In [50]:

```
import numpy as np
from sklearn.impute import KNNImputer
```

In [51]:

```
import pandas as pd
import seaborn as sns

data = pd.read_csv('air quality assessment.csv.csv')

print(data.head())
```

```
   Stn Code Sampling Date       State City/Town/Village/Area  \
0        38      01-02-14  Tamil Nadu                Chennai   
1        38      01-07-14  Tamil Nadu                Chennai   
2        38      21-01-14  Tamil Nadu                Chennai   
3        38      23-01-14  Tamil Nadu                Chennai   
4        38      28-01-14  Tamil Nadu                Chennai   

                     Location of Monitoring Station  \
0  Kathivakkam, Municipal Kalyana Mandapam, Chennai   
1  Kathivakkam, Municipal Kalyana Mandapam, Chennai   
2  Kathivakkam, Municipal Kalyana Mandapam, Chennai   
3  Kathivakkam, Municipal Kalyana Mandapam, Chennai   
4  Kathivakkam, Municipal Kalyana Mandapam, Chennai   

                                    Agency Type of Location   SO2   NO2  \
0  Tamilnadu State Pollution Control Board  Industrial Area  11.0  17.0   
1  Tamilnadu State Pollution Control Board  Industrial Area  13.0  17.0   
2  Tamilnadu State Pollution Control Board  Industrial Area  12.0  18.0   
3  Tamilnadu State Pollution Control Board  Industrial Area  15.0  16.0   
4  Tamilnadu State Pollution Control Board  Industrial Area  13.0  14.0   

   RSPM/PM10  PM 2.5
0       55.0     NaN
1       45.0     NaN
2       50.0     NaN
3       46.0     NaN
4       42.0     NaN
```

In [62]:

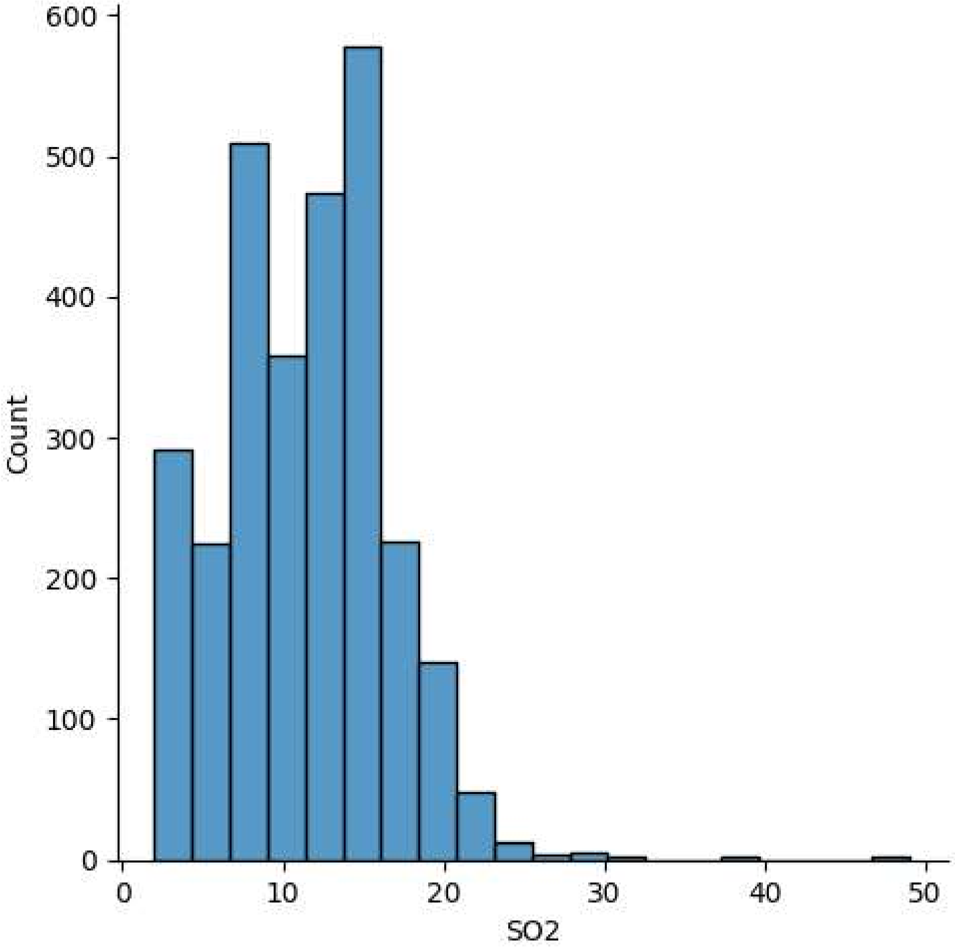
```
data = data.drop(["PM 2.5"],axis=1)
```

In [63]:

```
sns.displot(data["SO2"],bins=20)
```

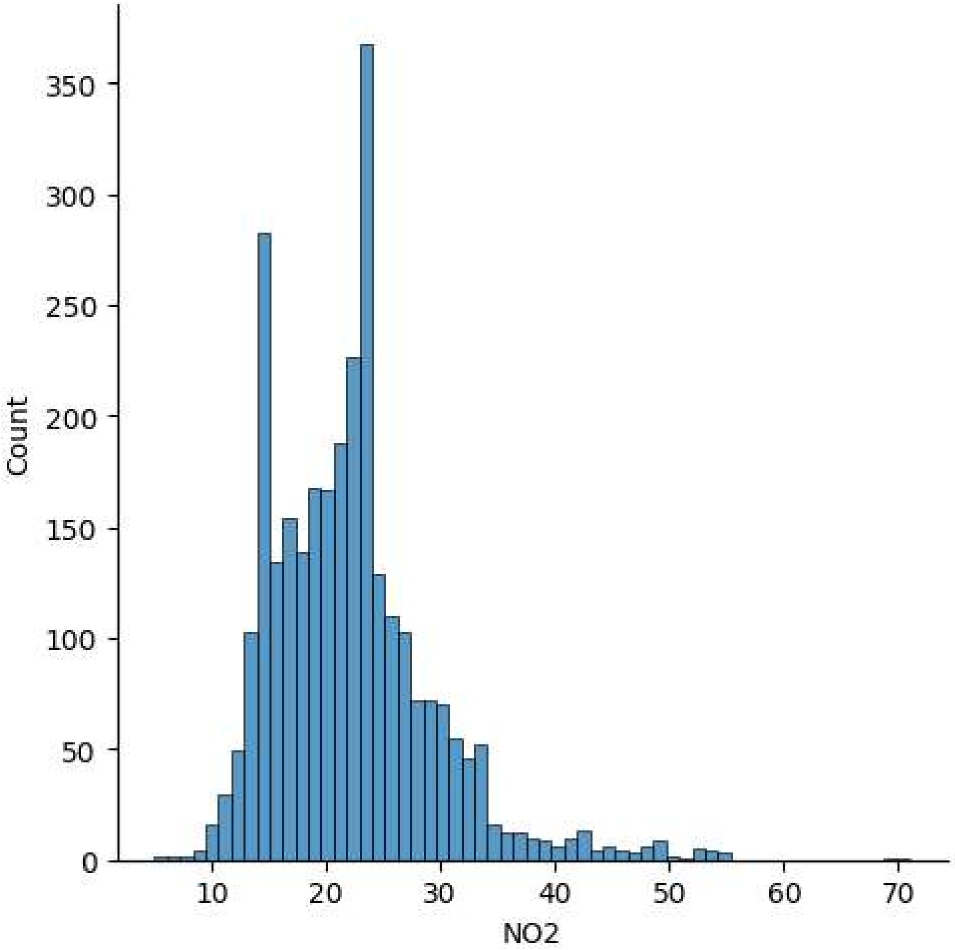
Out[63]:

```
<seaborn.axisgrid.FacetGrid at 0x1ed982cdb10>
```



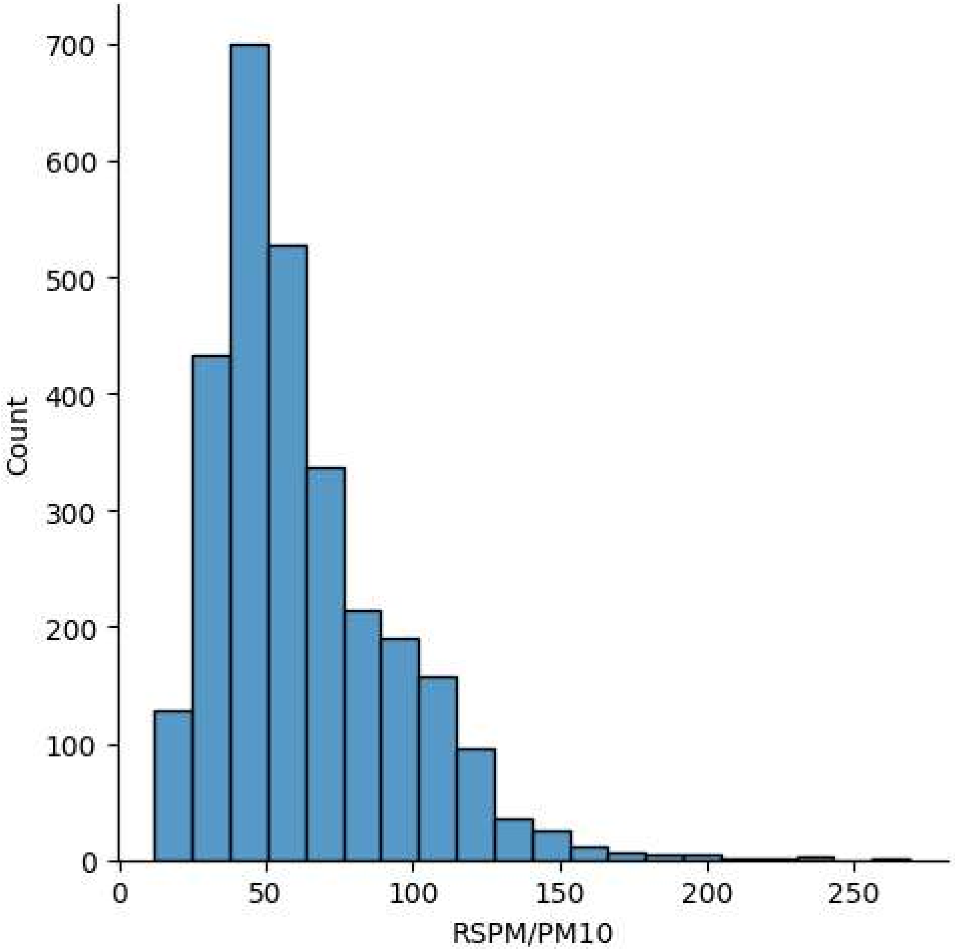
In [64]:
```
sns.displot(data["NO2"])
```

Out[64]:
```
<seaborn.axisgrid.FacetGrid at 0x1ed93f01e10>
```



```
In [65]: sns.displot(data["RSPM/PM10"],bins=20)
```

```
Out[65]: <seaborn.axisgrid.FacetGrid at 0x1ed98354390>
```



In [66]:
```
print(data.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2879 entries, 0 to 2878
Data columns (total 10 columns):
 #   Column                             Non-Null Count  Dtype  
---  ------                             --------------  -----  
 0   Stn Code                           2879 non-null   int64  
 1   Sampling Date                      2879 non-null   object 
 2   State                              2879 non-null   object 
 3   City/Town/Village/Area             2879 non-null   object 
 4   Location of Monitoring Station     2879 non-null   object 
 5   Agency                             2879 non-null   object 
 6   Type of Location                   2879 non-null   object 
 7   SO2                                2879 non-null   float64
 8   NO2                                2879 non-null   float64
 9   RSPM/PM10                          2879 non-null   float64
dtypes: float64(3), int64(1), object(6)
memory usage: 225.1+ KB
None
```

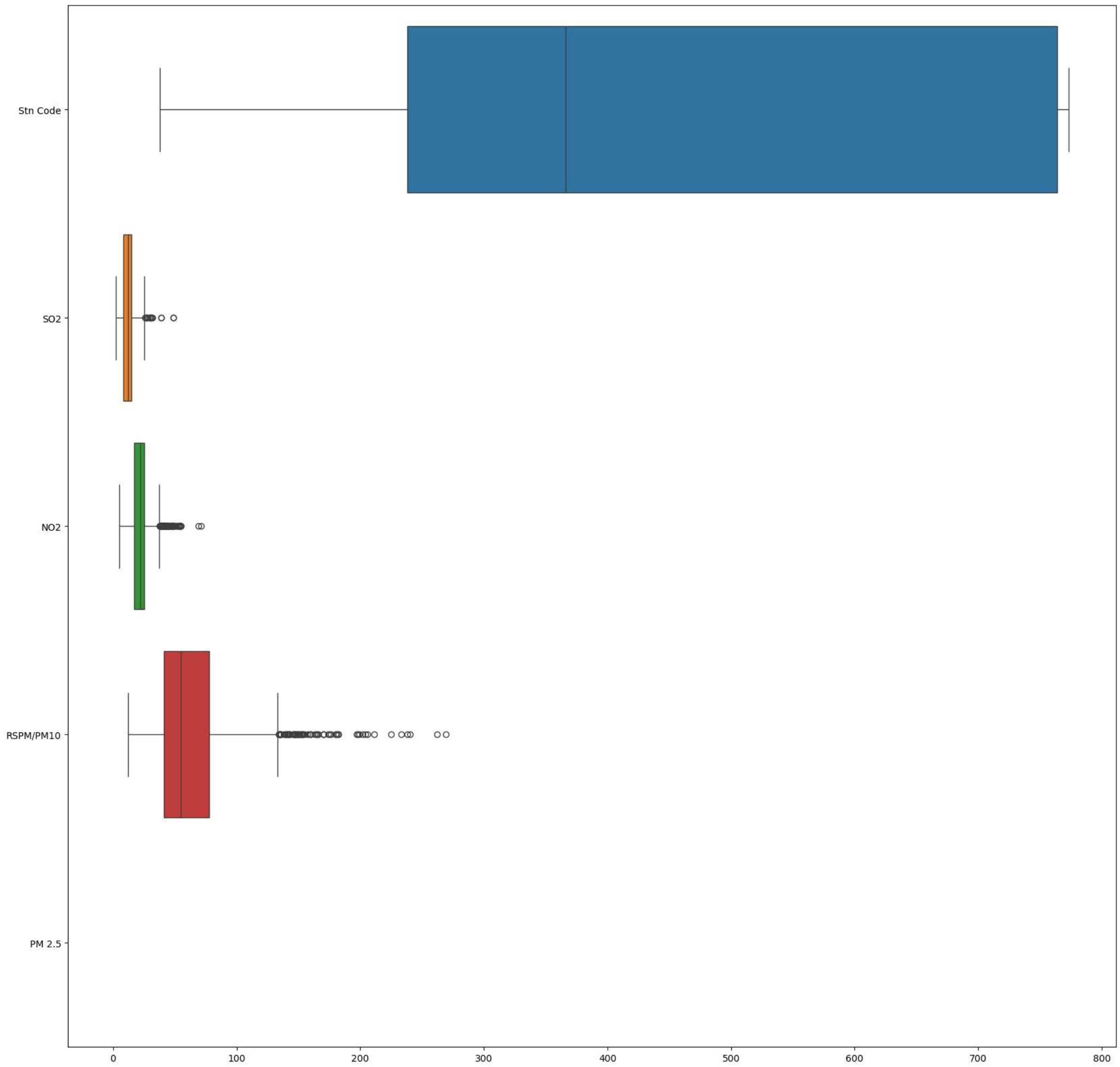
In [67]:
```
imputer = KNNImputer(n_neighbors=3)
data[["SO2","NO2","RSPM/PM10"]]=imputer.fit_transform(data[["SO2","NO2","RSPM/PM10"
```

In [68]:
```
print(data.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2879 entries, 0 to 2878
Data columns (total 10 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   Stn Code                        2879 non-null   int64  
 1   Sampling Date                   2879 non-null   object 
 2   State                           2879 non-null   object 
 3   City/Town/Village/Area          2879 non-null   object 
 4   Location of Monitoring Station  2879 non-null   object 
 5   Agency                          2879 non-null   object 
 6   Type of Location                2879 non-null   object 
 7   SO2                             2879 non-null   float64
 8   NO2                             2879 non-null   float64
 9   RSPM/PM10                       2879 non-null   float64
dtypes: float64(3), int64(1), object(6)
memory usage: 225.1+ KB
None
```

In [41]:

```
plt.figure(figsize=(20,20))
sns.boxplot(data,orient='h')
plt.show()
```



In [71]:
```
def handle_outlier(data):
    mean = np.mean(data)
    sd=np.std(data)
    max= mean+3*sd
    min= mean-3*sd
    data[data<min]=min
    data[data>max]=max
    return data
```

In [85]:
```
data["SO2"]=handle_outlier(data["SO2"])
handle_outlier(data["NO2"])
handle_outlier(data["RSPM/PM10"])
```

```
C:\Users\LAB2_61\AppData\Local\Temp\ipykernel_10600\2769896553.py:6: SettingWithCopy
Warning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/u
ser_guide/indexing.html#returning-a-view-versus-a-copy
  data[data<min]=min
C:\Users\LAB2_61\AppData\Local\Temp\ipykernel_10600\2769896553.py:7: SettingWithCopy
Warning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/u
ser_guide/indexing.html#returning-a-view-versus-a-copy
  data[data>max]=max
C:\Users\LAB2_61\AppData\Local\Temp\ipykernel_10600\2769896553.py:6: SettingWithCopy
Warning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/u
ser_guide/indexing.html#returning-a-view-versus-a-copy
  data[data<min]=min
C:\Users\LAB2_61\AppData\Local\Temp\ipykernel_10600\2769896553.py:7: SettingWithCopy
Warning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/u
ser_guide/indexing.html#returning-a-view-versus-a-copy
  data[data>max]=max
C:\Users\LAB2_61\AppData\Local\Temp\ipykernel_10600\2769896553.py:6: SettingWithCopy
Warning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/u
ser_guide/indexing.html#returning-a-view-versus-a-copy
  data[data<min]=min
C:\Users\LAB2_61\AppData\Local\Temp\ipykernel_10600\2769896553.py:7: SettingWithCopy
Warning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/u
ser_guide/indexing.html#returning-a-view-versus-a-copy
  data[data>max]=max
```

```
Out[85]: 0        55.0
         1        45.0
         2        50.0
         3        46.0
         4        42.0
                  ...
         2874    102.0
         2875     91.0
         2876    100.0
         2877     95.0
         2878     94.0
         Name: RSPM/PM10, Length: 2879, dtype: float64
```

In [ ]: