```
Assignment No:02
```

```
Aim : 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.

Reason and document your approach properly.
```

In [1]:

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

In [2]:

```
data=pd.read_csv("Desktop\StudentPerformance.csv")
```

In [3]:

```
data
```

Out[3]:

	math_score	reading_score	writing_score	placement_score	club_join_year	placement_offer_c
0	74	67	80	88	2016	
1	77	74	66	84	2025	
2	66	68	63	79	2025	
3	80	78	69	79	2024	
4	62	79	69	82	2024	
5	65	75	62	89	2024	
6	63	79	68	71	2022	
7	72	72	64	65	2024	
8	77	73	72	99	2022	
9	67	71	64	76	2023	
10	66	70	60	63	2025	
11	77	74	62	66	2015	
12	60	80	67	97	2024	
13	75	61	63	68	2021	
14	78	78	69	85	2020	
15	66	77	68	60	2021	
16	76	64	69	71	2017	
17	71	73	79	72	2024	
18	67	80	80	64	2016	
19	66	72	69	95	2021	
20	72	74	69	81	2015	
21	79	69	74	68	2025	
22	70	71	70	80	2015	
23	60	61	63	98	2024	
24	71	65	66	79	2023	
25	70	69	68	75	2015	
26	73	62	63	94	2022	
27	70	65	71	71	2016	
28	74	72	74	83	2016	
29	67	72	72	82	2016	

In [4]:

```
data.isnull()
```

Out[4]:

	math_score	reading_score	writing_score	placement_score	club_join_year	placement_offer_c
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	False	False	False	False	
6	False	False	False	False	False	
7	False	False	False	False	False	
8	False	False	False	False	False	
9	False	False	False	False	False	
10	False	False	False	False	False	
11	False	False	False	False	False	
12	False	False	False	False	False	
13	False	False	False	False	False	
14	False	False	False	False	False	
15	False	False	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	
20	False	False	False	False	False	
21	False	False	False	False	False	
22	False	False	False	False	False	
23	False	False	False	False	False	
24	False	False	False	False	False	
25	False	False	False	False	False	
26	False	False	False	False	False	
27	False	False	False	False	False	
28	False	False	False	False	False	
29	False	False	False	False	False	

In [9]:

```python
series = pd.isnull(data['math_score '])
data[series]
```

Out[9]:

math_score	reading_score	writing_score	placement_score	club_join_year	placement_offer_cou

In [14]:

```python
print(data.columns)
```

```
Index(['math_score ', 'reading_score', 'writing_score', 'placement_score',
       'club_join_year', 'placement_offer_count'],
      dtype='object')
```

In [10]:

```
data.notnull()
```

Out[10]:

	math_score	reading_score	writing_score	placement_score	club_join_year	placement_offer_c
0	True	True	True	True	True	
1	True	True	True	True	True	
2	True	True	True	True	True	
3	True	True	True	True	True	
4	True	True	True	True	True	
5	True	True	True	True	True	
6	True	True	True	True	True	
7	True	True	True	True	True	
8	True	True	True	True	True	
9	True	True	True	True	True	
10	True	True	True	True	True	
11	True	True	True	True	True	
12	True	True	True	True	True	
13	True	True	True	True	True	
14	True	True	True	True	True	
15	True	True	True	True	True	
16	True	True	True	True	True	
17	True	True	True	True	True	
18	True	True	True	True	True	
19	True	True	True	True	True	
20	True	True	True	True	True	
21	True	True	True	True	True	
22	True	True	True	True	True	
23	True	True	True	True	True	
24	True	True	True	True	True	
25	True	True	True	True	True	
26	True	True	True	True	True	
27	True	True	True	True	True	
28	True	True	True	True	True	
29	True	True	True	True	True	

In [11]:

```python
series1 = pd.notnull(data['math_score '])
data[series1]

```

Out[11]:

	math_score	reading_score	writing_score	placement_score	club_join_year	placement_offer_c
0	74	67	80	88	2016	
1	77	74	66	84	2025	
2	66	68	63	79	2025	
3	80	78	69	79	2024	
4	62	79	69	82	2024	
5	65	75	62	89	2024	
6	63	79	68	71	2022	
7	72	72	64	65	2024	
8	77	73	72	99	2022	
9	67	71	64	76	2023	
10	66	70	60	63	2025	
11	77	74	62	66	2015	
12	60	80	67	97	2024	
13	75	61	63	68	2021	
14	78	78	69	85	2020	
15	66	77	68	60	2021	
16	76	64	69	71	2017	
17	71	73	79	72	2024	
18	67	80	80	64	2016	
19	66	72	69	95	2021	
20	72	74	69	81	2015	
21	79	69	74	68	2025	
22	70	71	70	80	2015	
23	60	61	63	98	2024	
24	71	65	66	79	2023	
25	70	69	68	75	2015	
26	73	62	63	94	2022	
27	70	65	71	71	2016	
28	74	72	74	83	2016	
29	67	72	72	82	2016	

In [16]:

```
print(data.columns)
```

```
Index(['math_score ', 'reading_score', 'writing_score', 'placement_score',
       'club_join_year', 'placement_offer_count'],
      dtype='object')
```

In [6]:

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

In [7]:

```
data=pd.read_csv("Desktop\StudentPerformance.csv")
```

In [8]:

```
data
```

Out[8]:

	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placemer
0	female	74.0	67.0	80.0	88.0	NaN	
1	male	77.0	74.0	66.0	84.0	2025.0	
2	male	66.0	68.0	63.0	79.0	2025.0	
3	female	80.0	78.0	69.0	79.0	2024.0	
4	male	62.0	79.0	69.0	82.0	2024.0	
5	female	65.0	75.0	NaN	89.0	2024.0	
6	male	NaN	79.0	68.0	71.0	2022.0	
7	female	72.0	72.0	64.0	65.0	2024.0	
8	female	77.0	73.0	72.0	99.0	2022.0	
9	male	67.0	71.0	64.0	NaN	2023.0	
10	male	66.0	70.0	60.0	63.0	2025.0	
11	male	77.0	74.0	62.0	66.0	2015.0	
12	female	60.0	80.0	67.0	97.0	2024.0	
13	male	75.0	NaN	63.0	68.0	2021.0	
14	male	78.0	78.0	69.0	85.0	2020.0	
15	female	66.0	77.0	68.0	NaN	2021.0	
16	female	76.0	64.0	69.0	71.0	2017.0	
17	male	71.0	73.0	79.0	72.0	2024.0	
18	female	67.0	80.0	80.0	64.0	2016.0	
19	male	66.0	72.0	69.0	95.0	2021.0	
20	male	72.0	74.0	69.0	81.0	2015.0	
21	female	79.0	69.0	74.0	68.0	2025.0	
22	male	70.0	71.0	70.0	80.0	2015.0	
23	male	60.0	61.0	63.0	NaN	2024.0	
24	male	71.0	65.0	66.0	79.0	2023.0	
25	female	70.0	69.0	68.0	75.0	2015.0	
26	male	73.0	62.0	63.0	94.0	2022.0	
27	male	70.0	65.0	71.0	71.0	2016.0	
28	male	74.0	72.0	74.0	83.0	2016.0	
29	female	67.0	72.0	72.0	82.0	2016.0	

In [9]:

```python
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
data['gender'] = le.fit_transform(data['gender'])
newdata=data
data
```

Out[9]:

	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placemer
0	0	74.0	67.0	80.0	88.0	NaN	
1	1	77.0	74.0	66.0	84.0	2025.0	
2	1	66.0	68.0	63.0	79.0	2025.0	
3	0	80.0	78.0	69.0	79.0	2024.0	
4	1	62.0	79.0	69.0	82.0	2024.0	
5	0	65.0	75.0	NaN	89.0	2024.0	
6	1	NaN	79.0	68.0	71.0	2022.0	
7	0	72.0	72.0	64.0	65.0	2024.0	
8	0	77.0	73.0	72.0	99.0	2022.0	
9	1	67.0	71.0	64.0	NaN	2023.0	
10	1	66.0	70.0	60.0	63.0	2025.0	
11	1	77.0	74.0	62.0	66.0	2015.0	
12	0	60.0	80.0	67.0	97.0	2024.0	
13	1	75.0	NaN	63.0	68.0	2021.0	
14	1	78.0	78.0	69.0	85.0	2020.0	
15	0	66.0	77.0	68.0	NaN	2021.0	
16	0	76.0	64.0	69.0	71.0	2017.0	
17	1	71.0	73.0	79.0	72.0	2024.0	
18	0	67.0	80.0	80.0	64.0	2016.0	
19	1	66.0	72.0	69.0	95.0	2021.0	
20	1	72.0	74.0	69.0	81.0	2015.0	
21	0	79.0	69.0	74.0	68.0	2025.0	
22	1	70.0	71.0	70.0	80.0	2015.0	
23	1	60.0	61.0	63.0	NaN	2024.0	
24	1	71.0	65.0	66.0	79.0	2023.0	
25	0	70.0	69.0	68.0	75.0	2015.0	
26	1	73.0	62.0	63.0	94.0	2022.0	
27	1	70.0	65.0	71.0	71.0	2016.0	
28	1	74.0	72.0	74.0	83.0	2016.0	
29	0	67.0	72.0	72.0	82.0	2016.0	

In [10]:

```python
series = pd.isnull(data["math_score "])
data[series]
```

Out[10]:

	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placement
6	1	NaN	79.0	68.0	71.0	2022.0	

In [11]:

```python
series = pd.isnull(data["placement_score"])
data[series]
```

Out[11]:

	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placemer
9	1	67.0	71.0	64.0	NaN	2023.0	
15	0	66.0	77.0	68.0	NaN	2021.0	
23	1	60.0	61.0	63.0	NaN	2024.0	

In [12]:

```
data.notnull()
```

Out[12]:

	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placemer
0	True	True	True	True	True	False	
1	True	True	True	True	True	True	
2	True	True	True	True	True	True	
3	True	True	True	True	True	True	
4	True	True	True	True	True	True	
5	True	True	True	False	True	True	
6	True	False	True	True	True	True	
7	True	True	True	True	True	True	
8	True	True	True	True	True	True	
9	True	True	True	True	False	True	
10	True	True	True	True	True	True	
11	True	True	True	True	True	True	
12	True	True	True	True	True	True	
13	True	True	False	True	True	True	
14	True	True	True	True	True	True	
15	True	True	True	True	False	True	
16	True	True	True	True	True	True	
17	True	True	True	True	True	True	
18	True	True	True	True	True	True	
19	True	True	True	True	True	True	
20	True	True	True	True	True	True	
21	True	True	True	True	True	True	
22	True	True	True	True	True	True	
23	True	True	True	True	False	True	
24	True	True	True	True	True	True	
25	True	True	True	True	True	True	
26	True	True	True	True	True	True	
27	True	True	True	True	True	True	
28	True	True	True	True	True	True	
29	True	True	True	True	True	True	

In [32]:

```
series = pd.notnull(data["math_score "])
data[series]
```

Out[32]:

	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placemer
0	female	74.0	67.0	80.0	88.0	NaN	
1	male	77.0	74.0	66.0	84.0	2025.0	
2	male	66.0	68.0	63.0	79.0	2025.0	
3	female	80.0	78.0	69.0	79.0	2024.0	
4	male	62.0	79.0	69.0	82.0	2024.0	
5	female	65.0	75.0	NaN	89.0	2024.0	
7	female	72.0	72.0	64.0	65.0	2024.0	
8	female	77.0	73.0	72.0	99.0	2022.0	
9	male	67.0	71.0	64.0	NaN	2023.0	
10	male	66.0	70.0	60.0	63.0	2025.0	
11	male	77.0	74.0	62.0	66.0	2015.0	
12	female	60.0	80.0	67.0	97.0	2024.0	
13	male	75.0	NaN	63.0	68.0	2021.0	
14	male	78.0	78.0	69.0	85.0	2020.0	
15	female	66.0	77.0	68.0	NaN	2021.0	
16	female	76.0	64.0	69.0	71.0	2017.0	
17	male	71.0	73.0	79.0	72.0	2024.0	
18	female	67.0	80.0	80.0	64.0	2016.0	
19	male	66.0	72.0	69.0	95.0	2021.0	
20	male	72.0	74.0	69.0	81.0	2015.0	
21	female	79.0	69.0	74.0	68.0	2025.0	
22	male	70.0	71.0	70.0	80.0	2015.0	
23	male	60.0	61.0	63.0	NaN	2024.0	
24	male	71.0	65.0	66.0	79.0	2023.0	
25	female	70.0	69.0	68.0	75.0	2015.0	
26	male	73.0	62.0	63.0	94.0	2022.0	
27	male	70.0	65.0	71.0	71.0	2016.0	
28	male	74.0	72.0	74.0	83.0	2016.0	
29	female	67.0	72.0	72.0	82.0	2016.0	

In [13]:

```python
missing_values = ["Na", "na"]
data= pd.read_csv("Desktop\StudentPerformance.csv", na_values =
missing_values)
data
```

Out[13]:

	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placemer
0	female	74.0	67.0	80.0	88.0	NaN	
1	male	77.0	74.0	66.0	84.0	2025.0	
2	male	66.0	68.0	63.0	79.0	2025.0	
3	female	80.0	78.0	69.0	79.0	2024.0	
4	male	62.0	79.0	69.0	82.0	2024.0	
5	female	65.0	75.0	NaN	89.0	2024.0	
6	male	NaN	79.0	68.0	71.0	2022.0	
7	female	72.0	72.0	64.0	65.0	2024.0	
8	female	77.0	73.0	72.0	99.0	2022.0	
9	male	67.0	71.0	64.0	NaN	2023.0	
10	male	66.0	70.0	60.0	63.0	2025.0	
11	male	77.0	74.0	62.0	66.0	2015.0	
12	female	60.0	80.0	67.0	97.0	2024.0	
13	male	75.0	NaN	63.0	68.0	2021.0	
14	male	78.0	78.0	69.0	85.0	2020.0	
15	female	66.0	77.0	68.0	NaN	2021.0	
16	female	76.0	64.0	69.0	71.0	2017.0	
17	male	71.0	73.0	79.0	72.0	2024.0	
18	female	67.0	80.0	80.0	64.0	2016.0	
19	male	66.0	72.0	69.0	95.0	2021.0	
20	male	72.0	74.0	69.0	81.0	2015.0	
21	female	79.0	69.0	74.0	68.0	2025.0	
22	male	70.0	71.0	70.0	80.0	2015.0	
23	male	60.0	61.0	63.0	NaN	2024.0	
24	male	71.0	65.0	66.0	79.0	2023.0	
25	female	70.0	69.0	68.0	75.0	2015.0	
26	male	73.0	62.0	63.0	94.0	2022.0	
27	male	70.0	65.0	71.0	71.0	2016.0	
28	male	74.0	72.0	74.0	83.0	2016.0	
29	female	67.0	72.0	72.0	82.0	2016.0	

In [14]:

```python
ndf=data
ndf.fillna(1)
```

Out[14]:

	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placemer
0	female	74.0	67.0	80.0	88.0	1.0	
1	male	77.0	74.0	66.0	84.0	2025.0	
2	male	66.0	68.0	63.0	79.0	2025.0	
3	female	80.0	78.0	69.0	79.0	2024.0	
4	male	62.0	79.0	69.0	82.0	2024.0	
5	female	65.0	75.0	1.0	89.0	2024.0	
6	male	1.0	79.0	68.0	71.0	2022.0	
7	female	72.0	72.0	64.0	65.0	2024.0	
8	female	77.0	73.0	72.0	99.0	2022.0	
9	male	67.0	71.0	64.0	1.0	2023.0	
10	male	66.0	70.0	60.0	63.0	2025.0	
11	male	77.0	74.0	62.0	66.0	2015.0	
12	female	60.0	80.0	67.0	97.0	2024.0	
13	male	75.0	1.0	63.0	68.0	2021.0	
14	male	78.0	78.0	69.0	85.0	2020.0	
15	female	66.0	77.0	68.0	1.0	2021.0	
16	female	76.0	64.0	69.0	71.0	2017.0	
17	male	71.0	73.0	79.0	72.0	2024.0	
18	female	67.0	80.0	80.0	64.0	2016.0	
19	male	66.0	72.0	69.0	95.0	2021.0	
20	male	72.0	74.0	69.0	81.0	2015.0	
21	female	79.0	69.0	74.0	68.0	2025.0	
22	male	70.0	71.0	70.0	80.0	2015.0	
23	male	60.0	61.0	63.0	1.0	2024.0	
24	male	71.0	65.0	66.0	79.0	2023.0	
25	female	70.0	69.0	68.0	75.0	2015.0	
26	male	73.0	62.0	63.0	94.0	2022.0	
27	male	70.0	65.0	71.0	71.0	2016.0	
28	male	74.0	72.0	74.0	83.0	2016.0	
29	female	67.0	72.0	72.0	82.0	2016.0	

In [15]:

```python
m_v=data['math_score '].mean()
data['math_score '].fillna(value=m_v, inplace=True)
data
```

Out[15]:

	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placemer
0	female	74.00000	67.0	80.0	88.0	NaN	
1	male	77.00000	74.0	66.0	84.0	2025.0	
2	male	66.00000	68.0	63.0	79.0	2025.0	
3	female	80.00000	78.0	69.0	79.0	2024.0	
4	male	62.00000	79.0	69.0	82.0	2024.0	
5	female	65.00000	75.0	NaN	89.0	2024.0	
6	male	70.62069	79.0	68.0	71.0	2022.0	
7	female	72.00000	72.0	64.0	65.0	2024.0	
8	female	77.00000	73.0	72.0	99.0	2022.0	
9	male	67.00000	71.0	64.0	NaN	2023.0	
10	male	66.00000	70.0	60.0	63.0	2025.0	
11	male	77.00000	74.0	62.0	66.0	2015.0	
12	female	60.00000	80.0	67.0	97.0	2024.0	
13	male	75.00000	NaN	63.0	68.0	2021.0	
14	male	78.00000	78.0	69.0	85.0	2020.0	
15	female	66.00000	77.0	68.0	NaN	2021.0	
16	female	76.00000	64.0	69.0	71.0	2017.0	
17	male	71.00000	73.0	79.0	72.0	2024.0	
18	female	67.00000	80.0	80.0	64.0	2016.0	
19	male	66.00000	72.0	69.0	95.0	2021.0	
20	male	72.00000	74.0	69.0	81.0	2015.0	
21	female	79.00000	69.0	74.0	68.0	2025.0	
22	male	70.00000	71.0	70.0	80.0	2015.0	
23	male	60.00000	61.0	63.0	NaN	2024.0	
24	male	71.00000	65.0	66.0	79.0	2023.0	
25	female	70.00000	69.0	68.0	75.0	2015.0	
26	male	73.00000	62.0	63.0	94.0	2022.0	
27	male	70.00000	65.0	71.0	71.0	2016.0	
28	male	74.00000	72.0	74.0	83.0	2016.0	
29	female	67.00000	72.0	72.0	82.0	2016.0	

In [16]:

```
ndf.replace(to_replace = np.nan, value = -99)
```

Out[16]:

	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placemer
0	female	74.00000	67.0	80.0	88.0	-99.0	
1	male	77.00000	74.0	66.0	84.0	2025.0	
2	male	66.00000	68.0	63.0	79.0	2025.0	
3	female	80.00000	78.0	69.0	79.0	2024.0	
4	male	62.00000	79.0	69.0	82.0	2024.0	
5	female	65.00000	75.0	-99.0	89.0	2024.0	
6	male	70.62069	79.0	68.0	71.0	2022.0	
7	female	72.00000	72.0	64.0	65.0	2024.0	
8	female	77.00000	73.0	72.0	99.0	2022.0	
9	male	67.00000	71.0	64.0	-99.0	2023.0	
10	male	66.00000	70.0	60.0	63.0	2025.0	
11	male	77.00000	74.0	62.0	66.0	2015.0	
12	female	60.00000	80.0	67.0	97.0	2024.0	
13	male	75.00000	-99.0	63.0	68.0	2021.0	
14	male	78.00000	78.0	69.0	85.0	2020.0	
15	female	66.00000	77.0	68.0	-99.0	2021.0	
16	female	76.00000	64.0	69.0	71.0	2017.0	
17	male	71.00000	73.0	79.0	72.0	2024.0	
18	female	67.00000	80.0	80.0	64.0	2016.0	
19	male	66.00000	72.0	69.0	95.0	2021.0	
20	male	72.00000	74.0	69.0	81.0	2015.0	
21	female	79.00000	69.0	74.0	68.0	2025.0	
22	male	70.00000	71.0	70.0	80.0	2015.0	
23	male	60.00000	61.0	63.0	-99.0	2024.0	
24	male	71.00000	65.0	66.0	79.0	2023.0	
25	female	70.00000	69.0	68.0	75.0	2015.0	
26	male	73.00000	62.0	63.0	94.0	2022.0	
27	male	70.00000	65.0	71.0	71.0	2016.0	
28	male	74.00000	72.0	74.0	83.0	2016.0	
29	female	67.00000	72.0	72.0	82.0	2016.0	

In [17]:

```
ndf.dropna()
```

Out[17]:

	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placemer
1	male	77.0	74.0	66.0	84.0	2025.0	
3	female	80.0	78.0	69.0	79.0	2024.0	
7	female	72.0	72.0	64.0	65.0	2024.0	
14	male	78.0	78.0	69.0	85.0	2020.0	
17	male	71.0	73.0	79.0	72.0	2024.0	
19	male	66.0	72.0	69.0	95.0	2021.0	
22	male	70.0	71.0	70.0	80.0	2015.0	
26	male	73.0	62.0	63.0	94.0	2022.0	
27	male	70.0	65.0	71.0	71.0	2016.0	
29	female	67.0	72.0	72.0	82.0	2016.0	

In [18]:

```
ndf.dropna(how = 'all')
```

Out[18]:

	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placemer
0	female	74.00000	67.0	80.0	88.0	NaN	
1	male	77.00000	74.0	66.0	84.0	2025.0	
2	male	66.00000	68.0	63.0	79.0	2025.0	
3	female	80.00000	78.0	69.0	79.0	2024.0	
4	male	62.00000	79.0	69.0	82.0	2024.0	
5	female	65.00000	75.0	NaN	89.0	2024.0	
6	male	70.62069	79.0	68.0	71.0	2022.0	
7	female	72.00000	72.0	64.0	65.0	2024.0	
8	female	77.00000	73.0	72.0	99.0	2022.0	
9	male	67.00000	71.0	64.0	NaN	2023.0	
10	male	66.00000	70.0	60.0	63.0	2025.0	
11	male	77.00000	74.0	62.0	66.0	2015.0	
12	female	60.00000	80.0	67.0	97.0	2024.0	
13	male	75.00000	NaN	63.0	68.0	2021.0	
14	male	78.00000	78.0	69.0	85.0	2020.0	
15	female	66.00000	77.0	68.0	NaN	2021.0	
16	female	76.00000	64.0	69.0	71.0	2017.0	
17	male	71.00000	73.0	79.0	72.0	2024.0	
18	female	67.00000	80.0	80.0	64.0	2016.0	
19	male	66.00000	72.0	69.0	95.0	2021.0	
20	male	72.00000	74.0	69.0	81.0	2015.0	
21	female	79.00000	69.0	74.0	68.0	2025.0	
22	male	70.00000	71.0	70.0	80.0	2015.0	
23	male	60.00000	61.0	63.0	NaN	2024.0	
24	male	71.00000	65.0	66.0	79.0	2023.0	
25	female	70.00000	69.0	68.0	75.0	2015.0	
26	male	73.00000	62.0	63.0	94.0	2022.0	
27	male	70.00000	65.0	71.0	71.0	2016.0	
28	male	74.00000	72.0	74.0	83.0	2016.0	
29	female	67.00000	72.0	72.0	82.0	2016.0	

In [19]:

```
ndf.dropna(axis = 1)
```

Out[19]:

	gender	math_score	placement_offer_count
0	female	74.00000	3
1	male	77.00000	2
2	male	66.00000	2
3	female	80.00000	2
4	male	62.00000	2
5	female	65.00000	3
6	male	70.62069	1
7	female	72.00000	1
8	female	77.00000	3
9	male	67.00000	2
10	male	66.00000	1
11	male	77.00000	1
12	female	60.00000	3
13	male	75.00000	1
14	male	78.00000	3
15	female	66.00000	1
16	female	76.00000	1
17	male	71.00000	1
18	female	67.00000	1
19	male	66.00000	3
20	male	72.00000	2
21	female	79.00000	1
22	male	70.00000	2
23	male	60.00000	3
24	male	71.00000	2
25	female	70.00000	2
26	male	73.00000	3
27	male	70.00000	1
28	male	74.00000	2
29	female	67.00000	2

In [20]:

```
new_data = ndf.dropna(axis = 0, how ='any')
new_data
```

Out[20]:

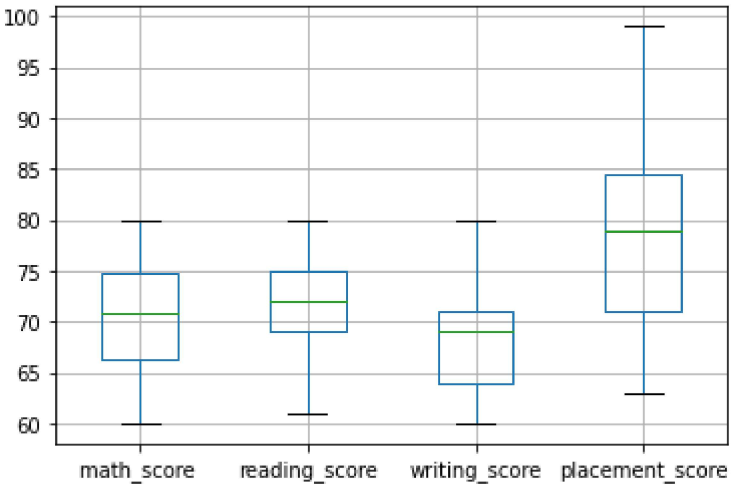
	gender	math_score	reading_score	writing_score	placement_score	club_join_year	placemer
1	male	77.0	74.0	66.0	84.0	2025.0	
3	female	80.0	78.0	69.0	79.0	2024.0	
7	female	72.0	72.0	64.0	65.0	2024.0	
14	male	78.0	78.0	69.0	85.0	2020.0	
17	male	71.0	73.0	79.0	72.0	2024.0	
19	male	66.0	72.0	69.0	95.0	2021.0	
22	male	70.0	71.0	70.0	80.0	2015.0	
26	male	73.0	62.0	63.0	94.0	2022.0	
27	male	70.0	65.0	71.0	71.0	2016.0	
29	female	67.0	72.0	72.0	82.0	2016.0	

In [21]:

```
col = ['math_score ', 'reading_score' , 'writing_score','placement_score']
data.boxplot(col)
```

Out[21]:

```
<AxesSubplot:>
```



In [22]:

```
print(np.where(data['math_score ']>90))
```

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

In [23]:

```
print(np.where(data['reading_score']<25))
```

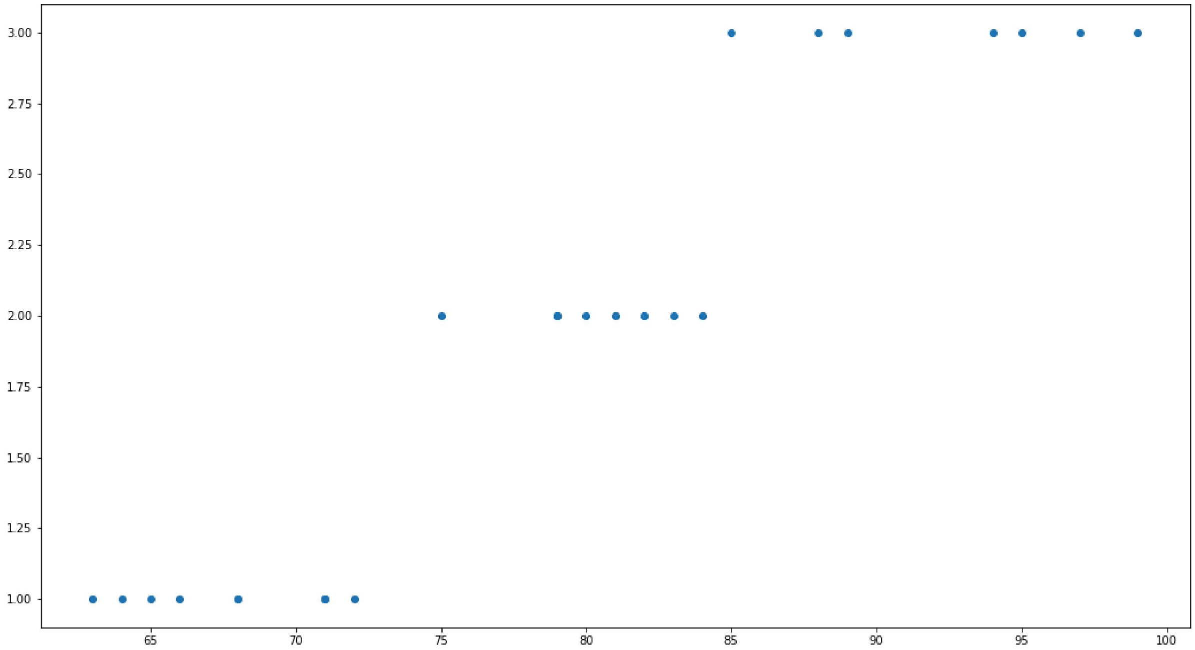
```
(array([], dtype=int64),)
```

In [24]:

```
import matplotlib.pyplot as plt
```

In [25]:

```python
fig, ax = plt.subplots(figsize = (18,10))
ax.scatter(data['placement_score'], data['placement_offer_count'])
plt.show()
ax.set_xlabel('(Proportion non-retail business acres)/(town)')
ax.set_ylabel('(Full-value property-tax rate)/( $10,000)')
```



Out[25]:

```
Text(3.200000000000017, 0.5, '(Full-value property-tax rate)/( $10,000)')
```

In [26]:

```python
print(np.where((data['placement_score']<50) & (data['placement_offer_count
print(np.where((data['placement_score']>85) & (data['placement_offer_count
```

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

In [27]:

```python
import numpy as np
from scipy import stats
```

In [28]:

```python
z = np.abs(stats.zscore(data['math_score ']))
```

In [29]:

```
print(z)
```

```
0     0.626505
1     1.182688
2     0.856650
3     1.738871
4     1.598227
5     1.042044
6     0.000000
7     0.255716
8     1.182688
9     0.671255
10    0.856650
11    1.182688
12    1.969015
13    0.811899
14    1.368082
15    0.856650
16    0.997294
17    0.070322
18    0.671255
19    0.856650
20    0.255716
21    1.553476
22    0.115072
23    1.969015
24    0.070322
25    0.115072
26    0.441111
27    0.115072
28    0.626505
29    0.671255
Name: math_score , dtype: float64
```

In [30]:

```
threshold = 0.18
```

In [31]:

```
sample_outliers = np.where(z <threshold)
sample_outliers
```

Out[31]:

```
(array([ 6, 17, 22, 24, 25, 27], dtype=int64),)
```

In [32]:

```
sorted_rscore= sorted(data['reading_score'])
```

In [33]:

```
sorted_rscore
```

Out[33]:

```
[61.0,
 62.0,
 64.0,
 65.0,
 65.0,
 67.0,
 68.0,
 69.0,
 69.0,
 70.0,
 71.0,
 71.0,
 72.0,
 72.0,
 72.0,
 72.0,
 73.0,
 73.0,
 74.0,
 74.0,
 74.0,
 75.0,
 77.0,
 78.0,
 78.0,
 79.0,
 79.0,
 80.0,
 nan,
 80.0]
```

In [34]:

```
q1 = np.percentile(sorted_rscore, 25)
q3 = np.percentile(sorted_rscore, 75)
print(q1,q3)
```

```
nan nan
```

In [35]:

```
IQR = q3-q1
```

In [36]:

```
lwr_bound = q1-(1.5*IQR)
upr_bound = q3+(1.5*IQR)
print(lwr_bound, upr_bound)
```

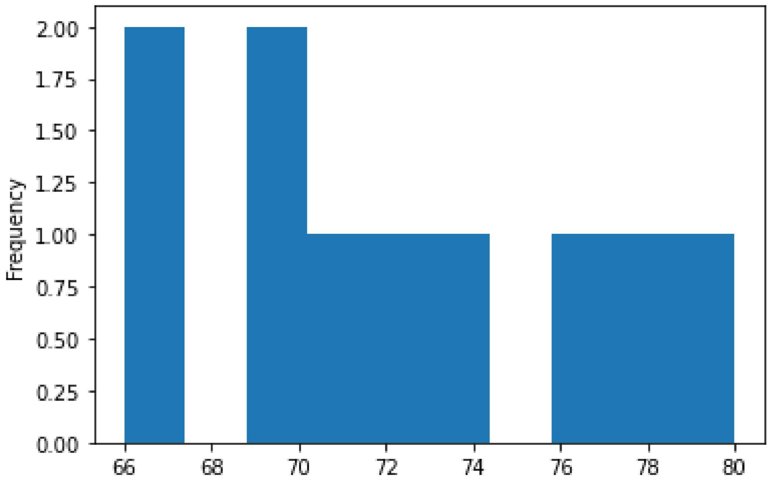
```
nan nan
```

In [39]:

```python
import matplotlib.pyplot as plt
new_data['math_score '].plot(kind = 'hist')
```

Out[39]:

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



In [40]:

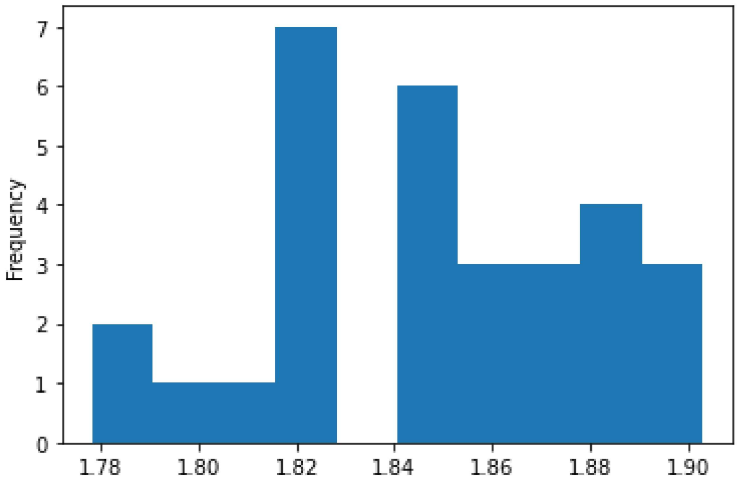
```python
data['log_math'] = np.log10(data['math_score '])
```

In [41]:

```python
data['log_math'].plot(kind = 'hist')
```

Out[41]:

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



Name:Sneha Navgire
Roll no :13246