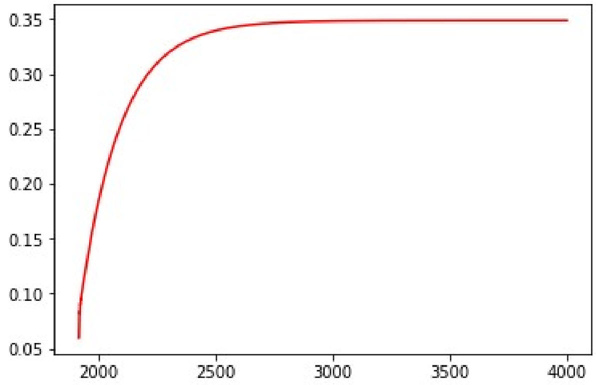
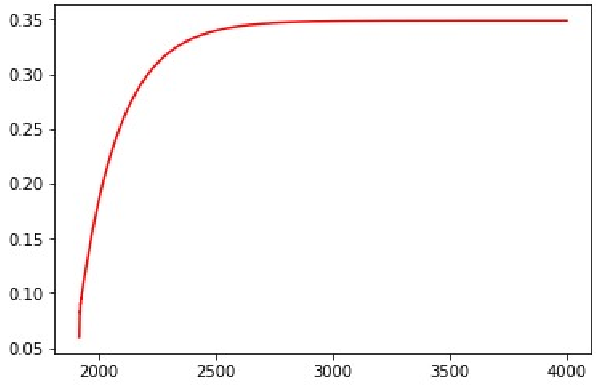
```
  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\dates.py", line 1230, in
__call__
    self.refresh()

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\dates.py", line 1250, in
refresh
    dmin, dmax = self.viewlim_to_dt()

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\dates.py", line 1001, in
viewlim_to_dt
    .format(vmin))

ValueError: view limit minimum -34805.54947916667 is less than 1 and is an invalid
Matplotlib date value. This often happens if you pass a non-datetime value to an axis that
has datetime units

Traceback (most recent call last):

  File "C:\Users\admin\Anaconda3\lib\site-packages\IPython\core\formatters.py", line 341,
in __call__
    return printer(obj)

  File "C:\Users\admin\Anaconda3\lib\site-packages\IPython\core\pylabtools.py", line 244,
in <lambda>
    png_formatter.for_type(Figure, lambda fig: print_figure(fig, 'png', **kwargs))

  File "C:\Users\admin\Anaconda3\lib\site-packages\IPython\core\pylabtools.py", line 128,
in print_figure
    fig.canvas.print_figure(bytes_io, **kw)

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\backend_bases.py", line
2049, in print_figure
    **kwargs)

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\backends\backend_agg.py",
line 510, in print_png
    FigureCanvasAgg.draw(self)

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\backends\backend_agg.py",
line 402, in draw
    self.figure.draw(self.renderer)

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\artist.py", line 50, in
draw_wrapper
    return draw(artist, renderer, *args, **kwargs)

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\figure.py", line 1649, in
draw
    renderer, self, artists, self.suppressComposite)

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\image.py", line 138, in
_draw_list_compositing_images
    a.draw(renderer)

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\artist.py", line 50, in
```

```
draw_wrapper
    return draw(artist, renderer, *args, **kwargs)

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\axes\_base.py", line 2628,
in draw
    mimage._draw_list_compositing_images(renderer, self, artists)

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\image.py", line 138, in
_draw_list_compositing_images
    a.draw(renderer)

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\artist.py", line 50, in
draw_wrapper
    return draw(artist, renderer, *args, **kwargs)

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\axis.py", line 1185, in draw
    ticks_to_draw = self._update_ticks(renderer)

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\axis.py", line 1023, in
_update_ticks
    tick_tups = list(self.iter_ticks())  # iter_ticks calls the locator

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\axis.py", line 967, in
iter_ticks
    majorLocs = self.major.locator()

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\dates.py", line 1230, in
__call__
    self.refresh()

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\dates.py", line 1250, in
refresh
    dmin, dmax = self.viewlim_to_dt()

  File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\dates.py", line 1001, in
viewlim_to_dt
    .format(vmin))

ValueError: view limit minimum -34805.54947916667 is less than 1 and is an invalid
Matplotlib date value. This often happens if you pass a non-datetime value to an axis that
has datetime units

<Figure size 1584x720 with 1 Axes>

In [147]:

In [147]:

In [147]: forecast = sc_arima_fit.predict(start=1919, end=4000)
     ...:
C:\Users\admin\Anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.py:531:
ValueWarning: No supported index is available. Prediction results will be given with an
integer index beginning at `start`.
  ValueWarning)
```

```
In [148]: mp.plot(forecast,color='red')
     ...:
Out[148]: [<matplotlib.lines.Line2D at 0x22891f70320>]
```

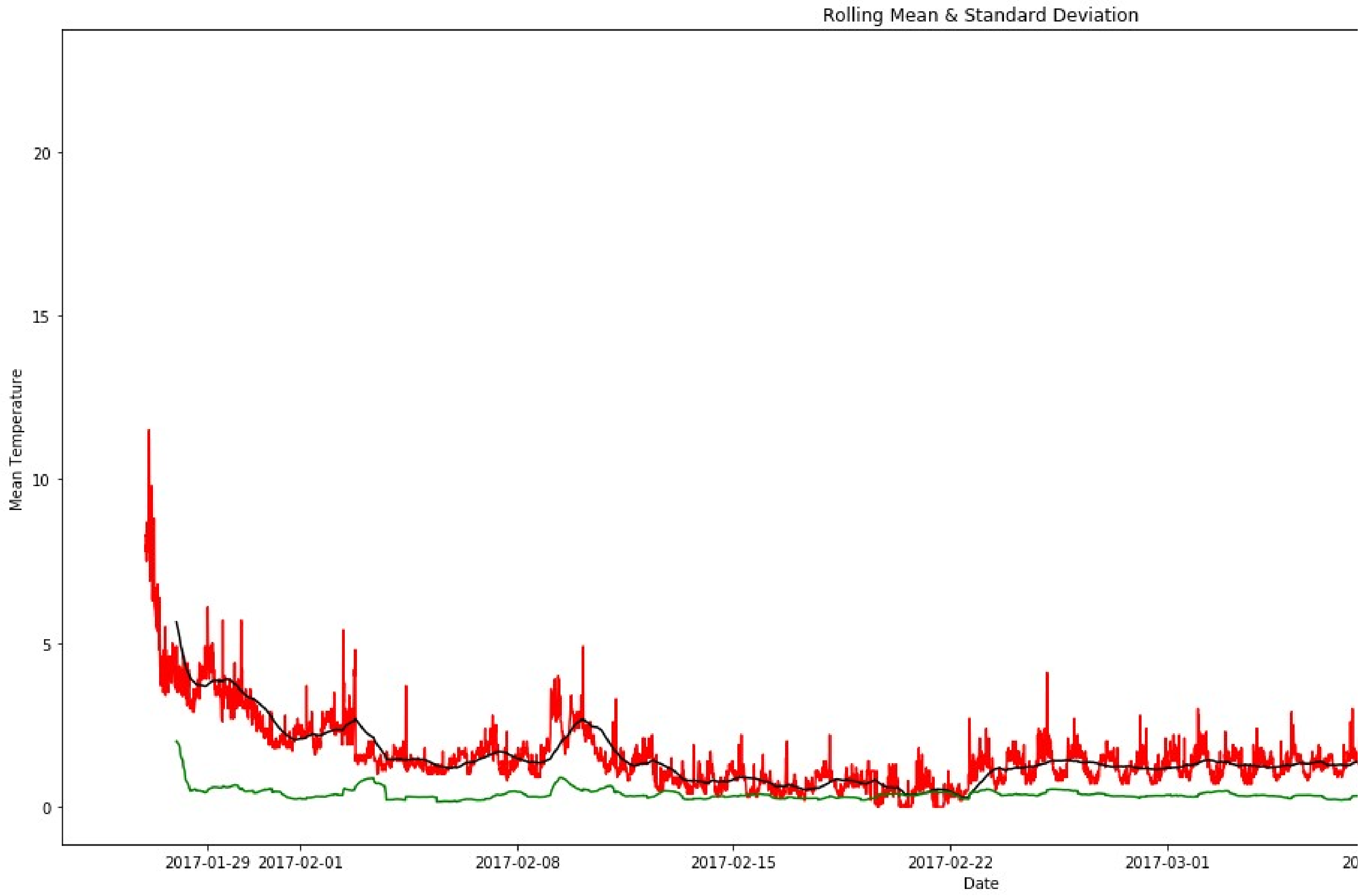


```
In [149]: predictions = sc_arima_fit.predict(start=1919, end=3000)
     ...:
C:\Users\admin\Anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.py:531:
ValueWarning: No supported index is available. Prediction results will be given with an
integer index beginning at `start`.
  ValueWarning)

In [150]: mp.plot(forecast,color='red')
     ...:
Out[150]: [<matplotlib.lines.Line2D at 0x22891fc7c18>]
```



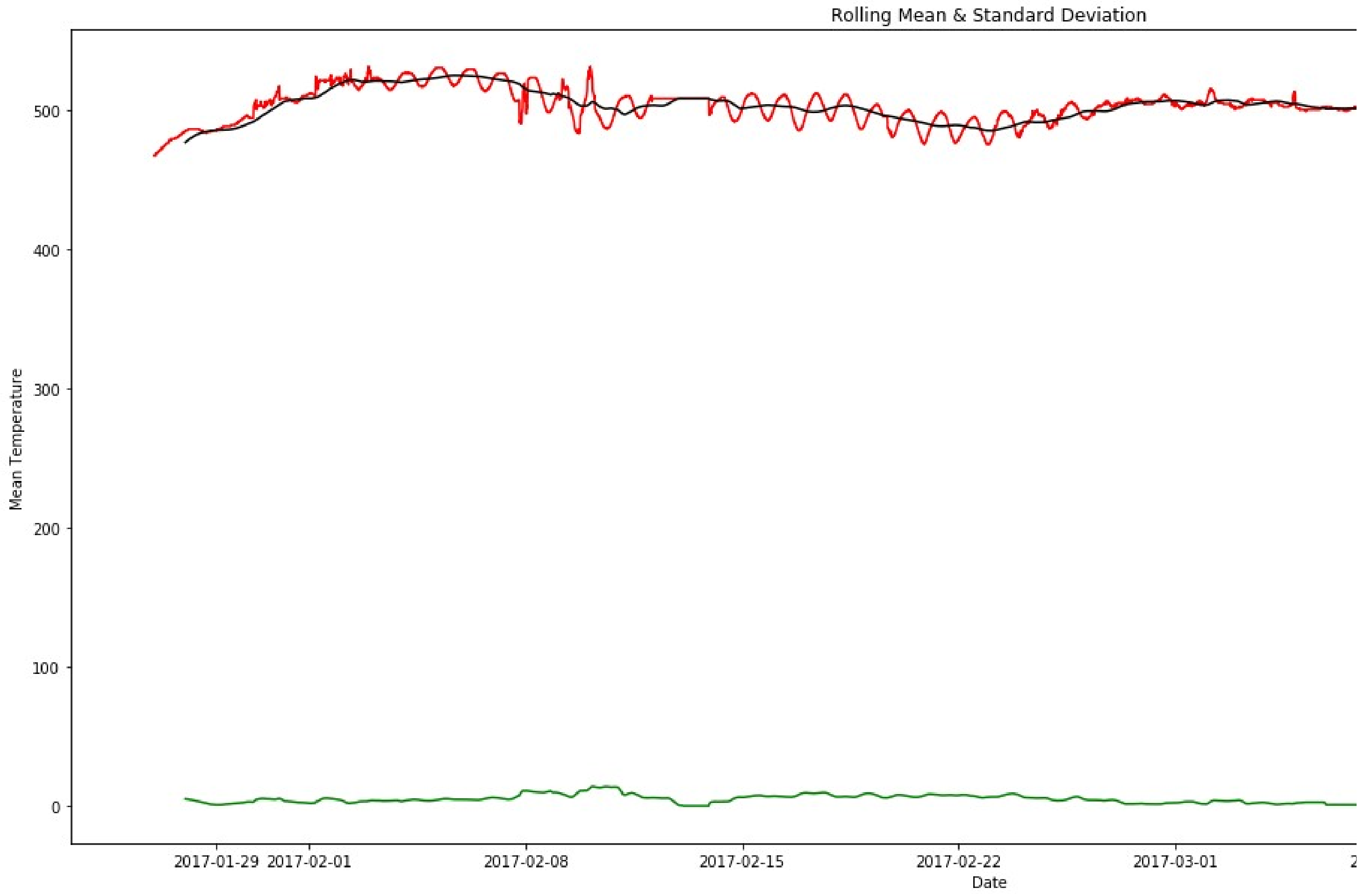
```
In [151]: def check_mean_std(ts):
     ...:     #Rolling statistics
     ...:     rolmean = pd.rolling_mean(ts, window=96)
     ...:     rolstd = pd.rolling_std(ts, window=96)
     ...:     plt.figure(figsize=(22,10))
     ...:     orig = plt.plot(ts, color='red',label='Original')
     ...:     mean = plt.plot(rolmean, color='black', label='Rolling Mean')
     ...:     std = plt.plot(rolstd, color='green', label = 'Rolling Std')
     ...:     plt.xlabel("Date")
     ...:     plt.ylabel("Mean Temperature")
     ...:     plt.title('Rolling Mean & Standard Deviation')
     ...:     plt.legend()
     ...:     plt.show()

In [152]: check_adfuller(turb['Turb(FNU)'])
```

```
     ...: check_mean_std(turb['Turb(FNU)'])
Results of Dickey Fuller Test:
--------For a stationary time series Test statistic is less than critical values
Test Statistic                   -5.039398
p-value                           0.000019
#Lags Used                       31.000000
Number of Observations Used    5139.000000
Critical Value (1%)              -3.431623
Critical Value (5%)              -2.862103
Critical Value (10%)             -2.567069
dtype: float64
Traceback (most recent call last):

  File "<ipython-input-152-acecb6fd4374>", line 2, in <module>
    check_mean_std(turb['Turb(FNU)'])

  File "<ipython-input-151-6532399e584e>", line 3, in check_mean_std
    rolmean = pd.rolling_mean(ts, window=96)

AttributeError: module 'pandas' has no attribute 'rolling_mean'


In [153]:

In [153]: def check_mean_std(ts):
     ...:     #Rolling statistics
     ...:     rolmean = ts.rolling(window=96).mean()
     ...:     rolstd = ts.rolling(window=96).std()
     ...:     plt.figure(figsize=(22,10))
     ...:     orig = plt.plot(ts, color='red',label='Original')
     ...:     mean = plt.plot(rolmean, color='black', label='Rolling Mean')
     ...:     std = plt.plot(rolstd, color='green', label = 'Rolling Std')
     ...:     plt.xlabel("Date")
     ...:     plt.ylabel("Mean Temperature")
     ...:     plt.title('Rolling Mean & Standard Deviation')
     ...:     plt.legend()
     ...:     plt.show()

In [154]: check_adfuller(turb['Turb(FNU)'])
     ...: check_mean_std(turb['Turb(FNU)'])
Results of Dickey Fuller Test:
--------For a stationary time series Test statistic is less than critical values
Test Statistic                   -5.039398
p-value                           0.000019
#Lags Used                       31.000000
Number of Observations Used    5139.000000
Critical Value (1%)              -3.431623
Critical Value (5%)              -2.862103
Critical Value (10%)             -2.567069
dtype: float64
```

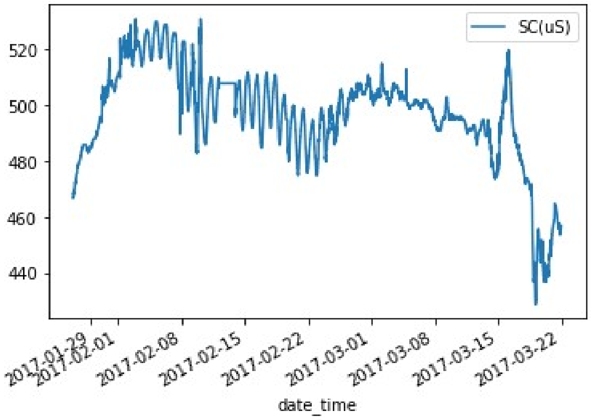


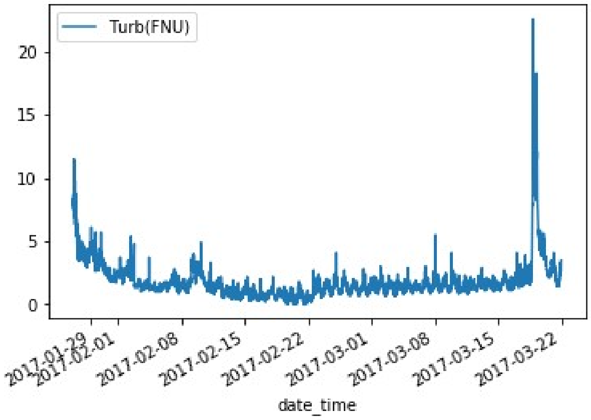
```
In [155]: check_adfuller(turb['SC(uS)'])
     ...: check_mean_std(turb['SC(uS)'])
Traceback (most recent call last):

  File "<ipython-input-155-e6dace5b0fbb>", line 1, in <module>
    check_adfuller(turb['SC(uS)'])

  File "C:\Users\admin\Anaconda3\lib\site-packages\pandas\core\frame.py", line 2688, in
__getitem__
    return self._getitem_column(key)

  File "C:\Users\admin\Anaconda3\lib\site-packages\pandas\core\frame.py", line 2695, in
_getitem_column
    return self._get_item_cache(key)

  File "C:\Users\admin\Anaconda3\lib\site-packages\pandas\core\generic.py", line 2489, in
_get_item_cache
    values = self._data.get(item)

  File "C:\Users\admin\Anaconda3\lib\site-packages\pandas\core\internals.py", line 4115,
in get
    loc = self.items.get_loc(item)

  File "C:\Users\admin\Anaconda3\lib\site-packages\pandas\core\indexes\base.py", line
```

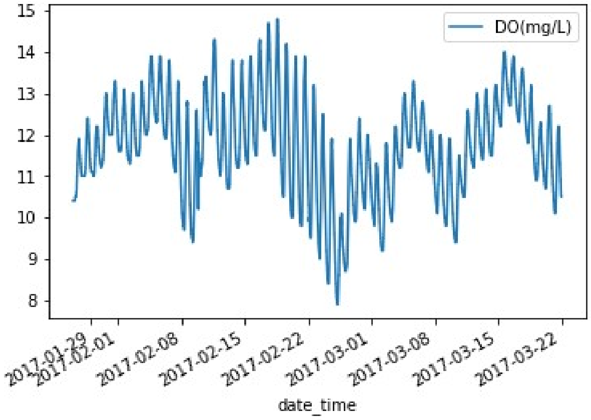
```
3080, in get_loc
    return self._engine.get_loc(self._maybe_cast_indexer(key))

  File "pandas\_libs\index.pyx", line 140, in pandas._libs.index.IndexEngine.get_loc

  File "pandas\_libs\index.pyx", line 162, in pandas._libs.index.IndexEngine.get_loc

  File "pandas\_libs\hashtable_class_helper.pxi", line 1492, in
pandas._libs.hashtable.PyObjectHashTable.get_item

  File "pandas\_libs\hashtable_class_helper.pxi", line 1500, in
pandas._libs.hashtable.PyObjectHashTable.get_item

KeyError: 'SC(uS)'


In [156]:

In [156]: check_adfuller(sc['SC(uS)'])
     ...: check_mean_std(sc['SC(uS)'])
Results of Dickey Fuller Test:
--------For a stationary time series Test statistic is less than critical values
Test Statistic                   -3.268401
p-value                           0.016351
#Lags Used                       27.000000
Number of Observations Used    5143.000000
Critical Value (1%)              -3.431622
Critical Value (5%)              -2.862102
Critical Value (10%)             -2.567069
dtype: float64
```

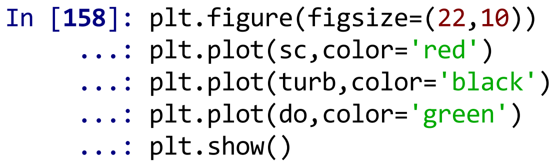


```
In [157]: plt.figure(figsize=(22,10))
     ...: sc.plot()
     ...: turb.plot()
     ...: do.plot()
Out[157]: <matplotlib.axes._subplots.AxesSubplot at 0x2288f831ef0><Figure size 1584x720
with 0 Axes>
```

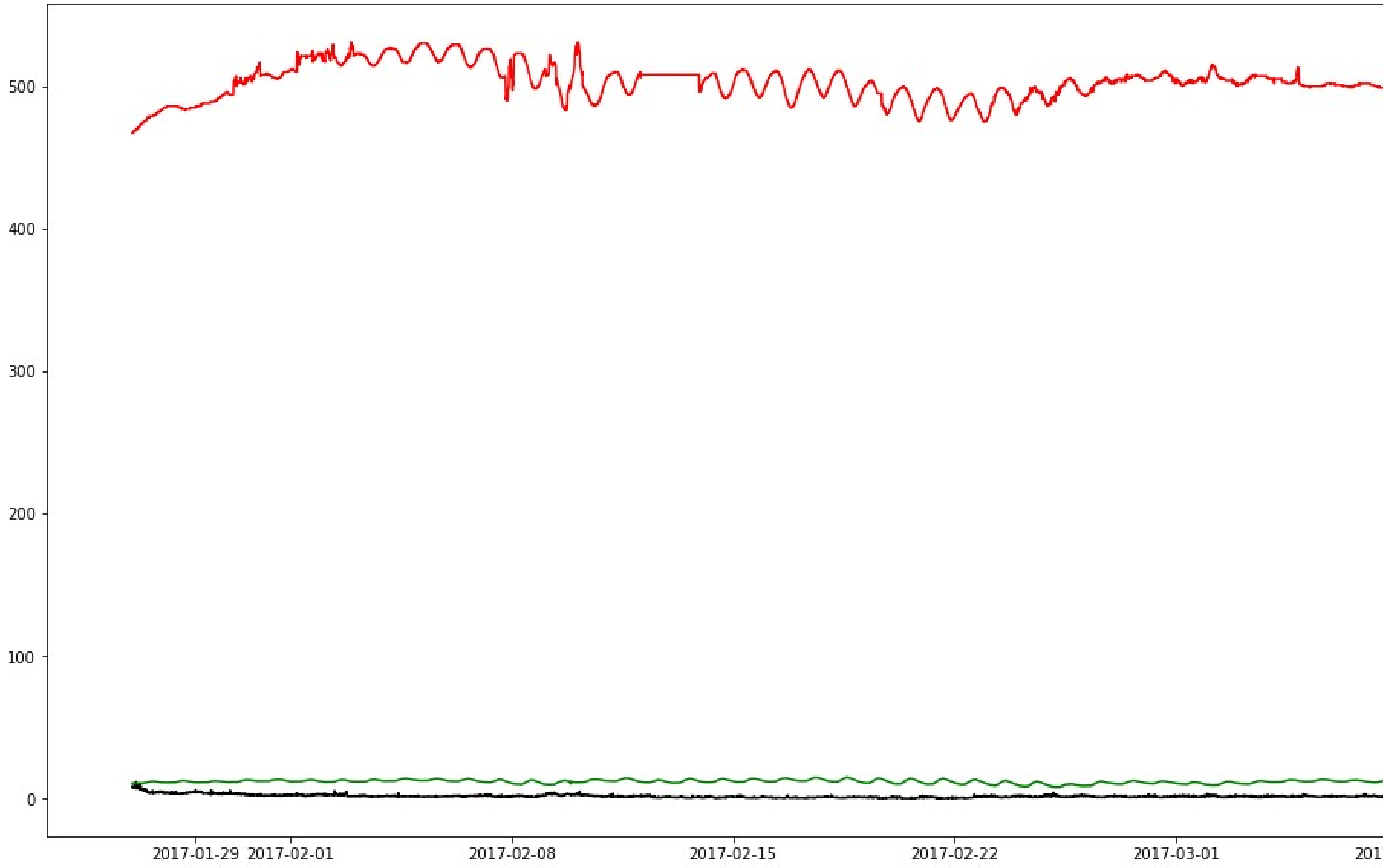






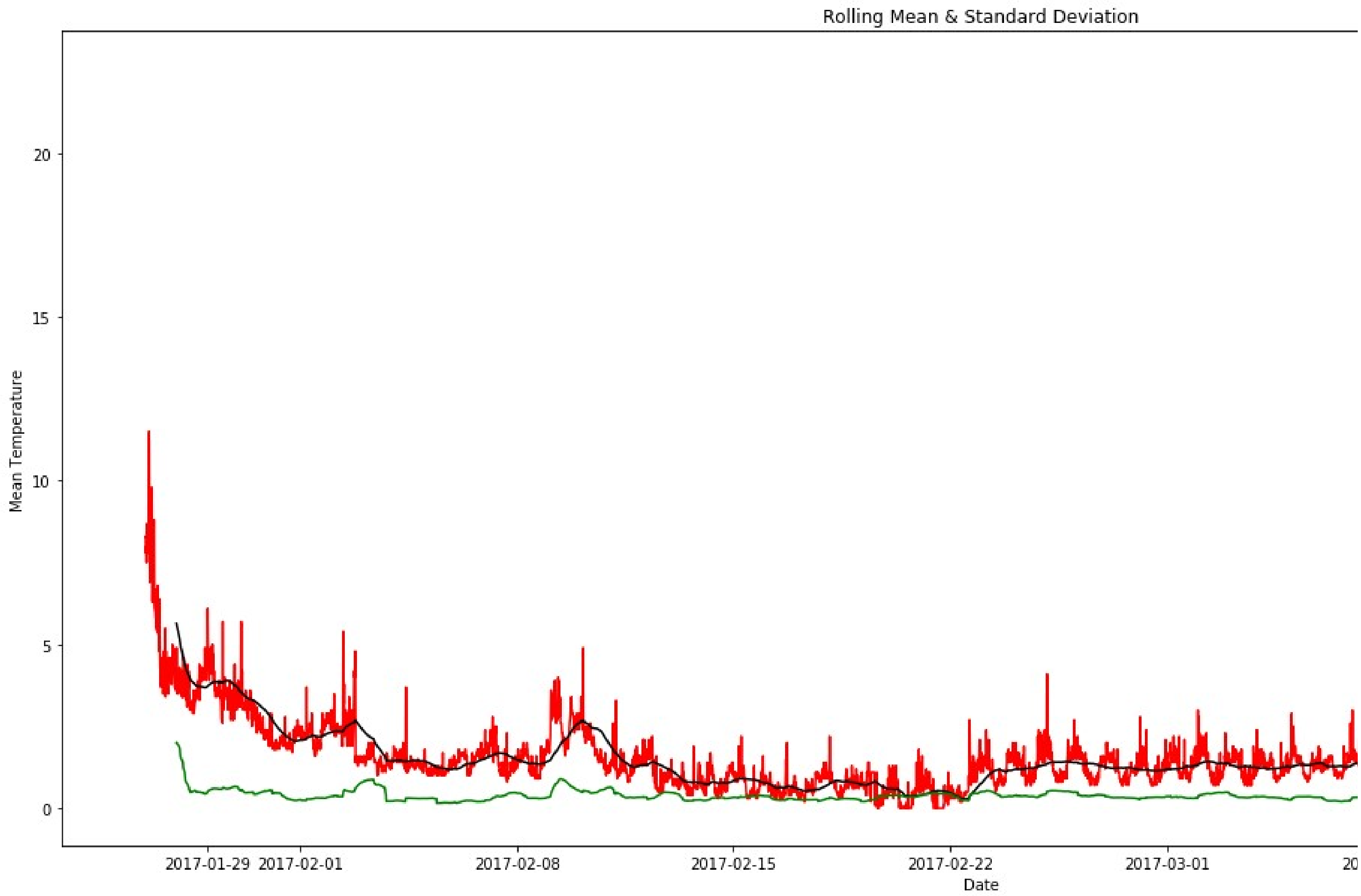


```
In [158]: plt.figure(figsize=(22,10))
     ...: plt.plot(sc,color='red')
     ...: plt.plot(turb,color='black')
     ...: plt.plot(do,color='green')
     ...: plt.show()
```

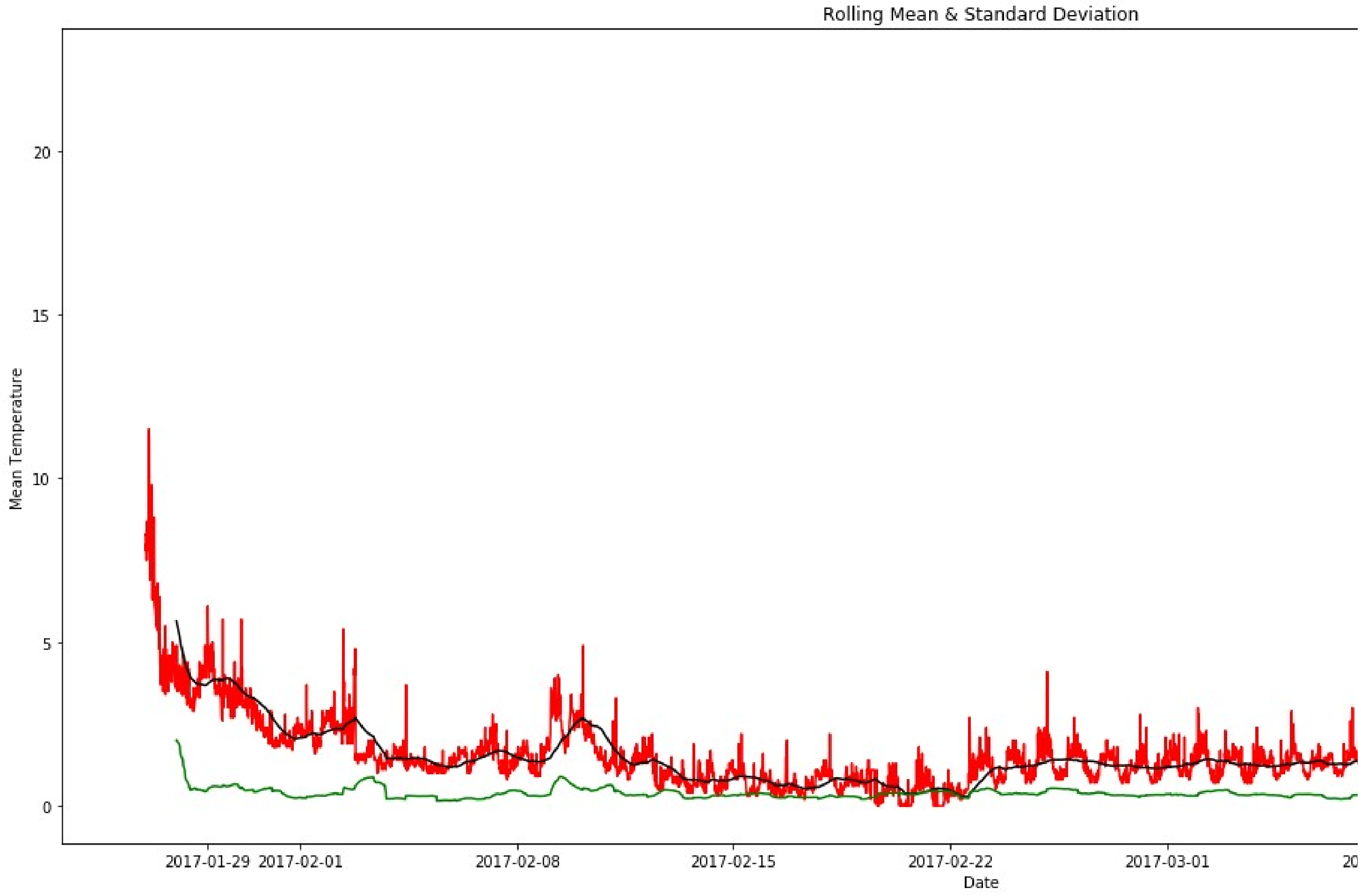


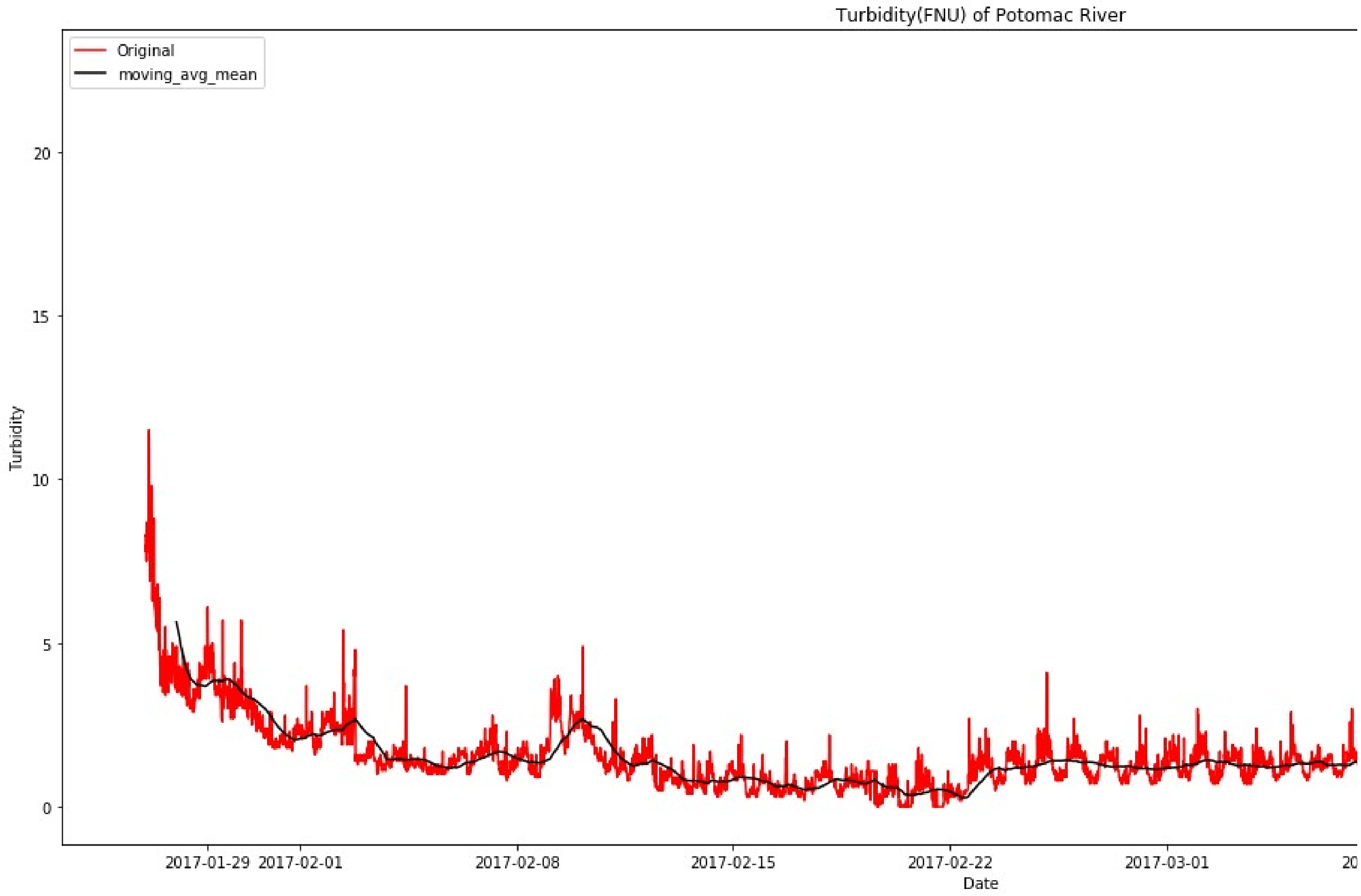
```
In [159]: check_adfuller(turb['Turb(FNU)'])
     ...: check_mean_std(turb['Turb(FNU)'])
     ...:
     ...: # Moving average method for turbidity
     ...: window_size = 96
     ...: moving_avg = pd.rolling_mean(turb,window_size)
     ...: plt.figure(figsize=(22,10))
     ...: plt.plot(turb, color = "red",label = "Original")
     ...: plt.plot(moving_avg, color='black', label = "moving_avg_mean")
     ...: plt.title("Turbidity(FNU) of Potomac River")
     ...: plt.xlabel("Date")
     ...: plt.ylabel("Turbidity")
     ...: plt.legend()
     ...: plt.show()
     ...: turb_moving_avg_diff = turb - turb_ma
     ...: turb_moving_avg_diff.dropna(inplace=True) # first 6 is nan value due to window
size
     ...:
     ...: # check stationary: mean, variance(std)and adfuller test
     ...: check_mean_std(ts_moving_avg_diff)
     ...: check_adfuller(ts_moving_avg_diff.MeanTemp)
Results of Dickey Fuller Test:
--------For a stationary time series Test statistic is less than critical values
Test Statistic                 -5.039398
p-value                         0.000019
```

```
#Lags Used                        31.000000
Number of Observations Used     5139.000000
Critical Value (1%)               -3.431623
Critical Value (5%)               -2.862103
Critical Value (10%)              -2.567069
dtype: float64
```

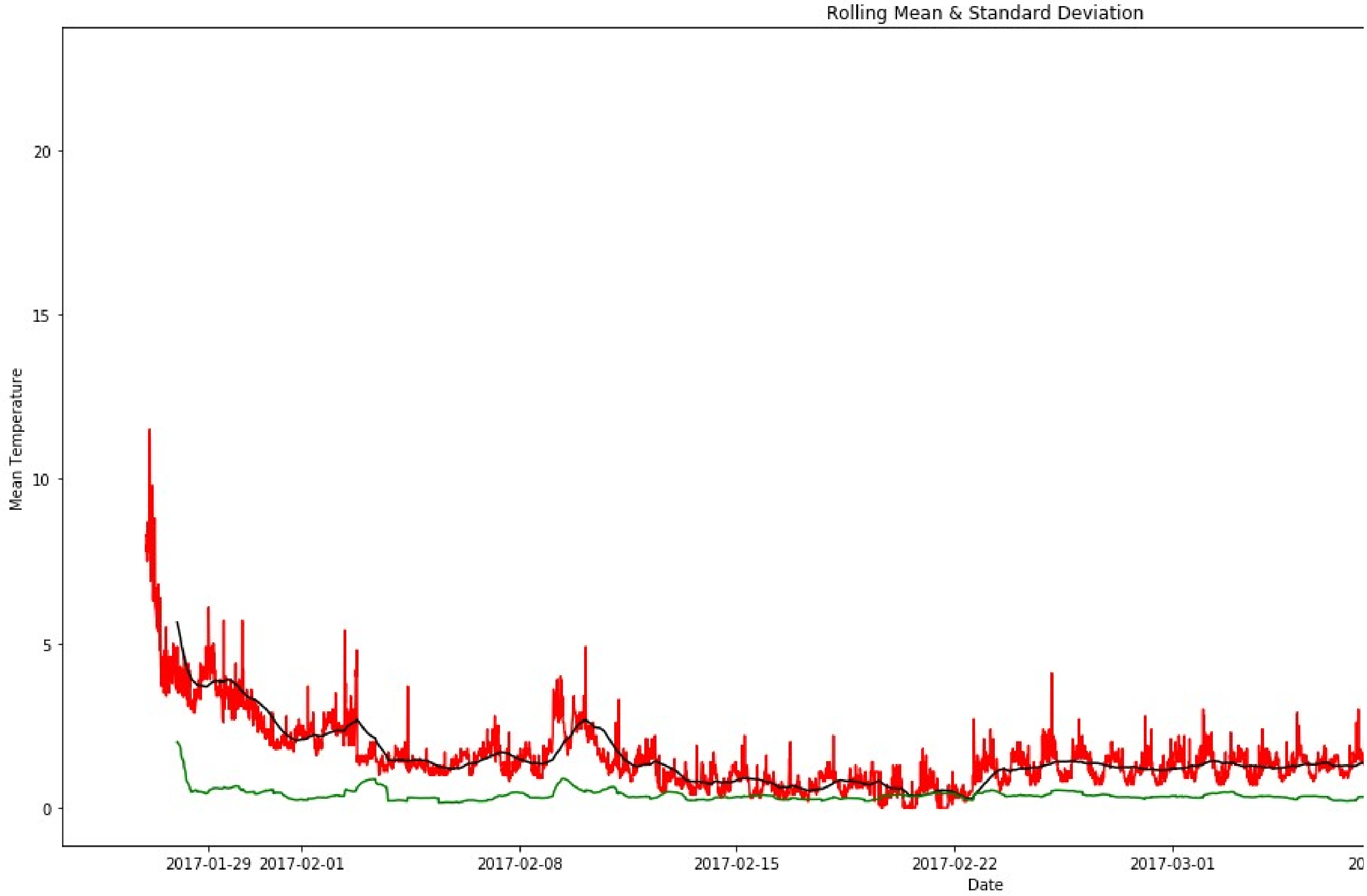


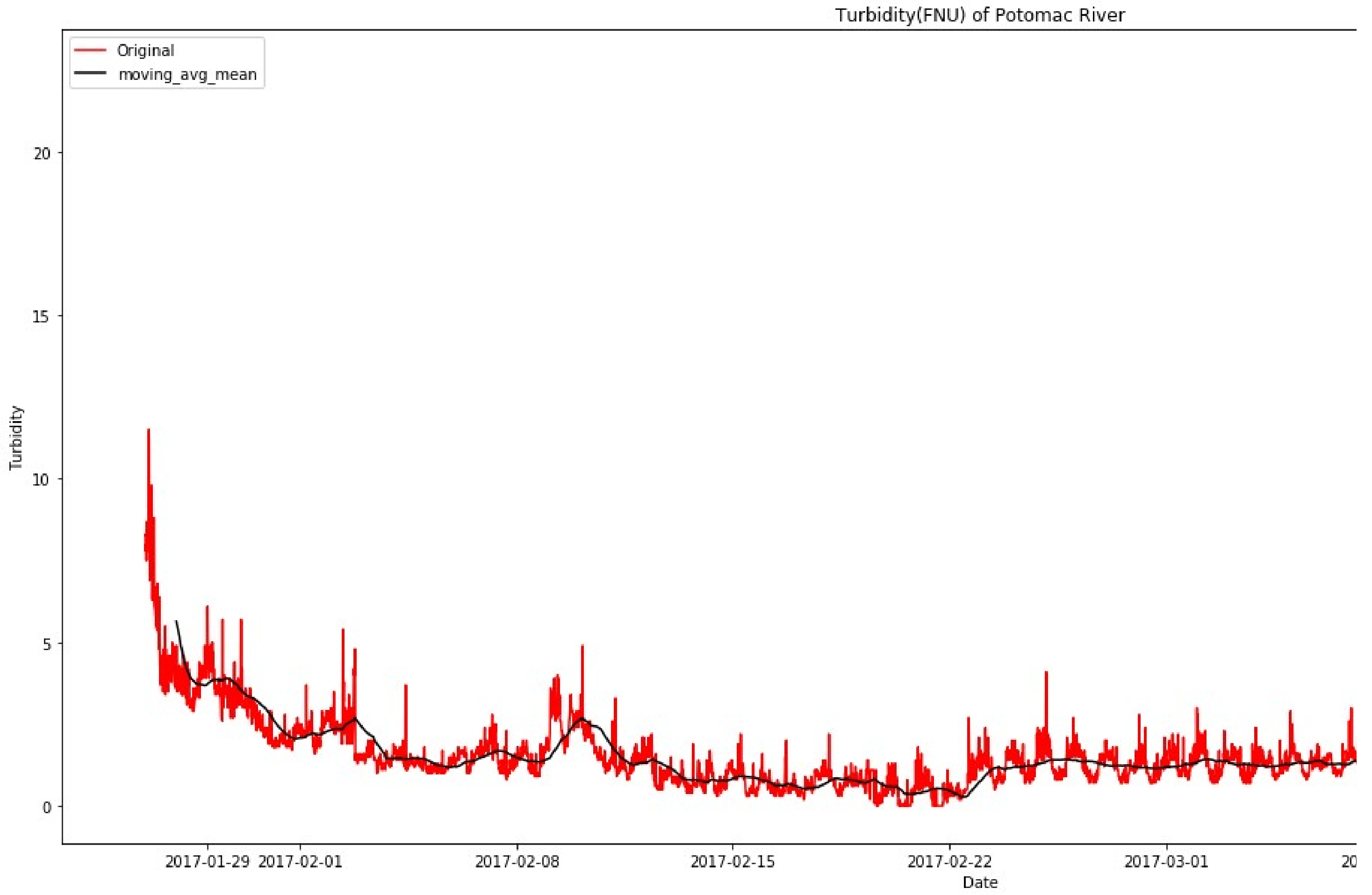
```
Traceback (most recent call last):

  File "<ipython-input-159-b489c42f8417>", line 6, in <module>
    moving_avg = pd.rolling_mean(turb,window_size)

AttributeError: module 'pandas' has no attribute 'rolling_mean'


In [160]:

In [160]: check_adfuller(turb['Turb(FNU)'])
     ...: check_mean_std(turb['Turb(FNU)'])
     ...:
     ...: # Moving average method for turbidity
     ...: window_size = 96
     ...: moving_avg = turb.rolling(window=window_size).mean()
     ...: plt.figure(figsize=(22,10))
     ...: plt.plot(turb, color = "red",label = "Original")
     ...: plt.plot(moving_avg, color='black', label = "moving_avg_mean")
     ...: plt.title("Turbidity(FNU) of Potomac River")
     ...: plt.xlabel("Date")
     ...: plt.ylabel("Turbidity")
     ...: plt.legend()
     ...: plt.show()
     ...: turb_moving_avg_diff = turb - turb_ma
     ...: turb_moving_avg_diff.dropna(inplace=True) # first 6 is nan value due to window
size
     ...:
     ...: # check stationary: mean, variance(std)and adfuller test
     ...: check_mean_std(ts_moving_avg_diff)
     ...: check_adfuller(ts_moving_avg_diff.MeanTemp)
Results of Dickey Fuller Test:
--------For a stationary time series Test statistic is less than critical values
Test Statistic                   -5.039398
p-value                           0.000019
#Lags Used                       31.000000
Number of Observations Used    5139.000000
Critical Value (1%)              -3.431623
Critical Value (5%)              -2.862103
Critical Value (10%)             -2.567069
dtype: float64
```

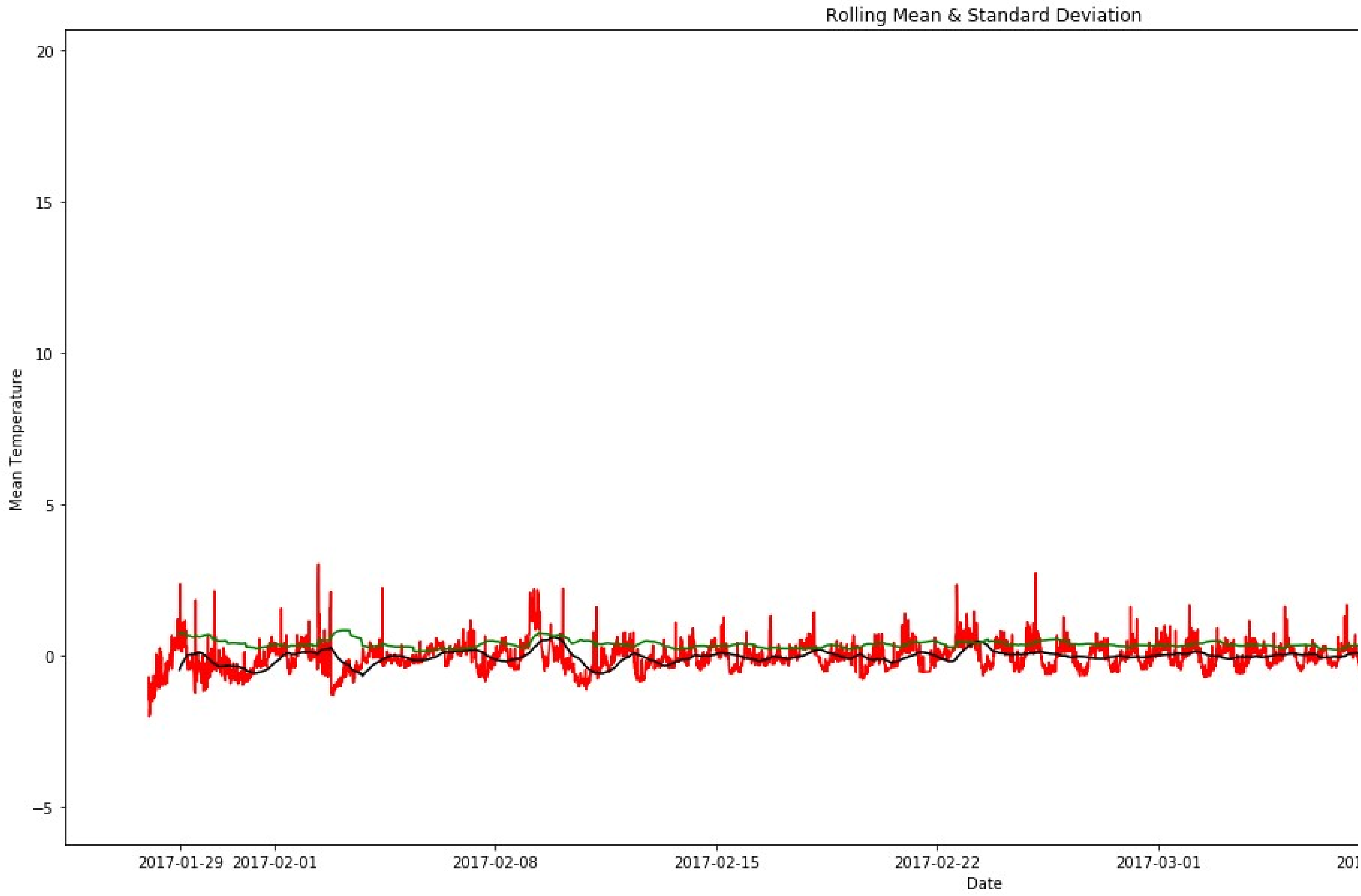




```
Traceback (most recent call last):

  File "<ipython-input-160-3c8d8659b590>", line 15, in <module>
    turb_moving_avg_diff = turb - turb_ma

NameError: name 'turb_ma' is not defined


In [161]:

In [161]: check_adfuller(turb['Turb(FNU)'])
     ...: check_mean_std(turb['Turb(FNU)'])
     ...:
     ...: # Moving average method for turbidity
     ...: window_size = 96
     ...: turb_ma = turb.rolling(window=window_size).mean()
     ...: plt.figure(figsize=(22,10))
     ...: plt.plot(turb, color = "red",label = "Original")
     ...: plt.plot(turb_ma, color='black', label = "moving_avg_mean")
     ...: plt.title("Turbidity(FNU) of Potomac River")
     ...: plt.xlabel("Date")
     ...: plt.ylabel("Turbidity")
     ...: plt.legend()
     ...: plt.show()
     ...: turb_moving_avg_diff = turb - turb_ma
     ...: turb_moving_avg_diff.dropna(inplace=True) # first 6 is nan value due to window size
     ...:
     ...: # check stationary: mean, variance(std)and adfuller test
     ...: check_mean_std(turb_moving_avg_diff)
     ...: check_adfuller(turb_moving_avg_diff['Turb(FNU)'])
Results of Dickey Fuller Test:
--------For a stationary time series Test statistic is less than critical values
Test Statistic                   -5.039398
p-value                           0.000019
#Lags Used                       31.000000
Number of Observations Used    5139.000000
Critical Value (1%)              -3.431623
Critical Value (5%)              -2.862103
Critical Value (10%)             -2.567069
dtype: float64
```

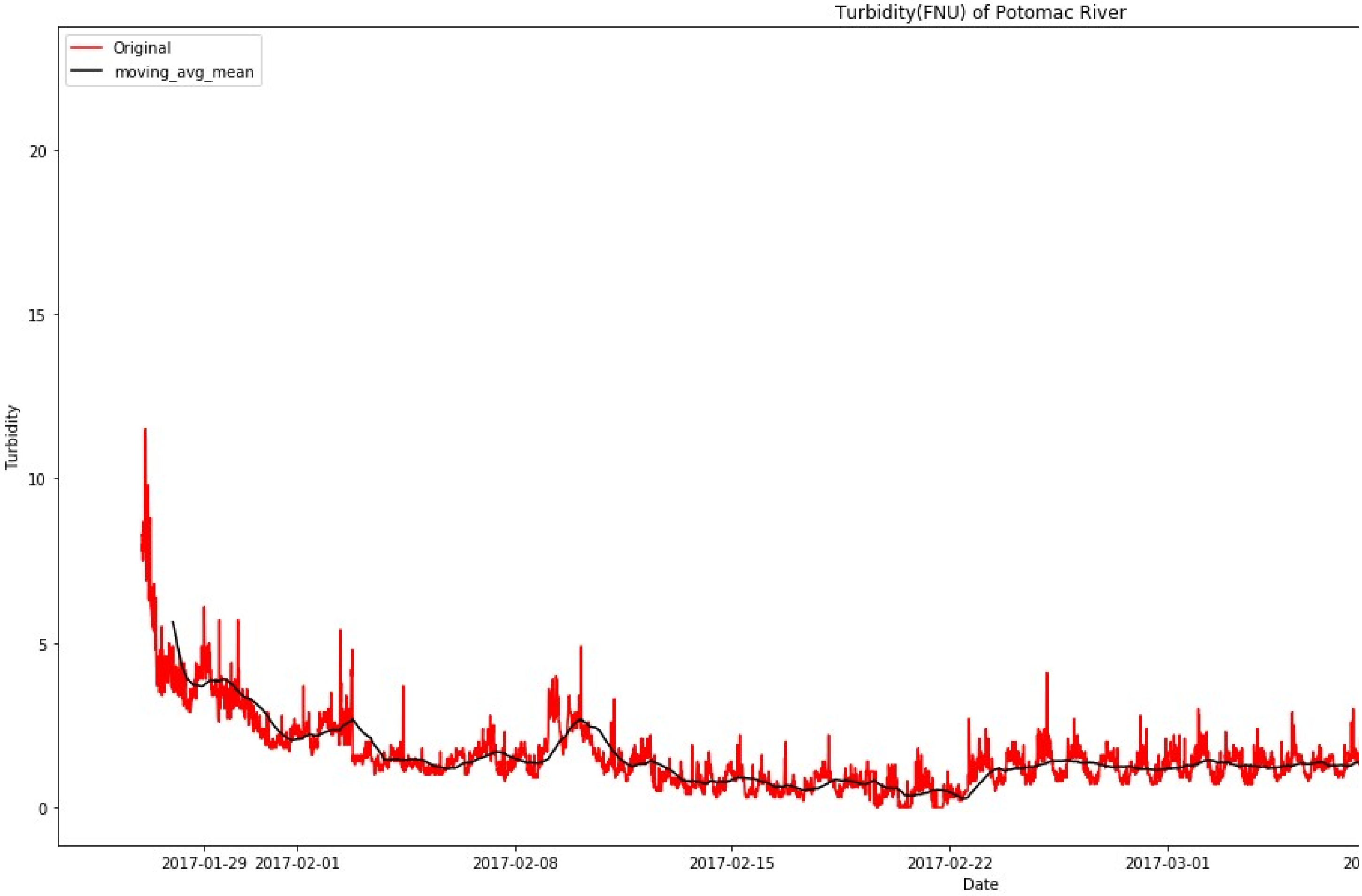


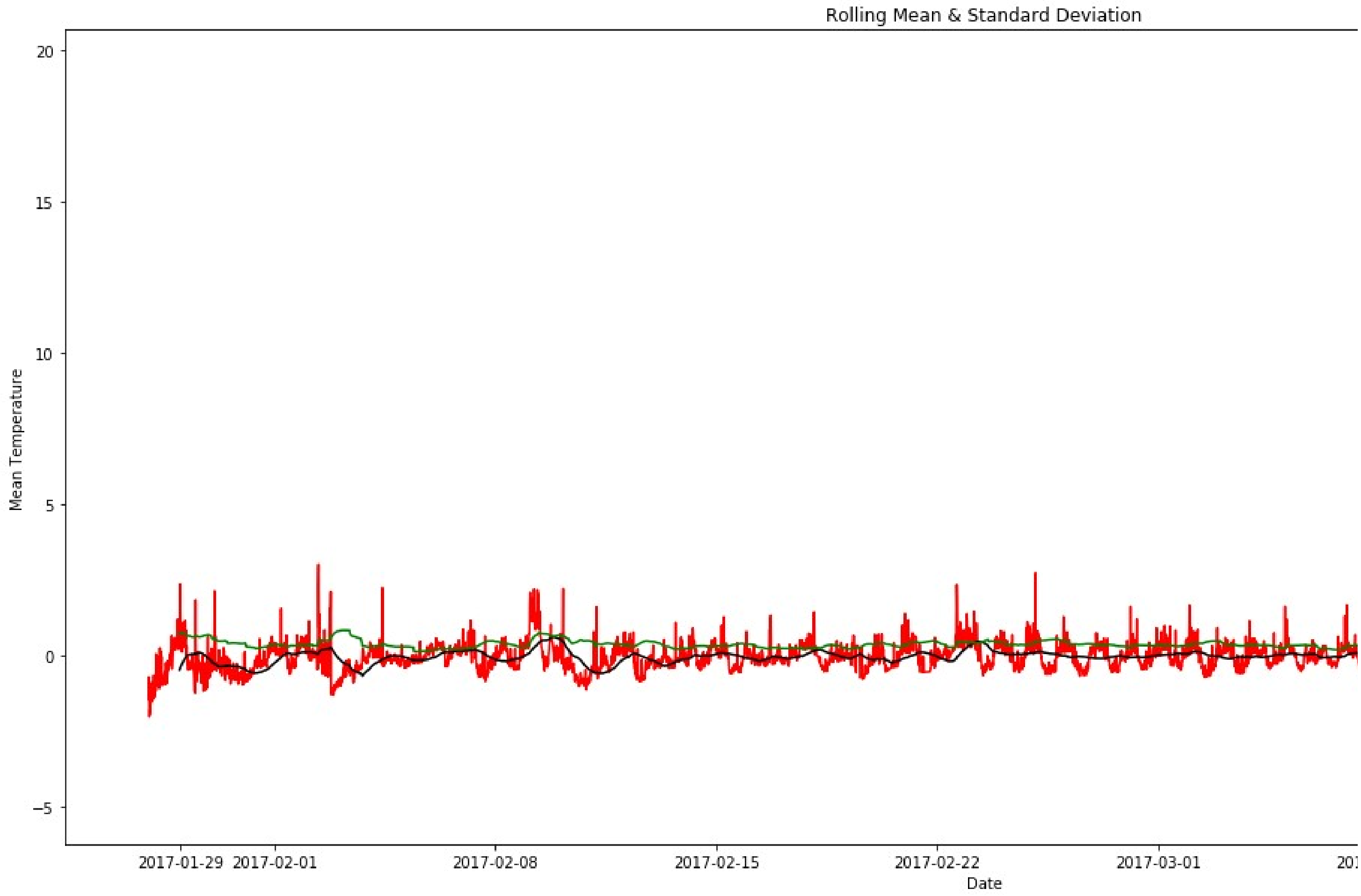

Turbidity(FNU) of Potomac River
Original
moving_avg_mean
Turbidity
20
15
10
5
0
2017-01-29
2017-02-01
2017-02-08
2017-02-15
2017-02-22
2017-03-01
20
Date



```
Results of Dickey Fuller Test:
--------For a stationary time series Test statistic is less than critical values
Test Statistic                -1.156621e+01
p-value                        3.181840e-21
#Lags Used                     3.000000e+00
Number of Observations Used    5.072000e+03
Critical Value (1%)           -3.431640e+00
Critical Value (5%)           -2.862110e+00
Critical Value (10%)          -2.567073e+00
dtype: float64

In [162]: window_size = 96
     ...: turb_ma = turb.rolling(window=window_size).mean()
     ...: plt.figure(figsize=(22,10))
     ...: plt.plot(turb, color = "red",label = "Original")
     ...: plt.plot(turb_ma, color='black', label = "moving_avg_mean")
     ...: plt.title("Turbidity(FNU) of Potomac River")
     ...: plt.xlabel("Date")
     ...: plt.ylabel("Turbidity")
     ...: plt.legend()
     ...: plt.show()
     ...: turb_moving_avg_diff = turb - turb_ma
     ...: turb_moving_avg_diff.dropna(inplace=True) # first 6 is nan value due to window
size
     ...:
```

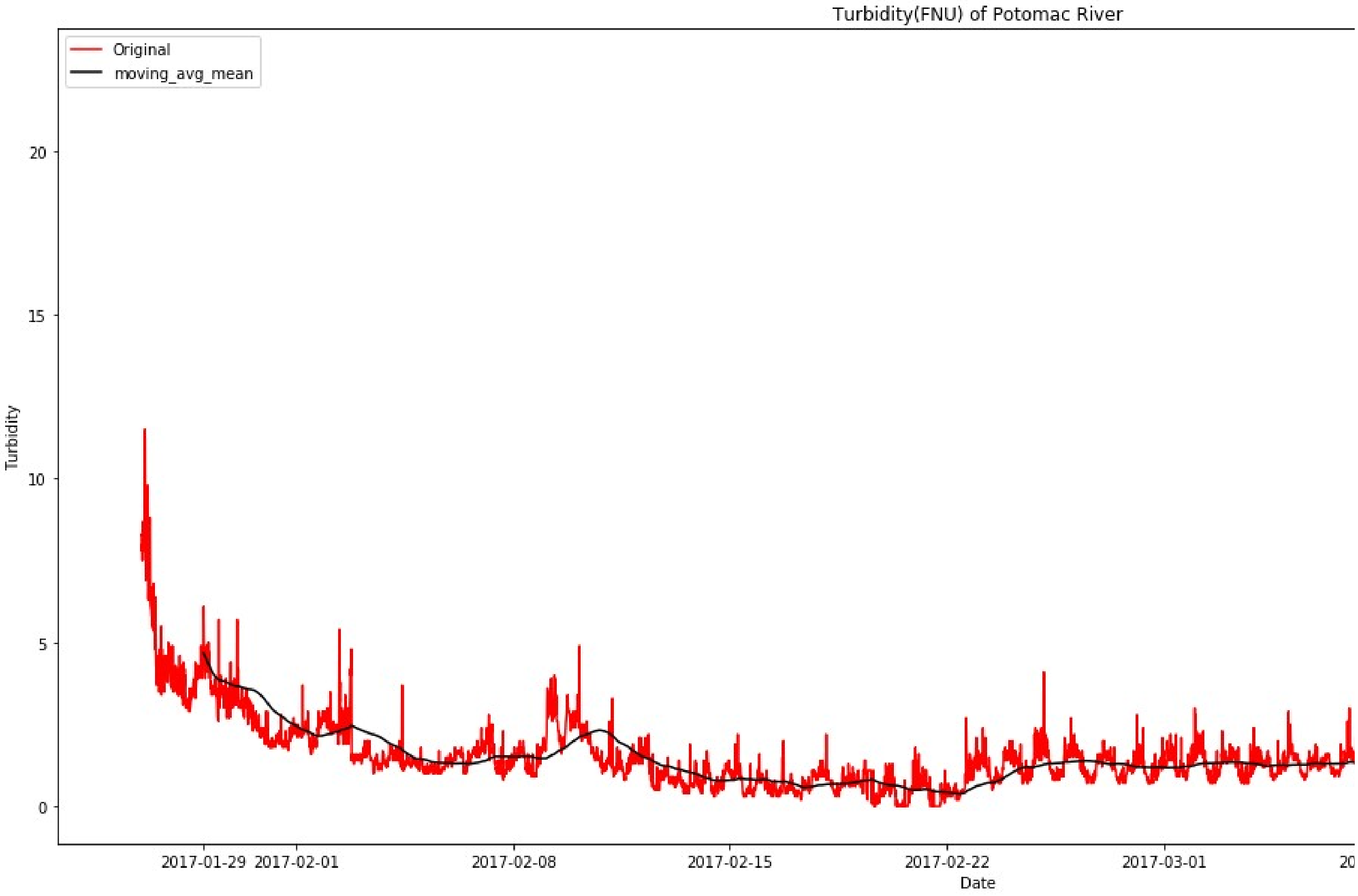
```
...: # check stationary: mean, variance(std)and adfuller test
...: check_mean_std(turb_moving_avg_diff)
...: check_adfuller(turb_moving_avg_diff['Turb(FNU)'])
```

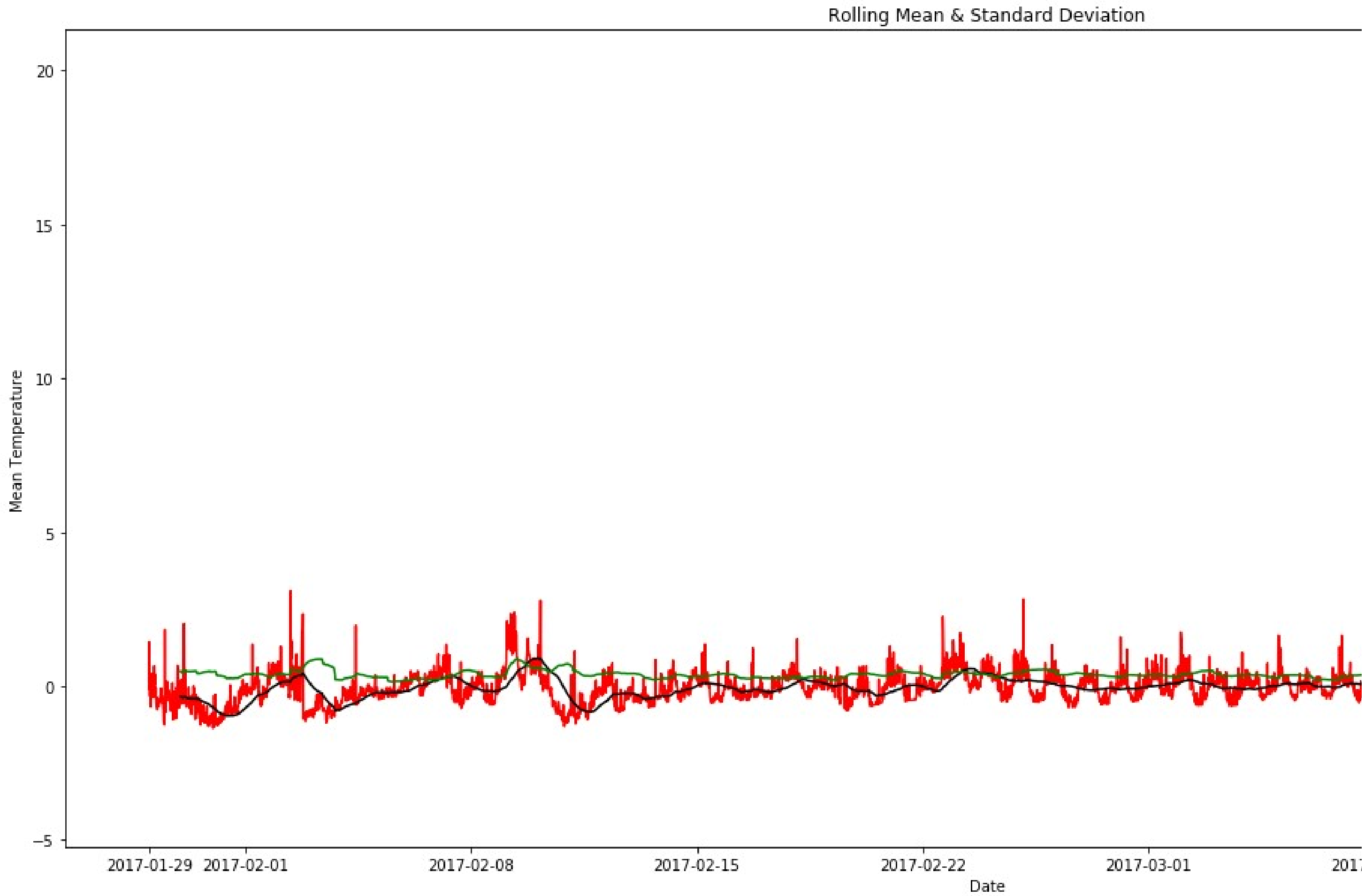




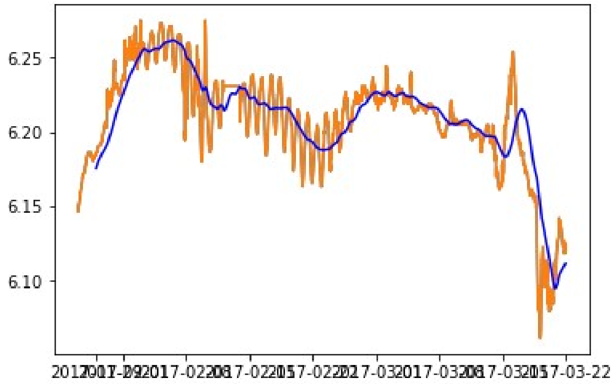
```
Results of Dickey Fuller Test:
--------For a stationary time series Test statistic is less than critical values
Test Statistic                -1.156621e+01
p-value                        3.181840e-21
#Lags Used                     3.000000e+00
Number of Observations Used    5.072000e+03
Critical Value (1%)           -3.431640e+00
Critical Value (5%)           -2.862110e+00
Critical Value (10%)          -2.567073e+00
dtype: float64

In [163]: window_size = 192
     ...: turb_ma = turb.rolling(window=window_size).mean()
     ...: plt.figure(figsize=(22,10))
     ...: plt.plot(turb, color = "red",label = "Original")
     ...: plt.plot(turb_ma, color='black', label = "moving_avg_mean")
     ...: plt.title("Turbidity(FNU) of Potomac River")
     ...: plt.xlabel("Date")
     ...: plt.ylabel("Turbidity")
     ...: plt.legend()
     ...: plt.show()
     ...: turb_moving_avg_diff = turb - turb_ma
     ...: turb_moving_avg_diff.dropna(inplace=True) # first 6 is nan value due to window
size
     ...:
```

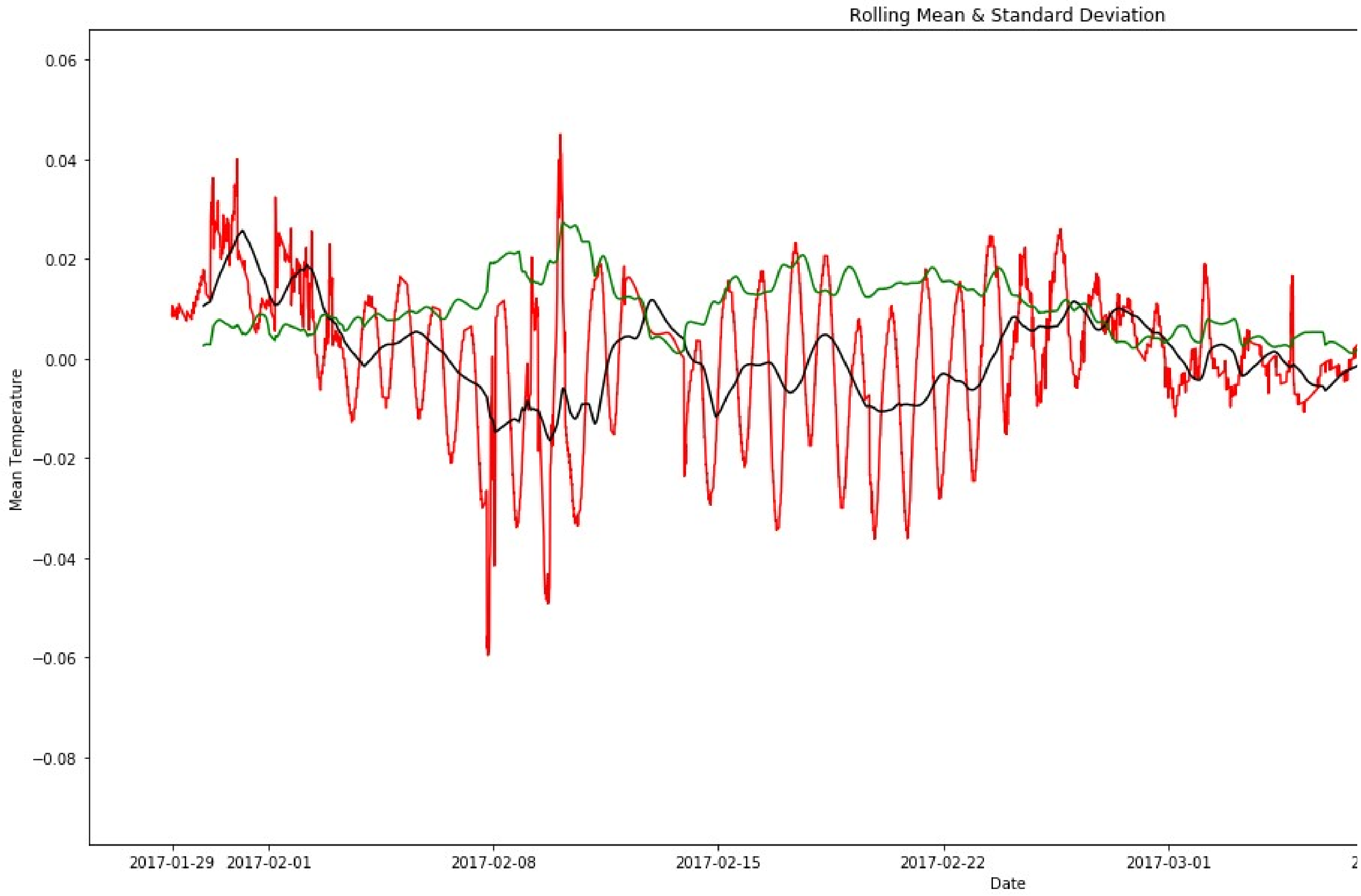
```
...: # check stationary: mean, variance(std)and adfuller test
...: check_mean_std(turb_moving_avg_diff)
...: check_adfuller(turb_moving_avg_diff['Turb(FNU)'])
```



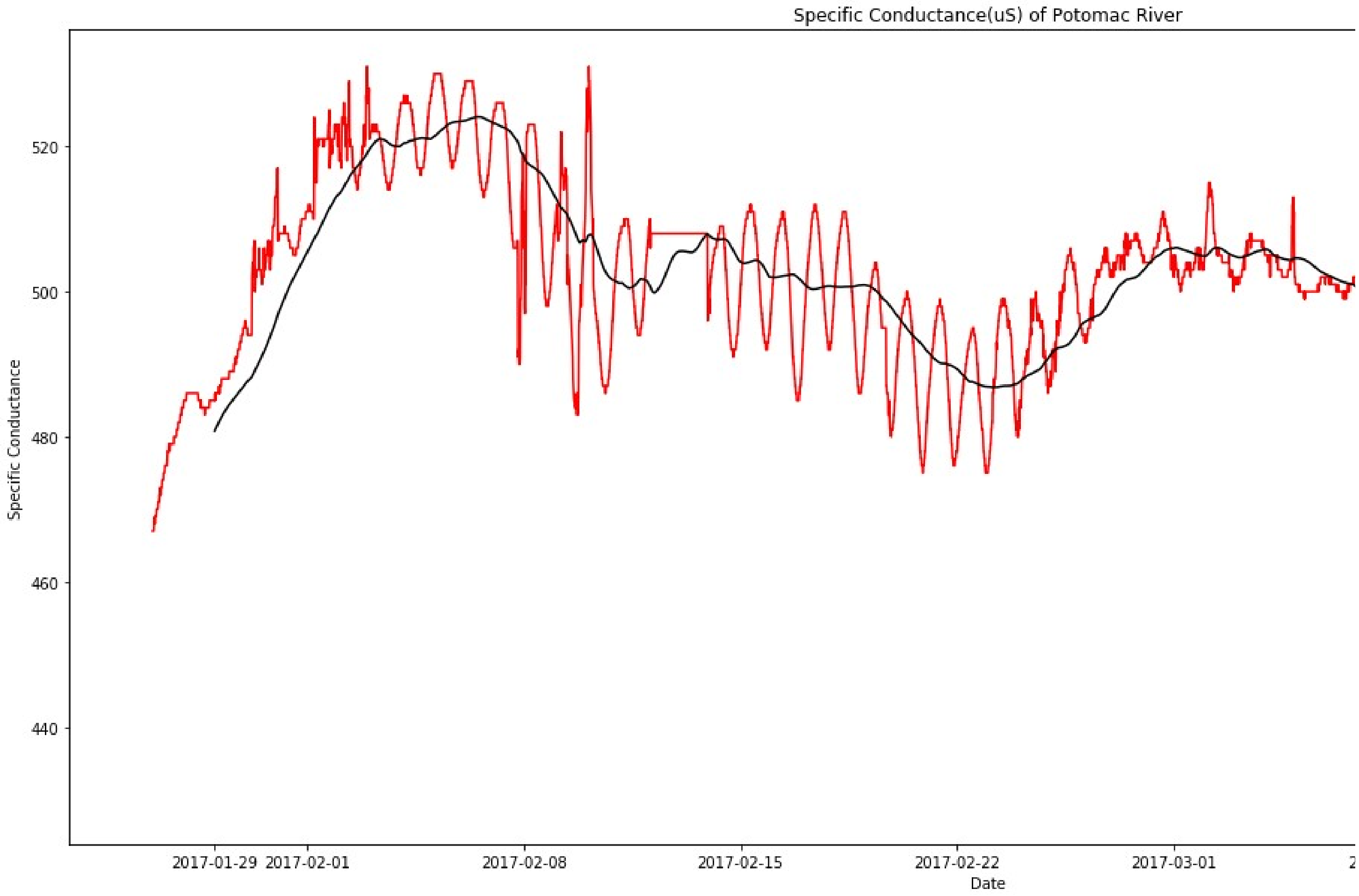


```
Results of Dickey Fuller Test:
--------For a stationary time series Test statistic is less than critical values
Test Statistic                  -5.544052
p-value                          0.000002
#Lags Used                      31.000000
Number of Observations Used    4948.000000
Critical Value (1%)             -3.431672
Critical Value (5%)             -2.862124
Critical Value (10%)            -2.567081
dtype: float64

In [164]: sc_logScale = np.log(sc)
     ...: plt.plot(sc_logScale)
     ...:
     ...: sc_moving_Average = sc_logScale.rolling(window=192).mean() #window size 12
denotes 12 months, giving rolling mean at yearly level
     ...: sc_movingSTD = sc_logScale.rolling(window=192).std()
     ...:
     ...: plt.plot(sc_logScale)
     ...: plt.plot(sc_moving_Average, color='blue')
     ...:
     ...: sc_LogScaleMinusMovingAverage = sc_logScale - sc_moving_Average
     ...: sc_LogScaleMinusMovingAverage.head(100)
     ...:
     ...: sc_LogScaleMinusMovingAverage.dropna(inplace=True)
     ...: #print(sc_rolmean,sc_rolstd)
     ...:
     ...: check_adfuller(sc_LogScaleMinusMovingAverage['SC(uS)'])
     ...: check_mean_std(sc_LogScaleMinusMovingAverage)
Results of Dickey Fuller Test:
--------For a stationary time series Test statistic is less than critical values
Test Statistic                 -8.252439e+00
p-value                         5.346032e-13
#Lags Used                      2.500000e+01
Number of Observations Used     4.954000e+03
Critical Value (1%)            -3.431671e+00
Critical Value (5%)            -2.862124e+00
Critical Value (10%)           -2.567081e+00
dtype: float64
```

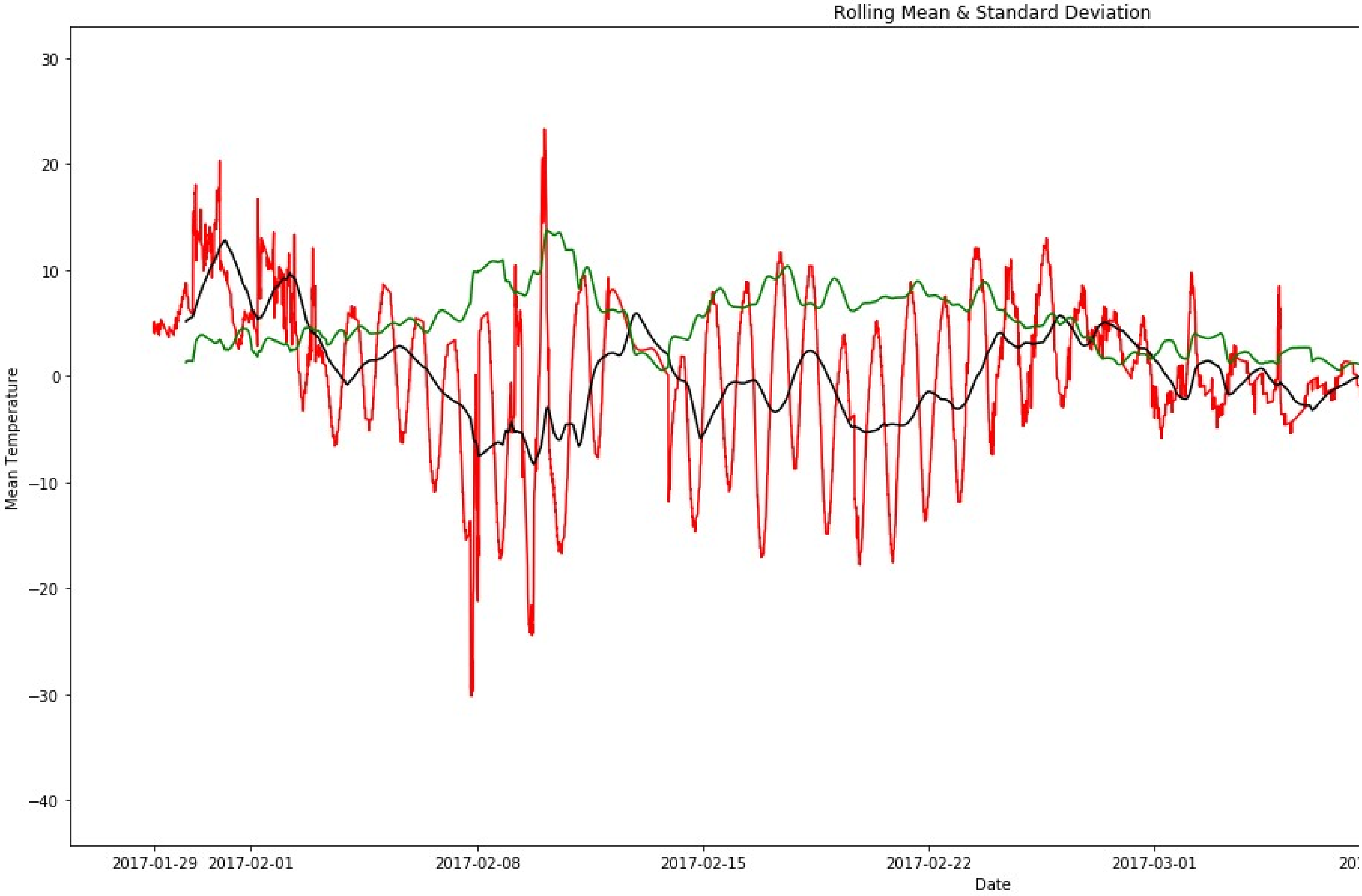




```
In [165]: sc_ma = sc.rolling(window=192).mean() #window size 12 denotes 12 months, giving
rolling mean at yearly level
     ...: #sc_movingSTD = sc_logScale.rolling(window=192).std()
     ...:
     ...: #plt.plot(sc_logScale)
     ...: #plt.plot(sc_moving_Average, color='blue')
     ...: plt.figure(figsize=(22,10))
     ...: plt.plot(sc, color = "red",label = "Original")
     ...: plt.plot(sc_ma, color='black', label = "moving_avg_mean")
     ...: plt.title("Specific Conductance(uS) of Potomac River")
     ...: plt.xlabel("Date")
     ...: plt.ylabel("Specific Conductance")
     ...: plt.legend()
     ...: plt.show()
     ...: sc_ma_diff = sc - sc_ma
     ...: #sc_LogScaleMinusMovingAverage.head(100)
     ...:
     ...: sc_ma_diff.dropna(inplace=True)
     ...: #print(sc_rolmean,sc_rolstd)
     ...:
     ...: check_adfuller(sc_ma_diff['SC(uS)'])
     ...: check_mean_std(sc_ma_diff)
```



```
Results of Dickey Fuller Test:
--------For a stationary time series Test statistic is less than critical values
Test Statistic                -8.211565e+00
p-value                        6.796513e-13
#Lags Used                     2.600000e+01
Number of Observations Used    4.953000e+03
Critical Value (1%)           -3.431671e+00
Critical Value (5%)           -2.862124e+00
Critical Value (10%)          -2.567081e+00
dtype: float64
```


Rolling Mean & Standard Deviation
30
20
10
0
−10
−20
−30
−40
Mean Temperature
2017-01-29
2017-02-01
2017-02-08
2017-02-15
2017-02-22
2017-03-01
201
Date

```
In [166]:
```