## **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
            78
                          90.0
                                            75
                                                            95.0
2019
                                            62
            62
                          84.0
                                                            88.0
2020
                                            65
            74
                          78.0
                                                            79.0
2018
                                            72
3
            69
                          87.0
                                                            92.0
2021
            76
                          91.0
                                            70
                                                            96.0
2019
                                            68
                                                            85.0
            65
                           NaN
2020
            80
                          88.0
                                            78
                                                           100.0
2021
                                            77
            64
                          95.0
                                                            81.0
2018
            70
                          89.0
                                            74
                                                             NaN
2019
            77
                                            79
                                                            94.0
9
                          84.0
2020
10
            68
                          83.0
                                            67
                                                            87.0
2021
            75
                          86.0
                                            70
                                                             NaN
11
2018
                                            64
12
            61
                          79.0
                                                            80.0
2019
            79
                                            77
                                                            98.0
13
                           NaN
2020
                          85.0
                                            65
14
            63
                                                            84.0
2021
15
            72
                                            76
                           NaN
                                                            91.0
2018
16
            66
                          88.0
                                            70
                                                            93.0
2019
17
            73
                          87.0
                                            74
                                                            89.0
2020
            78
                                            79
                          90.0
                                                            99.0
18
```

2021 19 2018	60	81.0	63	82.0	
Plac 0 1 2 3 4 5	ement_0	ffer_Count 3 2 1 3 4 2 5 2 3 4			
7 8 9 10 11 12 13 14 15 16 17 18		2 3 4 2 4 1 5 2 3 3 2 5 1			
df.isnul		Dooding Coope	Maiting Coope	Diagonant Cooks	
Club_Joi 0	_Score .n_Date False	\ False	False	Placement_Score 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	False	True	False	False	
6 False	False	False	False	False	
7 False	False	False	False	False	
8 False	False	False	False	True	
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				
Pla	cement_Offe	r Count		
	_	False		
0 1		False		
2		Falso		

	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')

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

19 2018	60	81.00		63	82.0
	+ Offer Com	at Durati	a m		
0 1 2 3 4 5 6 7	t_Offer_Cou	3 2 1 3 4 2 5	6 5 7 4 6 5 4 7		
8 9 10 11 12 13 14 15 16 17 18		3 4 2 4 1 5 2 3 3 2 5 1	6 5 4 7 6 5 4 7 6 5 4 7		
<pre>df['Placemen inplace=True df</pre>		illna(df['	Placemen <sup>-</sup>	t_Score'].	mean(),
FutureWarnin Series throu The behavior	g: A value : gh chained a will change the interme	is trying assignment e in panda ediate obje	to be se using an s 3.0. Th	t on a copy n inplace i his inplace	579713478.py:1: y of a DataFrame or method. e method will never e setting values
using 'df.me	thod({col: vod(value) i	value}, inp	place=Tr	ue)' or df	ace=True)', try [col] = tion inplace on the
<pre>df['Placem inplace=True</pre>	ent_Score'] )	.fillna(df	['Placeme	ent_Score'	].mean(),
		_Score Wr: 90.00	iting_Sco	ore Place 75	ment_Score 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	09	87.00		12	92.00000
4	76	91.00		70	96.000000
2019 5	65	86.18		68	85.000000
2020					
6 2021	80	88.00		78	100.000000
7	64	95.00		77	81.000000
2018					
8 2019	70	89.00		74	89.611111
9	77	84.00		79	94.000000
2020					
10 2021	68	83.00		67	87.000000
11	75	86.00		70	89.611111
2018					
12 2019	61	79.00		64	80.000000
13	79	86.18		77	98.000000
2020					
14 2021	63	85.00		65	84.000000
15	72	86.18		76	91.000000
2018	6.6	00.00		7.0	03.00000
16 2019	66	88.00		70	93.000000
17	73	87.00		74	89.000000
2020	70	00.00		70	00.00000
18 2021	78	90.00		79	99.000000
19	60	81.00		63	82.000000
2018					
Placem	ent_Offer_C	ount Durat	ion		
0 1		3	6		
1		2 1	5 7		
2		3	4		
4		4	6		
5 6 7		2 5	5 4 7		
		2			
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 17		3 2	6 5		
18		5	4		
19		3 1	7		
		_	•		
df.des	cribe()				
count	Math_Score 20.00000	Reading_Score 20.000000	Writing_Score 20.000000	Placement_Score 20.000000	\
mean std	70.50000 6.56546	86.177000 4.055345	71.250000 5.627704	89.611111 6.383687	
J - U	0.505.0	. 10555 15	31027701	31303007	

	Math_Score	Reading_Score	Writing_Score	Placement_Score	\
cou	nt 20.00000	$20.\overline{0}00000$	20.000000	$20.\overline{0}00000$	
mea	n 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	
mea std min 25% 50% 75%	n 70.50000 6.56546 60.00000 64.75000 71.00000 76.25000	86.177000 4.055345 78.000000 84.000000 86.180000 88.250000	71.250000 5.627704 62.000000 66.500000 71.000000 76.250000	89.611111 6.383687 79.000000 84.750000 89.611111 94.250000	

	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		_Score	Writing_Score	Placement_Score	
Club_Join_Date		00 00	7.5	05 000000	
0 78	8	90.00	75	95.000000	
2019					
1 6	2	84.00	62	88.00000	
2020					
2 7	4	78.00	65	79.000000	
2018					
3 6	9	87.00	72	92.000000	
2021					
4 7	6	91.00	70	96.000000	
2019					
5 6.	5	86.18	68	85.000000	

2020				
2020 6	80	88.00	78	100.000000
2021	6.4	05.00	77	01 000000
7 2018	64	95.00	77	81.000000
8	70	89.00	74	89.611111
2019 9	77	84.00	79	94.000000
2020	7.7	84.00	79	94.000000
10	68	83.00	67	87.000000
2021 11	75	86.00	70	89.611111
2018	73	00.00	70	09.011111
12	61	79.00	64	80.000000
2019 13	79	86.18	77	98.000000
2020	79	00.10	11	90.00000
14	63	85.00	65	84.000000
2021 15	72	86.18	76	91.000000
2018	12	00.10	70	91.000000
16	66	88.00	70	93.000000
2019 17	73	87.00	74	89.000000
2020	75	07.00	74	09.00000
18	78	90.00	79	99.000000
2021 19	60	81.00	63	82.000000
2018	00	81.00	03	02.00000
	D1	Carrate Danielia	Math. Carre	
0	Placement_Offer_ ი.ი	_	n Math Score 1.892095	
1 2		366513	5 1.792392	
			7 1.869232	
3 4			1.838849 1.880814	
			1.812913	
5 6 7	0.4	175885 <i>4</i>	1.903090	
			1.806180	
8 9		)94048 ( 326634 !	1.845098 1.886491	
10			1.832509	
11			7 1.875061	
12			1.785330	
13		175885	1.897627	
14			1.799341	
15			1.857332	
16			1.819544	
17	-0.3	366513	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'>
```

