```
In [249… import pandas as pd
         import numpy as np
         import matplotlib.pyplot as plt
```

```
In [250… df = pd.read_csv("D:/DSBDA/StudentsPerformance_Prac2_100.csv")
```

```
In [251… df
```

Out[251…

…

	Math_Score	Reading_Score	Writing_Score	Placement_Score	Club_Join_Date
0	64	NaN	70.0	88.0	2020.0
1	68	94.0	62.0	79.0	2020.0
2	72	88.0	77.0	90.0	2018.0
3	77	80.0	63.0	99.0	2020.0
4	80	93.0	60.0	96.0	2020.0
...	...	...	...	...	
95	61	84.0	69.0	68.0	NaN
96	72	75.0	86.0	NaN	2020.0
97	95	85.0	80.0	61.0	2021.0
98	69	88.0	55.0	99.0	NaN
99	67	84.0	62.0	81.0	NaN

100 rows × 6 columns

```
In [252… df.head()
```

Out[252…

	Math_Score	Reading_Score	Writing_Score	Placement_Score	Club_Join_Date
0	64	NaN	70.0	88.0	2020.0
1	68	94.0	62.0	79.0	2020.0
2	72	88.0	77.0	90.0	2018.0
3	77	80.0	63.0	99.0	2020.0
4	80	93.0	60.0	96.0	2020.0

```
In [253… df.shape
Out[253…  (100, 6)
In [254… df.columns
Out[254… Index(['Math_Score', 'Reading_Score', 'Writing_Score', 'Placement_Score',
        'Club_Join_Date', 'Placement_Offer_Count'],
       dtype='object')
```

In [255…

```
df.dtypes
```

Out[255…

```
Math_Score                 int64
Reading_Score            float64
Writing_Score            float64
Placement_Score          float64
Club_Join_Date           float64
Placement_Offer_Count      int64
dtype: object
```

In [256…

```
df.isnull().sum()
```

Out[256…

```
Math_Score                0
Reading_Score             9
Writing_Score             7
Placement_Score           3
Club_Join_Date           22
Placement_Offer_Count     0
dtype: int64
```

In [257…

```
df.isnull().sum().sum()
```

Out[257…

```
np.int64(41)
```

In [258…

```
series = pd.isnull(df["Math_Score"])
df[series]
```

Math_Score	Reading_Score	Writing_Score	Placement_Score	Club_Join_Date

Out[258…P

In [259…

```
df['Math_Score'] = df['Math_Score'].fillna(df['Math_Score'].mean())
df['Reading_Score'] =
df['Reading_Score'].fillna(df['Reading_Score'].median()) df['Writing_Score']
= df['Writing_Score'].fillna(df['Writing_Score'].mode())
df['Placement_Score'] =
df['Placement_Score'].fillna(df['Placement_Score'].mea
```

In [260…

```
df
```

Out[260…

	Math_Score	Reading_Score	Writing_Score	Placement_Score	Club_Join_Date
0	64	85.0	70.0	88.000000	2020.0

1	68	94.0	62.0	79.000000	2020.0
2	72	88.0	77.0	90.000000	2018.0
3	77	80.0	63.0	99.000000	2020.0
4	80	93.0	60.0	96.000000	2020.0
... ...	...	...	...	...	
95	61	84.0	69.0	68.000000	NaN
96	72	75.0	86.0	87.804124	2020.0
97	95	85.0	80.0	61.000000	2021.0
98	69	88.0	55.0	99.000000	NaN
99	67	84.0	62.0	81.000000	NaN

100 rows × 6 columns

```
In [261… df['Club_Join_Date'] =
          df['Club_Join_Date'].fillna(df['Club_Join_Date'].mode()  df
```

Out[261…

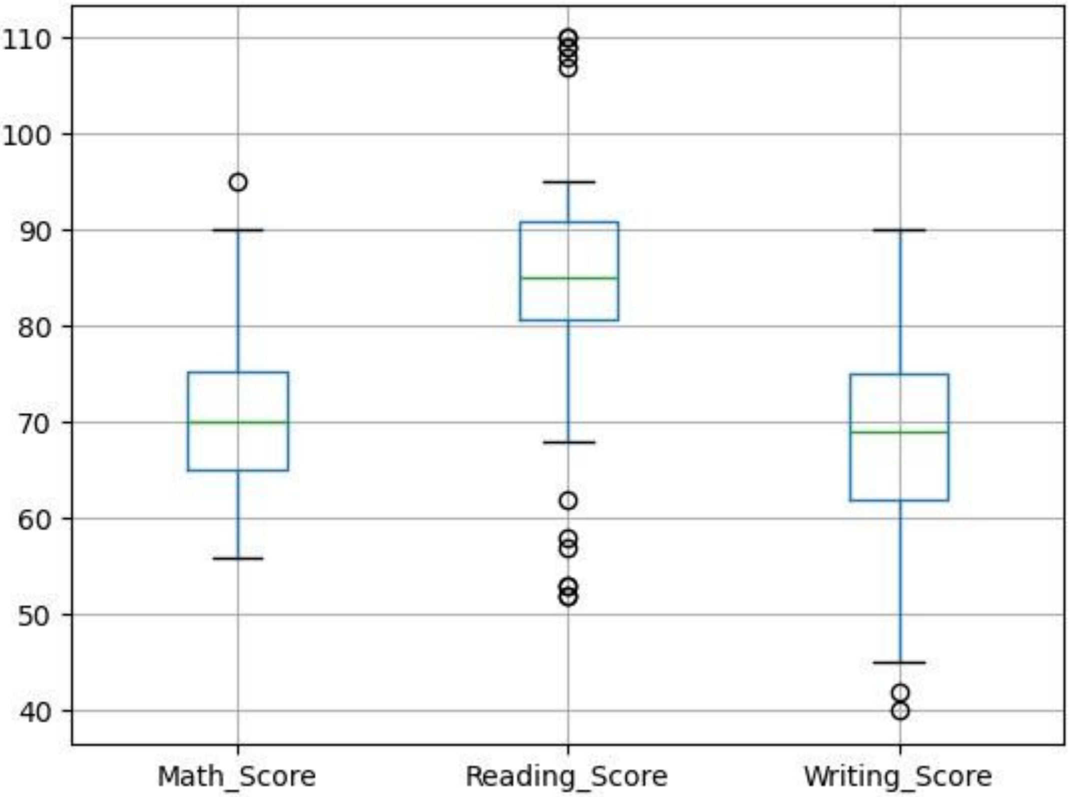
	Math_Score	Reading_Score	Writing_Score	Placement_Score	Club_Join_Date
0	64	85.0	70.0	88.000000	2020.0
1	68	94.0	62.0	79.000000	2020.0
2	72	88.0	77.0	90.000000	2018.0
3	77	80.0	63.0	99.000000	2020.0
... 4	80	93.0	60.0	96.000000	2020.0
...	...	...	...	...	
95	61	84.0	69.0	68.000000	2020.0
96	72	75.0	86.0	87.804124	2020.0
97	95	85.0	80.0	61.000000	2021.0

98	69	88.0	55.0	99.000000	2020.0
99	67	84.0	62.0	81.000000	2020.0

In [262…

100 rows × 6 columns

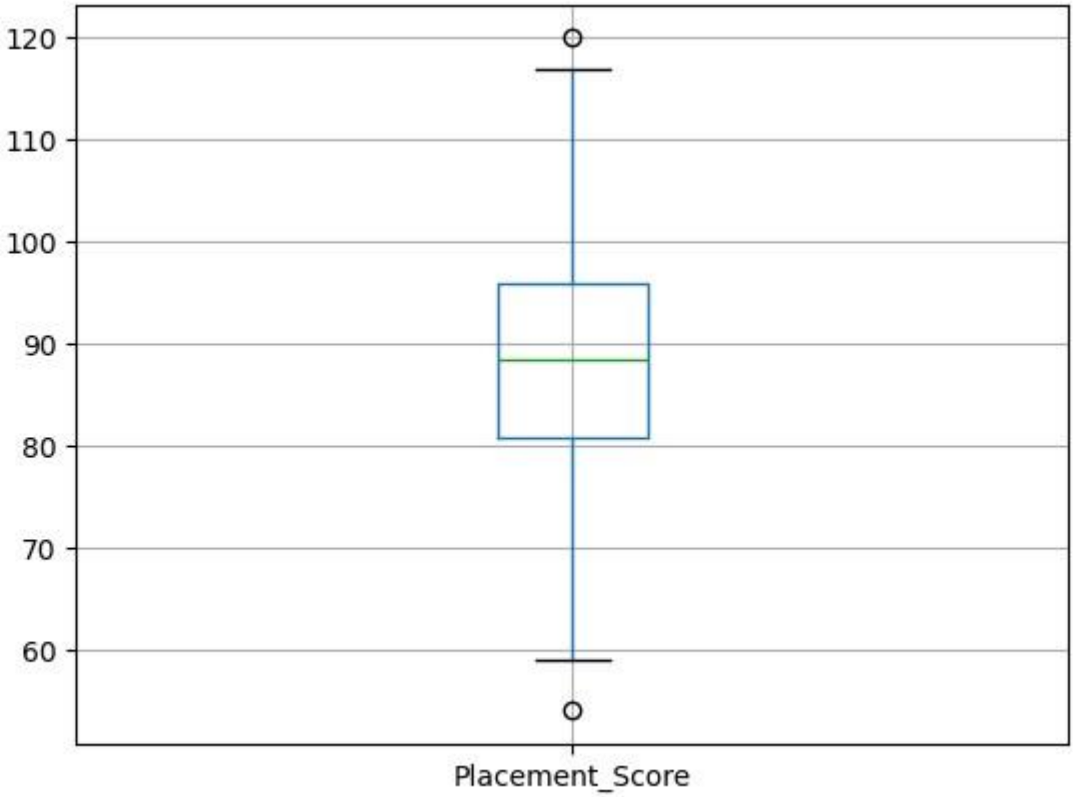
```
col = ['Math_Score', 'Reading_Score',
'Writing_Score'] df.boxplot(col) plt.show()
```



In [263…

```
col = ['Placement_Score']
df.boxplot(col)
plt.show()
```

```
import numpy as np
from scipy import
stats
z = np.abs(stats.zscore(df['Math_Score']))
print(z)
```



In [264…

```
[0.89567475 0.31594999 0.26377476 0.98843071 1.42322428 0.84349952
 0.26377476 1.04060594 0.75074356 0.31594999 0.26377476 0.1710188
 1.04060594 0.98843071 0.84349952 1.33046832 1.47539951 0.40870595
 0.98843071 0.69856833 0.98843071 0.60581237 2.05512426 0.60581237
 1.04060594 1.18553713 0.84349952 1.18553713 1.27829309 0.26377476
 0.11884358 1.33046832 1.27829309 0.40870595 0.84349952 1.1333619
 0.02608761 0.26377476 1.42322428 0.75074356 0.84349952 1.42322428
 0.75074356 1.18553713 0.40870595 0.11884358 0.1710188  1.18553713
 0.40870595 1.27829309 0.02608761 1.42322428 0.26377476 0.11884358
 0.75074356 1.18553713 0.31594999 0.31594999 0.46088118 0.40870595
 0.69856833 1.33046832 1.47539951 0.02608761 0.69856833 1.1333619
 1.27829309 1.18553713 0.11884358 0.98843071 0.46088118 0.89567475
 0.46088118 0.02608761 1.04060594 0.60581237 1.42322428 1.04060594
 0.26377476 0.84349952 1.27829309 0.69856833 0.02608761 1.47539951
 0.75074356 1.33046832 0.02608761 0.40870595 1.18553713 2.87253617
 0.26377476 1.6203307  0.26377476 0.60581237 0.60581237 1.33046832
 0.26377476 3.59719212 0.1710188  0.46088118]
```

In [284…

```
np.where
```

Out[284…

```
<function where at 0x00000214EC9B1630>
```

In [285…

```
threshold = 3
np.where(z > threshold)
```

```
sorted_rscore =
sorted(df['Reading_Score']) q1 =
np.percentile(sorted_rscore, 25) q3 =
np.percentile(sorted_rscore, 75)
IQR = q3 - q1 lower =
q1 - (1.5 * IQR) upper
= q3 + (1.5 * IQR)

print("Lower Bound:", lower)
print("Upper Bound:", upper)
```

Out[285…

```
(array([97]),)
```

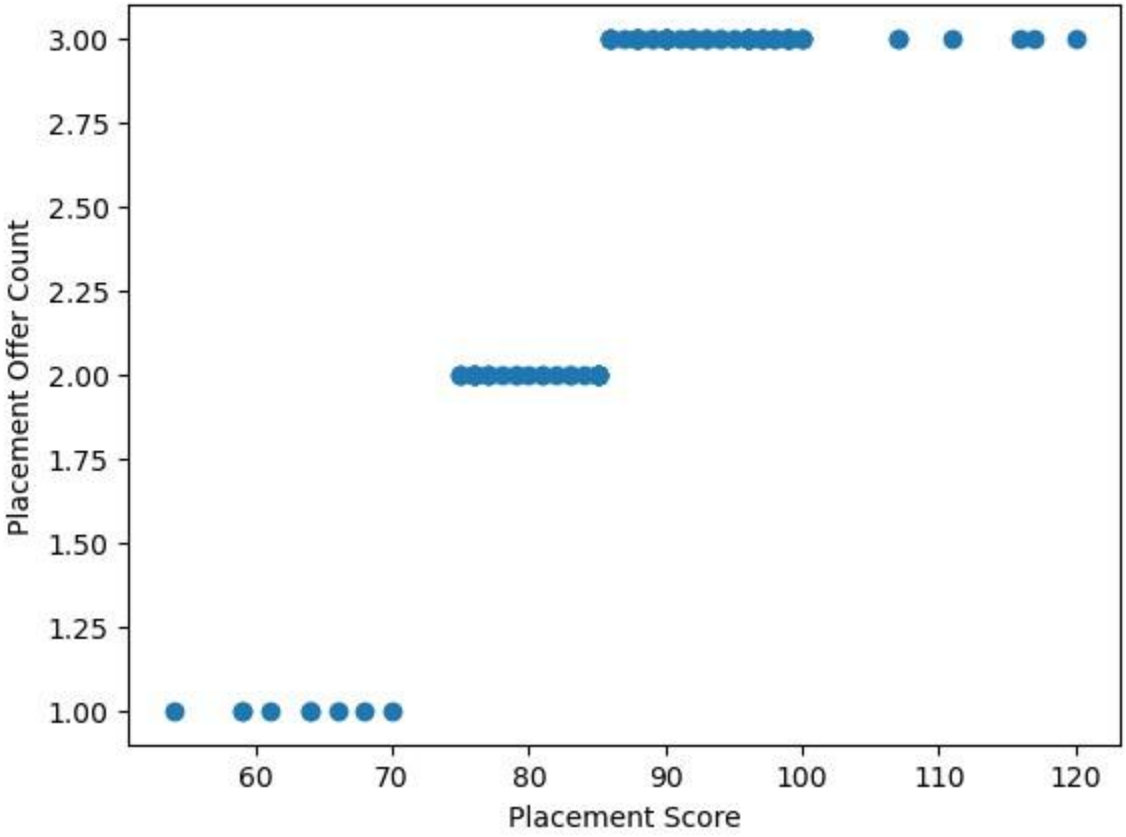
In [286…

```
Lower Bound: 73.0
Upper Bound: 97.0
```

```
In [287…
outliers = [] for i in
sorted_rscore:
    if i < lower or i > upper:
        outliers.append(i)
outliers
```

```
Out[287… [68.0]
```

```
In [288…
plt.scatter(df['Placement_Score'],
df['Placement_Offer_Count']) plt.xlabel("Placement Score")
plt.ylabel("Placement Offer Count") plt.show()
```



```
In
[289…3
print(np.where((df['Placement_Score'] > 85) & (df['Placement_Offer_Count']
<
```

```
(array([], dtype=int64),)
```

```
In [290…
Q1 = df['Reading_Score'].quantile(0.25)
Q3 = df['Reading_Score'].quantile(0.75)
IQR = Q3 - Q1 lower =
Q1 - 1.5 * IQR upper
= Q3 + 1.5 * IQR
```

In [291…

Out[291…

```
median = np.median(sorted_rscore)

df['Reading_Score'] = np.where(
    df['Reading_Score'] > upper,
    median,
    df['Reading_Score']
)

df['Reading_Score'] = np.where(
    df['Reading_Score'] < lower,
    median,

    df['Reading_Score']
)
df
```

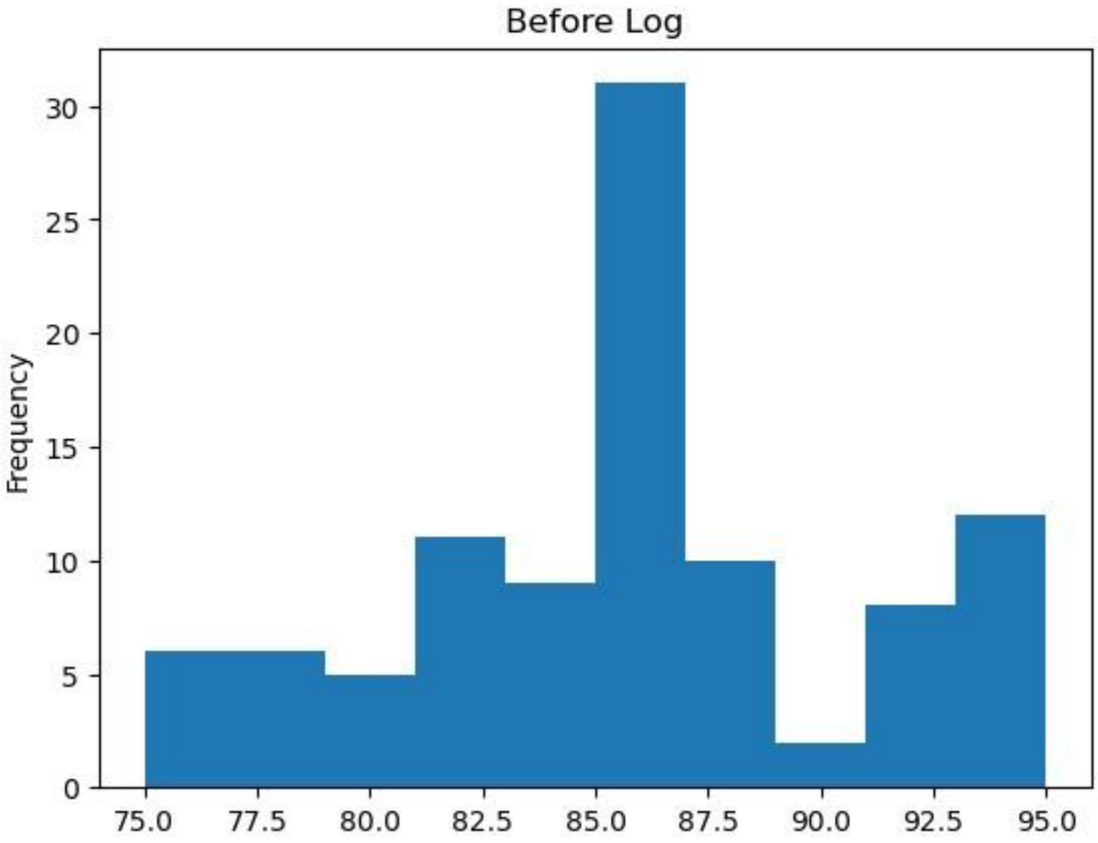
	Math_Score	Reading_Score	Writing_Score	Placement_Score	Club_Join_Date
0	64	85.0	70.0	88.000000	2020.0
1	68	94.0	62.0	79.000000	2020.0
2	72	88.0	77.0	90.000000	2018.0
3	77	80.0	63.0	99.000000	2020.0
4	80	93.0	60.0	96.000000	2020.0
...	...	...	...	...	
95	61	84.0	69.0	68.000000	2020.0
96	72	75.0	86.0	87.804124	2020.0
97	95	85.0	80.0	61.000000	2021.0
98	69	88.0	55.0	99.000000	2020.0
99	67	84.0	62.0	81.000000	2020.0

100 rows × 8 columns

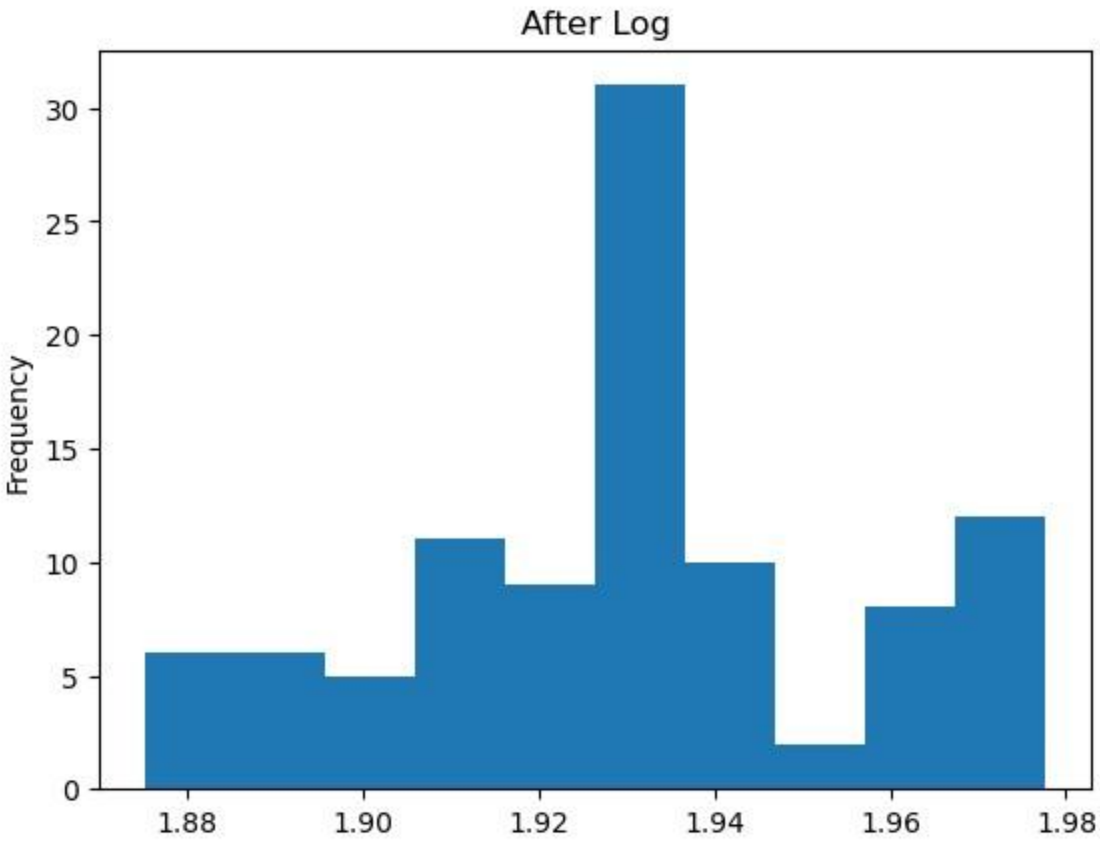
In [292…

```
df.loc[df['Placement_Score'] < 75, 'Placement_Offer_Count'] = 1

df.loc[
    (df['Placement_Score'] >= 75) & (df['Placement_Score'] <= 85),
    'Placement_Offer_Count'
] = 2

df.loc[df['Placement_Score'] > 85, 'Placement_Offer_Count'] = 3
```

```
In [295… df['Reading_Score'].plot(kind='hist') plt.title("Before
         Log")
plt.show()
```



```
In [294… df['log_reading'] = np.log10(df['Reading_Score'])

         df['log_reading'].plot(kind='hist')
         plt.title("After Log")
         plt.show()
```

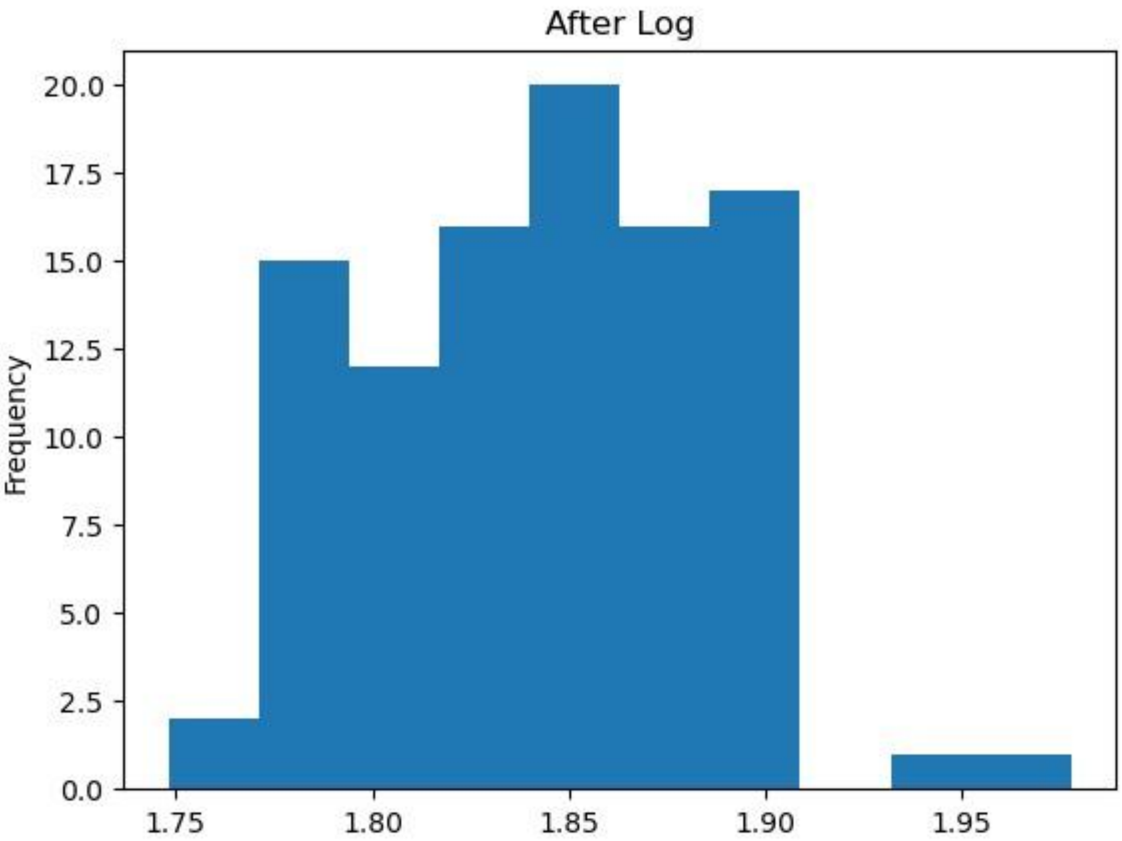


In [296…

```
df.loc[df['Placement_Score'] > 85, 'Placement_Offer_Count'] = 3
```

In [297…

```
df['log_math'] = np.log10(df['Math_Score'])

df['log_math'].plot(kind='hist')
plt.title("After Log")
plt.show()
```


After Log
Frequency
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
1.75
1.80
1.85
1.90
1.95

```
In [ ]:
```