```
import pandas as pd
import numpy as np

missing_values = ["Na", "na"]

df =
pd.read_excel("/home/ubuntu/TCOB35/Student_Performance.xlsx",na_values
= missing_values)
df
```

```
    math_score  reading_score  writing_score  placement_score
club_join_year  \
0           77           73.0           69.0             65.0
2020
1           75           70.0           58.0             94.0
2018
2           75           63.0           61.0             69.0
2019
3           69           70.0           71.0             73.0
2019
4           97           50.0           64.0             77.0
2018
5           78           75.0           62.0             78.0
2019
6           77           71.0           74.0             66.0
2019
7           66           70.0            NaN             77.0
2019
8           70           67.0           69.0             62.0
2019
9           55           20.0           74.0             75.0
2021
10          73           77.0           77.0             60.0
2018
11          64           60.0           73.0             75.0
2021
12          73           60.0           73.0             95.0
2021
13          72           71.0           61.0             80.0
2019
14          74           69.0           68.0             70.0
2021
15          73           63.0           67.0              NaN
2018
16          65           86.0           61.0             77.0
2019
17          87           61.0           90.0             77.0
2018
18          71           69.0           78.0              NaN
2019
```

```
19           67            77.0            69.0            79.0
2020
20           62             NaN            78.0            75.0
2021
21           67            68.0            64.0            64.0
2018
22           66            69.0            72.0            79.0
2021
23           63            62.0            61.0            77.0
2019
24           67            76.0            70.0            65.0
2018
25           55            74.0            66.0            77.0
2020
26           96            60.0            20.0            69.0
2021
27           75            79.0            73.0            80.0
2019
28           70            79.0            57.0            63.0
2019
29           70            74.0            75.0            69.0
2020
```

```
    Placement-offer-count
0                       2
1                       0
2                       2
3                       2
4                       0
5                       0
6                       2
7                       0
8                       2
9                       0
10                      2
11                      0
12                      0
13                      0
14                      2
15                      0
16                      0
17                      0
18                      2
19                      0
20                      0
21                      2
22                      0
23                      0
24                      2
```

```
25                       0
26                       2
27                       0
28                       2
29                       2

series = pd.isnull(df["placement_score"])
df[series]

    math_score  reading_score  writing_score  placement_score
club_join_year  \
15          73           63.0           67.0              NaN
2018
18          71           69.0           78.0              NaN
2019

    Placement-offer-count
15                      0
18                      2

df.notnull()

    math_score  reading_score  writing_score  placement_score
club_join_year  \
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          False             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             False
True
16         True             True             True              True
True
17         True             True             True              True
True
18         True             True             True             False
True
19         True             True             True              True
True
20         True            False             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

    Placement-offer-count
0                    True
1                    True
2                    True
3                    True
4                    True
5                    True
6                    True
7                    True
8                    True
9                    True
10                   True
11                   True
12                   True
```

```
13                    True
14                    True
15                    True
16                    True
17                    True
18                    True
19                    True
20                    True
21                    True
22                    True
23                    True
24                    True
25                    True
26                    True
27                    True
28                    True
29                    True
```

```
series1 = pd.notnull(df["math_score"])
df[series1]
```

```
    math_score  reading_score  writing_score  placement_score
club_join_year  \
0           77           73.0           69.0             65.0
2020
1           75           70.0           58.0             94.0
2018
2           75           63.0           61.0             69.0
2019
3           69           70.0           71.0             73.0
2019
4           97           50.0           64.0             77.0
2018
5           78           75.0           62.0             78.0
2019
6           77           71.0           74.0             66.0
2019
7           66           70.0            NaN             77.0
2019
8           70           67.0           69.0             62.0
2019
9           55           20.0           74.0             75.0
2021
10          73           77.0           77.0             60.0
2018
11          64           60.0           73.0             75.0
2021
12          73           60.0           73.0             95.0
2021
13          72           71.0           61.0             80.0
```

```
2019
14           74            69.0            68.0            70.0
2021
15           73            63.0            67.0             NaN
2018
16           65            86.0            61.0            77.0
2019
17           87            61.0            90.0            77.0
2018
18           71            69.0            78.0             NaN
2019
19           67            77.0            69.0            79.0
2020
20           62             NaN            78.0            75.0
2021
21           67            68.0            64.0            64.0
2018
22           66            69.0            72.0            79.0
2021
23           63            62.0            61.0            77.0
2019
24           67            76.0            70.0            65.0
2018
25           55            74.0            66.0            77.0
2020
26           96            60.0            20.0            69.0
2021
27           75            79.0            73.0            80.0
2019
28           70            79.0            57.0            63.0
2019
29           70            74.0            75.0            69.0
2020
```

| | Placement-offer-count |
|---|---|
| 0 | 2 |
| 1 | 0 |
| 2 | 2 |
| 3 | 2 |
| 4 | 0 |
| 5 | 0 |
| 6 | 2 |
| 7 | 0 |
| 8 | 2 |
| 9 | 0 |
| 10 | 2 |
| 11 | 0 |
| 12 | 0 |
| 13 | 0 |

```
14                      2
15                      0
16                      0
17                      0
18                      2
19                      0
20                      0
21                      2
22                      0
23                      0
24                      2
25                      0
26                      2
27                      0
28                      2
29                      2
```

```
ndf=df
ndf.fillna(0)
```

```
    math_score  reading_score  writing_score  placement_score
club_join_year  \
0           77           73.0           69.0             65.0
2020
1           75           70.0           58.0             94.0
2018
2           75           63.0           61.0             69.0
2019
3           69           70.0           71.0             73.0
2019
4           97           50.0           64.0             77.0
2018
5           78           75.0           62.0             78.0
2019
6           77           71.0           74.0             66.0
2019
7           66           70.0            0.0             77.0
2019
8           70           67.0           69.0             62.0
2019
9           55           20.0           74.0             75.0
2021
10          73           77.0           77.0             60.0
2018
11          64           60.0           73.0             75.0
2021
12          73           60.0           73.0             95.0
2021
13          72           71.0           61.0             80.0
2019
```

```
14           74            69.0            68.0            70.0
2021
15           73            63.0            67.0             0.0
2018
16           65            86.0            61.0            77.0
2019
17           87            61.0            90.0            77.0
2018
18           71            69.0            78.0             0.0
2019
19           67            77.0            69.0            79.0
2020
20           62             0.0            78.0            75.0
2021
21           67            68.0            64.0            64.0
2018
22           66            69.0            72.0            79.0
2021
23           63            62.0            61.0            77.0
2019
24           67            76.0            70.0            65.0
2018
25           55            74.0            66.0            77.0
2020
26           96            60.0            20.0            69.0
2021
27           75            79.0            73.0            80.0
2019
28           70            79.0            57.0            63.0
2019
29           70            74.0            75.0            69.0
2020

    Placement-offer-count
0                       2
1                       0
2                       2
3                       2
4                       0
5                       0
6                       2
7                       0
8                       2
9                       0
10                      2
11                      0
12                      0
13                      0
14                      2
```

```
15                      0
16                      0
17                      0
18                      2
19                      0
20                      0
21                      2
22                      0
23                      0
24                      2
25                      0
26                      2
27                      0
28                      2
29                      2
```

```
# df['reading_score'] =
df['reading_score'].fillna(df['reading_score'].mean())
# df['reading_score'] =
df['reading_score'].fillna(df['reading_score'].median())
# df['reading_score'] =
df['reading_score'].fillna(df['reading_score'].std())
# df['reading_score'] =
df['reading_score'].fillna(df['reading_score'].min())
df['reading_score'] =
df['reading_score'].fillna(df['reading_score'].max())


df
```

```
    math_score  reading_score  writing_score  placement_score
club_join_year  \
0           77           73.0           69.0             65.0
2020
1           75           70.0           58.0             94.0
2018
2           75           63.0           61.0             69.0
2019
3           69           70.0           71.0             73.0
2019
4           97           50.0           64.0             77.0
2018
5           78           75.0           62.0             78.0
2019
6           77           71.0           74.0             66.0
2019
7           66           70.0            NaN             77.0
2019
8           70           67.0           69.0             62.0
2019
```

```
9            55            20.0            74.0            75.0
2021
10           73            77.0            77.0            60.0
2018
11           64            60.0            73.0            75.0
2021
12           73            60.0            73.0            95.0
2021
13           72            71.0            61.0            80.0
2019
14           74            69.0            68.0            70.0
2021
15           73            63.0            67.0             NaN
2018
16           65            86.0            61.0            77.0
2019
17           87            61.0            90.0            77.0
2018
18           71            69.0            78.0             NaN
2019
19           67            77.0            69.0            79.0
2020
20           62            86.0            78.0            75.0
2021
21           67            68.0            64.0            64.0
2018
22           66            69.0            72.0            79.0
2021
23           63            62.0            61.0            77.0
2019
24           67            76.0            70.0            65.0
2018
25           55            74.0            66.0            77.0
2020
26           96            60.0            20.0            69.0
2021
27           75            79.0            73.0            80.0
2019
28           70            79.0            57.0            63.0
2019
29           70            74.0            75.0            69.0
2020

    Placement-offer-count
0                       2
1                       0
2                       2
3                       2
4                       0
```

```
5                       0
6                       2
7                       0
8                       2
9                       0
10                      2
11                      0
12                      0
13                      0
14                      2
15                      0
16                      0
17                      0
18                      2
19                      0
20                      0
21                      2
22                      0
23                      0
24                      2
25                      0
26                      2
27                      0
28                      2
29                      2
```

```
m_v=df['placement_score'].mean()
df['placement_score'].fillna(value=m_v, inplace=True)
df
```

```
    math_score  reading_score  writing_score  placement_score
club_join_year  \
0           77           73.0           69.0        65.000000
2020
1           75           70.0           58.0        94.000000
2018
2           75           63.0           61.0        69.000000
2019
3           69           70.0           71.0        73.000000
2019
4           97           50.0           64.0        77.000000
2018
5           78           75.0           62.0        78.000000
2019
6           77           71.0           74.0        66.000000
2019
7           66           70.0            NaN        77.000000
2019
8           70           67.0           69.0        62.000000
2019
```

```
9            55            20.0            74.0        75.000000
2021
10           73            77.0            77.0        60.000000
2018
11           64            60.0            73.0        75.000000
2021
12           73            60.0            73.0        95.000000
2021
13           72            71.0            61.0        80.000000
2019
14           74            69.0            68.0        70.000000
2021
15           73            63.0            67.0        73.821429
2018
16           65            86.0            61.0        77.000000
2019
17           87            61.0            90.0        77.000000
2018
18           71            69.0            78.0        73.821429
2019
19           67            77.0            69.0        79.000000
2020
20           62            86.0            78.0        75.000000
2021
21           67            68.0            64.0        64.000000
2018
22           66            69.0            72.0        79.000000
2021
23           63            62.0            61.0        77.000000
2019
24           67            76.0            70.0        65.000000
2018
25           55            74.0            66.0        77.000000
2020
26           96            60.0            20.0        69.000000
2021
27           75            79.0            73.0        80.000000
2019
28           70            79.0            57.0        63.000000
2019
29           70            74.0            75.0        69.000000
2020

    Placement-offer-count
0                       2
1                       0
2                       2
3                       2
4                       0
```

```
5                       0
6                       2
7                       0
8                       2
9                       0
10                      2
11                      0
12                      0
13                      0
14                      2
15                      0
16                      0
17                      0
18                      2
19                      0
20                      0
21                      2
22                      0
23                      0
24                      2
25                      0
26                      2
27                      0
28                      2
29                      2
```

```
ndf1=df
ndf1.replace(to_replace = np.nan, value = -99)
ndf1=ndf1.replace(to_replace = 77, value = np.nan)
ndf1
```

```
    math_score  reading_score  writing_score  placement_score
club_join_year  \
0          NaN           73.0           69.0        65.000000
2020
1         75.0           70.0           58.0        94.000000
2018
2         75.0           63.0           61.0        69.000000
2019
3         69.0           70.0           71.0        73.000000
2019
4         97.0           50.0           64.0              NaN
2018
5         78.0           75.0           62.0        78.000000
2019
6          NaN           71.0           74.0        66.000000
2019
7         66.0           70.0            NaN              NaN
2019
8         70.0           67.0           69.0        62.000000
```

```
2019
9         55.0            20.0            74.0         75.000000
2021
10        73.0             NaN             NaN         60.000000
2018
11        64.0            60.0            73.0         75.000000
2021
12        73.0            60.0            73.0         95.000000
2021
13        72.0            71.0            61.0         80.000000
2019
14        74.0            69.0            68.0         70.000000
2021
15        73.0            63.0            67.0         73.821429
2018
16        65.0            86.0            61.0                NaN
2019
17        87.0            61.0            90.0                NaN
2018
18        71.0            69.0            78.0         73.821429
2019
19        67.0             NaN            69.0         79.000000
2020
20        62.0            86.0            78.0         75.000000
2021
21        67.0            68.0            64.0         64.000000
2018
22        66.0            69.0            72.0         79.000000
2021
23        63.0            62.0            61.0                NaN
2019
24        67.0            76.0            70.0         65.000000
2018
25        55.0            74.0            66.0                NaN
2020
26        96.0            60.0            20.0         69.000000
2021
27        75.0            79.0            73.0         80.000000
2019
28        70.0            79.0            57.0         63.000000
2019
29        70.0            74.0            75.0         69.000000
2020

    Placement-offer-count
0                       2
1                       0
2                       2
3                       2
```

```
4                        0
5                        0
6                        2
7                        0
8                        2
9                        0
10                       2
11                       0
12                       0
13                       0
14                       2
15                       0
16                       0
17                       0
18                       2
19                       0
20                       0
21                       2
22                       0
23                       0
24                       2
25                       0
26                       2
27                       0
28                       2
29                       2
```

```
# 1. Dropping rows with at least 1 null value
ndf1.dropna()
```

```
    math_score  reading_score  writing_score  placement_score
club_join_year  \
1         75.0           70.0           58.0        94.000000
2018
2         75.0           63.0           61.0        69.000000
2019
3         69.0           70.0           71.0        73.000000
2019
5         78.0           75.0           62.0        78.000000
2019
8         70.0           67.0           69.0        62.000000
2019
9         55.0           20.0           74.0        75.000000
2021
11        64.0           60.0           73.0        75.000000
2021
12        73.0           60.0           73.0        95.000000
2021
13        72.0           71.0           61.0        80.000000
2019
```

```
14         74.0           69.0           68.0         70.000000
2021
15         73.0           63.0           67.0         73.821429
2018
18         71.0           69.0           78.0         73.821429
2019
20         62.0           86.0           78.0         75.000000
2021
21         67.0           68.0           64.0         64.000000
2018
22         66.0           69.0           72.0         79.000000
2021
24         67.0           76.0           70.0         65.000000
2018
26         96.0           60.0           20.0         69.000000
2021
27         75.0           79.0           73.0         80.000000
2019
28         70.0           79.0           57.0         63.000000
2019
29         70.0           74.0           75.0         69.000000
2020

    Placement-offer-count
1                       0
2                       2
3                       2
5                       0
8                       2
9                       0
11                      0
12                      0
13                      0
14                      2
15                      0
18                      2
20                      0
21                      2
22                      0
24                      2
26                      2
27                      0
28                      2
29                      2

# 2. Dropping rows if all values in that row are missing
ndf1.dropna(how = 'all')

    math_score  reading_score  writing_score  placement_score
club_join_year  \
```

| | | | | | |
|---|---|---|---|---|---|
| 0 | NaN | 73.0 | 69.0 | 65.000000 | 2020 |
| 1 | 75.0 | 70.0 | 58.0 | 94.000000 | 2018 |
| 2 | 75.0 | 63.0 | 61.0 | 69.000000 | 2019 |
| 3 | 69.0 | 70.0 | 71.0 | 73.000000 | 2019 |
| 4 | 97.0 | 50.0 | 64.0 | NaN | 2018 |
| 5 | 78.0 | 75.0 | 62.0 | 78.000000 | 2019 |
| 6 | NaN | 71.0 | 74.0 | 66.000000 | 2019 |
| 7 | 66.0 | 70.0 | NaN | NaN | 2019 |
| 8 | 70.0 | 67.0 | 69.0 | 62.000000 | 2019 |
| 9 | 55.0 | 20.0 | 74.0 | 75.000000 | 2021 |
| 10 | 73.0 | NaN | NaN | 60.000000 | 2018 |
| 11 | 64.0 | 60.0 | 73.0 | 75.000000 | 2021 |
| 12 | 73.0 | 60.0 | 73.0 | 95.000000 | 2021 |
| 13 | 72.0 | 71.0 | 61.0 | 80.000000 | 2019 |
| 14 | 74.0 | 69.0 | 68.0 | 70.000000 | 2021 |
| 15 | 73.0 | 63.0 | 67.0 | 73.821429 | 2018 |
| 16 | 65.0 | 86.0 | 61.0 | NaN | 2019 |
| 17 | 87.0 | 61.0 | 90.0 | NaN | 2018 |
| 18 | 71.0 | 69.0 | 78.0 | 73.821429 | 2019 |
| 19 | 67.0 | NaN | 69.0 | 79.000000 | 2020 |
| 20 | 62.0 | 86.0 | 78.0 | 75.000000 | 2021 |
| 21 | 67.0 | 68.0 | 64.0 | 64.000000 | 2018 |
| 22 | 66.0 | 69.0 | 72.0 | 79.000000 | 2021 |
| 23 | 63.0 | 62.0 | 61.0 | NaN | 2019 |
| 24 | 67.0 | 76.0 | 70.0 | 65.000000 | |

```
2018
25        55.0            74.0            66.0               NaN
2020
26        96.0            60.0            20.0         69.000000
2021
27        75.0            79.0            73.0         80.000000
2019
28        70.0            79.0            57.0         63.000000
2019
29        70.0            74.0            75.0         69.000000
2020
```

```
    Placement-offer-count
0                       2
1                       0
2                       2
3                       2
4                       0
5                       0
6                       2
7                       0
8                       2
9                       0
10                      2
11                      0
12                      0
13                      0
14                      2
15                      0
16                      0
17                      0
18                      2
19                      0
20                      0
21                      2
22                      0
23                      0
24                      2
25                      0
26                      2
27                      0
28                      2
29                      2
```

```
# 3. Dropping columns with at least 1 null value.
ndf1.dropna(axis = 1)
```

```
    club_join_year  Placement-offer-count
0             2020                      2
1             2018                      0
```

```
2              2019                        2
3              2019                        2
4              2018                        0
5              2019                        0
6              2019                        2
7              2019                        0
8              2019                        2
9              2021                        0
10             2018                        2
11             2021                        0
12             2021                        0
13             2019                        0
14             2021                        2
15             2018                        0
16             2019                        0
17             2018                        0
18             2019                        2
19             2020                        0
20             2021                        0
21             2018                        2
22             2021                        0
23             2019                        0
24             2018                        2
25             2020                        0
26             2021                        2
27             2019                        0
28             2019                        2
29             2020                        2
```

```
# 4. Dropping Rows with at least 1 null value in CSV file
new_data = ndf1.dropna(axis = 0, how ='any')
new_data
```

```
    math_score  reading_score  writing_score  placement_score
club_join_year  \
1         75.0           70.0           58.0        94.000000
2018
2         75.0           63.0           61.0        69.000000
2019
3         69.0           70.0           71.0        73.000000
2019
5         78.0           75.0           62.0        78.000000
2019
8         70.0           67.0           69.0        62.000000
2019
9         55.0           20.0           74.0        75.000000
2021
11        64.0           60.0           73.0        75.000000
2021
12        73.0           60.0           73.0        95.000000
```

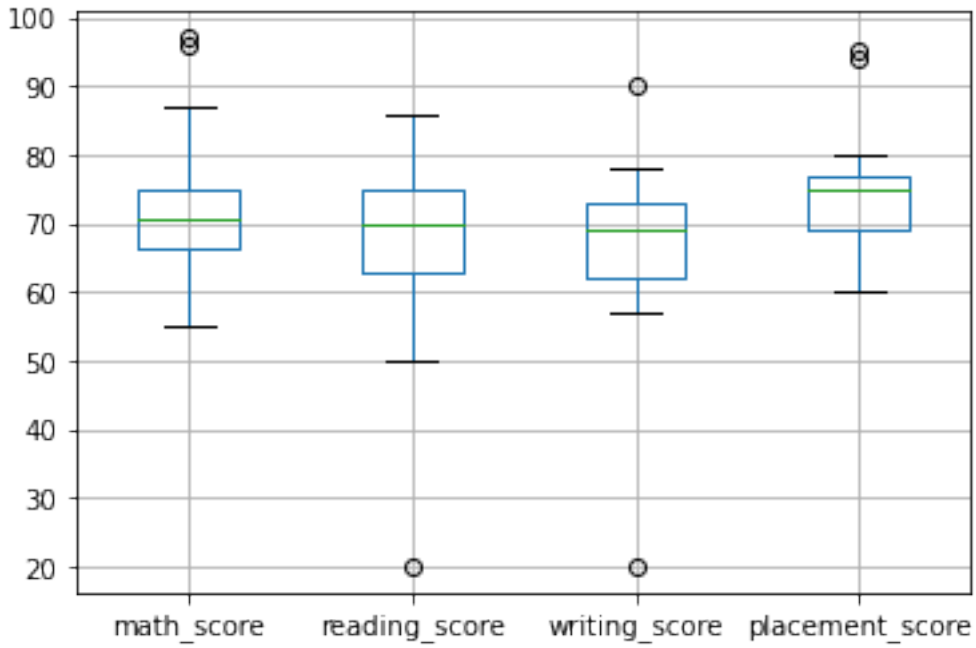
```
2021
13         72.0            71.0            61.0         80.000000
2019
14         74.0            69.0            68.0         70.000000
2021
15         73.0            63.0            67.0         73.821429
2018
18         71.0            69.0            78.0         73.821429
2019
20         62.0            86.0            78.0         75.000000
2021
21         67.0            68.0            64.0         64.000000
2018
22         66.0            69.0            72.0         79.000000
2021
24         67.0            76.0            70.0         65.000000
2018
26         96.0            60.0            20.0         69.000000
2021
27         75.0            79.0            73.0         80.000000
2019
28         70.0            79.0            57.0         63.000000
2019
29         70.0            74.0            75.0         69.000000
2020
```

```
    Placement-offer-count
1                       0
2                       2
3                       2
5                       0
8                       2
9                       0
11                      0
12                      0
13                      0
14                      2
15                      0
18                      2
20                      0
21                      2
22                      0
24                      2
26                      2
27                      0
28                      2
29                      2
```

```
#Outliers
d1= df
```

```
#Boxplot
col = ['math_score', 'reading_score' ,
'writing_score','placement_score']
df.boxplot(col)
```

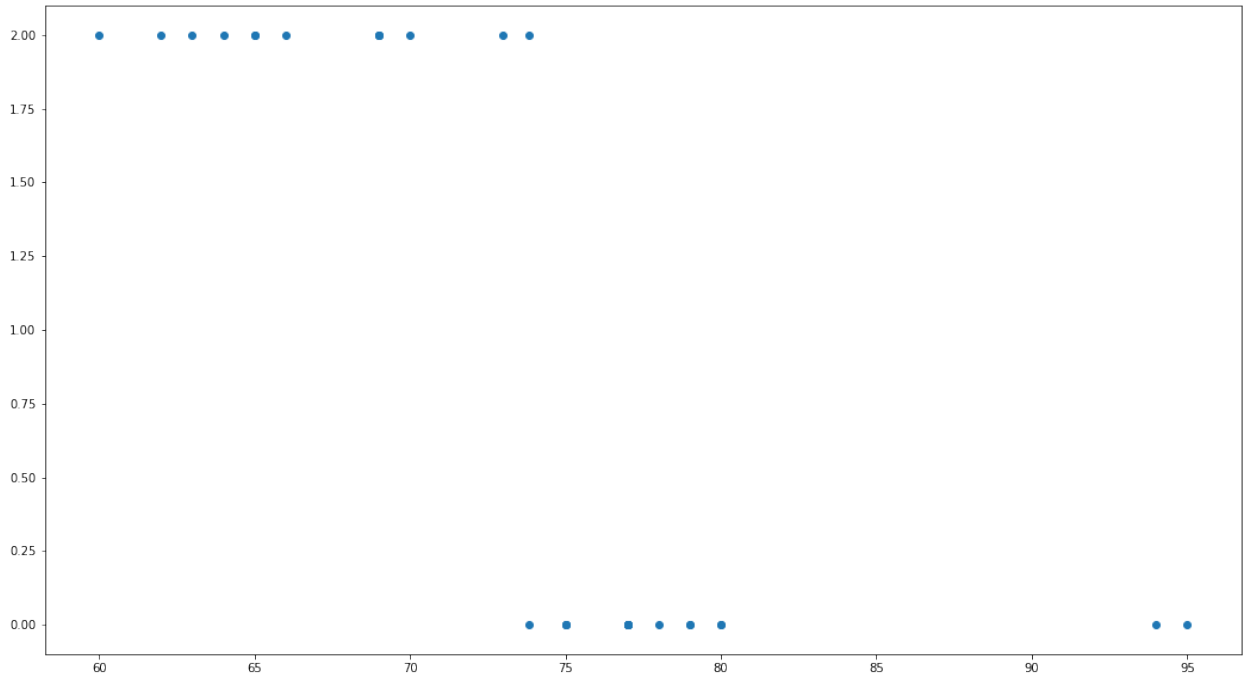
```
<AxesSubplot:>
```




```
print(np.where(df['math_score']>90))
print(np.where(df['reading_score']<25))
print(np.where(df['writing_score']<30))
```

```
(array([ 4, 26]),)
(array([9]),)
(array([26]),)
```

```
import matplotlib.pyplot as plt

#scatterplot
fig, ax = plt.subplots(figsize = (18,10))
ax.scatter(d1['placement_score'], d1['Placement-offer-count'])
plt.show()
ax.set_xlabel('(Proportion non-retail business acres)/(town)')
ax.set_ylabel('(Full-value property-tax rate)/($10,000)')
```




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

df

    math_score  reading_score  writing_score  placement_score
club_join_year  \
0           77           73.0           69.0        65.000000
2020
1           75           70.0           58.0        94.000000
2018
2           75           63.0           61.0        69.000000
2019
3           69           70.0           71.0        73.000000
2019
4           97           50.0           64.0        77.000000
2018
5           78           75.0           62.0        78.000000
2019
6           77           71.0           74.0        66.000000
2019
7           66           70.0            NaN        77.000000
2019
8           70           67.0           69.0        62.000000
2019
9           55           20.0           74.0        75.000000
2021
10          73           77.0           77.0        60.000000
2018
```

```
11          64          60.0          73.0        75.000000
2021
12          73          60.0          73.0        95.000000
2021
13          72          71.0          61.0        80.000000
2019
14          74          69.0          68.0        70.000000
2021
15          73          63.0          67.0        73.821429
2018
16          65          86.0          61.0        77.000000
2019
17          87          61.0          90.0        77.000000
2018
18          71          69.0          78.0        73.821429
2019
19          67          77.0          69.0        79.000000
2020
20          62          86.0          78.0        75.000000
2021
21          67          68.0          64.0        64.000000
2018
22          66          69.0          72.0        79.000000
2021
23          63          62.0          61.0        77.000000
2019
24          67          76.0          70.0        65.000000
2018
25          55          74.0          66.0        77.000000
2020
26          96          60.0          20.0        69.000000
2021
27          75          79.0          73.0        80.000000
2019
28          70          79.0          57.0        63.000000
2019
29          70          74.0          75.0        69.000000
2020
```

```
    Placement-offer-count
0                       2
1                       0
2                       2
3                       2
4                       0
5                       0
6                       2
7                       0
8                       2
```

```
9                       0
10                      2
11                      0
12                      0
13                      0
14                      2
15                      0
16                      0
17                      0
18                      2
19                      0
20                      0
21                      2
22                      0
23                      0
24                      2
25                      0
26                      2
27                      0
28                      2
29                      2
```

```
print(np.where((df['placement_score']<50) & (df['Placement-offer-
count']>1)))
print(np.where((df['placement_score']>85) & (df['Placement-offer-
count']<3)))
```

```
(array([], dtype=int64),)
(array([ 1, 12]),)
```

```
#Z-score
from scipy import stats
z = np.abs(stats.zscore(df['math_score']))
print(z)
```

```
0     0.578370
1     0.362828
2     0.362828
3     0.283797
4     2.733787
5     0.686141
6     0.578370
7     0.607109
8     0.176026
9     1.792588
10    0.147287
11    0.822651
12    0.147287
13    0.039516
14    0.255058
```

```
15    0.147287
16    0.714880
17    1.656078
18    0.068255
19    0.499338
20    1.038192
21    0.499338
22    0.607109
23    0.930421
24    0.499338
25    1.792588
26    2.626016
27    0.362828
28    0.176026
29    0.176026
Name: math_score, dtype: float64
```

```
threshold = 0.18
sample_outliers = np.where(z <threshold)
sample_outliers
```

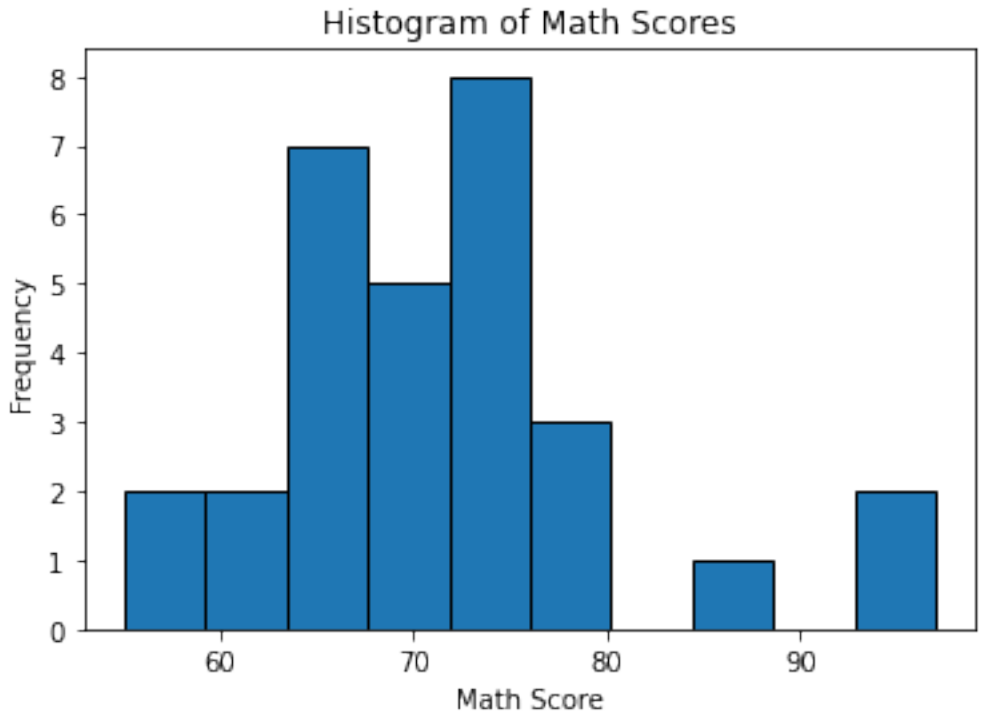
```
(array([ 6, 15, 24, 25]),)
```

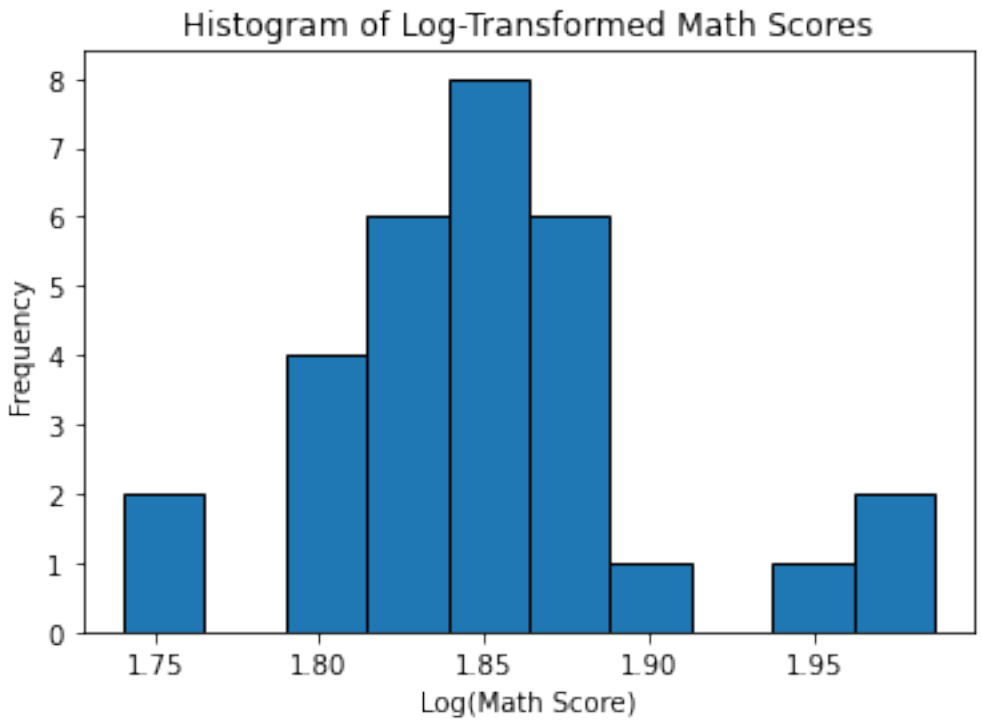
```
# Detecting outliers using Inter Quantile Range(IQR):

sorted_rscore= sorted(d1['reading_score'])
sorted_rscore
q1 = np.percentile(sorted_rscore, 25)

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

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

r_outliers = []
for i in sorted_rscore:
    if (i<lwr_bound or i>upr_bound):
        r_outliers.append(i)

print(r_outliers)
```

```
63.0 74.75
45.375 92.375
[20.0]
```

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

```
df =
pd.read_excel("/home/ubuntu/TCOB35/DSBDAL/Student_Performance.xlsx")

df['math_score'].plot(kind='hist', title='Histogram of Math Scores',
edgecolor='black')
plt.xlabel('Math Score')
plt.ylabel('Frequency')
plt.show()

df['log_math'] = np.log10(df['math_score'])

df['log_math'].plot(kind='hist', title='Histogram of Log-Transformed
Math Scores', edgecolor='black')
plt.xlabel('Log(Math Score)')
plt.ylabel('Frequency')
plt.show()
```






```
#Handling of Outliers
sample_outliers=[9,1,26,4] #contains index of row where outlier is
present
# 1. Trimming/Removing the Outliers:
new_df = df
for i in sample_outliers:
    new_df.drop(i, inplace=True)
new_df
```

```
    math_score  reading_score  writing_score placement_score
club_join_year  \
0           77           73.0           69.0              65
2020
2           75           63.0           61.0              69
2019
3           69           70.0           71.0              73
2019
5           78           75.0           62.0              78
2019
6           77           71.0           74.0              66
2019
7           66           70.0            NaN              77
2019
8           70           67.0           69.0              62
2019
10          73           77.0           77.0              60
2018
```

```
11           64            60.0            73.0                75
2021
12           73            60.0            73.0                95
2021
13           72            71.0            61.0                80
2019
14           74            69.0            68.0                70
2021
15           73            63.0            67.0                na
2018
16           65            86.0            61.0                77
2019
17           87            61.0            90.0                77
2018
18           71            69.0            78.0               NaN
2019
19           67            77.0            69.0                79
2020
20           62             NaN            78.0                75
2021
21           67            68.0            64.0                64
2018
22           66            69.0            72.0                79
2021
23           63            62.0            61.0                77
2019
24           67            76.0            70.0                65
2018
25           55            74.0            66.0                77
2020
27           75            79.0            73.0                80
2019
28           70            79.0            57.0                63
2019
29           70            74.0            75.0                69
2020

    Placement-offer-count  log_math
0                       2  1.886491
2                       2  1.875061
3                       2  1.838849
5                       0  1.892095
6                       2  1.886491
7                       0  1.819544
8                       2  1.845098
10                      2  1.863323
11                      0  1.806180
12                      0  1.863323
13                      0  1.857332
```

```
14                      2  1.869232
15                      0  1.863323
16                      0  1.812913
17                      0  1.939519
18                      2  1.851258
19                      0  1.826075
20                      0  1.792392
21                      2  1.826075
22                      0  1.819544
23                      0  1.799341
24                      2  1.826075
25                      0  1.740363
27                      0  1.875061
28                      2  1.845098
29                      2  1.845098
```

```
# 2. Quantile-Based Flooring and Capping:
import numpy as np

df_stud = df
ninetieth_percentile = np.percentile(df_stud['math_score'], 90)

# Cap values above the 90th percentile
b = np.where(df_stud['math_score'] > ninetieth_percentile,
ninetieth_percentile, df_stud['math_score'])

print("New array:", b)

# Insert the capped data back into the DataFrame
df_stud.insert(1, "m score", b, True)
df_stud
```

```
New array: [77. 75. 69. 77. 77. 66. 70. 73. 64. 73. 72. 74. 73. 65.
77. 71. 67. 62.
 67. 66. 63. 67. 55. 75. 70. 70.]
```

```
    math_score  m score  reading_score  writing_score placement_score
\
0           77     77.0           73.0           69.0               65

2           75     75.0           63.0           61.0               69

3           69     69.0           70.0           71.0               73

5           78     77.0           75.0           62.0               78

6           77     77.0           71.0           74.0               66

7           66     66.0           70.0            NaN               77

8           70     70.0           67.0           69.0               62
```

| | | | | | |
|---|---|---|---|---|---|
| 10 | 73 | 73.0 | 77.0 | 77.0 | 60 |
| 11 | 64 | 64.0 | 60.0 | 73.0 | 75 |
| 12 | 73 | 73.0 | 60.0 | 73.0 | 95 |
| 13 | 72 | 72.0 | 71.0 | 61.0 | 80 |
| 14 | 74 | 74.0 | 69.0 | 68.0 | 70 |
| 15 | 73 | 73.0 | 63.0 | 67.0 | na |
| 16 | 65 | 65.0 | 86.0 | 61.0 | 77 |
| 17 | 87 | 77.0 | 61.0 | 90.0 | 77 |
| 18 | 71 | 71.0 | 69.0 | 78.0 | NaN |
| 19 | 67 | 67.0 | 77.0 | 69.0 | 79 |
| 20 | 62 | 62.0 | NaN | 78.0 | 75 |
| 21 | 67 | 67.0 | 68.0 | 64.0 | 64 |
| 22 | 66 | 66.0 | 69.0 | 72.0 | 79 |
| 23 | 63 | 63.0 | 62.0 | 61.0 | 77 |
| 24 | 67 | 67.0 | 76.0 | 70.0 | 65 |
| 25 | 55 | 55.0 | 74.0 | 66.0 | 77 |
| 27 | 75 | 75.0 | 79.0 | 73.0 | 80 |
| 28 | 70 | 70.0 | 79.0 | 57.0 | 63 |
| 29 | 70 | 70.0 | 74.0 | 75.0 | 69 |

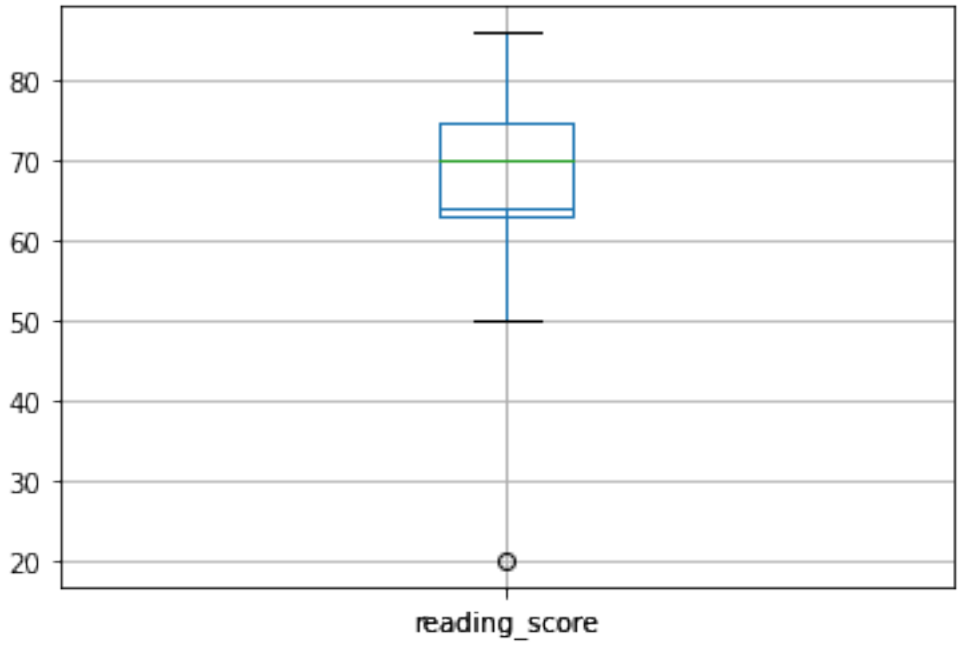
| | club_join_year | Placement-offer-count | log_math |
|---|---|---|---|
| 0 | 2020 | 2 | 1.886491 |
| 2 | 2019 | 2 | 1.875061 |
| 3 | 2019 | 2 | 1.838849 |
| 5 | 2019 | 0 | 1.892095 |
| 6 | 2019 | 2 | 1.886491 |
| 7 | 2019 | 0 | 1.819544 |
| 8 | 2019 | 2 | 1.845098 |
| 10 | 2018 | 2 | 1.863323 |
| 11 | 2021 | 0 | 1.806180 |
| 12 | 2021 | 0 | 1.863323 |

```
13              2019                        0  1.857332
14              2021                        2  1.869232
15              2018                        0  1.863323
16              2019                        0  1.812913
17              2018                        0  1.939519
18              2019                        2  1.851258
19              2020                        0  1.826075
20              2021                        0  1.792392
21              2018                        2  1.826075
22              2021                        0  1.819544
23              2019                        0  1.799341
24              2018                        2  1.826075
25              2020                        0  1.740363
27              2019                        0  1.875061
28              2019                        2  1.845098
29              2020                        2  1.845098
```

```
# 3. Mean/Median Imputation:
import numpy as np

col = ['reading_score']
df.boxplot(col)

sorted_rscore = sorted(df['reading_score'])
median = np.median(sorted_rscore)
print("Median:", median)

refined_df = df.copy()
refined_df['reading_score'] = np.where(refined_df['reading_score'] >
upr_bound, median, refined_df['reading_score'])

refined_df['reading_score'] = np.where(refined_df['reading_score'] <
lwr_bound, median, refined_df['reading_score'])

refined_df.boxplot(col)
```

```
Median: 70.0
```

```
<AxesSubplot:>
```


80
70
60
50
40
30
20
reading_score