Date 04-07-2021

Line Charts Part-2

In [1]:
```
import pandas as pd
import numpy as np
import plotly.offline as pyo
import plotly.graph_objs as go
```

In [2]:
```
data_solar_power_csv=pd.read_csv("Solar_Power_Generation_Data.csv",
                                 usecols=["DATE_TIME", "AC_POWER"])
data_solar_power_csv
```

Out[2]:

	DATE_TIME	AC_POWER
0	15-05-2020 00:00	0.0
1	15-05-2020 00:00	0.0
2	15-05-2020 00:00	0.0
3	15-05-2020 00:00	0.0
4	15-05-2020 00:00	0.0
...	...	...
67693	17-06-2020 23:45	0.0
67694	17-06-2020 23:45	0.0
67695	17-06-2020 23:45	0.0
67696	17-06-2020 23:45	0.0
67697	17-06-2020 23:45	0.0

67698 rows × 2 columns

In [3]:
```
data_solar_power_time_numpy_array = data_solar_power_csv["DATE_TIME"].to_numpy()
data_solar_power_time_numpy_array
```

Out[3]:
```
array(['15-05-2020 00:00', '15-05-2020 00:00', '15-05-2020 00:00', ...,
       '17-06-2020 23:45', '17-06-2020 23:45', '17-06-2020 23:45'],
      dtype=object)
```

In [4]:
```
data_solar_power_time_numpy_array[data_solar_power_time_numpy_array.size-1] = str(data_solar_power_time_numpy_array[data_solar_power_time_numpy_array.size-1]).split(' ')[1]
data_solar_power_time_numpy_array[data_solar_power_time_numpy_array.size-1]
```

Out[4]:
```
'23:45'
```

In [5]:
```
type(data_solar_power_time_numpy_array[0])
```

Out[5]:
```
str
```

In [6]:
```
for i in range(len(data_solar_power_time_numpy_array)-1):
    data_solar_power_time_numpy_array[i] = str(data_solar_power_time_numpy_array[i]).split(' ')[1]
```

In [7]:
```
data_solar_power_time_numpy_array
```

Out[7]:
```
array(['00:00', '00:00', '00:00', ..., '23:45', '23:45', '23:45'],
      dtype=object)
```

In [8]:
```
data_solar_power_csv["DATE_TIME"] = data_solar_power_time_numpy_array
```

In [9]:
```
data_solar_power_csv.columns=["TIME", "AC_POWER"]
```

In [10]:
```
data_solar_power_csv.set_index("TIME",inplace=True)
data_solar_power_csv
```

Out[10]:

	AC_POWER
TIME	
00:00	0.0
00:00	0.0
00:00	0.0
00:00	0.0
00:00	0.0
...	...
23:45	0.0
23:45	0.0
23:45	0.0
23:45	0.0
23:45	0.0

67698 rows × 1 columns

In [11]:
```
data_solar_power_csv.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 67698 entries, 00:00 to 23:45
Data columns (total 1 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   AC_POWER  67698 non-null  float64
dtypes: float64(1)
memory usage: 1.0+ MB
```

In [12]:
```
data_solar_power_csv.describe()
```

Out[12]:

	AC_POWER
count	67698.000000
mean	241.277825
std	362.112118
min	0.000000
25%	0.000000
50%	0.000000
75%	438.215000
max	1385.420000

In [13]:
```
data_solar_power_csv.index.values
```

Out[13]:
```
array(['00:00', '00:00', '00:00', ..., '23:45', '23:45', '23:45'],
      dtype=object)
```

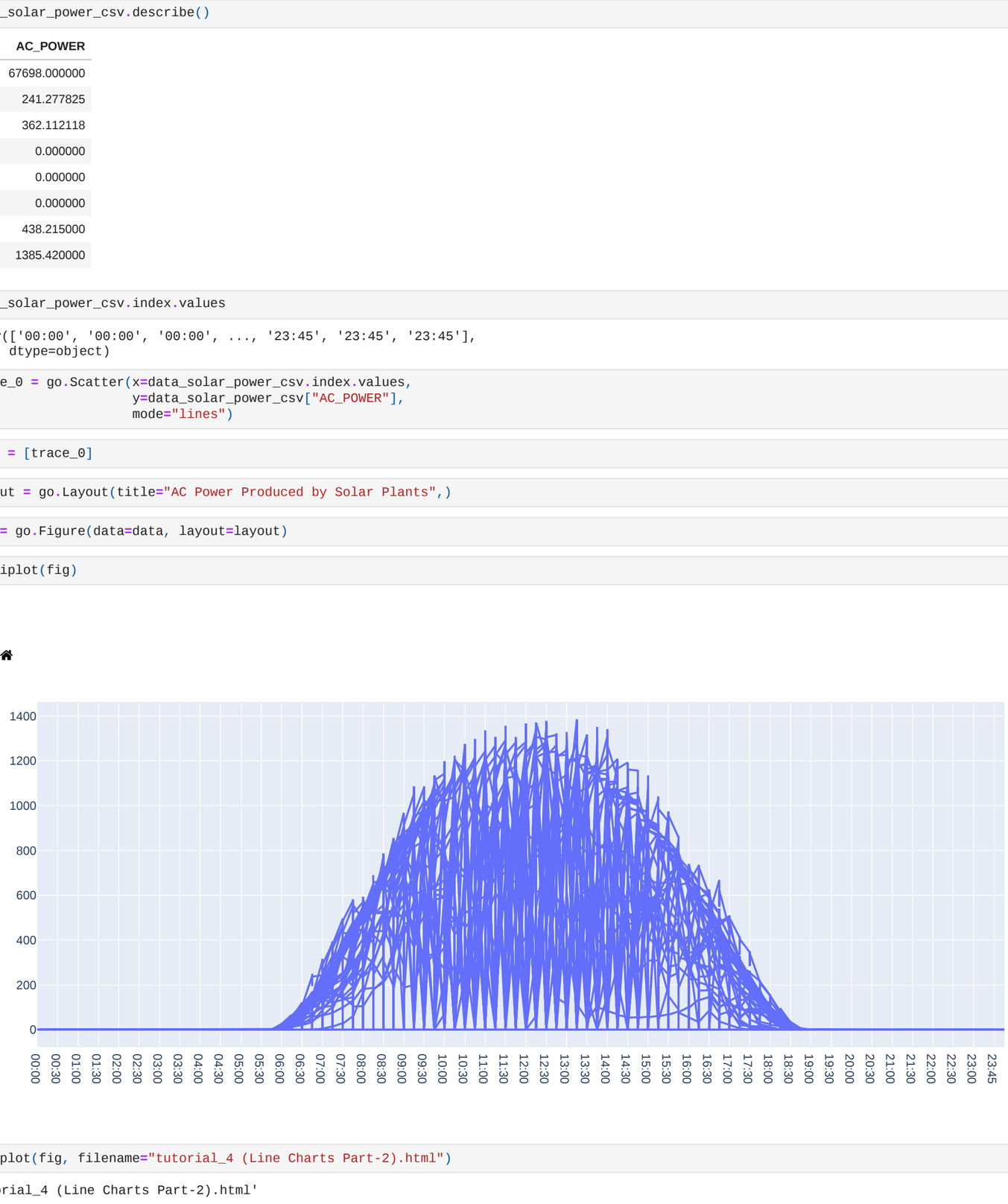
In [14]:
```
trace_0 = go.Scatter(x=data_solar_power_csv.index.values,
                     y=data_solar_power_csv["AC_POWER"],
                     mode="lines")
```

In [15]:
```
data = [trace_0]
```

In [16]:
```
layout = go.Layout(title="AC Power Produced by Solar Plants",)
```

In [17]:
```
fig = go.Figure(data=data, layout=layout)
```

In [18]:
```
pyo.iplot(fig)
```



In [19]:
```
pyo.plot(fig, filename="tutorial_4 (Line Charts Part-2).html")
```

Out[19]:
```
'tutorial_4 (Line Charts Part-2).html'
```