Import packages

In [23]:
```python
import pandas as pd
import numpy as np
import datetime
from datetime import timedelta
import time
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline
```

Read file

In [3]:
```python
Result = pd.read_csv('multizoneout.csv')
Result
```

Out[3]:

	Date/Time	ZONE_PERI0:Zone Lights Electric Energy [J] (Monthly)	ZONE_PERI1:Zone Lights Electric Energy [J] (Monthly)	ZONE_PERI2:Zone Lights Electric Energy [J] (Monthly)	ZONE_PERI3:Zone Lights Electric Energy [J] (Monthly)	ZONE_CORE:Zone Lights Electric Energy [J] (Monthly)	ZONE_PERI0 IDEAL LOADS AIR:Zone Ideal Loads Zone Total Heating Energy [J] (Monthly)
0	January	2.882522e+08	562446255.5	464236718.6	2.900251e+08	1505433600	0.000044
1	February	1.869031e+08	347760352.0	296057707.9	1.785919e+08	1368576000	0.000061
2	March	1.620329e+08	272353500.3	258378449.5	1.400399e+08	1573862400	1011.257795
3	April	8.409551e+07	148269682.2	147264435.5	8.722369e+07	1368576000	2716.874886
4	May	9.259360e+07	160021079.0	161012553.0	9.442220e+07	1573862400	1345.737606
5	June	8.880083e+07	156003241.2	157203944.3	9.184060e+07	1505433600	0.000115
6	July	8.389293e+07	144470645.1	145010798.8	8.508432e+07	1437004800	0.000124
7	August	9.421761e+07	161683754.3	162150521.3	9.574912e+07	1573862400	0.000083
8	September	9.637661e+07	165241889.1	158860104.7	9.489405e+07	1437004800	248.816110
9	October	1.739372e+08	336222977.2	283812305.6	1.739299e+08	1505433600	3302.688920
10	November	2.690519e+08	500243796.2	428936131.8	2.582498e+08	1505433600	4569.693026
11	December	3.161101e+08	614539558.9	489034803.0	3.020278e+08	1437004800	290.472639

In [4]:
```python
## check all columes name
Result.columns.values
```

Out[4]:
```
array(['Date/Time', 'ZONE_PERI0:Zone Lights Electric Energy [J](Monthly)',
       'ZONE_PERI1:Zone Lights Electric Energy [J](Monthly)',
       'ZONE_PERI2:Zone Lights Electric Energy [J](Monthly)',
       'ZONE_PERI3:Zone Lights Electric Energy [J](Monthly)',
       'ZONE_CORE:Zone Lights Electric Energy [J](Monthly)',
       'ZONE_PERI0 IDEAL LOADS AIR:Zone Ideal Loads Zone Total Heating Energy [J](Monthly)',
       'ZONE_PERI0 IDEAL LOADS AIR:Zone Ideal Loads Zone Total Cooling Energy [J](Monthly)',
       'ZONE_PERI1 IDEAL LOADS AIR:Zone Ideal Loads Zone Total Heating Energy [J](Monthly)',
       'ZONE_PERI1 IDEAL LOADS AIR:Zone Ideal Loads Zone Total Cooling Energy [J](Monthly)',
       'ZONE_PERI2 IDEAL LOADS AIR:Zone Ideal Loads Zone Total Heating Energy [J](Monthly)',
       'ZONE_PERI2 IDEAL LOADS AIR:Zone Ideal Loads Zone Total Cooling Energy [J](Monthly)',
       'ZONE_PERI3 IDEAL LOADS AIR:Zone Ideal Loads Zone Total Heating Energy [J](Monthly)',
       'ZONE_PERI3 IDEAL LOADS AIR:Zone Ideal Loads Zone Total Cooling Energy [J](Monthly)',
       'ZONE_CORE IDEAL LOADS AIR:Zone Ideal Loads Zone Total Heating Energy [J](Monthly)',
       'ZONE_CORE IDEAL LOADS AIR:Zone Ideal Loads Zone Total Cooling Energy [J](Monthly) '],
      dtype=object)
```

Selecting columns

In [5]:
```python
## Create colume list
ColumnsList = pd.Series(Result.columns)
ColumnsList
```

Out[5]:
```
0                                             Date/Time
1     ZONE_PERI0:Zone Lights Electric Energy [J](Mon...
2     ZONE_PERI1:Zone Lights Electric Energy [J](Mon...
3     ZONE_PERI2:Zone Lights Electric Energy [J](Mon...
4     ZONE_PERI3:Zone Lights Electric Energy [J](Mon...
5     ZONE_CORE:Zone Lights Electric Energy [J](Mont...
6     ZONE_PERI0 IDEAL LOADS AIR:Zone Ideal Loads Zo...
7     ZONE_PERI0 IDEAL LOADS AIR:Zone Ideal Loads Zo...
8     ZONE_PERI1 IDEAL LOADS AIR:Zone Ideal Loads Zo...
9     ZONE_PERI1 IDEAL LOADS AIR:Zone Ideal Loads Zo...
10    ZONE_PERI2 IDEAL LOADS AIR:Zone Ideal Loads Zo...
11    ZONE_PERI2 IDEAL LOADS AIR:Zone Ideal Loads Zo...
12    ZONE_PERI3 IDEAL LOADS AIR:Zone Ideal Loads Zo...
13    ZONE_PERI3 IDEAL LOADS AIR:Zone Ideal Loads Zo...
14    ZONE_CORE IDEAL LOADS AIR:Zone Ideal Loads Zon...
15    ZONE_CORE IDEAL LOADS AIR:Zone Ideal Loads Zon...
dtype: object
```

In [6]:
```python
## Selecting your target colume list
ZoneLightingPointList = list(ColumnsList[(ColumnsList.str.endswith("Zone Lights Electric Ene
rgy [J](Monthly)"))])
ZoneLightingPointList
```

Out[6]:
```
['ZONE_PERI0:Zone Lights Electric Energy [J](Monthly)',
 'ZONE_PERI1:Zone Lights Electric Energy [J](Monthly)',
 'ZONE_PERI2:Zone Lights Electric Energy [J](Monthly)',
 'ZONE_PERI3:Zone Lights Electric Energy [J](Monthly)',
 'ZONE_CORE:Zone Lights Electric Energy [J](Monthly)']
```

In [80]:
```python
# CoreLightingElec = list(ColumnsList[(ColumnsList.str.endswith("Zone Lights Electric Energy
[J](Monthly)"))&(ColumnsList.str.contains("CORE"))])
# PERI0LightingElec = list(ColumnsList[(ColumnsList.str.endswith("Zone Lights Electric Energ
y [J](Monthly)"))&(ColumnsList.str.contains("PERI0"))])
# PERI1LightingElec = list(ColumnsList[(ColumnsList.str.endswith("Zone Lights Electric Energ
y [J](Monthly)"))&(ColumnsList.str.contains("PERI1"))])
# PERI2LightingElec = list(ColumnsList[(ColumnsList.str.endswith("Zone Lights Electric Energ
y [J](Monthly)"))&(ColumnsList.str.contains("PERI2"))])
# PERI3LightingElec = list(ColumnsList[(ColumnsList.str.endswith("Zone Lights Electric Energ
y [J](Monthly)"))&(ColumnsList.str.contains("PERI3"))])
# CoreLightingElec
```

In [7]:
```python
## Apply target lists to the dataframe
ZoneLighting = Result[ZoneLightingPointList]

## Multiply numerical values to modify your list
ZoneLighting = ZoneLighting/3600000

ZoneLighting
```

Out[7]:

	ZONE_PERI0:Zone Lights Electric Energy [J](Monthly)	ZONE_PERI1:Zone Lights Electric Energy [J](Monthly)	ZONE_PERI2:Zone Lights Electric Energy [J](Monthly)	ZONE_PERI3:Zone Lights Electric Energy [J](Monthly)	ZONE_CORE:Zone Lights Electric Energy [J](Monthly)
0	80.070060	156.235071	128.954644	80.562533	418.176
1	51.917522	96.600098	82.238252	49.608874	380.160
2	45.009151	75.653750	71.771792	38.899966	437.184
3	23.359863	41.186023	40.906788	24.228802	380.160
4	25.720444	44.450300	44.725709	26.228389	437.184
5	24.666898	43.334234	43.667762	25.511278	418.176
6	23.303591	40.130735	40.280777	23.634532	399.168
7	26.171559	44.912154	45.041811	26.596977	437.184
8	26.771282	45.900525	44.127807	26.359459	399.168
9	48.315885	93.395271	78.836752	48.313850	418.176
10	74.736644	138.956610	119.148926	71.736068	418.176
11	87.808360	170.705433	135.843001	83.896612	399.168

In [8]:
```python
## Change your dataframe colume names
ZoneLighting = ZoneLighting.rename(columns={'ZONE_PERI0:Zone Lights Electric Energy [J](Mont
hly)': 'ZONE_PERI0','ZONE_PERI1:Zone Lights Electric Energy [J](Monthly)': 'ZONE_PERI1','ZON
E_PERI2:Zone Lights Electric Energy [J](Monthly)': 'ZONE_PERI2','ZONE_PERI3:Zone Lights Elec
tric Energy [J](Monthly)': 'ZONE_PERI3','ZONE_CORE:Zone Lights Electric Energy [J](Monthly)'
: 'ZONE_CORE' })
```

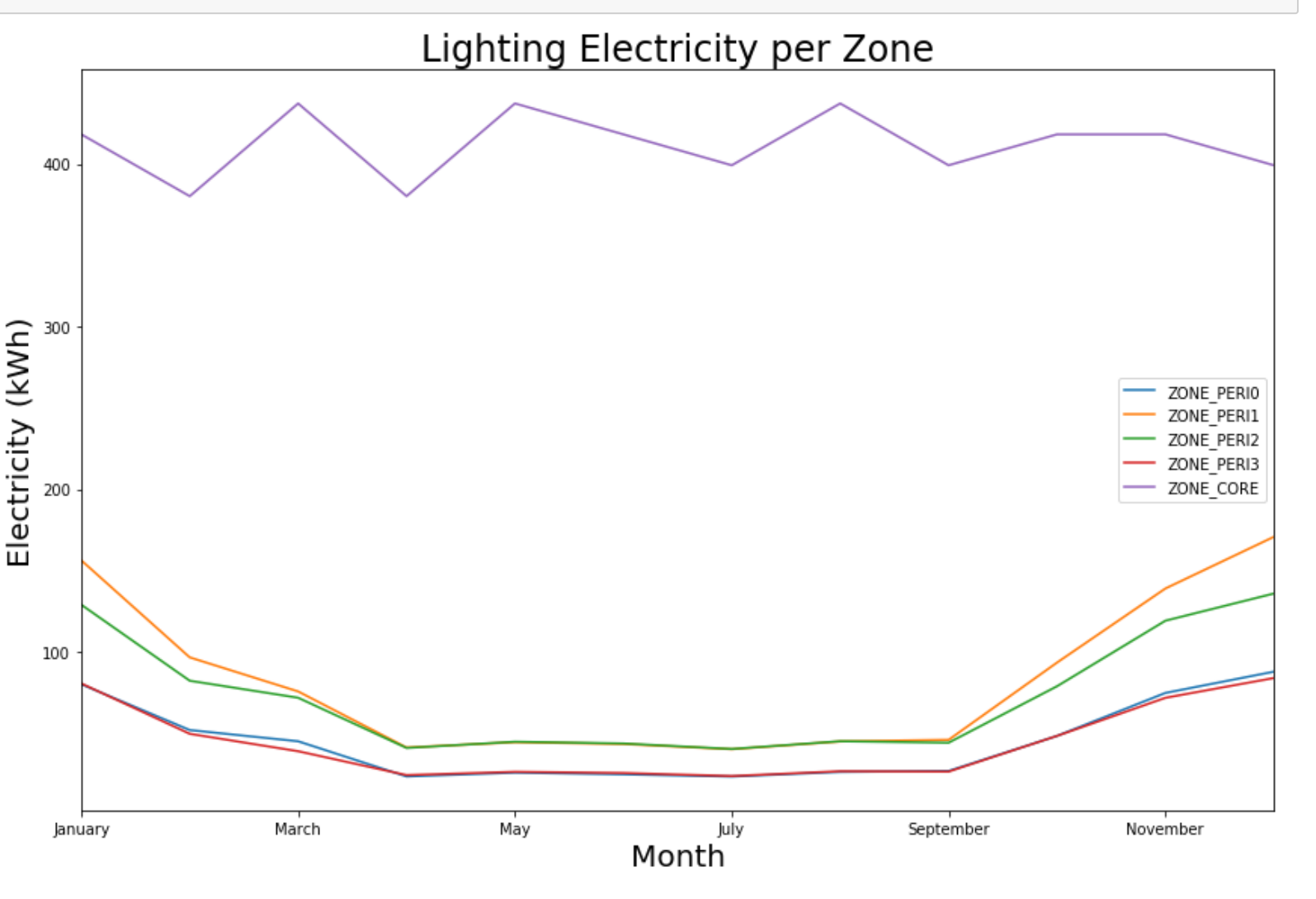
In [9]:
```python
## Set index
ZoneLighting.index = Result["Date/Time"]
ZoneLighting
```

Out[9]:

Date/Time	ZONE_PERI0	ZONE_PERI1	ZONE_PERI2	ZONE_PERI3	ZONE_CORE
January	80.070060	156.235071	128.954644	80.562533	418.176
February	51.917522	96.600098	82.238252	49.608874	380.160
March	45.009151	75.653750	71.771792	38.899966	437.184
April	23.359863	41.186023	40.906788	24.228802	380.160
May	25.720444	44.450300	44.725709	26.228389	437.184
June	24.666898	43.334234	43.667762	25.511278	418.176
July	23.303591	40.130735	40.280777	23.634532	399.168
August	26.171559	44.912154	45.041811	26.596977	437.184
September	26.771282	45.900525	44.127807	26.359459	399.168
October	48.315885	93.395271	78.836752	48.313850	418.176
November	74.736644	138.956610	119.148926	71.736068	418.176
December	87.808360	170.705433	135.843001	83.896612	399.168

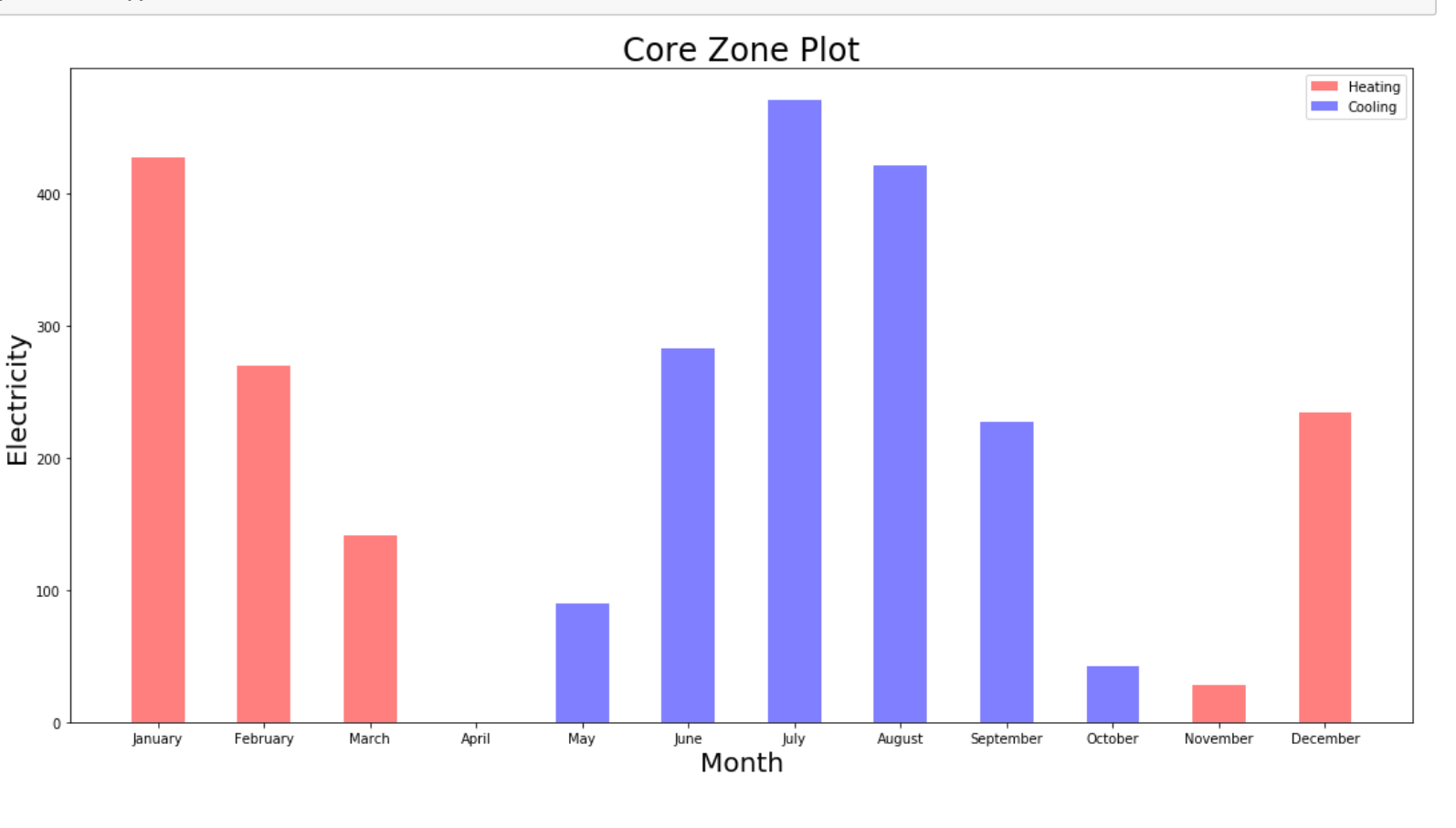
Visualization

In [10]:
```python
ZoneLighting.plot(figsize=(12,8))

plt.title("Lighting Electricity per Zone",fontsize = 24)
plt.xlabel('Month', fontsize = 20)
plt.ylabel("Electricity (kWh)",fontsize = 20)

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



In [11]:
```python
## Define your customized plot function
def VisBarPlotByAX(ax, xData, width=0.5, offset=0, color = "purple", title = 'title', YLable
="YLable" , path="", axLine1 = 0, axLine2 = 0,label='Heating' ):
    if(offset !=0):
        yTime = [i+offset for i in range(0,len(xData))]
    else:
        yTime = [i for i in range(0,len(xData))]

    ax.bar(Result["Date/Time"],xData,width=width, alpha=0.5, color=color,label=label)
    ax.set_title(title,fontsize = 24)
    ax.set_xlabel('Month',fontsize = 20)
    ax.set_ylabel(YLable,fontsize = 20)
    ax.legend(loc='best')
    if(axLine1 != 0 and axLine2 != 0 ):
        ax.set_axhline(axLine1, color='r', label=r'True $\beta_1$')
        ax.set_axhline(axLine2, color='r', label=r'True $\beta_1$')
        ax.set_axhspan(axLine1, axLine2, facecolor='0.5', alpha=0.5,label=r'True $\beta_1$')
```

In [12]:
```python
## Visuzlization
# 'ZONE_CORE IDEAL LOADS AIR:Zone Ideal Loads Zone Total Heating Energy [J](Monthly)',
#        'ZONE_CORE IDEAL LOADS AIR:Zone Ideal Loads Zone Total Cooling Energy [J](Monthly)
 '

## Set two variables to visualize
bar1 = Result['ZONE_CORE IDEAL LOADS AIR:Zone Ideal Loads Zone Total Heating Energy [J](Mont
hly)']/3600000
bar2 = Result['ZONE_CORE IDEAL LOADS AIR:Zone Ideal Loads Zone Total Cooling Energy [J](Mont
hly) ']/3600000

ind=Result["Date/Time"]
width =0.2

path = "barChartPlot"
fig = plt.figure(figsize=(15, 8))
ax = fig.add_subplot(1, 1, 1)

VisBarPlotByAX(ax, bar1, width = 0.5, color="red", YLable="Electricity",label='Heating', tit
le = "Core Zone Plot")
VisBarPlotByAX(ax, bar2, width=0.5,offset=0.2, color="blue", YLable="Electricity",label='Coo
ling', title = "Core Zone Plot")


plt.savefig(path)
plt.tight_layout()
plt.show()
```



Read multiple files

In [27]:
```python
from os import walk
import numpy as np

mypath = 'Desktop/params'
f = []
for (dirpath, dirnames, filenames) in walk(mypath):
    f.extend(filenames)
    break

print (f)
#f = f[1:]
#print (f)
```

```
['0.2.txt', '0.4.txt', '0.6.txt', '0.8.txt', '1.txt']
```

In [29]:
```python
dfs = []
for i in f:
    fn = mypath + '/' + i
    dfs.append(pd.read_csv(fn, header = None))
```

In [30]:
```python
dfs
```

Out[30]:
```
[           0          1  2             3
 0   54.031211   7.914282  3  4.873598e+07
 1   55.031211   7.914282  3  6.715399e+07
 2   56.031211   7.914282  3  9.396396e+07
 3   57.031211   7.914282  3  1.126246e+08
 4   58.031211   7.914282  3  1.483804e+08
 5   59.031211   7.914282  3  1.503511e+08
 6   54.031211   8.914282  3  1.623807e+08
 7   55.031211   8.914282  3  1.410284e+08
 8   56.031211   8.914282  3  1.059134e+08
 9   57.031211   8.914282  3  7.938370e+07
 10  58.031211   8.914282  3  5.249855e+07
 11  59.031211   8.914282  3  4.169823e+07
 12  54.031211   9.914282  3  4.860107e+07
 13  55.031211   9.914282  3  6.676811e+07
 14  56.031211   9.914282  3  9.286495e+07
 15  57.031211   9.914282  3  1.110360e+08
 16  58.031211   9.914282  3  1.466968e+08
 17  59.031211   9.914282  3  1.487503e+08
 18  54.031211  10.914282  3  1.604486e+08
 19  55.031211  10.914282  3  1.392035e+08
 20  56.031211  10.914282  3  1.043957e+08
 21  57.031211  10.914282  3  7.868045e+07
 22  58.031211  10.914282  3  5.231251e+07
 23  59.031211  10.914282  3  4.160872e+07
 24  54.031211  11.914282  3  4.835347e+07
 25  55.031211  11.914282  3  6.617352e+07
 26  56.031211  11.914282  3  9.146389e+07
 27  57.031211  11.914282  3  1.094248e+08
 28  58.031211  11.914282  3  1.444526e+08
 29  59.031211  11.914282  3  1.466347e+08,
            0          1  2             3
 0   54.031211   7.914282  3  4.873598e+07
 1   55.031211   7.914282  3  6.715399e+07
 2   56.031211   7.914282  3  9.396396e+07
 3   57.031211   7.914282  3  1.126246e+08
 4   58.031211   7.914282  3  1.483804e+08
 5   59.031211   7.914282  3  1.503511e+08
 6   54.031211   8.914282  3  1.623807e+08
 7   55.031211   8.914282  3  1.410284e+08
 8   56.031211   8.914282  3  1.059134e+08
 9   57.031211   8.914282  3  7.938370e+07
 10  58.031211   8.914282  3  5.249855e+07
 11  59.031211   8.914282  3  4.169823e+07
 12  54.031211   9.914282  3  4.860107e+07
 13  55.031211   9.914282  3  6.676811e+07
 14  56.031211   9.914282  3  9.286495e+07
 15  57.031211   9.914282  3  1.110360e+08
 16  58.031211   9.914282  3  1.466968e+08
 17  59.031211   9.914282  3  1.487503e+08
 18  54.031211  10.914282  3  1.604486e+08
 19  55.031211  10.914282  3  1.392035e+08
 20  56.031211  10.914282  3  1.043957e+08
 21  57.031211  10.914282  3  7.868045e+07
 22  58.031211  10.914282  3  5.231251e+07
 23  59.031211  10.914282  3  4.160872e+07
 24  54.031211  11.914282  3  4.835347e+07
 25  55.031211  11.914282  3  6.617352e+07
 26  56.031211  11.914282  3  9.146389e+07
 27  57.031211  11.914282  3  1.094248e+08
 28  58.031211  11.914282  3  1.444526e+08
 29  59.031211  11.914282  3  1.466347e+08,
            0          1  2             3
 0   54.031211   7.914282  3  4.873598e+07
 1   55.031211   7.914282  3  6.715399e+07
 2   56.031211   7.914282  3  9.396396e+07
 3   57.031211   7.914282  3  1.126246e+08
 4   58.031211   7.914282  3  1.483804e+08
 5   59.031211   7.914282  3  1.503511e+08
 6   54.031211   8.914282  3  1.623807e+08
 7   55.031211   8.914282  3  1.410284e+08
 8   56.031211   8.914282  3  1.059134e+08
 9   57.031211   8.914282  3  7.938370e+07
 10  58.031211   8.914282  3  5.249855e+07
 11  59.031211   8.914282  3  4.169823e+07
 12  54.031211   9.914282  3  4.860107e+07
 13  55.031211   9.914282  3  6.676811e+07
 14  56.031211   9.914282  3  9.286495e+07
 15  57.031211   9.914282  3  1.110360e+08
 16  58.031211   9.914282  3  1.466968e+08
 17  59.031211   9.914282  3  1.487503e+08
 18  54.031211  10.914282  3  1.604486e+08
 19  55.031211  10.914282  3  1.392035e+08
 20  56.031211  10.914282  3  1.043957e+08
 21  57.031211  10.914282  3  7.868045e+07
 22  58.031211  10.914282  3  5.231251e+07
 23  59.031211  10.914282  3  4.160872e+07
 24  54.031211  11.914282  3  4.835347e+07
 25  55.031211  11.914282  3  6.617352e+07
 26  56.031211  11.914282  3  9.146389e+07
 27  57.031211  11.914282  3  1.094248e+08
 28  58.031211  11.914282  3  1.444526e+08
 29  59.031211  11.914282  3  1.466347e+08,
            0          1  2             3
 0   54.031211   7.914282  3  4.873598e+07
 1   55.031211   7.914282  3  6.715399e+07
 2   56.031211   7.914282  3  9.396396e+07
 3   57.031211   7.914282  3  1.126246e+08
 4   58.031211   7.914282  3  1.483804e+08
 5   59.031211   7.914282  3  1.503511e+08
 6   54.031211   8.914282  3  1.623807e+08
 7   55.031211   8.914282  3  1.410284e+08
 8   56.031211   8.914282  3  1.059134e+08
 9   57.031211   8.914282  3  7.938370e+07
 10  58.031211   8.914282  3  5.249855e+07
 11  59.031211   8.914282  3  4.169823e+07
 12  54.031211   9.914282  3  4.860107e+07
 13  55.031211   9.914282  3  6.676811e+07
 14  56.031211   9.914282  3  9.286495e+07
 15  57.031211   9.914282  3  1.110360e+08
 16  58.031211   9.914282  3  1.466968e+08
 17  59.031211   9.914282  3  1.487503e+08
 18  54.031211  10.914282  3  1.604486e+08
 19  55.031211  10.914282  3  1.392035e+08
 20  56.031211  10.914282  3  1.043957e+08
 21  57.031211  10.914282  3  7.868045e+07
 22  58.031211  10.914282  3  5.231251e+07
 23  59.031211  10.914282  3  4.160872e+07
 24  54.031211  11.914282  3  4.835347e+07
 25  55.031211  11.914282  3  6.617352e+07
 26  56.031211  11.914282  3  9.146389e+07
 27  57.031211  11.914282  3  1.094248e+08
 28  58.031211  11.914282  3  1.444526e+08
 29  59.031211  11.914282  3  1.466347e+08,
            0          1  2             3
 0   54.031211   7.914282  3  4.873598e+07
 1   55.031211   7.914282  3  6.715399e+07
 2   56.031211   7.914282  3  9.396396e+07
 3   57.031211   7.914282  3  1.126246e+08
 4   58.031211   7.914282  3  1.483804e+08
 5   59.031211   7.914282  3  1.503511e+08
 6   54.031211   8.914282  3  1.623807e+08
 7   55.031211   8.914282  3  1.410284e+08
 8   56.031211   8.914282  3  1.059134e+08
 9   57.031211   8.914282  3  7.938370e+07
 10  58.031211   8.914282  3  5.249855e+07
 11  59.031211   8.914282  3  4.169823e+07
 12  54.031211   9.914282  3  4.860107e+07
 13  55.031211   9.914282  3  6.676811e+07
 14  56.031211   9.914282  3  9.286495e+07
 15  57.031211   9.914282  3  1.110360e+08
 16  58.031211   9.914282  3  1.466968e+08
 17  59.031211   9.914282  3  1.487503e+08
 18  54.031211  10.914282  3  1.604486e+08
 19  55.031211  10.914282  3  1.392035e+08
 20  56.031211  10.914282  3  1.043957e+08
 21  57.031211  10.914282  3  7.868045e+07
 22  58.031211  10.914282  3  5.231251e+07
 23  59.031211  10.914282  3  4.160872e+07
 24  54.031211  11.914282  3  4.835347e+07
 25  55.031211  11.914282  3  6.617352e+07
 26  56.031211  11.914282  3  9.146389e+07
 27  57.031211  11.914282  3  1.094248e+08
 28  58.031211  11.914282  3  1.444526e+08
 29  59.031211  11.914282  3  1.466347e+08]
```

In [97]:
```python
dfs[0]
```

Out[97]:

	0	1	2	3
0	54.031211	7.914282	3	4.917335e+07
1	55.031211	7.914282	3	6.827686e+07
2	56.031211	7.914282	3	9.948423e+07
3	57.031211	7.914282	3	1.197945e+08
4	58.031211	7.914282	3	1.539450e+08
5	59.031211	7.914282	3	1.534459e+08
6	54.031211	8.914282	3	1.668642e+08
7	55.031211	8.914282	3	1.474622e+08
8	56.031211	8.914282	3	1.129912e+08
9	57.031211	8.914282	3	8.177466e+07
10	58.031211	8.914282	3	5.303123e+07
11	59.031211	8.914282	3	4.204529e+07
12	54.031211	9.914282	3	4.917335e+07
13	55.031211	9.914282	3	6.827686e+07
14	56.031211	9.914282	3	9.948423e+07
15	57.031211	9.914282	3	1.197945e+08
16	58.031211	9.914282	3	1.539450e+08
17	59.031211	9.914282	3	1.534459e+08
18	54.031211	10.914282	3	1.668642e+08
19	55.031211	10.914282	3	1.474622e+08
20	56.031211	10.914282	3	1.129912e+08
21	57.031211	10.914282	3	8.177466e+07
22	58.031211	10.914282	3	5.303123e+07
23	59.031211	10.914282	3	4.204529e+07
24	54.031211	11.914282	3	4.917335e+07
25	55.031211	11.914282	3	6.827686e+07
26	56.031211	11.914282	3	9.948423e+07
27	57.031211	11.914282	3	1.197945e+08
28	58.031211	11.914282	3	1.539450e+08
29	59.031211	11.914282	3	1.534459e+08