

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 432x288 with 1 Axes>

In [145]:

In [145]:

In [145]: predictions

Out[145]:

1919	0.059764
1920	0.060740
1921	0.075906
1922	0.083076
1923	0.082281
1924	0.090794
1925	0.090164
1926	0.090651
1927	0.095590
1928	0.093994
1929	0.096303
1930	0.098989
1931	0.099137
1932	0.100801
1933	0.102842
1934	0.103480
1935	0.105352
1936	0.106833
1937	0.107857
1938	0.109601
1939	0.110868
1940	0.112115
1941	0.113661
1942	0.114945
1943	0.116236
1944	0.117679
1945	0.118942
1946	0.120268
1947	0.121627
1948	0.122891
...	
5142	0.348881
5143	0.348881
5144	0.348881
5145	0.348881
5146	0.348881
5147	0.348881
5148	0.348881
5149	0.348881
5150	0.348881
5151	0.348881
5152	0.348881

```

5153    0.348881
5154    0.348881
5155    0.348881
5156    0.348881
5157    0.348881
5158    0.348881
5159    0.348881
5160    0.348881
5161    0.348881
5162    0.348881
5163    0.348881
5164    0.348881
5165    0.348881
5166    0.348881
5167    0.348881
5168    0.348881
5169    0.348881
5170    0.348881
5171    0.348881
Length: 3253, dtype: float64

```

In [145]:

```

In [146]: start_index = parser("2017-02-16 00:00")
...: end_index = "2017-03-21 23:45"
...: forecast = sc_arma_fit.predict(start=1919, end=4000)
...:
...: plt.figure(figsize=(22,10))
...: plt.plot(sc_test,label = "original")
...: plt.plot(forecast,label = "predicted", color = 'red')

```

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)

Out[146]: [<matplotlib.lines.Line2D at 0x22891f10588>]Error in callback <function install_repl_displayhook.<locals>.post_execute at 0x0000022881225158> (for post_execute):
Traceback (most recent call last):

File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\pyplot.py", line 109, in post_execute
draw_all()

File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib_pylab_helpers.py", line 132, in draw_all
f_mgr.canvas.draw_idle()

File "C:\Users\admin\Anaconda3\lib\site-packages\matplotlib\backend_bases.py", line 1899, in draw_idle
self.draw(*args, **kwargs)

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

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_arma_fit.predict(start=1919, end=4000)

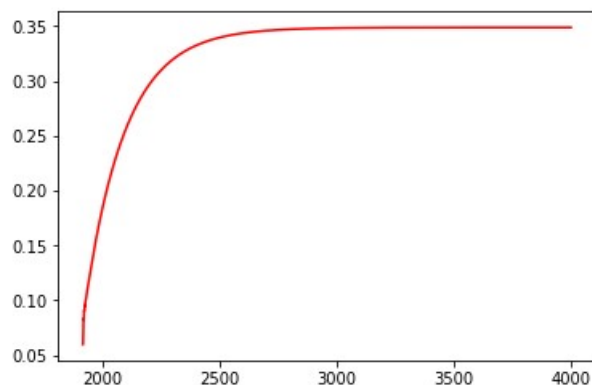
....:

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]

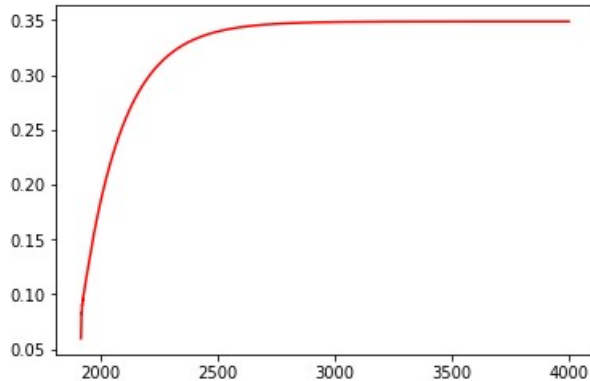


In [149]: predictions = sc_arma_fit.predict(start=1919, end=3000)

```
....:
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()
```

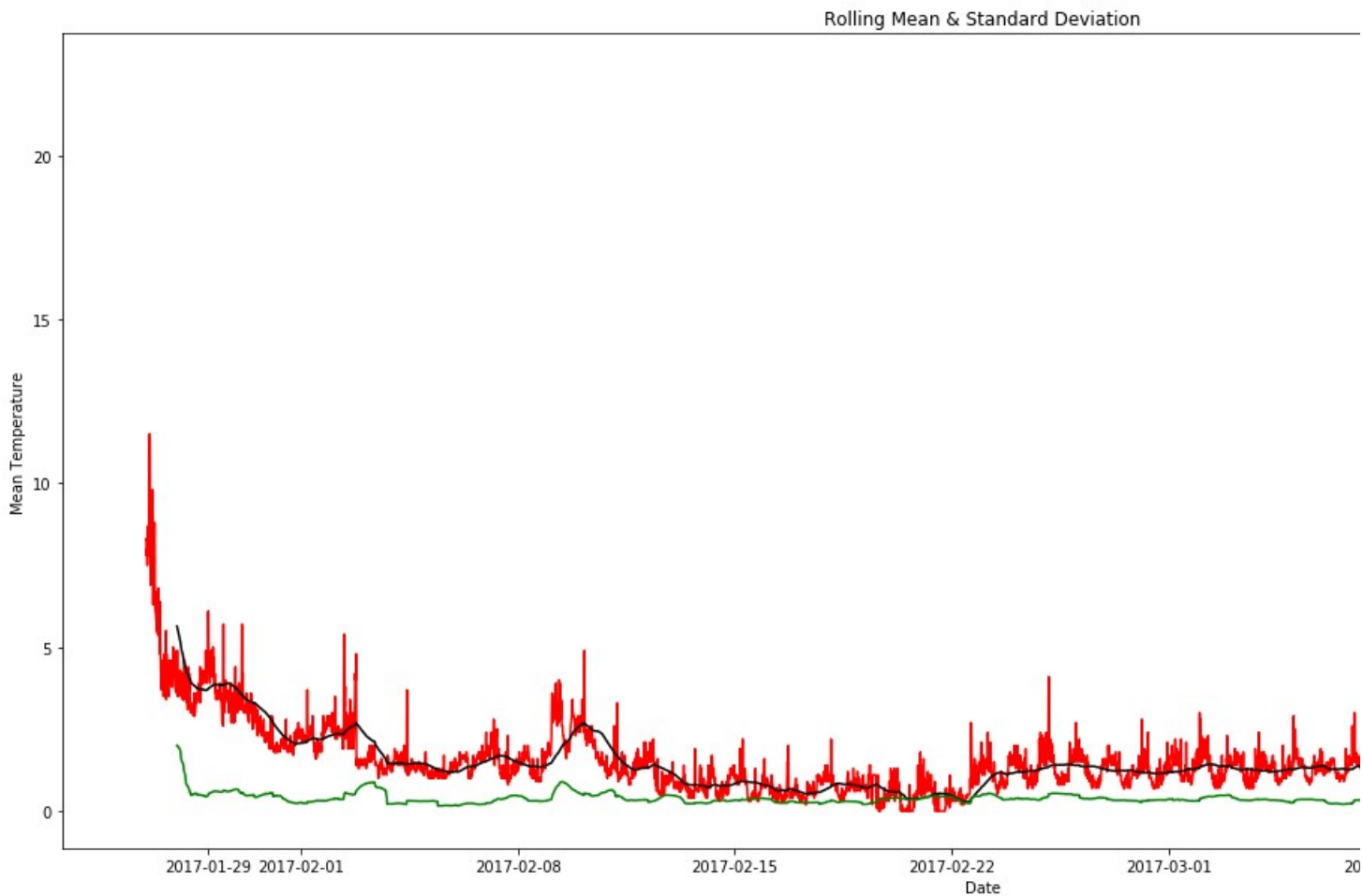
```
In [154]: check_adfuller(turb['Turb(FNU)'])
...: check_mean_std(turb['Turb(FNU)'])
```

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)'

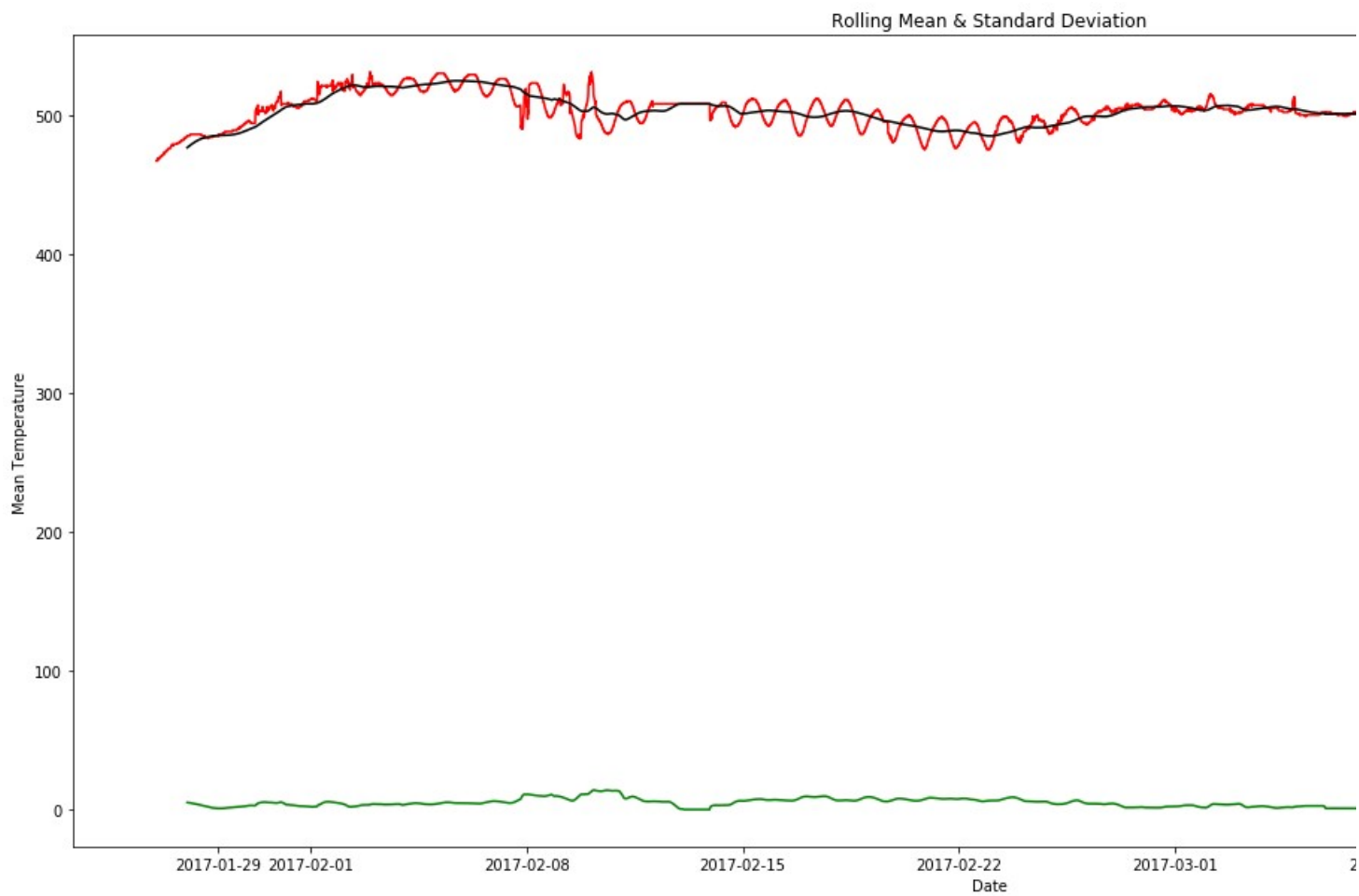
```

In [156]:

```

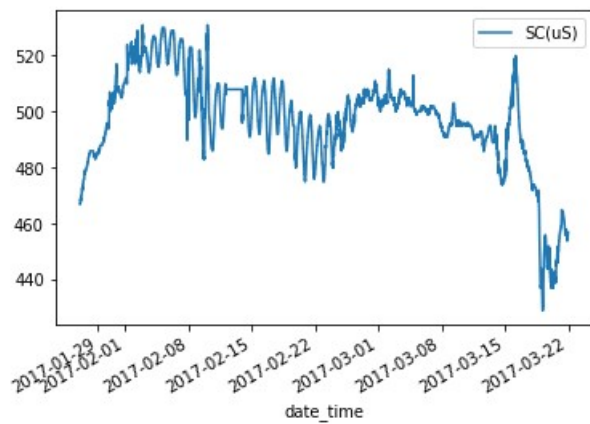
In [156]: check_adfuller(sc['SC(uS)'])
...: check_mean_std(sc['SC(uS)'])
Results of Dickey Fuller Test:
-----For a stationary time series Test statistic is less than critical values
Test Statistic          -3.268401
p-value                  0.016351
#Lags Used               27.000000
Number of Observations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

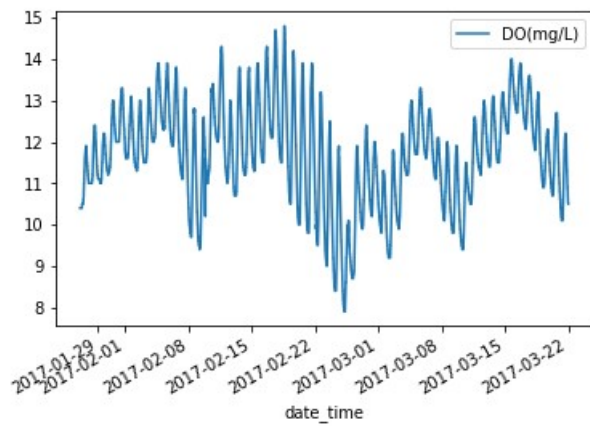
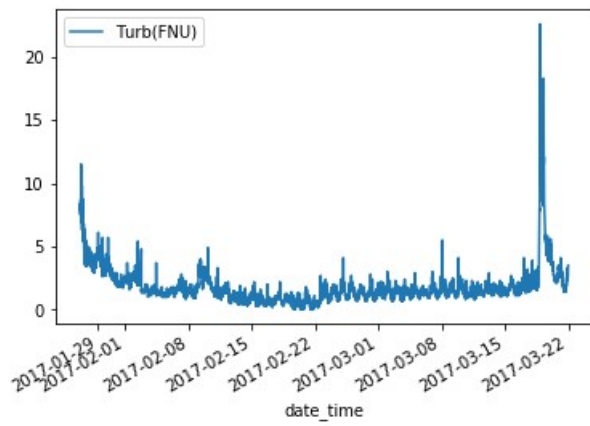
```



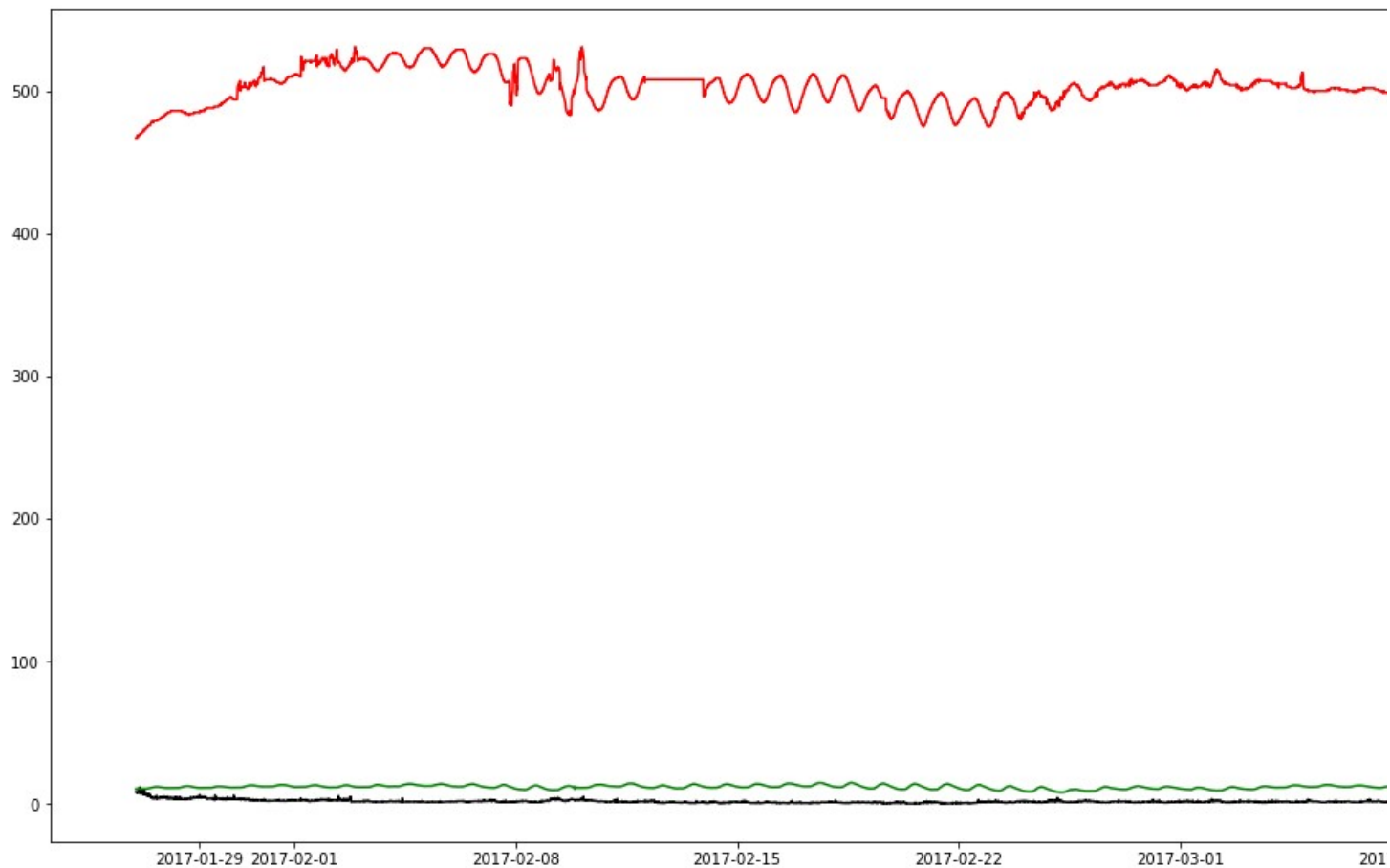
```
In [157]: plt.figure(figsize=(22,10))
...: sc.plot()
...: turb.plot()
...: do.plot()
```

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()
```

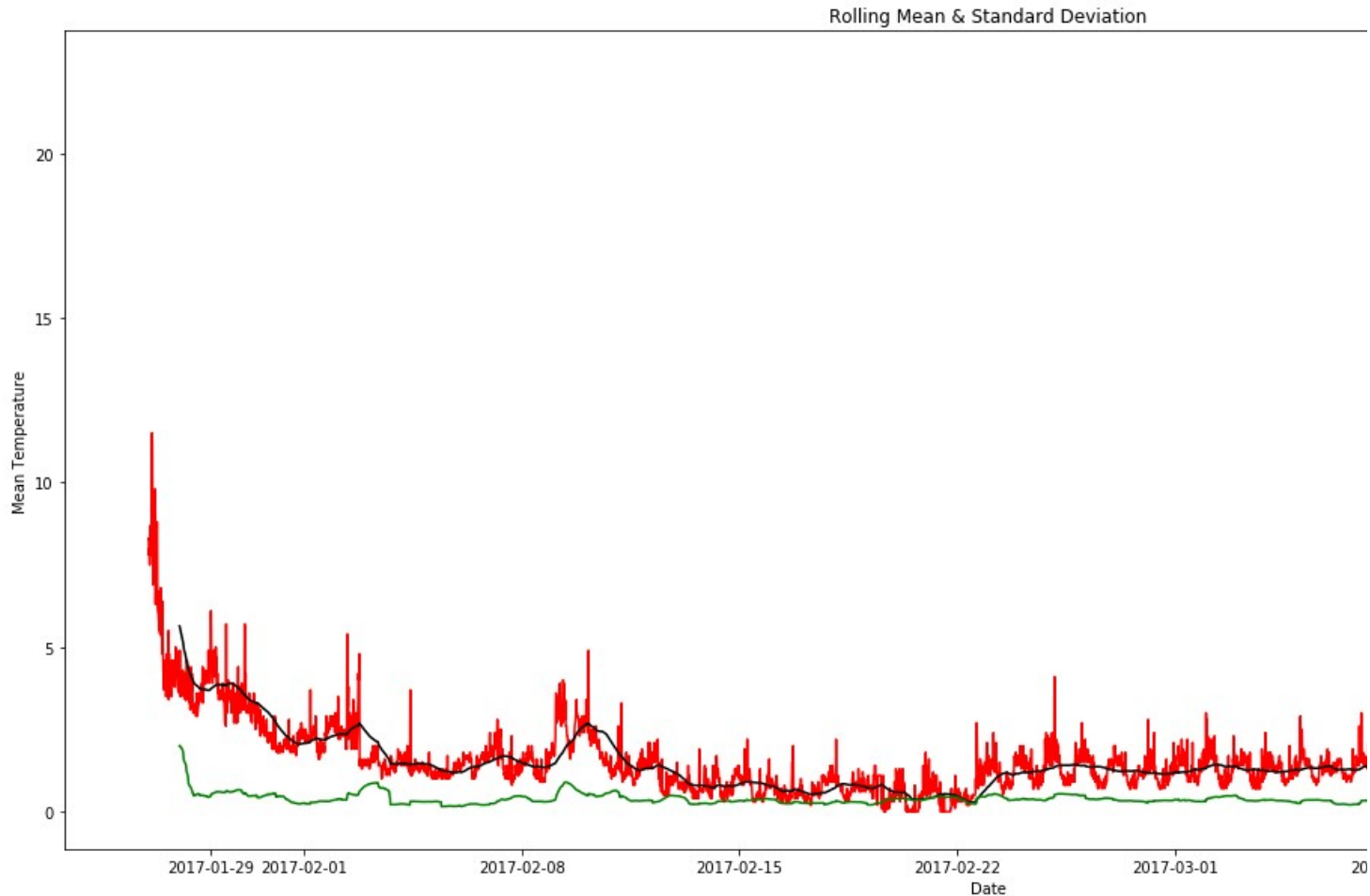


```
In [159]: check_adfuller(turb['Turb(FNU)'])
...: check_mean_std(turb['Turb(FNU)'])
...:
...: # Moving average method for turbidity
...: window_size = 96
...: moving_avg = pd.rolling_mean(turb,window_size)
...: plt.figure(figsize=(22,10))
...: plt.plot(turb, color = "red",label = "Original")
...: 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)

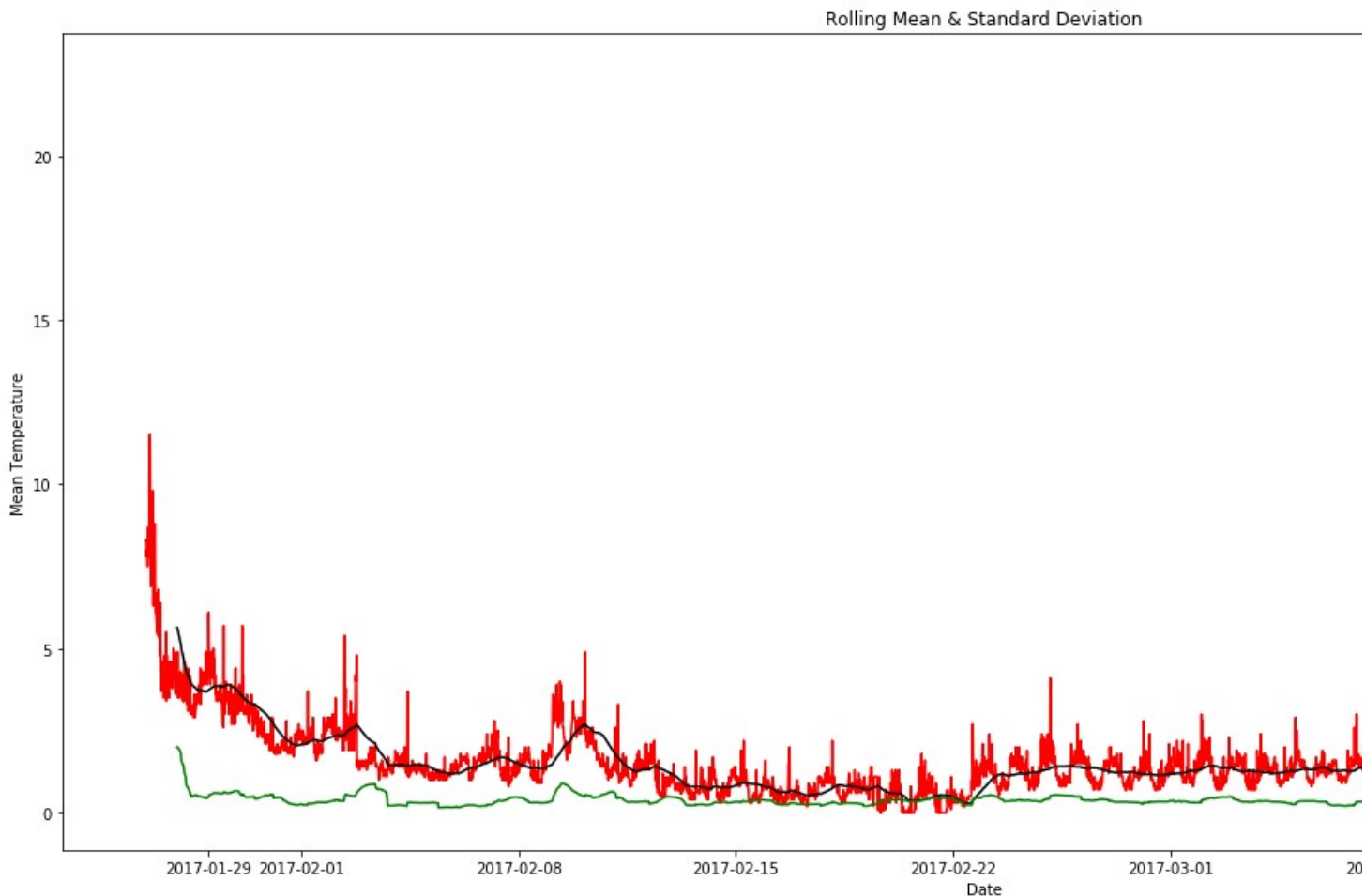
```

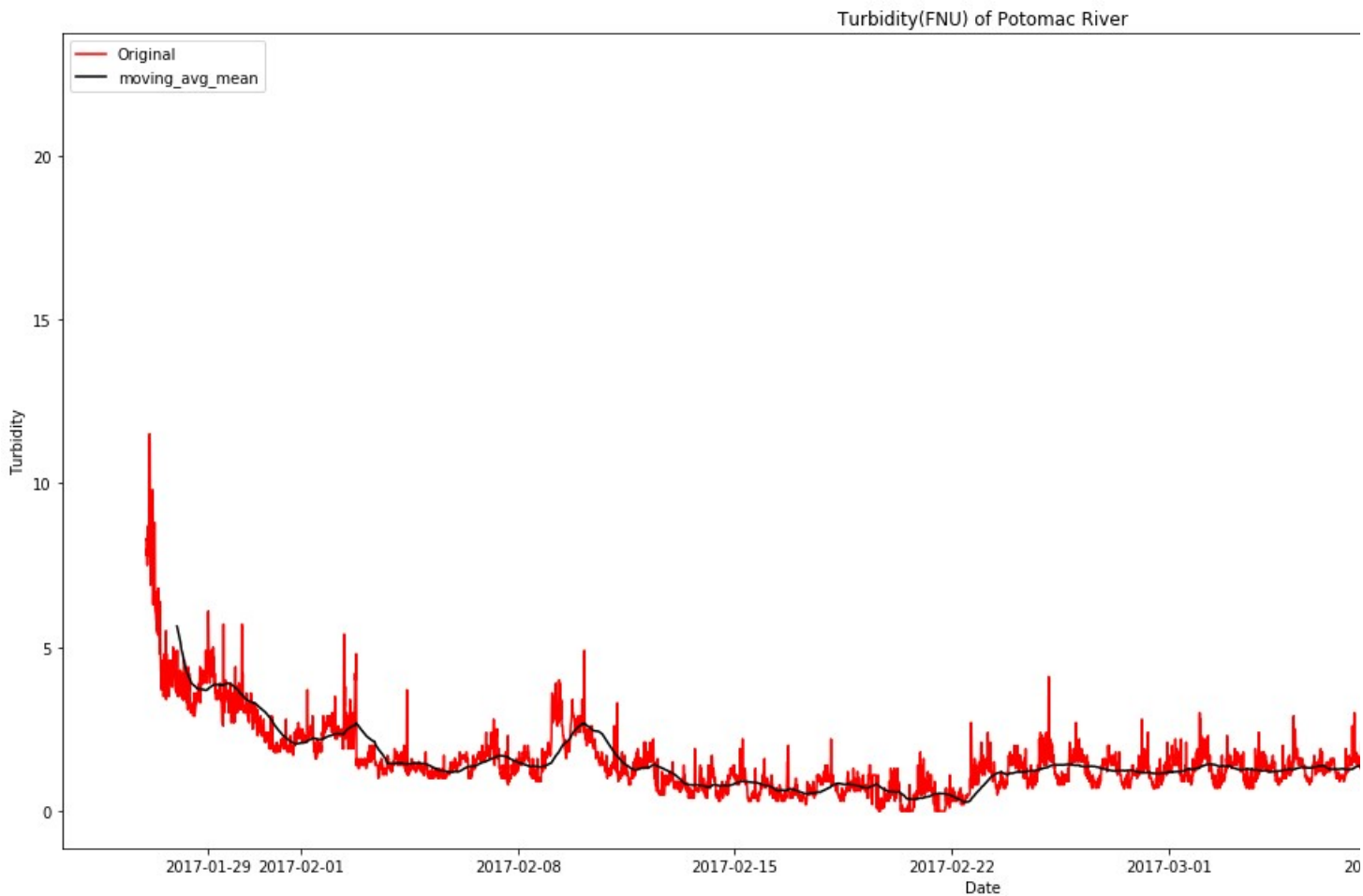
Results of Dickey Fuller Test:

-----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)'])

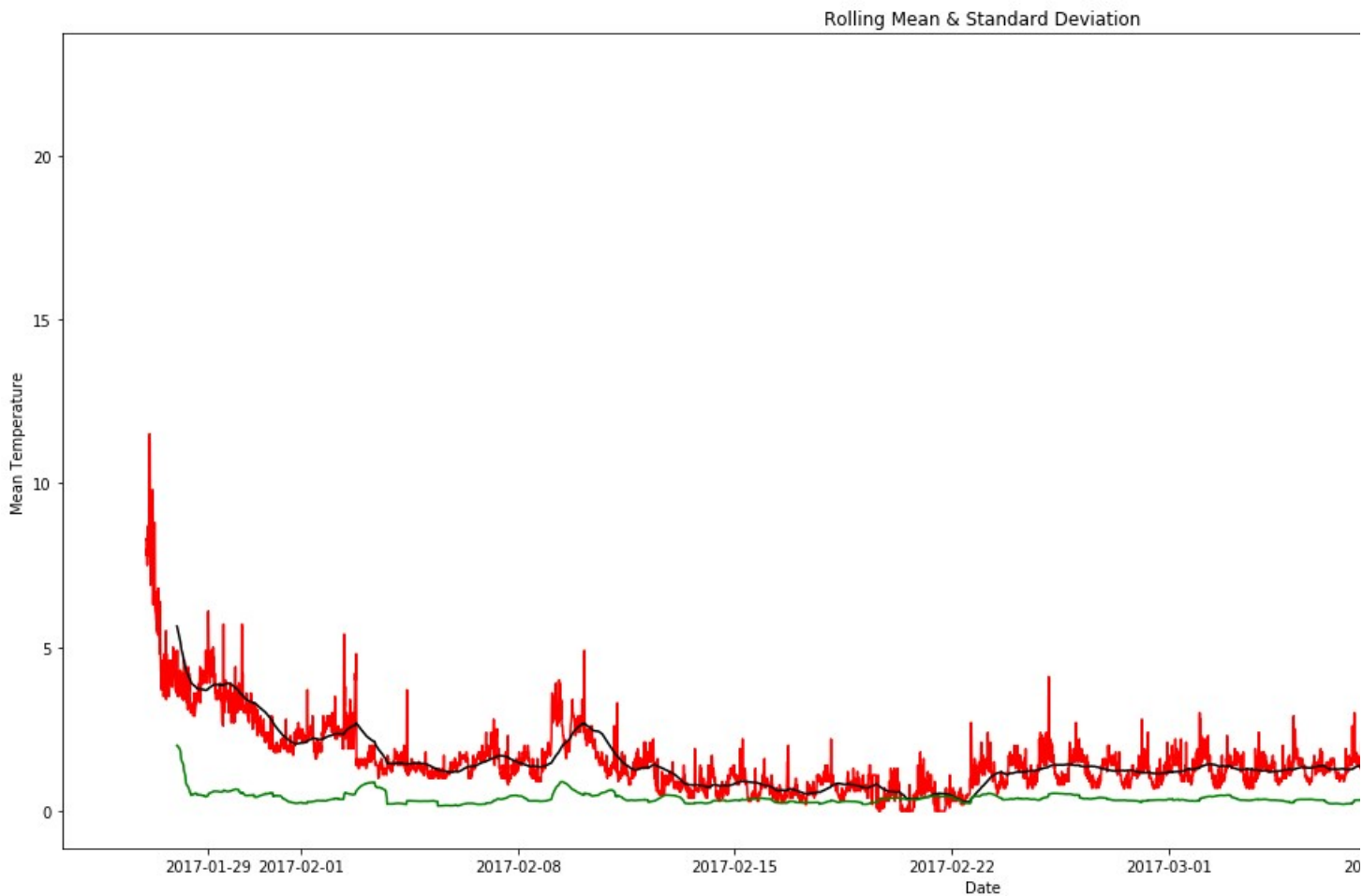
```

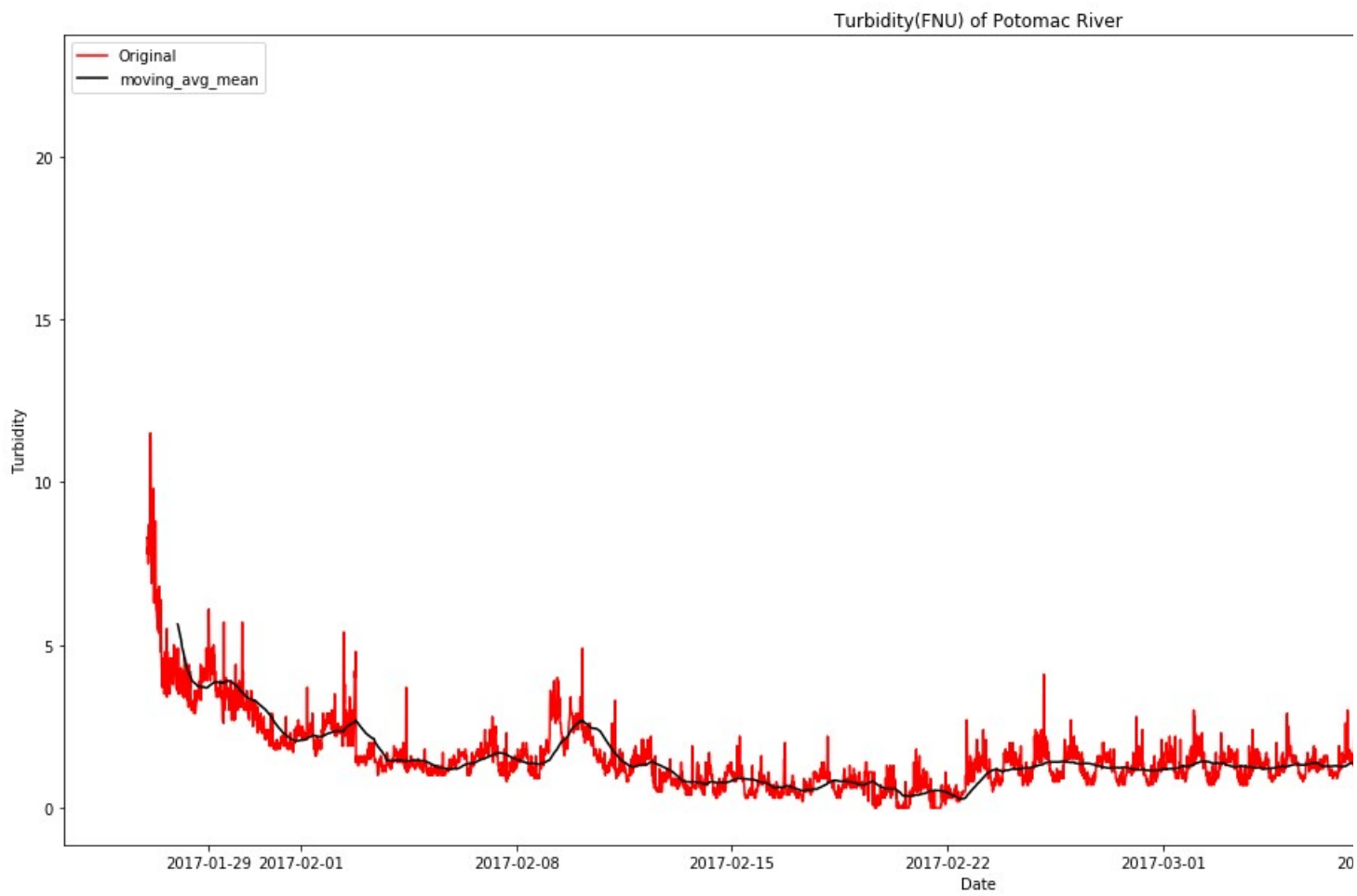
Results of Dickey Fuller Test:

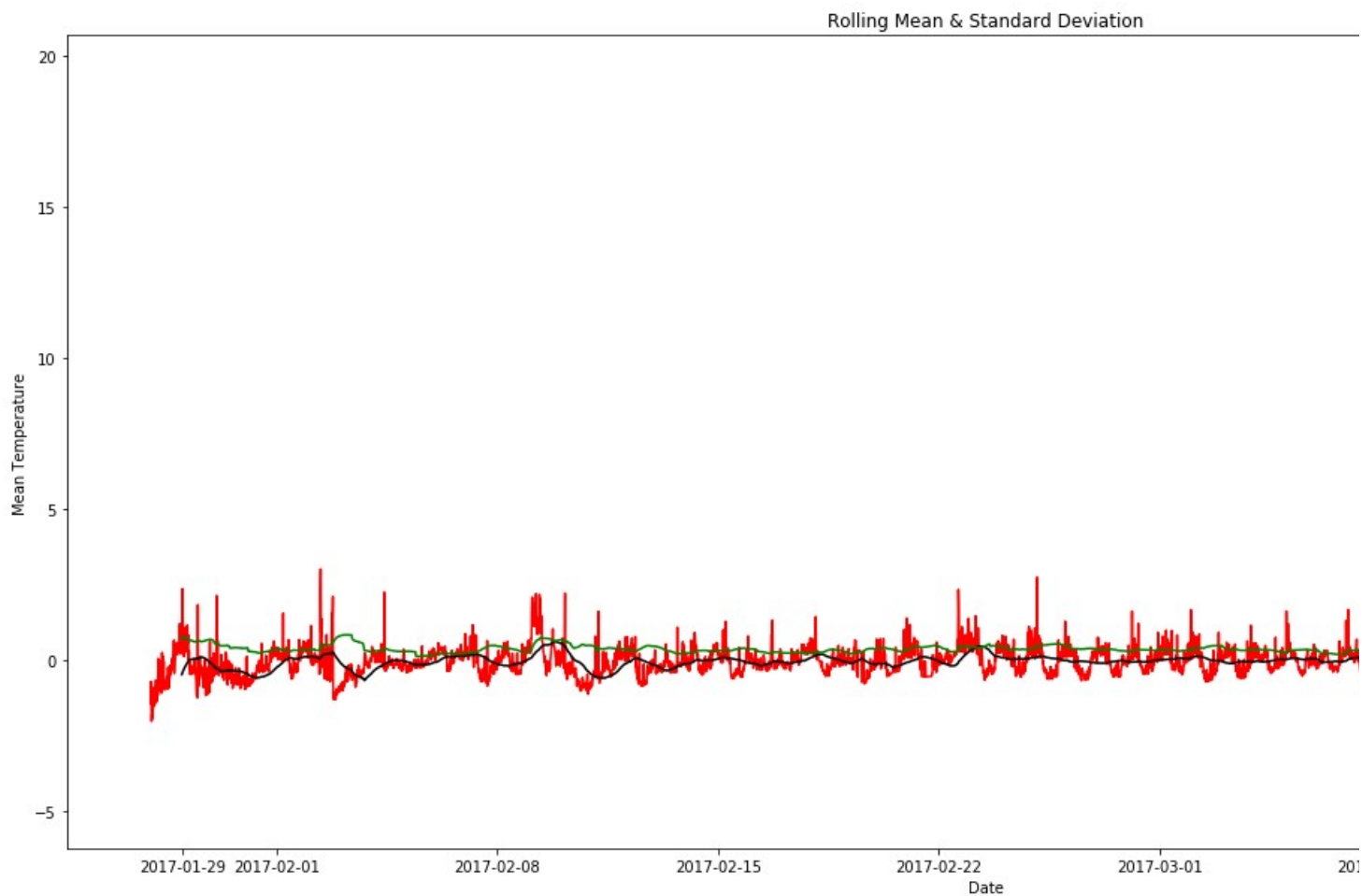
-----For a stationary time series Test statistic is less than critical values

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







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]: