ASSIGNMENT 2:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv("StudentsPerformance_20rows.csv")

df
```

```
    Math_Score  Reading_Score  Writing_Score  Placement_Score
Club_Join_Date  \
0           78           90.0             75             95.0
2019
1           62           84.0             62             88.0
2020
2           74           78.0             65             79.0
2018
3           69           87.0             72             92.0
2021
4           76           91.0             70             96.0
2019
5           65            NaN             68             85.0
2020
6           80           88.0             78            100.0
2021
7           64           95.0             77             81.0
2018
8           70           89.0             74              NaN
2019
9           77           84.0             79             94.0
2020
10          68           83.0             67             87.0
2021
11          75           86.0             70              NaN
2018
12          61           79.0             64             80.0
2019
13          79            NaN             77             98.0
2020
14          63           85.0             65             84.0
2021
15          72            NaN             76             91.0
2018
16          66           88.0             70             93.0
2019
17          73           87.0             74             89.0
2020
18          78           90.0             79             99.0
```

```
2021
19          60            81.0              63              82.0
2018

    Placement_Offer_Count
0                       3
1                       2
2                       1
3                       3
4                       4
5                       2
6                       5
7                       2
8                       3
9                       4
10                      2
11                      4
12                      1
13                      5
14                      2
15                      3
16                      3
17                      2
18                      5
19                      1

df.isnull()

    Math_Score  Reading_Score  Writing_Score  Placement_Score
Club_Join_Date  \
0        False          False          False            False
False
1        False          False          False            False
False
2        False          False          False            False
False
3        False          False          False            False
False
4        False          False          False            False
False
5        False           True          False            False
False
6        False          False          False            False
False
7        False          False          False            False
False
8        False          False          False             True
False
9        False          False          False            False
False
```

```
10       False          False          False             False
False
11       False          False          False              True
False
12       False          False          False             False
False
13       False           True          False             False
False
14       False          False          False             False
False
15       False           True          False             False
False
16       False          False          False             False
False
17       False          False          False             False
False
18       False          False          False             False
False
19       False          False          False             False
False
```

```
    Placement_Offer_Count
0                   False
1                   False
2                   False
3                   False
4                   False
5                   False
6                   False
7                   False
8                   False
9                   False
10                  False
11                  False
12                  False
13                  False
14                  False
15                  False
16                  False
17                  False
18                  False
19                  False
```

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

```
Math_Score               0
Reading_Score            3
Writing_Score            0
Placement_Score          2
Club_Join_Date           0
```

```
Placement_Offer_Count    0
dtype: int64
```

```
df.dropna(axis=1, how='any')
```

	Math_Score	Writing_Score	Club_Join_Date	Placement_Offer_Count
0	78	75	2019	3
1	62	62	2020	2
2	74	65	2018	1
3	69	72	2021	3
4	76	70	2019	4
5	65	68	2020	2
6	80	78	2021	5
7	64	77	2018	2
8	70	74	2019	3
9	77	79	2020	4
10	68	67	2021	2
11	75	70	2018	4
12	61	64	2019	1
13	79	77	2020	5
14	63	65	2021	2
15	72	76	2018	3
16	66	70	2019	3
17	73	74	2020	2
18	78	79	2021	5
19	60	63	2018	1

```
#handling null values
df['Reading_Score'].fillna(df['Reading_Score'].mean(), inplace=True)
```

```
C:\Users\omshi\AppData\Local\Temp\ipykernel_28468\1268933733.py:2:
FutureWarning: A value is trying to be set on a copy of a DataFrame or
Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never
work because the intermediate object on which we are setting values
always behaves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try
using 'df.method({col: value}, inplace=True)' or df[col] =
df[col].method(value) instead, to perform the operation inplace on the
original object.


  df['Reading_Score'].fillna(df['Reading_Score'].mean(), inplace=True)
```

```
df.notnull().sum()
```

```
Math_Score               20
Reading_Score            20
Writing_Score            20
Placement_Score          18
```

```
Club_Join_Date           20
Placement_Offer_Count    20
dtype: int64
```

```
from datetime import datetime
current_year = datetime.now().year
df['Duration'] = current_year - df['Club_Join_Date']
df = df.round({'Reading_Score':2,'Writing_Score':2})
df
```

```
    Math_Score  Reading_Score  Writing_Score  Placement_Score
Club_Join_Date  \
0           78          90.00             75             95.0
2019
1           62          84.00             62             88.0
2020
2           74          78.00             65             79.0
2018
3           69          87.00             72             92.0
2021
4           76          91.00             70             96.0
2019
5           65          86.18             68             85.0
2020
6           80          88.00             78            100.0
2021
7           64          95.00             77             81.0
2018
8           70          89.00             74              NaN
2019
9           77          84.00             79             94.0
2020
10          68          83.00             67             87.0
2021
11          75          86.00             70              NaN
2018
12          61          79.00             64             80.0
2019
13          79          86.18             77             98.0
2020
14          63          85.00             65             84.0
2021
15          72          86.18             76             91.0
2018
16          66          88.00             70             93.0
2019
17          73          87.00             74             89.0
2020
18          78          90.00             79             99.0
2021
```

```
19           60           81.00              63              82.0
2018

    Placement_Offer_Count  Duration
0                       3         6
1                       2         5
2                       1         7
3                       3         4
4                       4         6
5                       2         5
6                       5         4
7                       2         7
8                       3         6
9                       4         5
10                      2         4
11                      4         7
12                      1         6
13                      5         5
14                      2         4
15                      3         7
16                      3         6
17                      2         5
18                      5         4
19                      1         7
```

```
df['Placement_Score'].fillna(df['Placement_Score'].mean(),
inplace=True)
df
```

```
C:\Users\omshi\AppData\Local\Temp\ipykernel_28468\1579713478.py:1:
FutureWarning: A value is trying to be set on a copy of a DataFrame or
Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never
work because the intermediate object on which we are setting values
always behaves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try
using 'df.method({col: value}, inplace=True)' or df[col] =
df[col].method(value) instead, to perform the operation inplace on the
original object.


  df['Placement_Score'].fillna(df['Placement_Score'].mean(),
inplace=True)
```

```
    Math_Score  Reading_Score  Writing_Score  Placement_Score
Club_Join_Date  \
0           78          90.00             75        95.000000
2019
1           62          84.00             62        88.000000
```

```
2020
2           74           78.00              65         79.000000
2018
3           69           87.00              72         92.000000
2021
4           76           91.00              70         96.000000
2019
5           65           86.18              68         85.000000
2020
6           80           88.00              78        100.000000
2021
7           64           95.00              77         81.000000
2018
8           70           89.00              74         89.611111
2019
9           77           84.00              79         94.000000
2020
10          68           83.00              67         87.000000
2021
11          75           86.00              70         89.611111
2018
12          61           79.00              64         80.000000
2019
13          79           86.18              77         98.000000
2020
14          63           85.00              65         84.000000
2021
15          72           86.18              76         91.000000
2018
16          66           88.00              70         93.000000
2019
17          73           87.00              74         89.000000
2020
18          78           90.00              79         99.000000
2021
19          60           81.00              63         82.000000
2018
```

	Placement_Offer_Count	Duration
0	3	6
1	2	5
2	1	7
3	3	4
4	4	6
5	2	5
6	5	4
7	2	7
8	3	6
9	4	5

```
10                      2         4
11                      4         7
12                      1         6
13                      5         5
14                      2         4
15                      3         7
16                      3         6
17                      2         5
18                      5         4
19                      1         7
```

```
df.describe()
```

```
       Math_Score  Reading_Score  Writing_Score  Placement_Score  \
count    20.00000      20.000000      20.000000        20.000000
mean     70.50000      86.177000      71.250000        89.611111
std       6.56546       4.055345       5.627704         6.383687
min      60.00000      78.000000      62.000000        79.000000
25%      64.75000      84.000000      66.500000        84.750000
50%      71.00000      86.180000      71.000000        89.611111
75%      76.25000      88.250000      76.250000        94.250000
max      80.00000      95.000000      79.000000       100.000000

       Club_Join_Date  Placement_Offer_Count   Duration
count       20.000000              20.000000  20.000000
mean      2019.500000               2.850000   5.500000
std          1.147079               1.308877   1.147079
min       2018.000000               1.000000   4.000000
25%       2018.750000               2.000000   4.750000
50%       2019.500000               3.000000   5.500000
75%       2020.250000               4.000000   6.250000
max       2021.000000               5.000000   7.000000
```

```
# Applying data transformation on Placement_Offer_Count
df['Math Score'] = np.log10(df['Math_Score'])
df
```

```
    Math_Score  Reading_Score  Writing_Score  Placement_Score
Club_Join_Date  \
0           78          90.00             75        95.000000
2019
1           62          84.00             62        88.000000
2020
2           74          78.00             65        79.000000
2018
3           69          87.00             72        92.000000
2021
4           76          91.00             70        96.000000
2019
5           65          86.18             68        85.000000
```

```
2020
6            80          88.00              78       100.000000
2021
7            64          95.00              77        81.000000
2018
8            70          89.00              74        89.611111
2019
9            77          84.00              79        94.000000
2020
10           68          83.00              67        87.000000
2021
11           75          86.00              70        89.611111
2018
12           61          79.00              64        80.000000
2019
13           79          86.18              77        98.000000
2020
14           63          85.00              65        84.000000
2021
15           72          86.18              76        91.000000
2018
16           66          88.00              70        93.000000
2019
17           73          87.00              74        89.000000
2020
18           78          90.00              79        99.000000
2021
19           60          81.00              63        82.000000
2018
```

```
    Placement_Offer_Count  Duration  Math Score
0                0.094048         6    1.892095
1               -0.366513         5    1.792392
2                    -inf         7    1.869232
3                0.094048         4    1.838849
4                0.326634         6    1.880814
5               -0.366513         5    1.812913
6                0.475885         4    1.903090
7               -0.366513         7    1.806180
8                0.094048         6    1.845098
9                0.326634         5    1.886491
10              -0.366513         4    1.832509
11               0.326634         7    1.875061
12                   -inf         6    1.785330
13               0.475885         5    1.897627
14              -0.366513         4    1.799341
15               0.094048         7    1.857332
16               0.094048         6    1.819544
17              -0.366513         5    1.863323
```

```
18                0.475885          4    1.892095
19                    -inf          7    1.778151
```

```
#placing histogram
df['Math Score'].plot(kind= 'hist', edgecolor='white')
```

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