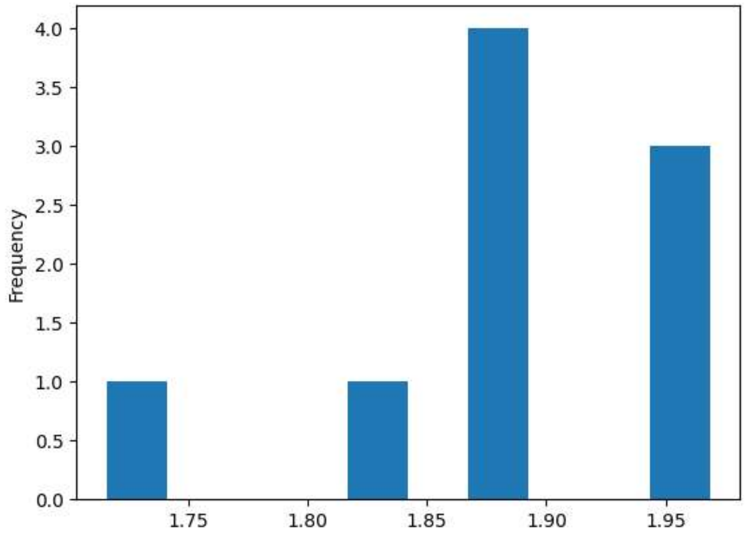
```
<Axes: ylabel='Frequency'>
```



Saved successfully!

```
df['log_math']=np.log10(df['writing score'])
df['log_math'].plot(kind='hist')
```

```
<Axes: ylabel='Frequency'>
```



```
df
```

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region	log_math
0	0	72	72	74.0	78.0	1	Pune	1.869232
1	0	69	90	88.0	NaN	2	na	1.944483
2	0	90	95	93.0	74.0	2	Nashik	1.968483
3	1	47	57	77.0	78.0	1	Na	1.886491
4	1	na	78	75.0	81.0	3	Pune	1.875061
5	0	71	Na	78.0	70.0	4	na	1.892095
6	1	12	44	52.0	12.0	2	Nashik	1.716003
7	1	NaN	65	67.0	49.0	1	Pune	1.826075
8	1	5	77	89.0	55.0	0	NaN	1.949390

Saved successfully!