In [57]:
```
##assignment 2
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
df=pd.read_csv("C:/Users/Pruthviraj/Downloads/StudentsPerformanceTest1.csv")
```

In [58]:
```
df.head()
```

Out[58]:

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	female	72	72	74.0	78.0	1	Pune
1	female	69	90	88.0	NaN	2	na
2	female	90	95	93.0	74.0	2	Nashik
3	male	47	57	NaN	78.0	1	Na
4	male	na	78	75.0	81.0	3	Pune

In [59]:
```
df.isnull()
```

Out[59]:

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	False	False	False	False	False	False	False
1	False	False	False	False	True	False	False
2	False	False	False	False	False	False	False
3	False	False	False	True	False	False	False
4	False	False	False	False	False	False	False
5	False	False	False	False	False	False	False
6	False	False	False	False	False	False	False
7	False	True	False	False	False	False	False
8	False	False	False	False	False	False	True

In [60]:
```
series = pd.isnull(df["math score"])
df[series]
```

Out[60]:

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
7	male	NaN	65	67.0	49.0	1	Pune

In [61]:
```
df.notnull()
```

Out[61]:

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	True	True	True	True	True	True	True
1	True	True	True	True	False	True	True
2	True	True	True	True	True	True	True
3	True	True	True	False	True	True	True
4	True	True	True	True	True	True	True
5	True	True	True	True	True	True	True
6	True	True	True	True	True	True	True
7	True	False	True	True	True	True	True
8	True	True	True	True	True	True	False

In [62]:
```
series1 = pd.notnull(df["math score"])
df[series1]
```

Out[62]:

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	female	72	72	74.0	78.0	1	Pune
1	female	69	90	88.0	NaN	2	na
2	female	90	95	93.0	74.0	2	Nashik
3	male	47	57	NaN	78.0	1	Na
4	male	na	78	75.0	81.0	3	Pune
5	female	71	Na	78.0	70.0	4	na
6	male	12	44	52.0	12.0	2	Nashik
8	male	5	77	89.0	55.0	0	NaN

In [63]:
```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
df['gender'] = le.fit_transform(df['gender'])
newdf=df
```

In [64]:
```
df.head()
```

Out[64]:

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	0	72	72	74.0	78.0	1	Pune
1	0	69	90	88.0	NaN	2	na
2	0	90	95	93.0	74.0	2	Nashik
3	1	47	57	NaN	78.0	1	Na
4	1	na	78	75.0	81.0	3	Pune

In [65]:
```
missing_values = ["Na", "na"]
df = pd.read_csv("StudentsPerformanceTest1.csv", na_values =
missing_values)
df.head()
```

Out[65]:

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	female	72.0	72.0	74.0	78.0	1	Pune
1	female	69.0	90.0	88.0	NaN	2	NaN
2	female	90.0	95.0	93.0	74.0	2	Nashik
3	male	47.0	57.0	NaN	78.0	1	NaN
4	male	NaN	78.0	75.0	81.0	3	Pune

In [66]:
```
ndf=df
ndf.fillna(0)
```

Out[66]:

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	female	72.0	72.0	74.0	78.0	1	Pune
1	female	69.0	90.0	88.0	0.0	2	0
2	female	90.0	95.0	93.0	74.0	2	Nashik
3	male	47.0	57.0	0.0	78.0	1	0
4	male	0.0	78.0	75.0	81.0	3	Pune
5	female	71.0	0.0	78.0	70.0	4	0
6	male	12.0	44.0	52.0	12.0	2	Nashik
7	male	0.0	65.0	67.0	49.0	1	Pune
8	male	5.0	77.0	89.0	55.0	0	0

In [67]:
```
m_v=df['math score'].mean()
df['math score'].fillna(value=m_v, inplace=True)
df.head()
```

Out[67]:

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	female	72.000000	72.0	74.0	78.0	1	Pune
1	female	69.000000	90.0	88.0	NaN	2	NaN
2	female	90.000000	95.0	93.0	74.0	2	Nashik
3	male	47.000000	57.0	NaN	78.0	1	NaN
4	male	52.285714	78.0	75.0	81.0	3	Pune

In [68]:
```
ndf.replace(to_replace = np.nan, value = -99)
```

Out[68]:

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	female	72.000000	72.0	74.0	78.0	1	Pune
1	female	69.000000	90.0	88.0	-99.0	2	-99
2	female	90.000000	95.0	93.0	74.0	2	Nashik
3	male	47.000000	57.0	-99.0	78.0	1	-99
4	male	52.285714	78.0	75.0	81.0	3	Pune
5	female	71.000000	-99.0	78.0	70.0	4	-99
6	male	12.000000	44.0	52.0	12.0	2	Nashik
7	male	52.285714	65.0	67.0	49.0	1	Pune
8	male	5.000000	77.0	89.0	55.0	0	-99

In [69]:
```
ndf.dropna()
```

Out[69]:

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	female	72.000000	72.0	74.0	78.0	1	Pune
2	female	90.000000	95.0	93.0	74.0	2	Nashik
4	male	52.285714	78.0	75.0	81.0	3	Pune
6	male	12.000000	44.0	52.0	12.0	2	Nashik
7	male	52.285714	65.0	67.0	49.0	1	Pune

In [70]:
```
ndf.dropna(how = 'all')
```

Out[70]:

	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	female	72.000000	72.0	74.0	78.0	1	Pune
1	female	69.000000	90.0	88.0	NaN	2	NaN
2	female	90.000000	95.0	93.0	74.0	2	Nashik
3	male	47.000000	57.0	NaN	78.0	1	NaN
4	male	52.285714	78.0	75.0	81.0	3	Pune
5	female	71.000000	NaN	78.0	70.0	4	NaN
6	male	12.000000	44.0	52.0	12.0	2	Nashik
7	male	52.285714	65.0	67.0	49.0	1	Pune
8	male	5.000000	77.0	89.0	55.0	0	NaN

In [71]:
```
ndf.dropna(axis = 1)
```

Out[71]:

	gender	math score	placement offer count
0	female	72.000000	1
1	female	69.000000	2
2	female	90.000000	2
3	male	47.000000	1
4	male	52.285714	3
5	female	71.000000	4
6	male	12.000000	2
7	male	52.285714	1
8	male	5.000000	0

In [72]:
```
new_data = ndf.dropna(axis = 0, how ='any')
new_data
```

Out[72]:

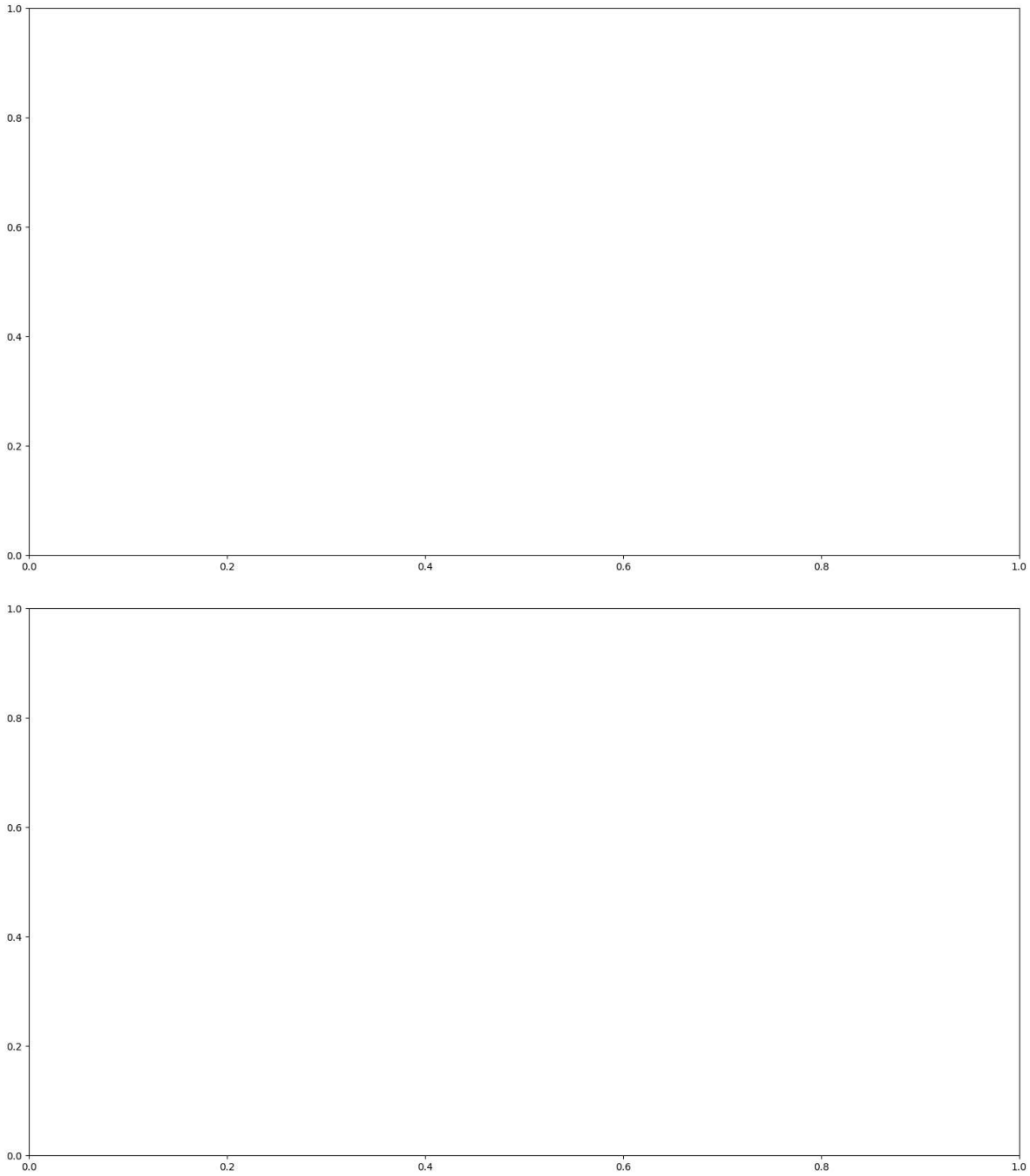
	gender	math score	reading score	writing score	Placement Score	placement offer count	Region
0	female	72.000000	72.0	74.0	78.0	1	Pune
2	female	90.000000	95.0	93.0	74.0	2	Nashik
4	male	52.285714	78.0	75.0	81.0	3	Pune
6	male	12.000000	44.0	52.0	12.0	2	Nashik
7	male	52.285714	65.0	67.0	49.0	1	Pune

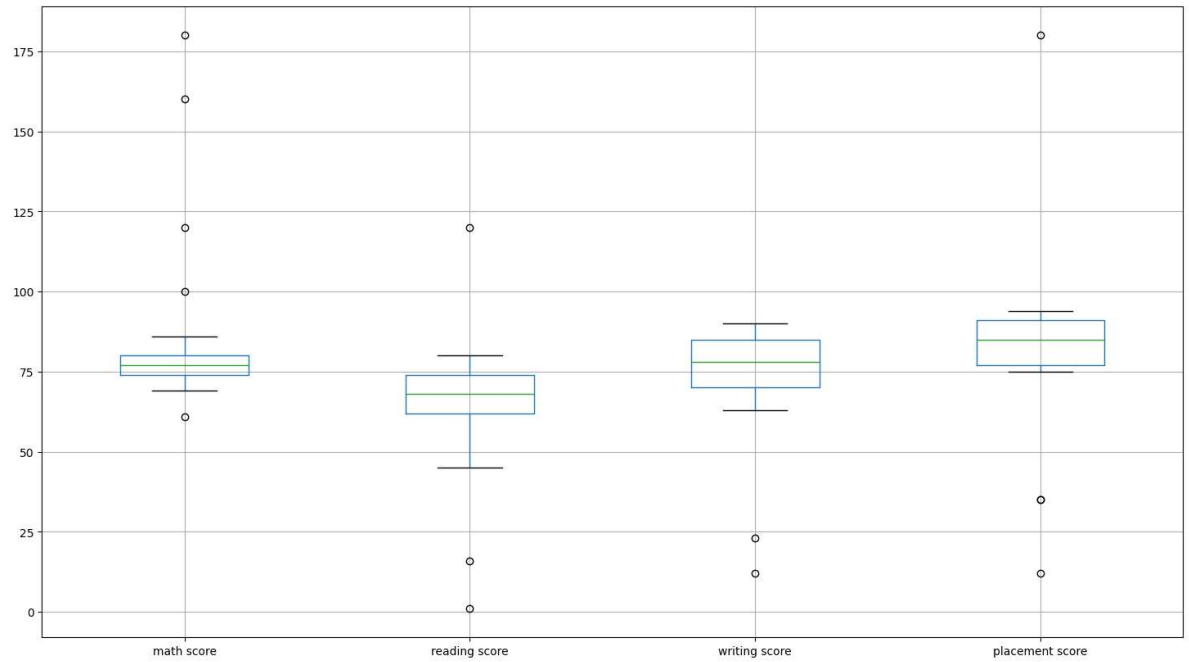
In [73]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import scipy
df=pd.read_csv("C:/Users/Pruthviraj/Downloads/demo1.csv")
df.head()
```

Out[73]:

	math score	reading score	writing score	placement score	placement offer count	club join year
0	80	68	70	89	3	2019
1	71	61	85	91	3	2019
2	79	16	87	77	2	2018
3	61	77	74	76	2	2020
4	78	71	67	90	3	2019

In [74]:
```
col = ['math score', 'reading score' , 'writing score','placement score']
df.boxplot(col)
plt.show()
```





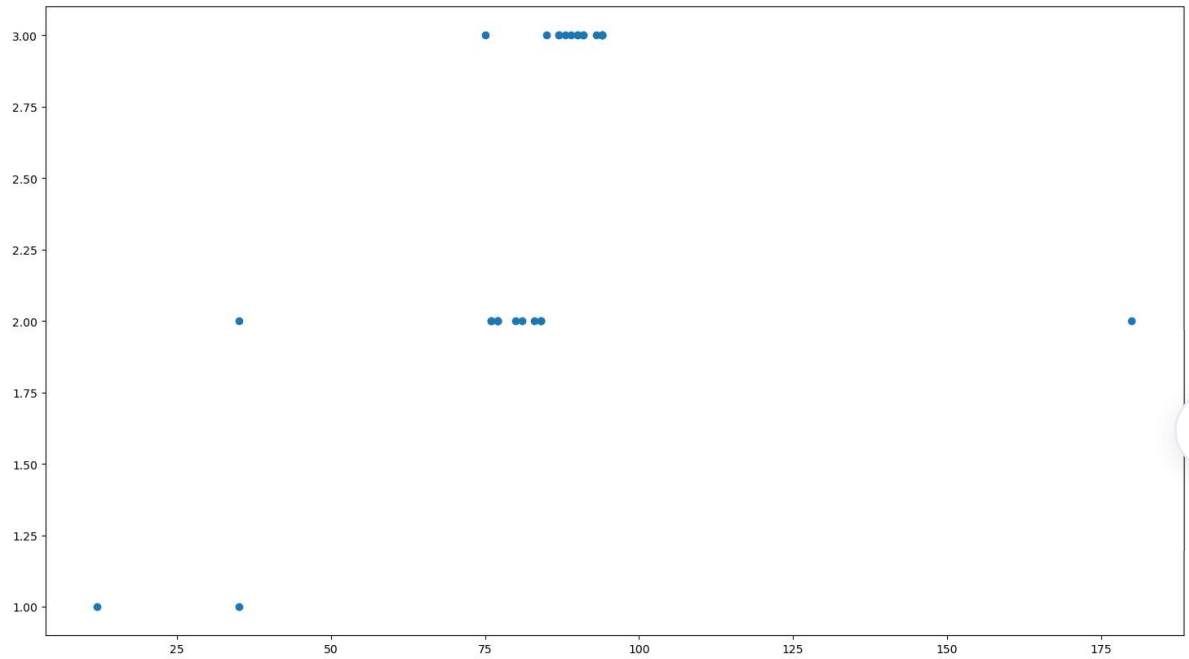
In [75]:
```
print(np.where(df['math score']>90))
print(np.where(df['reading score']<25))
print(np.where(df['writing score']<30))
```

```
(array([10, 21, 25, 28], dtype=int64),)
(array([ 2, 14], dtype=int64),)
(array([10, 17], dtype=int64),)
```

In [76]:
```
fig, ax = plt.subplots(figsize = (18,10))
ax.scatter(df['placement score'], df['placement offer count'])
```

Out[76]:
```
<matplotlib.collections.PathCollection at 0x1de678e82e0>
```

In [77]:
```
plt.show()
```



In [78]:
```
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([6], dtype=int64),)
(array([11], dtype=int64),)
```

In [79]:
```
threshold = 0.18
```

In [80]:
```
from scipy import stats
```

In [81]:
```
z = np.abs(stats.zscore(df['math score']))
```

In [82]:
```
sample_outliers = np.where(z < threshold)
sample_outliers
```

Out[82]:
```
(array([ 0, 12, 16, 17, 19], dtype=int64),)
```

In [83]:
```
print(z)
```

```
0     0.175646
1     0.528288
2     0.214828
3     0.920112
4     0.254010
5     0.449923
6     0.293193
7     0.410740
8     0.332375
9     0.371558
10    2.958952
11    0.214828
12    0.175646
13    0.254010
14    0.371558
15    0.254010
16    0.059449
17    0.175646
18    0.371558
19    0.097281
20    0.606653
21    0.608004
22    0.489105
23    0.410740
24    0.371558
25    3.742601
26    0.489105
27    0.528288
28    1.391653
Name: math score, dtype: float64
```

In [84]:
```
sorted_rscore= sorted(df['reading score'])
```

In [85]:
```
sorted_rscore
```

Out[85]:
```
[1,
 16,
 45,
 60,
 60,
 61,
 62,
 62,
 62,
 65,
 65,
 65,
 67,
 67,
 68,
 68,
 69,
 70,
 71,
 72,
 73,
 74,
 77,
 77,
 77,
 78,
 79,
 80,
 120]
```

In [86]:
```
q1 = np.percentile(sorted_rscore, 25)
q3 = np.percentile(sorted_rscore, 75)
print(q1,q3)
```

```
62.0 74.0
```

In [87]:
```
IQR = q3-q1
```

In [88]:
```
lwr_bound = q1-(1.5*IQR)
upr_bound = q3+(1.5*IQR)
print(lwr_bound, upr_bound)
```

```
44.0 92.0
```

In [89]:
```
r_outliers = []
for i in sorted_rscore:
    if (i<lwr_bound or i>upr_bound):
        r_outliers.append(i)
print(r_outliers)
```

```
[1, 16, 120]
```

In [90]:
```
new_df=df
for i in sample_outliers:
    new_df.drop(i, inplace=True)
    new_df
```

In [91]:
```
df_stud=df
ninetieth_percentile = np.percentile(df_stud['math score'], 90)
b = np.where(df_stud['math score']>ninetieth_percentile,
ninetieth_percentile, df_stud['math score'])
print("New array:",b)
```

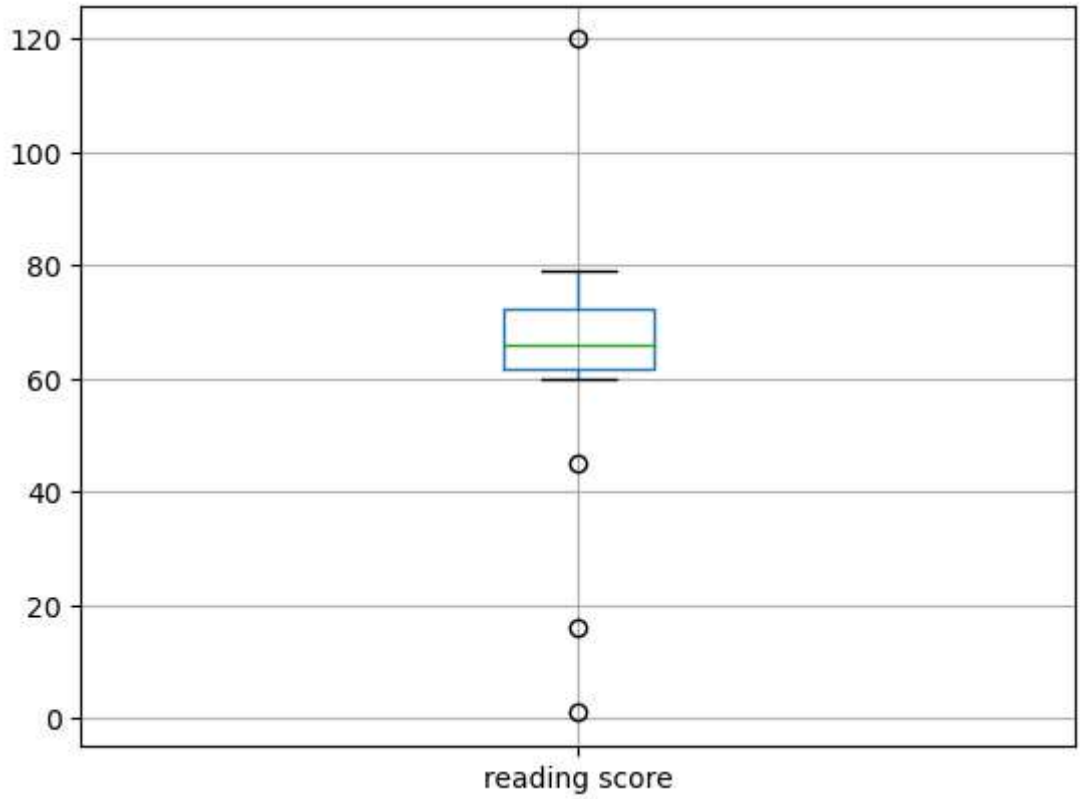
```
New array: [ 71.  79.  61.  78.  73.  77.  74.  76.  75. 114.  79.  78.  75.
78.
  75.  69. 100.  72.  74.  75. 114.  72.  71. 114.]
```

In [92]:
```
df_stud.insert(1,"m score",b,True)
df_stud.head()
```

Out[92]:

	math score	m score	reading score	writing score	placement score	placement offer count	club join year
1	71	71.0	61	85	91	3	2019
2	79	79.0	16	87	77	2	2018
3	61	61.0	77	74	76	2	2020
4	78	78.0	71	67	90	3	2019
5	73	73.0	68	90	80	2	2019

In [93]:
```
col = ['reading score']
df.boxplot(col)
plt.show()
```



In [94]:
```
median=np.median(sorted_rscore)
median
```

Out[94]: 68.0

In [95]:
```
refined_df=df
refined_df['reading score'] = np.where(refined_df['reading score'] >upr_bound,
```

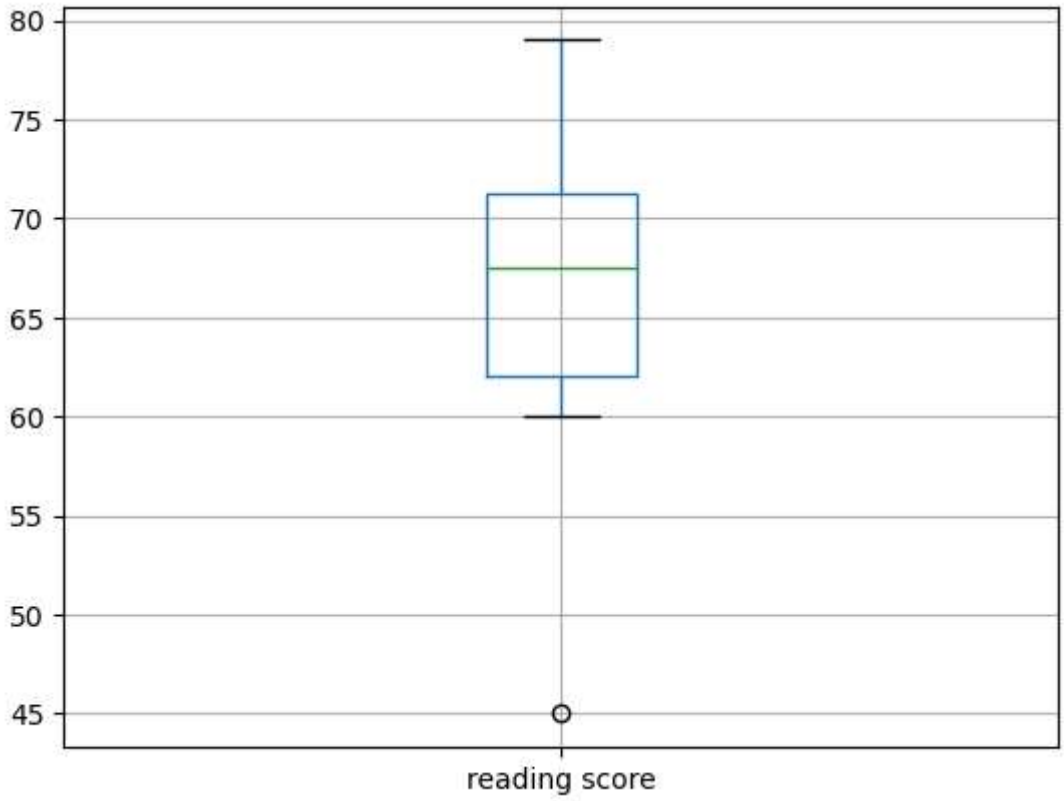
In [96]:
```
refined_df.head()
```

Out[96]:

	math score	m score	reading score	writing score	placement score	placement offer count	club join year
1	71	71.0	61.0	85	91	3	2019
2	79	79.0	16.0	87	77	2	2018
3	61	61.0	77.0	74	76	2	2020
4	78	78.0	71.0	67	90	3	2019
5	73	73.0	68.0	90	80	2	2019

In [97]:
```
refined_df['reading score'] = np.where(refined_df['reading score'] <lwr_bound,
```

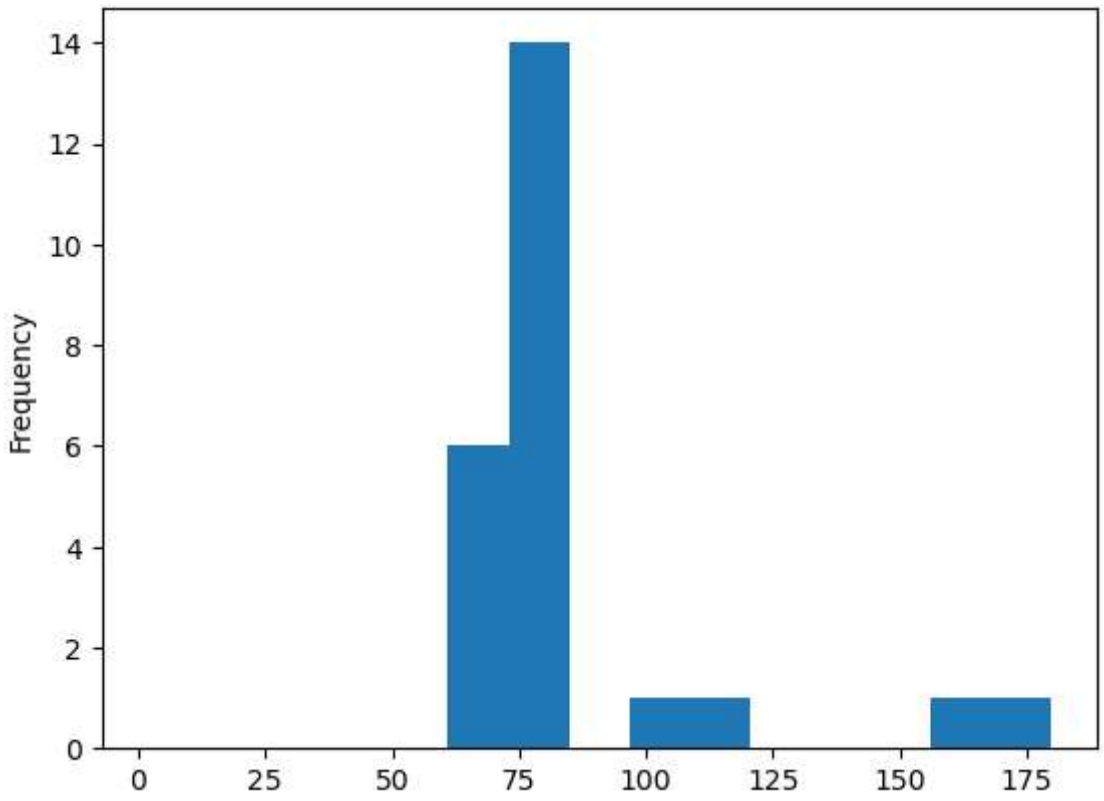
In [98]:
```
col = ['reading score']
refined_df.boxplot(col)
plt.show()
```



In [99]:
```
import matplotlib.pyplot as plt
new_df['math score'].plot(kind = 'hist')
```

Out[99]:
```
<Axes: ylabel='Frequency'>
```

In [100]:
```
df['log_math'] = np.log10(df['math score'])
df['log_math'].plot(kind = 'hist')
plt.show()
```



In [ ]: