AMBOMO TIGA GEDEON 21T2496

DEVOIR INF312: SERIE TEMPORELLE

1- Étude production de bière

In [ ]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

filepath_1 = "dataset/monthly-beer-production-in-austr.csv"
df  = pd.read_csv(filepath_1, nrows=500)
```

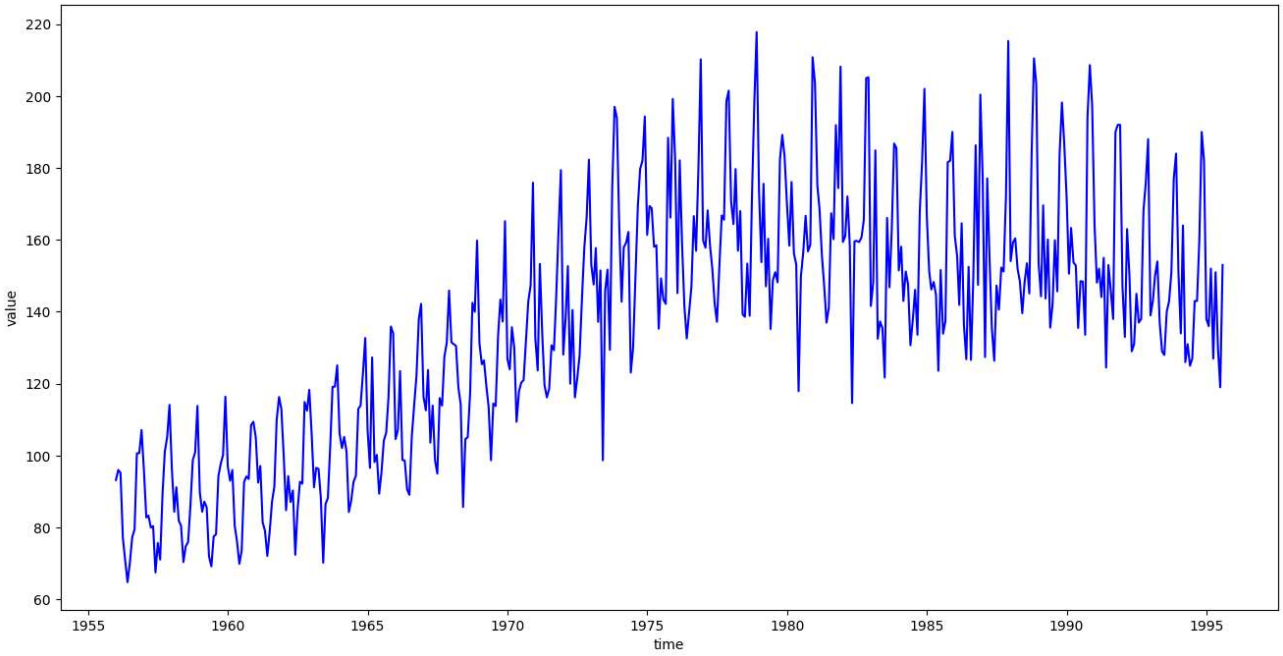
2- Moyenne, Variance et Ecart-type

In [ ]:
```
mean_1 = df['Monthly beer production'].mean()
var_1 = df['Monthly beer production'].var()
ect_1 = df['Monthly beer production'].std()
print(f"moyenne: {mean_1}\nVarieance: {var_1}\nEcart-type: {ect_1}")
```

```
moyenne: 136.3953781512605
Varieance: 1138.3015364882797
Ecart-type: 33.738724583011134
```

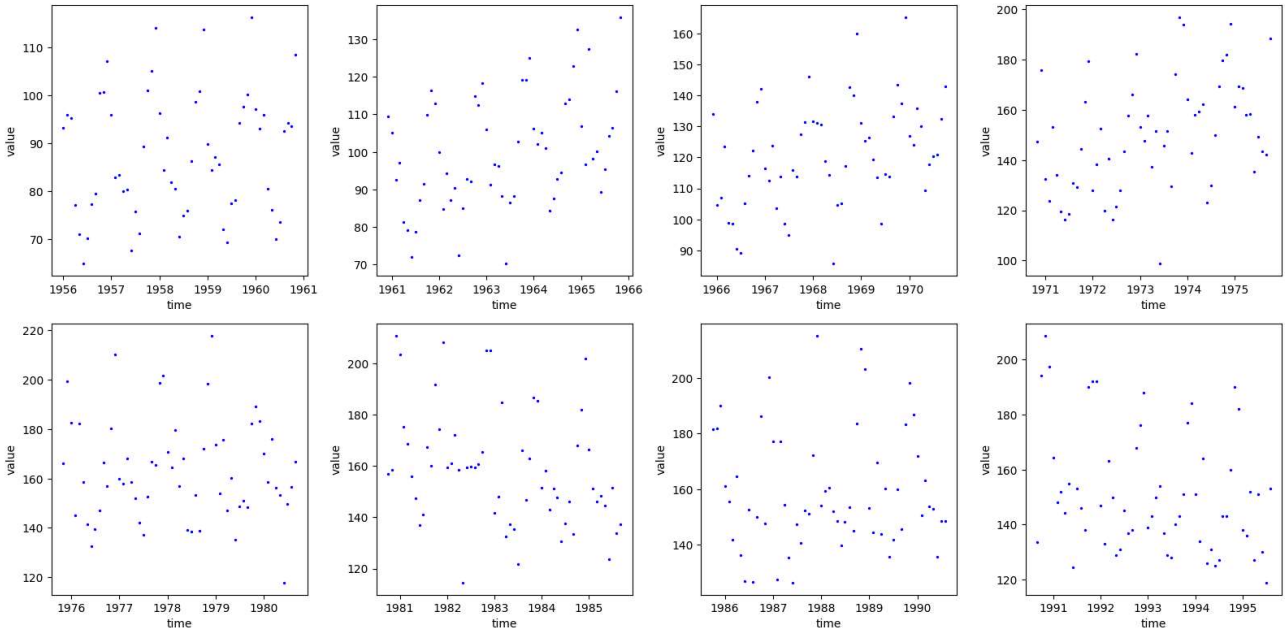
3- Représentation

In [ ]:
```
plt.figure(figsize=(16,8))
plt.plot(pd.to_datetime(df['Month']), df['Monthly beer production'], c='b')
plt.xlabel("time")
plt.ylabel("value")

# plt.title("Temporal serie for prooduction")
plt.show()
```



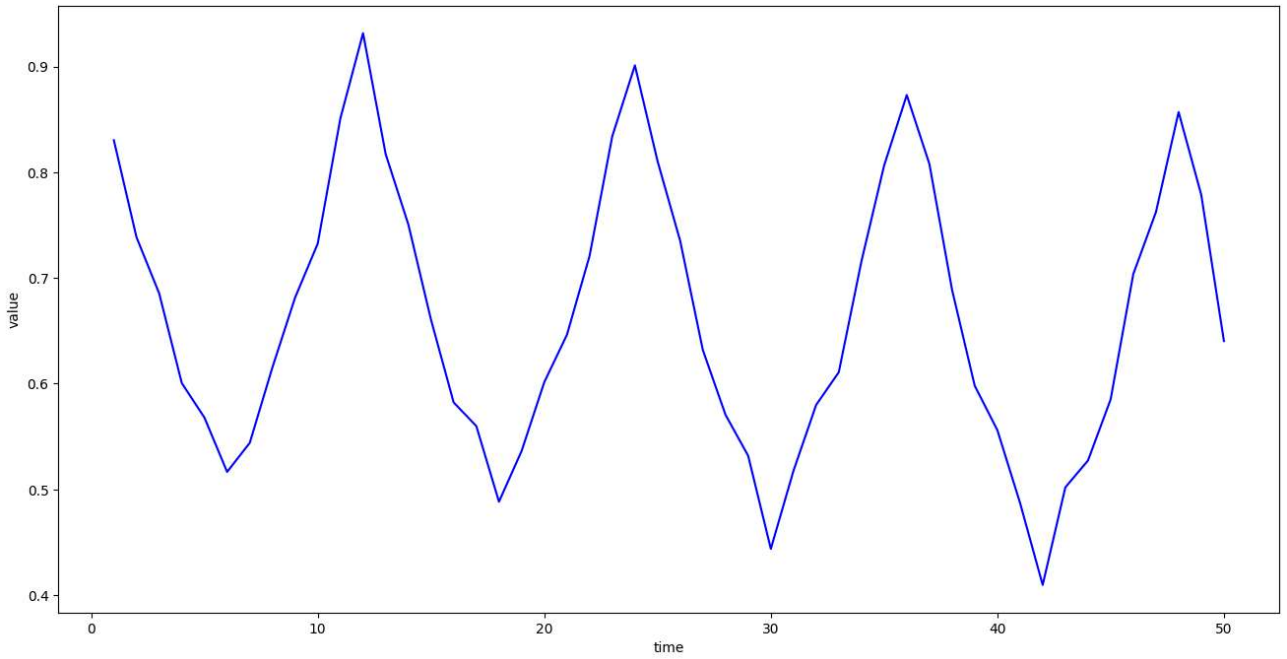
4- Nuage de points

In [ ]:
```
fig_1, axs_1 = plt.subplots(2, 4, figsize=(16, 8))

for i, ax in enumerate(axs_1.flat):
    start_idx = i * len(df['Month'])//8
    end_idx = (i+1)*len(df['Month'])//8
    ax.scatter(pd.to_datetime(df['Month'][start_idx:end_idx]), df['Monthly beer production'][start_idx:end_idx], s=2, c='b', label="Production serie")
    ax.set_xlabel("time")
    ax.set_ylabel("value")

plt.tight_layout()
plt.show()
```



5- Auto-correlations

In [ ]:
```
autocorr = [df['Monthly beer production'].autocorr(lag=lag) for lag in range(1, 51)]
lags = range(1, 51)
plt.figure(figsize=(16, 8))
plt.plot(lags, autocorr, c='b', label="Production series")
plt.xlabel("time")
plt.ylabel("value")
# plt.title("Temporal serie for prooduction")
plt.show()
```



1- Étude Bitcoin

In [ ]:
```
filepath_2 = "dataset/BTC-EUR.csv"
df_2  = pd.read_csv(filepath_2, nrows=500);
```

2- Moyenne, variance et Ecart-type

In [ ]:
```
mean_2 = df_2['Close'].mean()
var_2 = df_2['Close'].var()
ect_2 = df_2['Close'].std()
print(f"moyenne: {mean_2}\nVarieance: {var_2}\nEcart-type: {ect_2}")
```

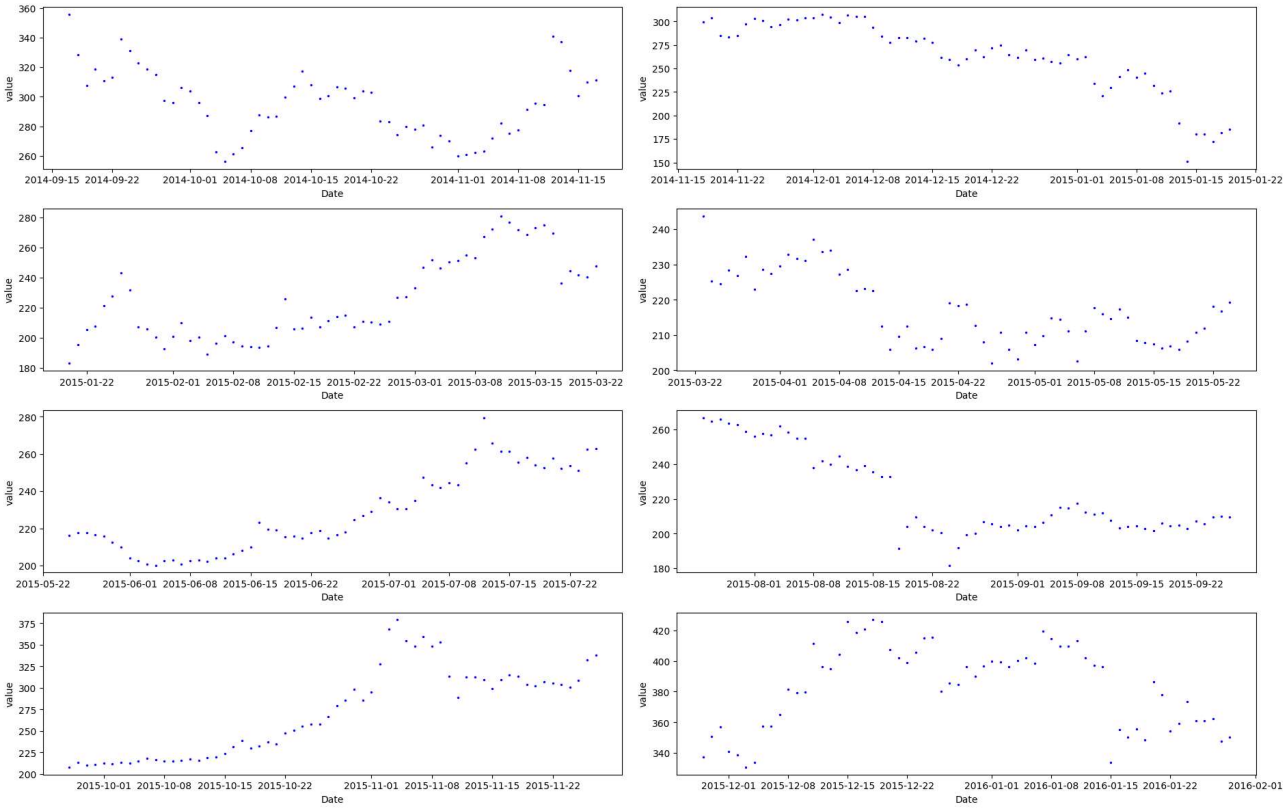
```
moyenne: 263.369604148
Varieance: 3671.2576510383046
Ecart-type: 60.59090402889121
```

3- Représentation

In [ ]:
```
plt.figure(figsize=(16,8))
plt.plot(pd.to_datetime(df_2['Date']), df_2['Close'], c='b');
plt.xlabel("Date")
plt.ylabel("value")
# plt.title("Temporal serie for prooduction")
plt.show()
```

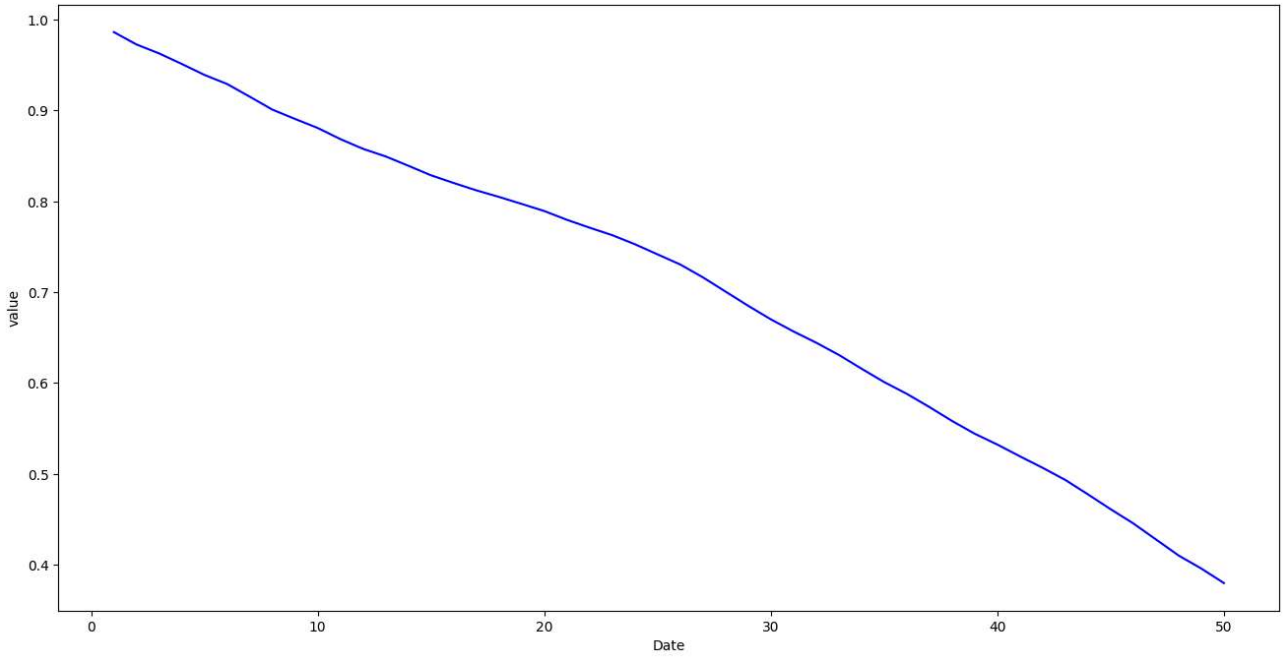
4- Nuage de points

In [ ]:
```
fig_2, axs_2 = plt.subplots(4, 2, figsize=(19, 12))

for i, ax in enumerate(axs_2.flat):
    start_idx = i*len(df_2['Date'])//8
    end_idx = (i+1)*len(df_2['Date'])//8
    ax.scatter(pd.to_datetime(df_2['Date'][start_idx:end_idx]), df_2['Close'][start_idx:end_idx], s=2, c='b')
    ax.set_xlabel("Date")
    ax.set_ylabel("value")
# plt.title("Temporal serie for prooduction")

plt.tight_layout()
plt.show()
```



5- Auto-correlation

In [ ]:
```
autocorr = [df_2['Close'].autocorr(lag=lag) for lag in range(1, 51)]
lags = range(1, 51)
plt.figure(figsize=(16, 8))
plt.plot(lags, autocorr, c='b')
plt.xlabel("Date")
plt.ylabel("value")
# plt.title("Temporal serie for prooduction")
plt.show()
```



1- Étude météo

In [ ]:
```
filepath_3 = "dataset/weather_data_kolkata_2015_2020.csv"
df_3  = pd.read_csv(filepath_3, nrows=700);
```

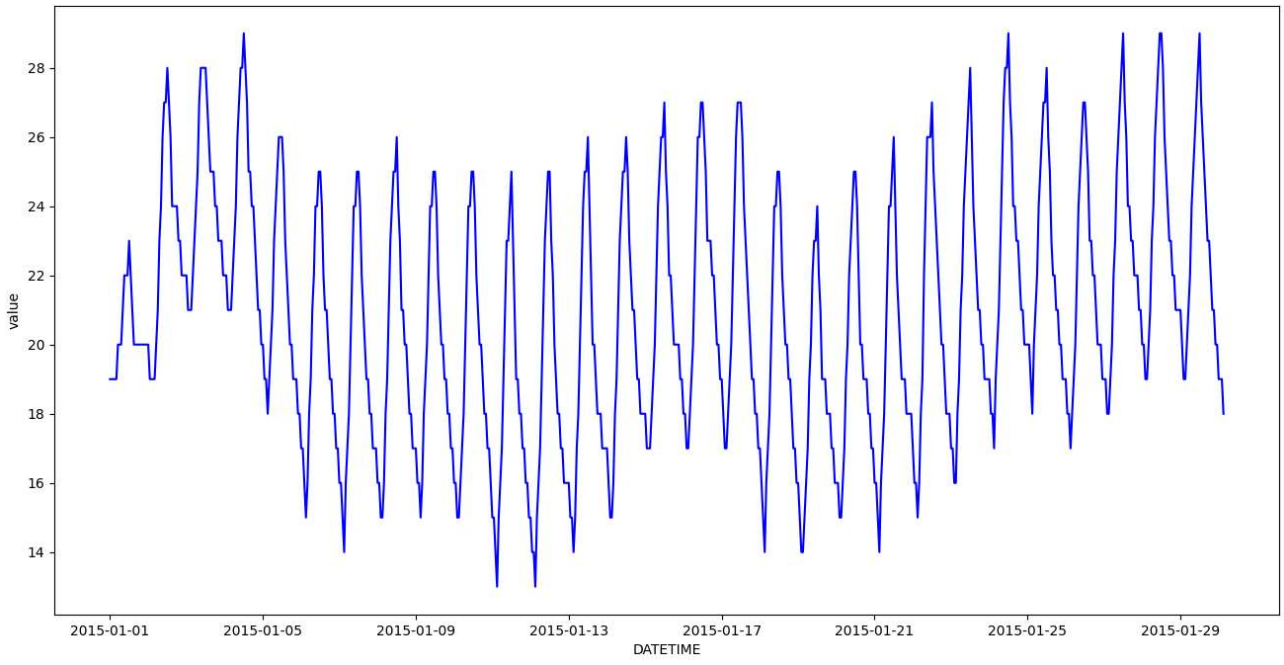
2- Moyenne, variance et Ecart-type

In [ ]:
```
mean_3 = df_3['TEMPERATURE'].mean()
var_3 = df_3['TEMPERATURE'].var()
ect_3 = df_3['TEMPERATURE'].std()
print(f"moyenne: {mean_3}\nVarieance: {var_3}\nEcart-type: {ect_3}")
```

```
moyenne: 20.87857142857143
Varieance: 12.524575924790518
Ecart-type: 3.5390077599223373
```

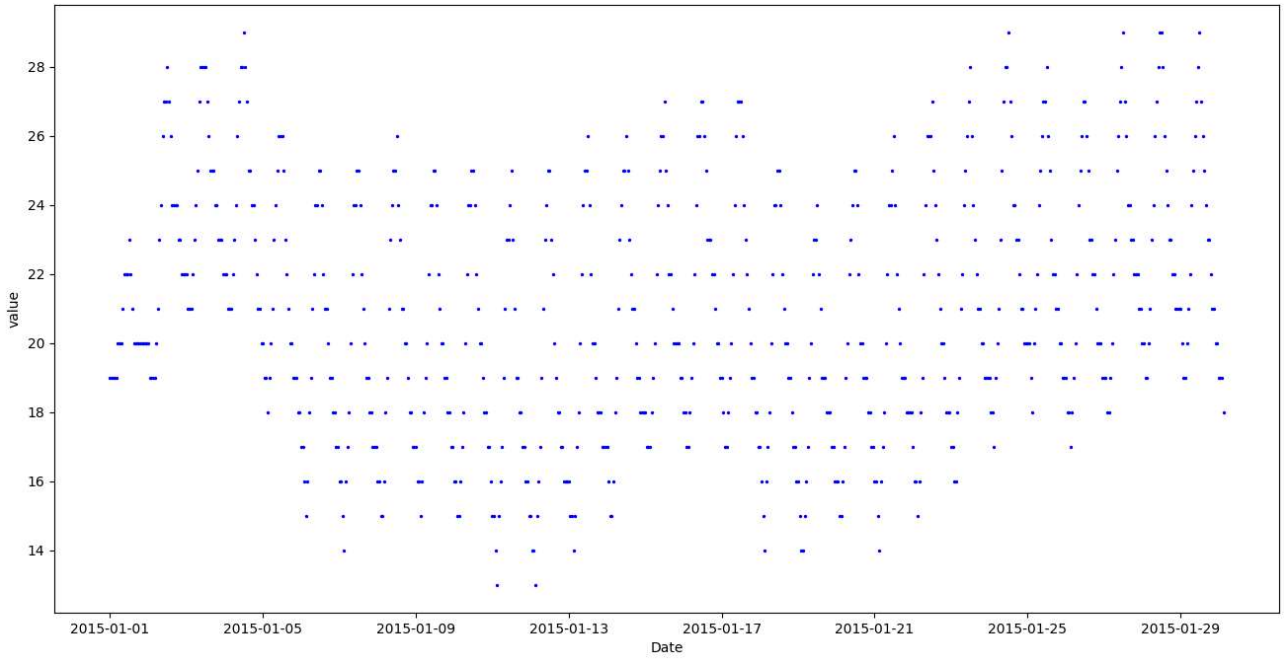
3- Représentation

In [ ]:
```
plt.figure(figsize=(16, 8))
plt.plot(pd.to_datetime(df_3['DATETIME']), df_3['TEMPERATURE'], c='b');
plt.xlabel("DATETIME")
plt.ylabel("value")
# plt.title("Temporal serie for prooduction")
plt.show()
```



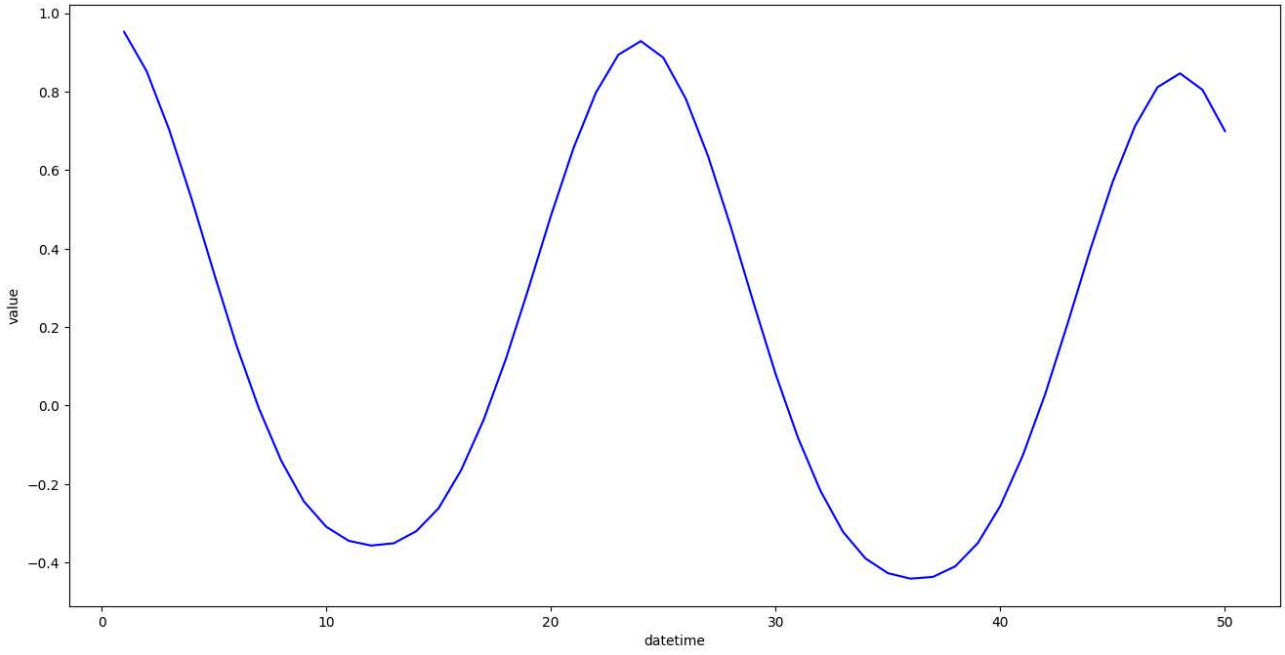
4- Nuage de points

In [ ]:
```
plt.figure(figsize=(16,8));
plt.scatter(pd.to_datetime(df_3['DATETIME']), df_3['TEMPERATURE'], s=2, c='b')
plt.xlabel("Date")
plt.ylabel("value")
# plt.title("Temporal serie for prooduction")
plt.show()
```



5- Auto-correlation

In [ ]:
```
autocorr = [df_3['TEMPERATURE'].autocorr(lag=lag) for lag in range(1, 51)]
lags = range(1, 51)
plt.figure(figsize=(16, 8))
plt.plot(lags, autocorr, c='b')
plt.xlabel("datetime")
plt.ylabel("value")
# plt.title("Temporal serie for prooduction")
plt.show()
```



1- Étude vente panneau solaire

In [ ]:
```
filepath_4 = "dataset/monthly-sunspots.csv"
df_4  = pd.read_csv(filepath_4, nrows=500);
```

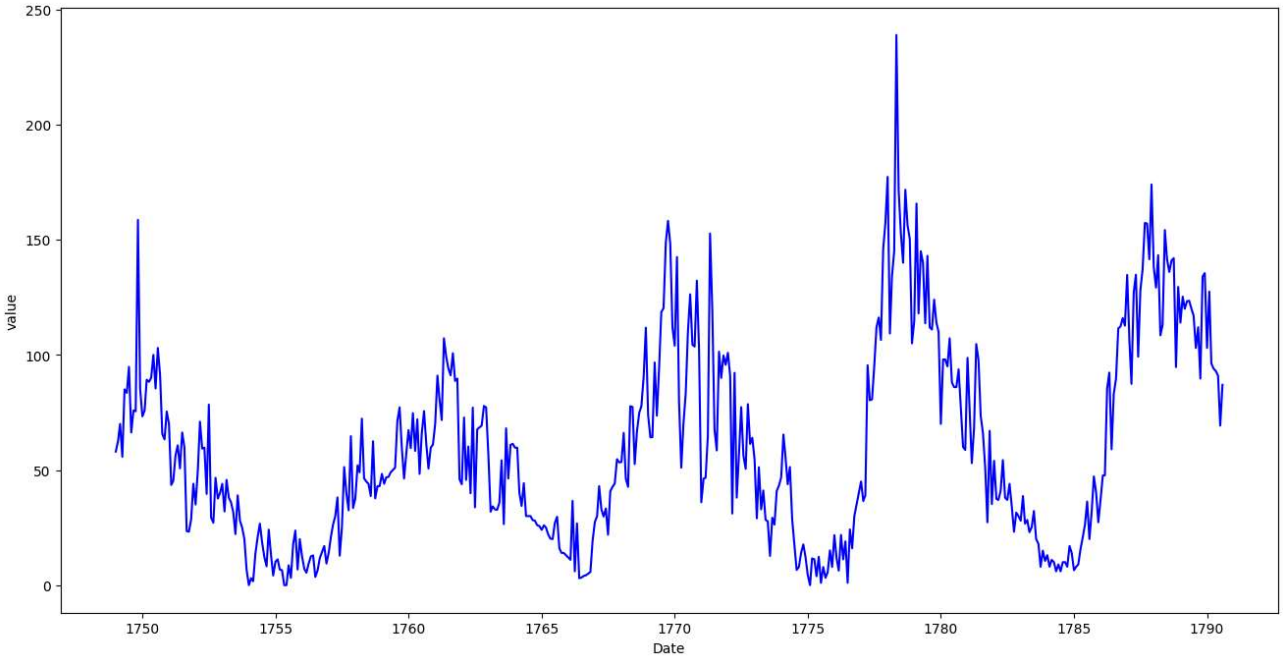
2- Moyenne, variance et Ecart-type

In [ ]:
```
mean_4 = df_4['Sunspots'].mean()
var_4 = df_4['Sunspots'].var()
ect_4 = df_4['Sunspots'].std()
print(f"moyenne: {mean_4}\nVarieance: {var_4}\nEcart-type: {ect_4}")
```

```
moyenne: 59.603199999999994
Varieance: 1808.524959679359
Ecart-type: 42.526755809482566
```

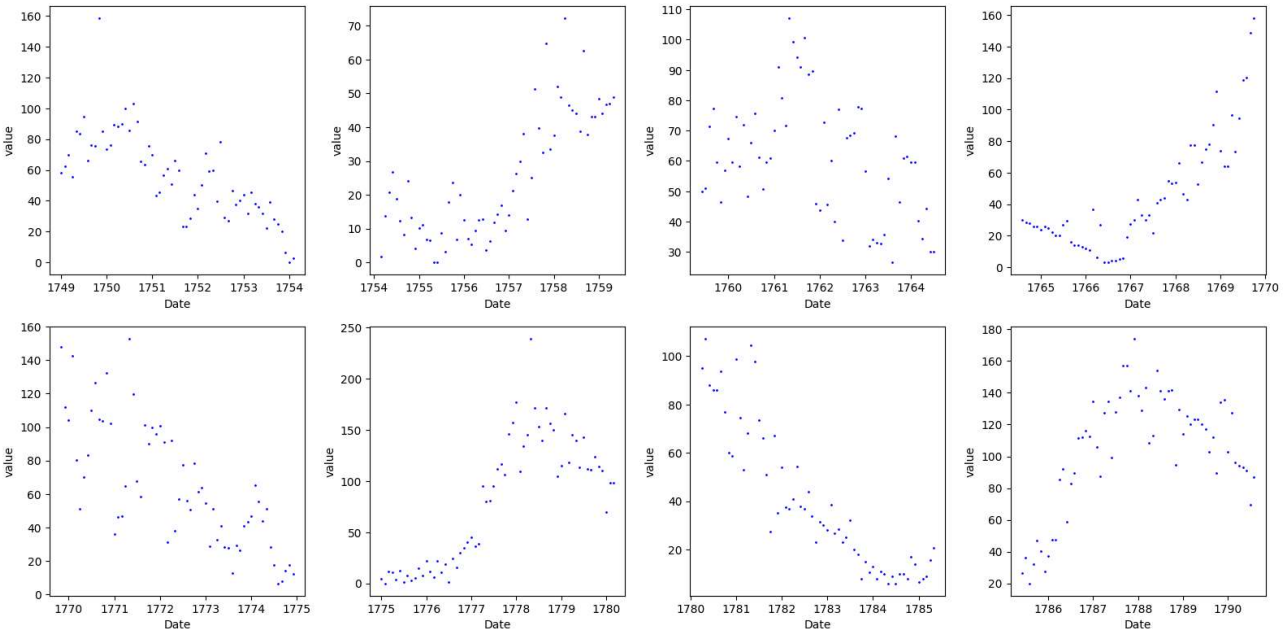
3- Représentation

In [ ]:
```
plt.figure(figsize=(16,8))
plt.plot(pd.to_datetime(df_4['Date']), df_4['Sunspots'], c='b')
plt.xlabel("Date")
plt.ylabel("value")
# plt.title("Temporal serie for prooduction")
plt.show()
```



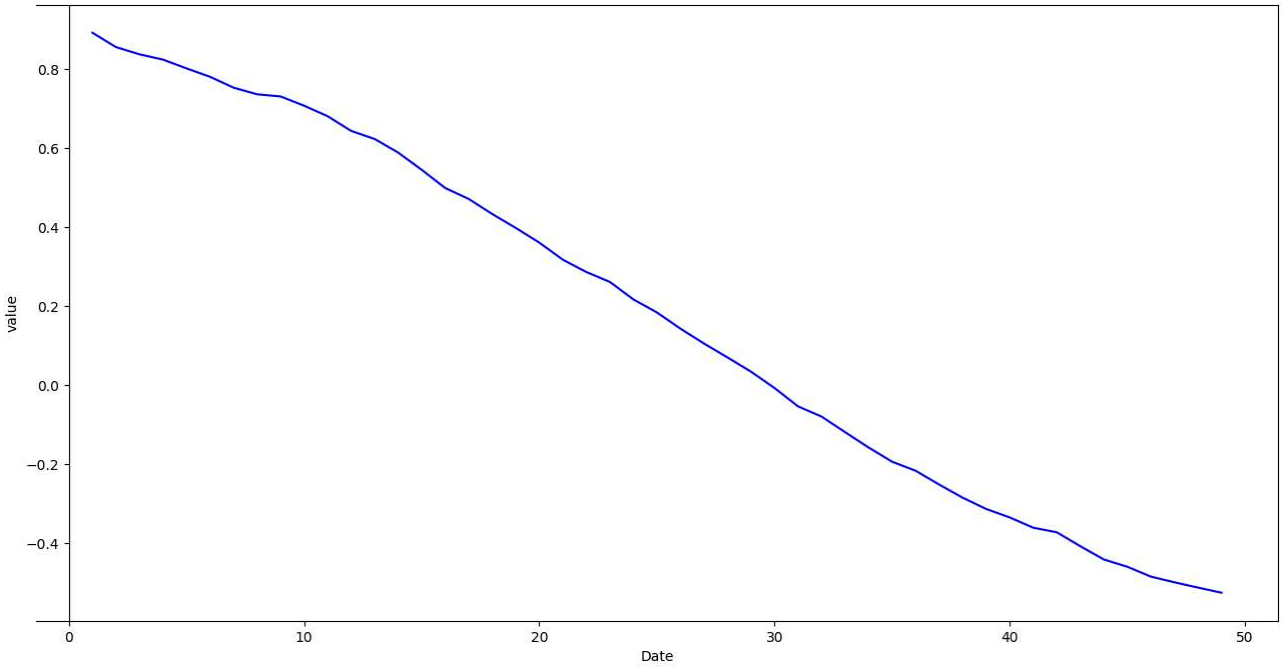
4- Nuage de points

In [ ]:
```
fig_4, axs_4 = plt.subplots(2, 4, figsize=(16, 8))

for i, ax in enumerate(axs_4.flat):
    start_idx = i * len(df_4['Date'])//8
    end_idx = (i+1)*len(df_4['Date'])//8
    ax.scatter(pd.to_datetime(df_4['Date'][start_idx:end_idx]), df_4['Sunspots'][start_idx:end_idx], s=1, c='b')
    ax.set_xlabel("Date")
    ax.set_ylabel("value")
# plt.title("Temporal serie for prooduction")
plt.tight_layout()
plt.show()
```



5- Auto-correlation

In [ ]:
```
fig, ax = plt.subplots(figsize=(16, 8))
autocorr = [df_4['Sunspots'].autocorr(lag=lag) for lag in range(1, 50)]
lags = range(1, 50)
ax.plot(lags, autocorr, c='b')
ax.spines['left'].set_position('zero')
plt.xlabel("Date")
plt.ylabel("value")
# plt.title("Temporal serie for prooduction")
plt.show()
```



1- Étude production électrique

In [ ]:
```
filepath_5 = "dataset/AirPassengers.csv"
df_5   = pd.read_csv(filepath_5);
```

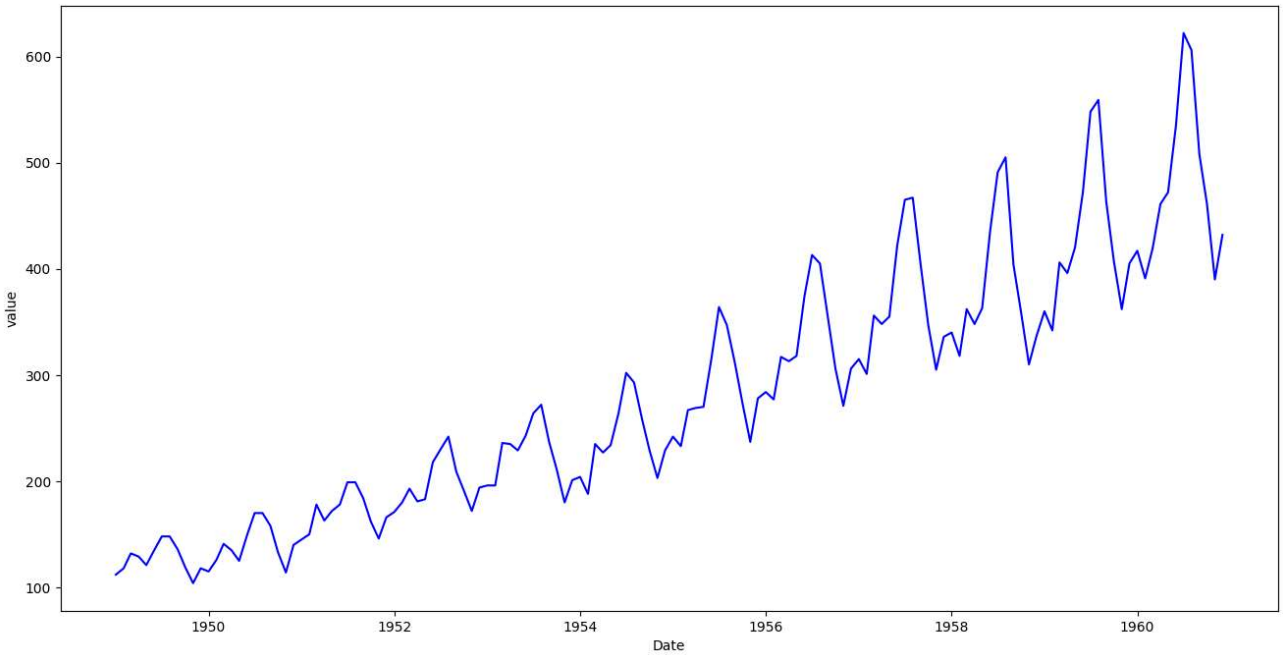
2- Moyenne, variance et Ecart-type

In [ ]:
```
mean_5 = df_5['Airpass'].mean()
var_5 = df_5['Airpass'].var()
ect_5 = df_5['Airpass'].std()
print(f"moyenne: {mean_5}\nVarieance: {var_5}\nEcart-type: {ect_5}")
```

```
moyenne: 280.2986111111111
Varieance: 14391.9172008547
Ecart-type: 119.9663169429432
```

3- Représentation

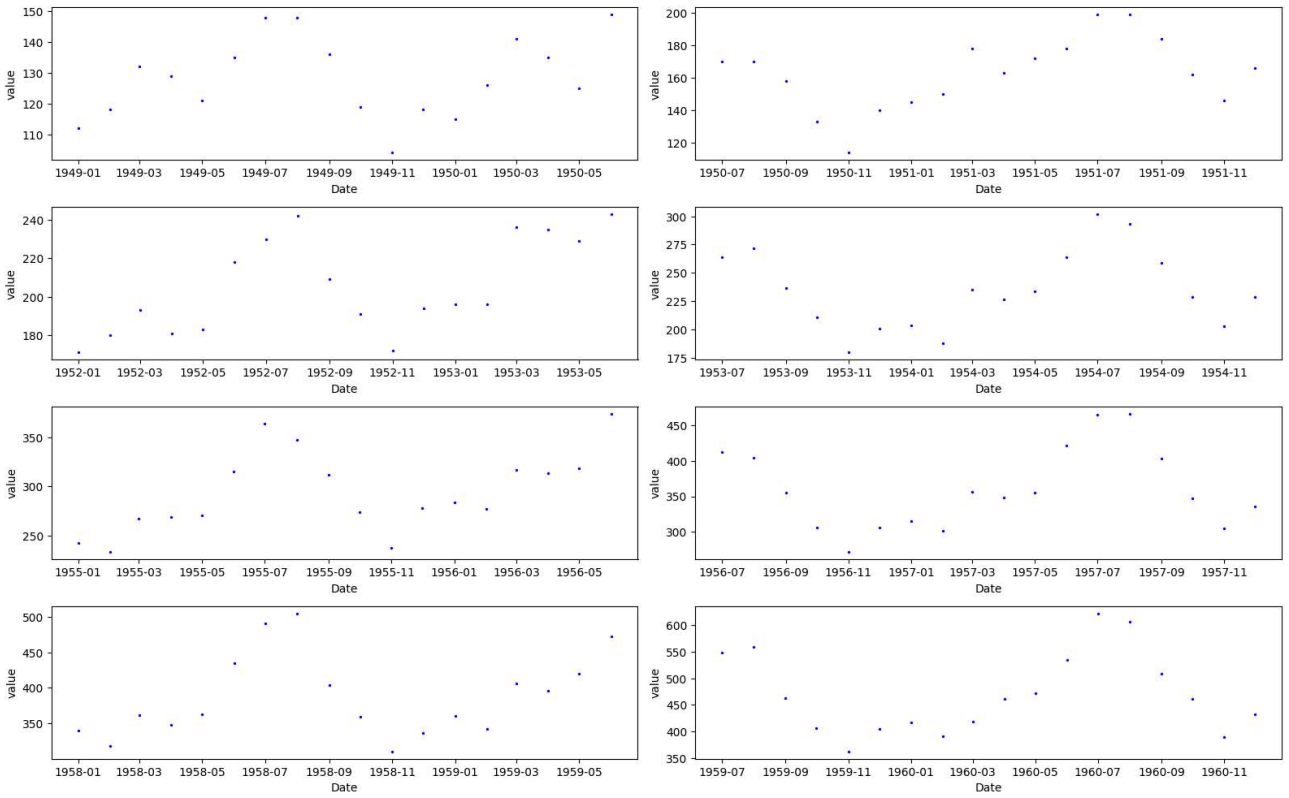
In [ ]:
```
plt.figure(figsize=(16, 8))
plt.plot(pd.to_datetime(df_5['t']), df_5['Airpass'], c='b')
plt.xlabel("Date")
plt.ylabel("value")
# plt.title("Temporal serie for prooduction")
plt.show()
```



4- Nuage de points

In [ ]:
```
fig_5, axs_5 = plt.subplots(4, 2, figsize=(16, 10))

for i, ax in enumerate(axs_5.flat):
    start_idx = i * len(df_5['t'])//8
    end_idx = (i+1)*len(df_5['t'])//8
    ax.scatter(pd.to_datetime(df_5['t'][start_idx:end_idx]), df_5['Airpass'][start_idx:end_idx], s=2, c='b')
    ax.set_xlabel("Date")
    ax.set_ylabel("value")
```

```
plt.tight_layout()
plt.show()
```



5- Auto-correlation

In [ ]:
```
autocorr = [df_5['Airpass'].autocorr(lag=lag) for lag in range(1, 50)]
lags = range(1, 50)
plt.figure(figsize=(16, 8))
plt.plot(lags, autocorr, c='b')
plt.xlabel("Date")
plt.ylabel("value")
# plt.title("Temporal serie for prooduction")
plt.show()
```