EXERCICE SUR LES SERIES TEMPORELLES

NIKOUM MODESTE LORENE 21T2580

In [13]:
```
import pandas as pd
import matplotlib.pyplot as plot
import numpy as num
```

In [14]:
```
data=pd.read_csv("./jeu_de_donnée/jeu_data1.csv")
```

In [15]:
```
data.head(7)
```

Out[15]:

	Period	Revenue	Sales_quantity	Average_cost	The_average_annual_payrol
0	01.01.2015	1.601007e+07	12729.0	1257.763541	
1	01.02.2015	1.580759e+07	11636.0	1358.507000	
2	01.03.2015	2.204715e+07	15922.0	1384.697024	
3	01.04.2015	1.881458e+07	15227.0	1235.606705	
4	01.05.2015	1.402148e+07	8620.0	1626.621765	
5	01.06.2015	1.678393e+07	13160.0	1275.374508	
6	01.07.2015	1.916189e+07	17254.0	1110.576805	

valeur statisque

In [21]:
```
data["Revenue"].describe()
```

Out[21]:
```
count    6.400000e+01
mean     3.236045e+07
std      1.164150e+07
min      1.402148e+07
25%      2.242655e+07
50%      3.209088e+07
75%      3.992999e+07
max      5.875647e+07
Name: Revenue, dtype: float64
```

In [22]:
```
moyenne=num.mean(data["Revenue"])
print('moyenne', moyenne)
variance=num.std(data["Revenue"])**2
print('variance',variance)
ecart_type=num.std(data["Revenue"])
print('ecart-type', ecart_type)
```

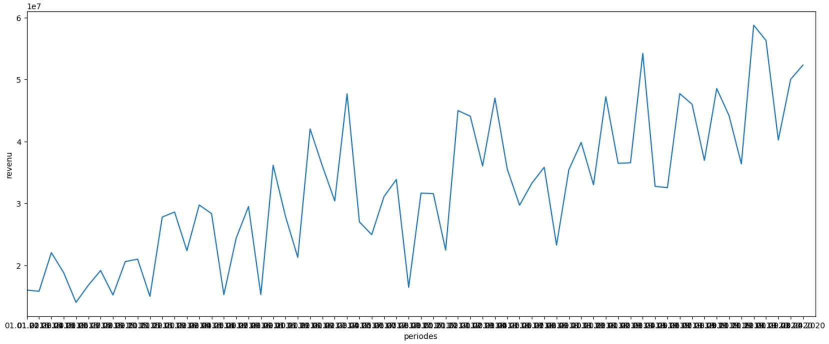
```
moyenne 32360452.25942575
variance 133406918096438.39
ecart-type 11550191.258002544
```

Representation de la serie temporelle

In [23]:
```
data.dropna(inplace=True)
#suppression des valeurs manquantes
```

In [24]:
```
plot.figure(figsize=[18,7])
plot.plot(data["Period"],data["Revenue"])
plot.xlim(0,64)
plot.xlabel("periodes")
plot.ylabel("revenu")
```

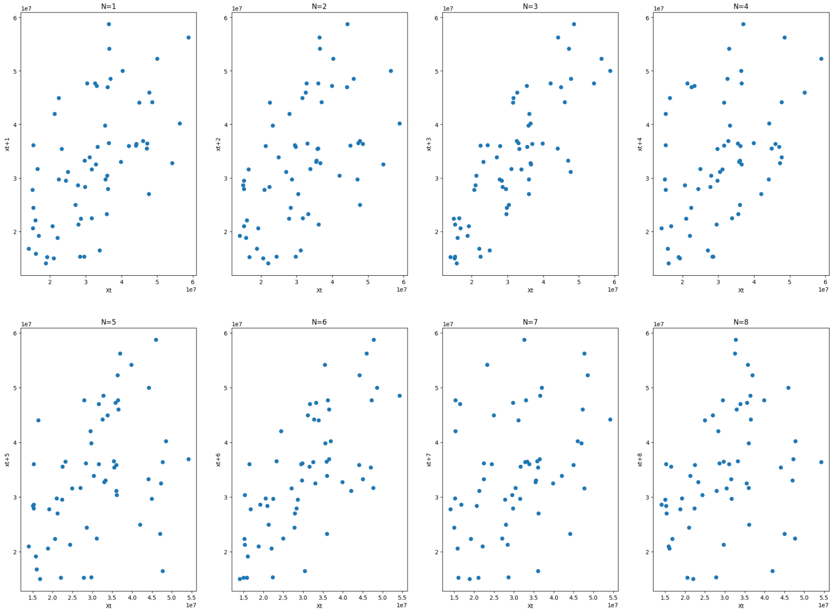
Out[24]:
```
Text(0, 0.5, 'revenu')
```



Différents nuages de point

In [22]:
```
#N=1: (0-62)(1,63)
#sur les revenues
```

In [25]:
```
plot.figure(figsize=[25,18])
for N in range(1,9):
    plot.subplot(2,4,N)
    debut=0
    fin=64
    plot.title(f"N={N}")
    plot.xlabel("Xt")
    plot.ylabel(f"xt+{N}")
    plot.scatter(data["Revenue"][debut:fin-N],data["Revenue"][debut+N:fin
```



Représenter la courbe des auto-corrélations ρ̂n (k) avec

k = 1...50 (50 valeurs de k)

In [26]:
```
moy=num.mean(data["Revenue"])
```

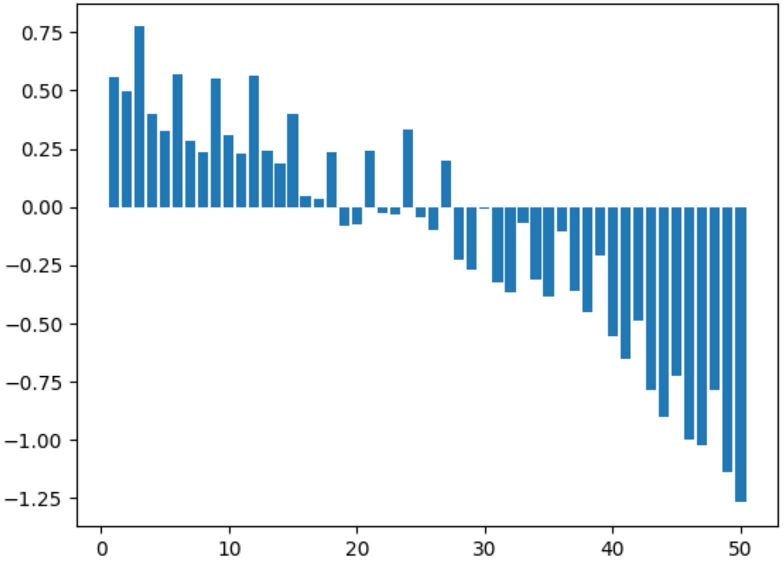
In [27]:
```
def auto_cov(data,k,moy):
    debut=0
    fin=len(data)
    xt=data[debut:fin-k]
    xt_k=data[debut+k:fin]
    cov=0
    for i in range(0,fin-k):
        cov=cov+(xt[i]-moy)*(xt_k[i]-moy)
        #calcul de L’auto-covariance empirique
    return (cov/(fin-k))
```

In [28]:
```
def auto_corr(data,k,moy):
    cov_0=auto_cov(data,0,moy)
    cov_k=auto_cov(data,k,moy)
    return (cov_k/cov_0)
```

In [29]:
```
correlation_k=[]
d=list(data["Revenue"])
for cov in range (1,51):
    res=auto_corr(d,cov,moy)
    correlation_k.append(res)
```

In [30]:
```
views=[i for i in range(1,51)]
plot.bar(views,correlation_k)
```

Out[30]:
```
<BarContainer object of 50 artists>
```



In [ ]: