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

In [ ]:
```python
dane = pd.read_csv('/reviews_courses.csv',parse_dates=['Timestamp'])
dane
```

Out[ ]:

	Course Name	Timestamp	Rating	Comment
0	The Python Course: AI/ML in Python	2021-04-02 06:25:52+00:00	4.0	NaN
1	The Python Course: AI/ML in Python	2021-04-02 05:12:34+00:00	4.0	NaN
2	The Python Course: AI/ML in Python	2021-04-02 05:11:03+00:00	4.0	NaN
3	The Python Course: AI/ML in Python	2021-04-02 03:33:24+00:00	5.0	NaN
4	The Python Course: AI/ML in Python	2021-04-02 03:31:49+00:00	4.5	NaN
...	...	...	...	...
44995	The Python Course: From Beginner to Expert	2018-01-01 01:11:26+00:00	4.0	NaN
44996	The Python Course: AI/ML in Python	2018-01-01 01:09:56+00:00	5.0	NaN
44997	The Python Course: AI/ML in Python	2018-01-01 01:08:11+00:00	5.0	NaN
44998	The Python Course: From Beginner to Expert	2018-01-01 01:05:26+00:00	5.0	NaN
44999	The Python Course: AI/ML in Python	2018-01-01 01:01:16+00:00	5.0	NaN

45000 rows × 4 columns

In [ ]:
```python
df_day = dane.groupby('Timestamp').mean()
df_day.columns #zwraca tylko Rating
df_day.index #zwraca daty
```

Out[ ]:
```
Index(['2018-01-01 01:01:16+00:00', '2018-01-01 01:05:26+00:00',
       '2018-01-01 01:08:11+00:00', '2018-01-01 01:09:56+00:00',
       '2018-01-01 01:11:26+00:00', '2018-01-01 01:19:24+00:00',
       '2018-01-01 01:22:21+00:00', '2018-01-01 01:27:26+00:00',
       '2018-01-01 03:03:38+00:00', '2018-01-01 03:21:18+00:00',
       ...
       '2021-04-01 20:13:27+00:00', '2021-04-01 21:58:40+00:00',
       '2021-04-01 23:42:02+00:00', '2021-04-02 00:44:54+00:00',
       '2021-04-02 01:10:06+00:00', '2021-04-02 03:31:49+00:00',
       '2021-04-02 03:33:24+00:00', '2021-04-02 05:11:03+00:00',
       '2021-04-02 05:12:34+00:00', '2021-04-02 06:25:52+00:00'],
      dtype='object', name='Timestamp', length=44988)
```

In [ ]:
```python
dane.columns
```

Out[ ]:
```
Index(['Course Name', 'Timestamp', 'Rating', 'Comment'], dtype='object')
```

In [ ]:
```python
dane['Day'] = dane['Timestamp'].dt.date
```

In [ ]:
```python
dane
```

Out[ ]:

	Course Name	Timestamp	Rating	Comment	Day
0	The Python Course: AI/ML in Python	2021-04-02 06:25:52+00:00	4.0	NaN	2021-04-02
1	The Python Course: AI/ML in Python	2021-04-02 05:12:34+00:00	4.0	NaN	2021-04-02
2	The Python Course: AI/ML in Python	2021-04-02 05:11:03+00:00	4.0	NaN	2021-04-02
3	The Python Course: AI/ML in Python	2021-04-02 03:33:24+00:00	5.0	NaN	2021-04-02
4	The Python Course: AI/ML in Python	2021-04-02 03:31:49+00:00	4.5	NaN	2021-04-02
...	...	...	...	...	...
44995	The Python Course: From Beginner to Expert	2018-01-01 01:11:26+00:00	4.0	NaN	2018-01-01
44996	The Python Course: AI/ML in Python	2018-01-01 01:09:56+00:00	5.0	NaN	2018-01-01
44997	The Python Course: AI/ML in Python	2018-01-01 01:08:11+00:00	5.0	NaN	2018-01-01
44998	The Python Course: From Beginner to Expert	2018-01-01 01:05:26+00:00	5.0	NaN	2018-01-01
44999	The Python Course: AI/ML in Python	2018-01-01 01:01:16+00:00	5.0	NaN	2018-01-01

45000 rows × 5 columns

In [ ]:
```python
df_day = dane.groupby(['Day']).mean()
```

In [ ]:
```python
df_day
```

Out[ ]:

Day	Rating
2018-01-01	4.532609
2018-01-02	4.122807
2018-01-03	4.360465
2018-01-04	4.531250
2018-01-05	4.423077
...	...
2021-03-29	4.240000
2021-03-30	4.428571
2021-03-31	4.453125
2021-04-01	4.592593
2021-04-02	4.357143

1188 rows × 1 columns

In [ ]:
```python
dane.shape[0]
df_day.shape[0]
```

Out[ ]:
```
1188
```

In [ ]:
```python
%matplotlib inline
```

In [ ]:
```python
plt.figure(figsize = (25,8))
```

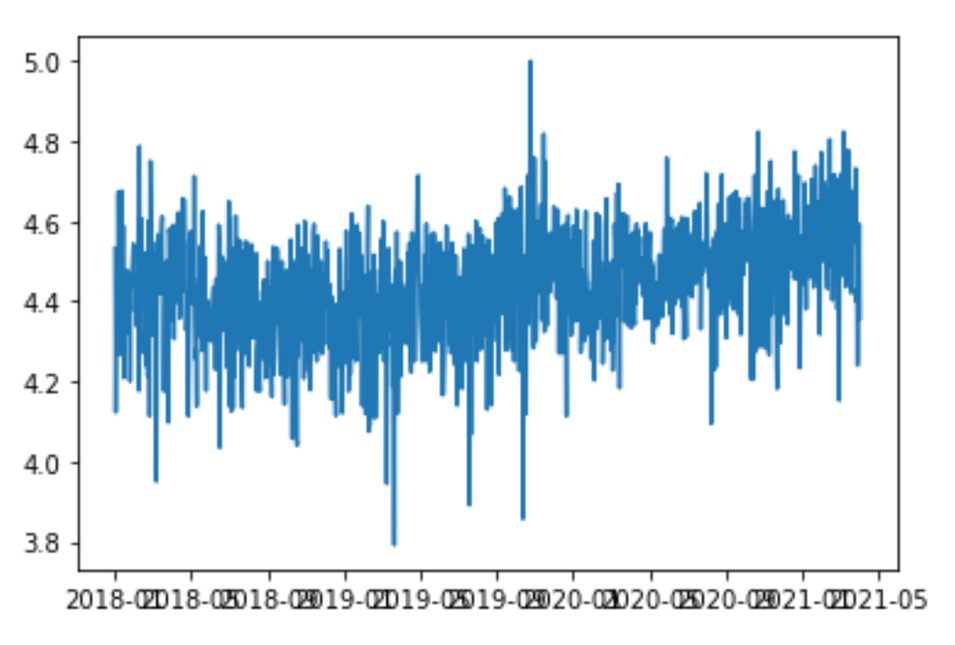
Out[ ]:
```
<Figure size 1800x576 with 0 Axes>
<Figure size 1800x576 with 0 Axes>
```

In [ ]:
```python
dane['Day'] = dane['Timestamp'].dt.date
```

In [ ]:
```python
df_2 = dane.groupby(['Day']).mean()
```

In [ ]:
```python
plt.plot(df_2)
```

Out[ ]:
```
[<matplotlib.lines.Line2D at 0x7f1995fcb8d0>]
```



In [ ]:
```python
dane_week = dane.groupby(['Week']).mean()
```

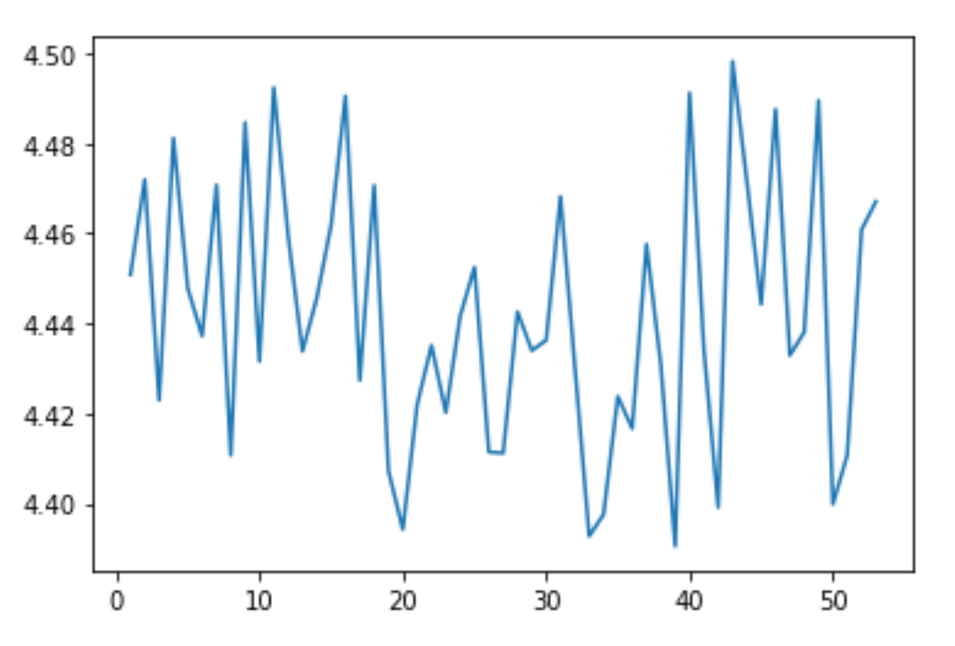
In [ ]:
```python
dane_week
```

Out[ ]:

Week	Rating
1	4.450845
2	4.471976
3	4.422941
4	4.481168
5	4.447789
6	4.437227
7	4.470739
8	4.410752
9	4.484642
10	4.431544
11	4.492391
12	4.459472
13	4.433804
14	4.445652
15	4.461895
16	4.490516
17	4.427374
18	4.470655
19	4.407242
20	4.394267
21	4.421959
22	4.435123
23	4.420213
24	4.441622
25	4.452491
26	4.411452
27	4.411221
28	4.442584
29	4.433974
30	4.436280
31	4.468215
32	4.430502
33	4.392760
34	4.397576
35	4.423767
36	4.416667
37	4.457592
38	4.430739
39	4.390561
40	4.491228
41	4.434319
42	4.399153
43	4.498236
44	4.471572
45	4.444248
46	4.487562
47	4.432917
48	4.438058
49	4.489510
50	4.399861
51	4.410741
52	4.460798
53	4.467078

In [ ]:
```python
plt.plot(dane_week)
```

Out[ ]:
```
[<matplotlib.lines.Line2D at 0x7f1995fc1750>]
```

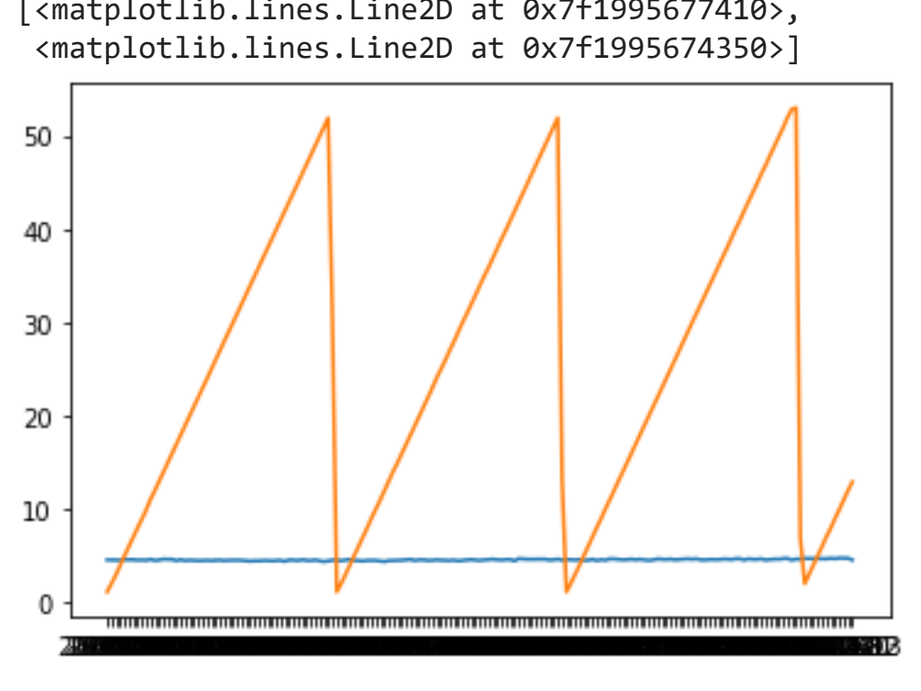


In [ ]:
```python
dane['Week2'] = dane['Timestamp'].dt.strftime("%Y-%U")
```

In [ ]:
```python
df_week2 = dane.groupby(['Week2']).mean()
```

In [ ]:
```python
plt.plot(df_week2)
```

Out[ ]:
```
[<matplotlib.lines.Line2D at 0x7f1995677410>,
 <matplotlib.lines.Line2D at 0x7f1995674350>]
```

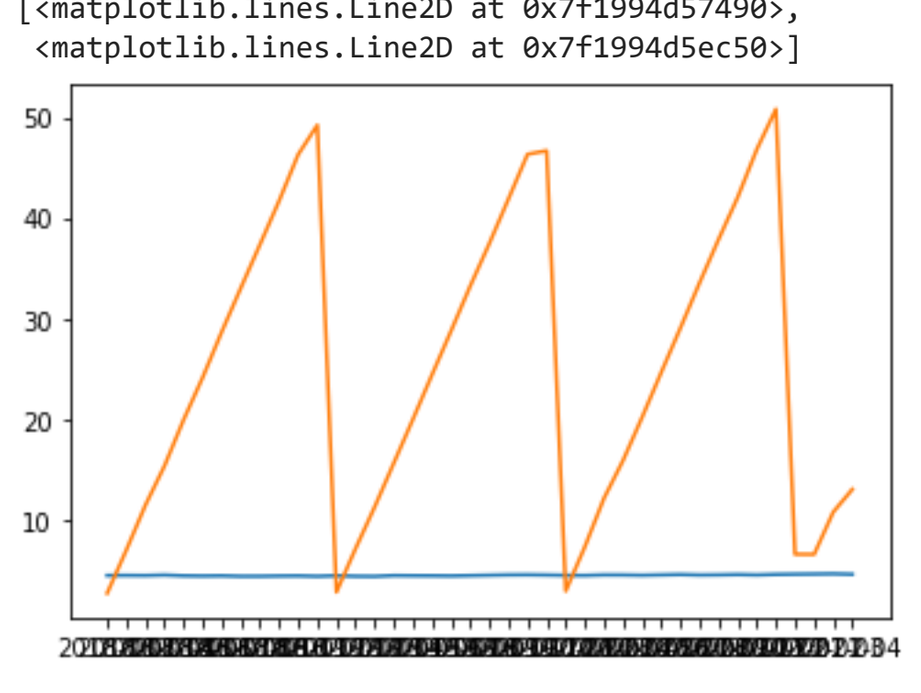


In [ ]:
```python
dane['Month'] = dane['Timestamp'].dt.strftime("%Y-%m")
```

In [ ]:
```python
df_month = dane.groupby(['Month']).mean()
```

In [ ]:
```python
plt.plot(df_month)
```

Out[ ]:
```
[<matplotlib.lines.Line2D at 0x7f1994d57490>,
 <matplotlib.lines.Line2D at 0x7f1994d5ec50>]
```



In [ ]:
```python
df_m = dane.groupby(['Month']).mean()
```

In [ ]:
```python
df_m2 = dane.groupby(['Month', 'Course Name']).mean()
```

In [ ]:
```python
df_m2
```

Out[ ]:

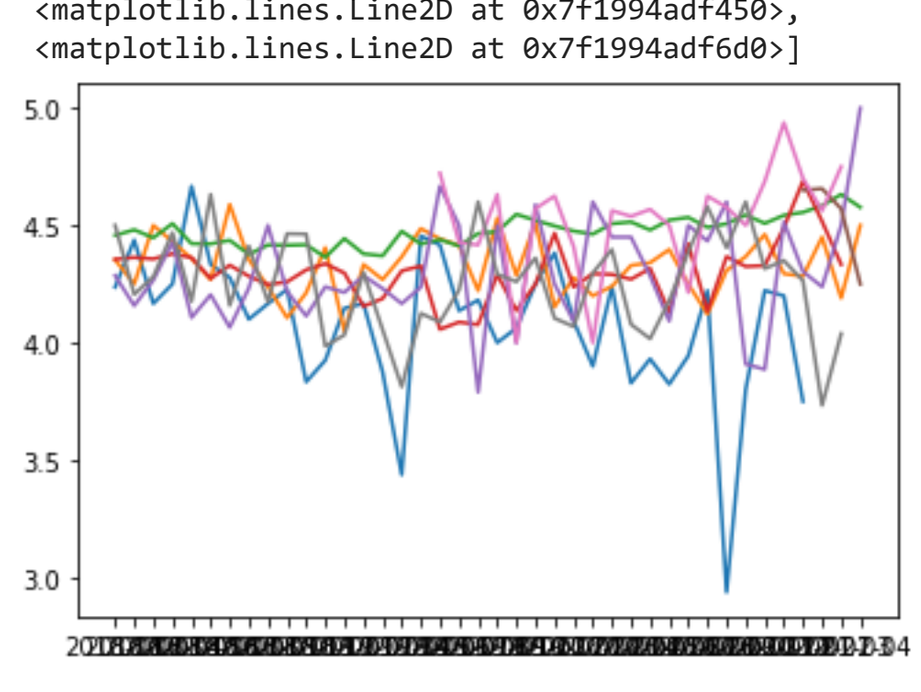
Month	Course Name	Rating	Week
2018-01	**Learn GIS in One Hour**	4.236842	3.578947
	The Python Course: 100 Excercises for Beginners	4.353448	2.741379
	The Python Course: AI/ML in Python	4.457368	2.583874
	The Python Course: From Beginner to Expert	4.355422	2.835341
	The Python Course: Interactive Visualizations	4.285714	2.942857
...	**...**	...	...
2021-03	**The Python Excercises:Data Processing by use of Python**	4.038462	11.538462
2021-04	**The Python Course: 100 Excercises for Beginners**	4.500000	13.000000
	The Python Course: AI/ML in Python	4.576923	13.000000
	The Python Course: Interactive Visualizations	5.000000	13.000000
	The Python Course: OpenCV in Python	4.250000	13.000000

262 rows × 2 columns

In [ ]:
```python
df_m3 = dane.groupby(['Month', 'Course Name']).mean().unstack()
```

In [ ]:
```python
plt.plot(df_m3.index, df_m3['Rating'])
```

Out[ ]:
```
[<matplotlib.lines.Line2D at 0x7f1994b56350>,
 <matplotlib.lines.Line2D at 0x7f1994adfbd0>,
 <matplotlib.lines.Line2D at 0x7f1994adf990>,
 <matplotlib.lines.Line2D at 0x7f1994adf890>,
 <matplotlib.lines.Line2D at 0x7f1994adf250>,
 <matplotlib.lines.Line2D at 0x7f1994adf150>,
 <matplotlib.lines.Line2D at 0x7f1994adf450>,
 <matplotlib.lines.Line2D at 0x7f1994adf6d0>]
```



In [ ]:
```python
dane['WeekDay'] = dane['Timestamp'].dt.strftime("%A-%w")
```

In [ ]:
```python
df_wd = dane.groupby(['WeekDay']).mean()
```

In [ ]:
```python
plt.plot(df_wd.index, df_wd['Rating'])
```

Out[ ]:
```
[<matplotlib.lines.Line2D at 0x7f1994451910>]
```