In [15]:
```python
%pylab inline
import pandas
import seaborn
```

Populating the interactive namespace from numpy and matplotlib

In [16]:
```python
data=pandas.read_csv('desktop/uber-raw-data-apr14.txt')
data
```

Out[16]:

	Date/Time	Lat	Lon	Base
0	4/1/2014 0:11:00	40.7690	-73.9549	B02512
1	4/1/2014 0:17:00	40.7267	-74.0345	B02512
2	4/1/2014 0:21:00	40.7316	-73.9873	B02512
3	4/1/2014 0:28:00	40.7588	-73.9776	B02512
4	4/1/2014 0:33:00	40.7594	-73.9722	B02512
...	...	...	...	...
564511	4/30/2014 23:22:00	40.7640	-73.9744	B02764
564512	4/30/2014 23:26:00	40.7629	-73.9672	B02764
564513	4/30/2014 23:31:00	40.7443	-73.9889	B02764
564514	4/30/2014 23:32:00	40.6756	-73.9405	B02764
564515	4/30/2014 23:48:00	40.6880	-73.9608	B02764

564516 rows × 4 columns

In [18]:
```python
data['Date/Time']=data['Date/Time'].map(pandas.to_datetime)
```

In [19]:
```python
data.head()
```

Out[19]:

	Date/Time	Lat	Lon	Base
0	2014-04-01 00:11:00	40.7690	-73.9549	B02512
1	2014-04-01 00:17:00	40.7267	-74.0345	B02512
2	2014-04-01 00:21:00	40.7316	-73.9873	B02512
3	2014-04-01 00:28:00	40.7588	-73.9776	B02512
4	2014-04-01 00:33:00	40.7594	-73.9722	B02512

In [20]:
```python
data['Date/Time'][0]
```

Out[20]: Timestamp('2014-04-01 00:11:00')

In [21]:
```python
def get_dom(dt):
    return dt.day
data['dom']=data['Date/Time'].map(get_dom)
data.head()
```

Out[21]:

	Date/Time	Lat	Lon	Base	dom
0	2014-04-01 00:11:00	40.7690	-73.9549	B02512	1
1	2014-04-01 00:17:00	40.7267	-74.0345	B02512	1
2	2014-04-01 00:21:00	40.7316	-73.9873	B02512	1
3	2014-04-01 00:28:00	40.7588	-73.9776	B02512	1
4	2014-04-01 00:33:00	40.7594	-73.9722	B02512	1

In [22]:
```python
def get_weekday(dt):
    return dt.weekday()
data['weekday']=data['Date/Time'].map(get_weekday)
def get_hour(dt):
    return dt.hour
data['hour']=data['Date/Time'].map(get_hour)
data.tail()
```

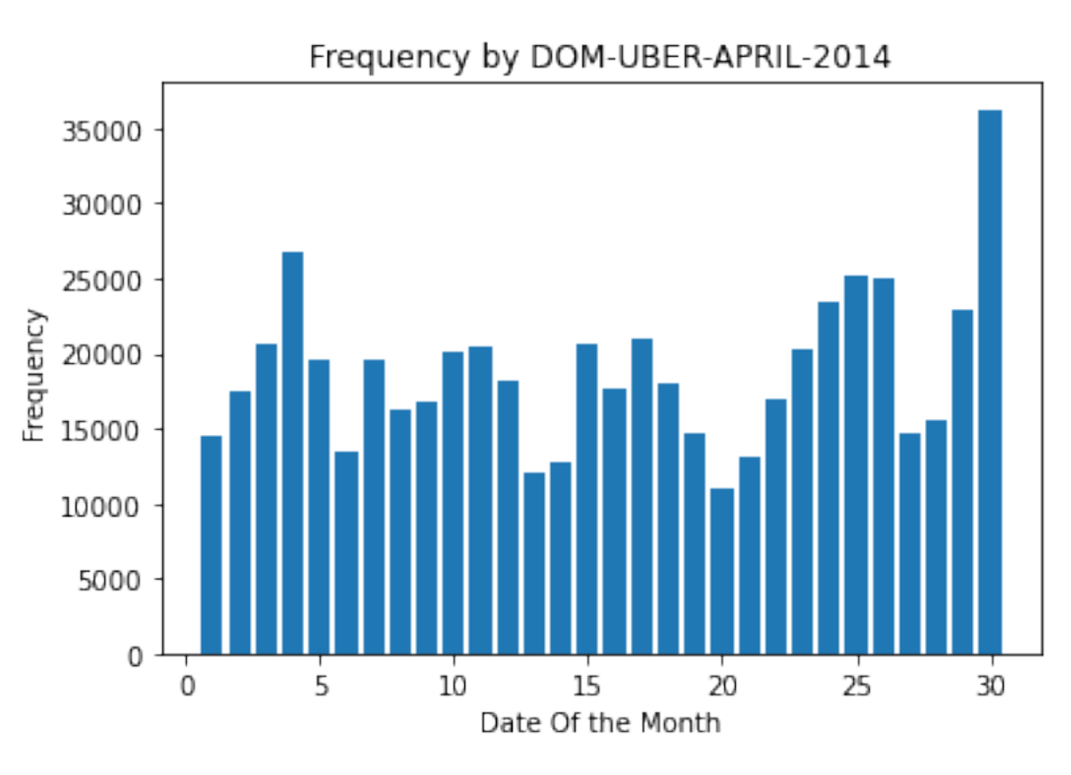
Out[22]:

	Date/Time	Lat	Lon	Base	dom	weekday	hour
564511	2014-04-30 23:22:00	40.7640	-73.9744	B02764	30	2	23
564512	2014-04-30 23:26:00	40.7629	-73.9672	B02764	30	2	23
564513	2014-04-30 23:31:00	40.7443	-73.9889	B02764	30	2	23
564514	2014-04-30 23:32:00	40.6756	-73.9405	B02764	30	2	23
564515	2014-04-30 23:48:00	40.6880	-73.9608	B02764	30	2	23

ANALYZE DOM

In [25]:
```python
hist(data.dom, bins=30,rwidth=.8, range=(0.5,30.5))
xlabel("Date Of the Month")
ylabel("Frequency")
title('Frequency by DOM-UBER-APRIL-2014')
```

Out[25]: Text(0.5, 1.0, 'Frequency by DOM-UBER-APRIL-2014')



In [26]:
```python
def count_rows(rows):
    return len(rows)
by_date=data.groupby('dom').apply(count_rows)
by_date
```

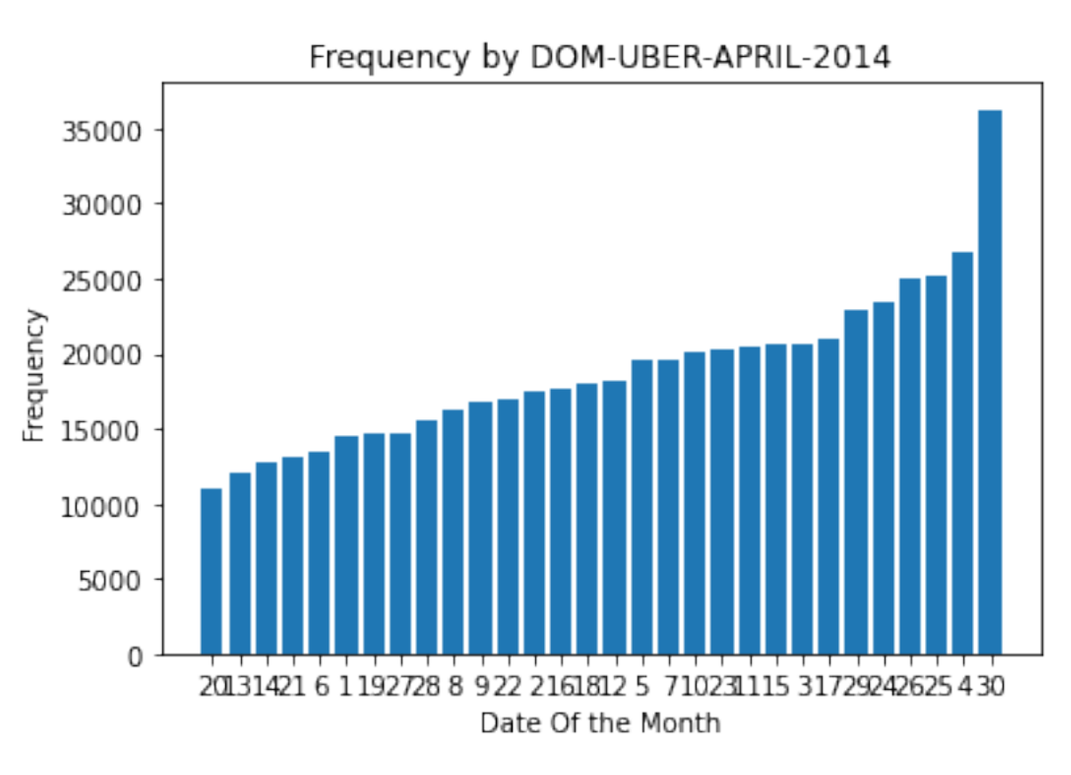
Out[26]:
```
dom
1     14546
2     17474
3     20701
4     26714
5     19521
6     13445
7     19550
8     16188
9     16843
10    20041
11    20420
12    18170
13    12112
14    12674
15    20641
16    17717
17    20973
18    18074
19    14602
20    11017
21    13162
22    16975
23    20346
24    23352
25    25095
26    24925
27    14677
28    15475
29    22835
30    36251
dtype: int64
```

In [33]:
```python
by_date_sorted=by_date.sort_values()
by_date_sorted
```

Out[33]:
```
dom
20    11017
13    12112
14    12674
21    13162
6     13445
1     14546
19    14602
27    14677
28    15475
8     16188
9     16843
22    16975
2     17474
16    17717
18    18074
12    18170
5     19521
7     19550
10    20041
23    20346
11    20420
15    20641
3     20701
17    20973
29    22835
24    23352
26    24925
25    25095
4     26714
30    36251
dtype: int64
```

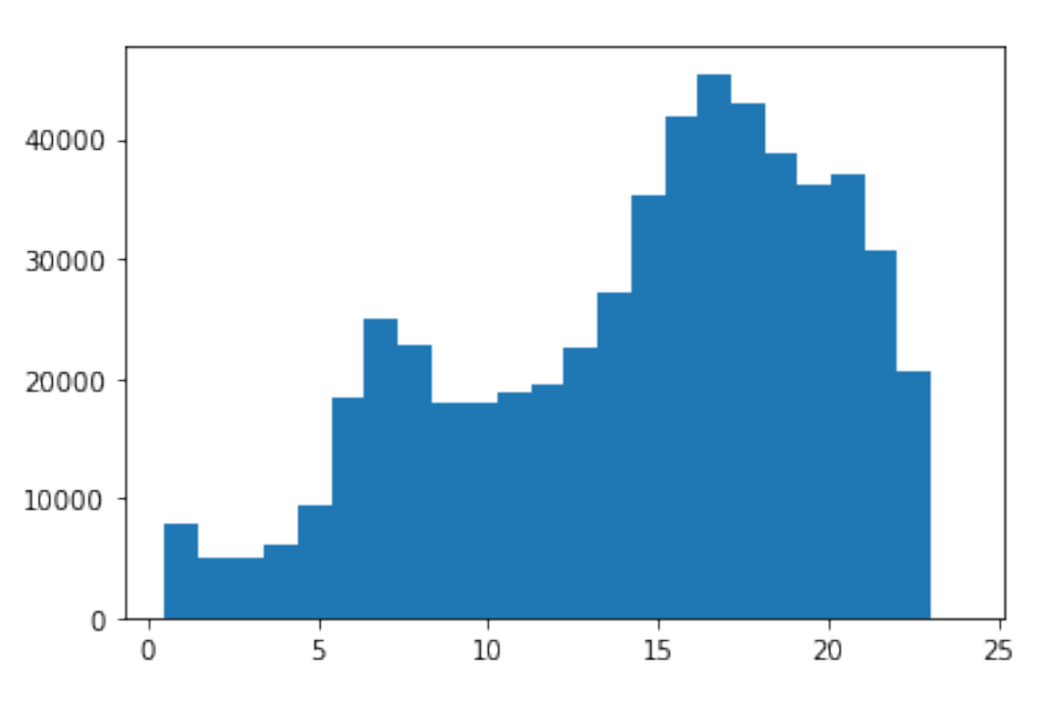
In [43]:
```python
bar( range( 1, 31), by_date_sorted)
xticks( range( 1, 31), by_date_sorted.index)
xlabel("Date Of the Month")
ylabel("Frequency")
title('Frequency by DOM-UBER-APRIL-2014')
;
```

Out[43]: ''



In [45]:
```python
hist(data.hour, bins=24, range=(0.5,24))
;
```

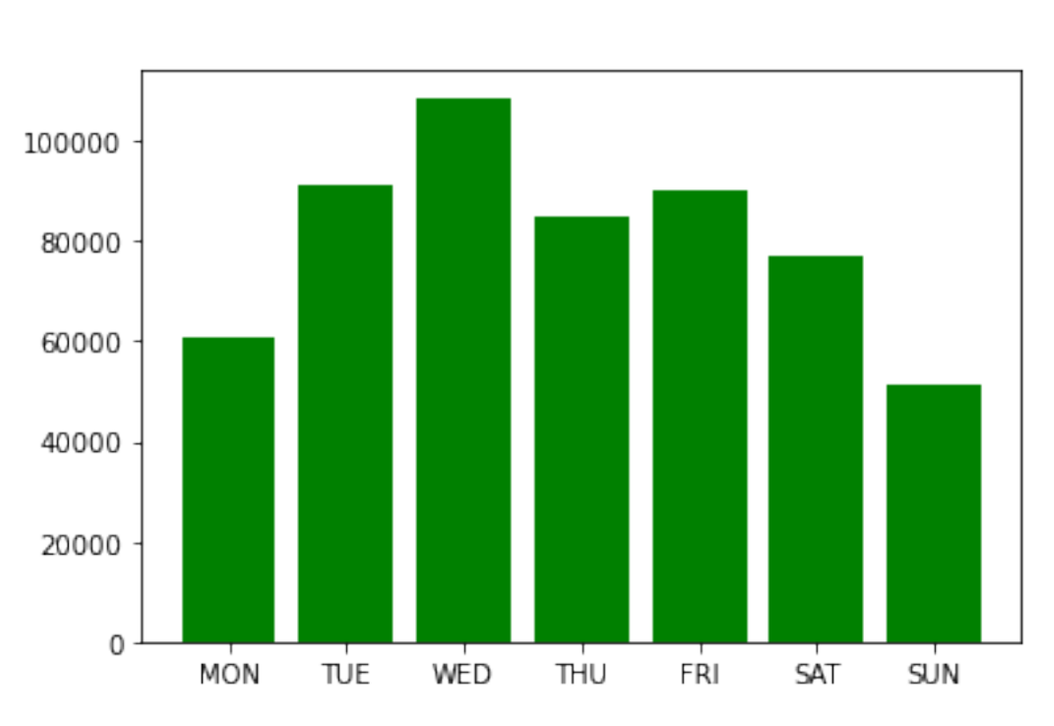
Out[45]: ''



Analyze the weekday

In [47]:
```python
hist(data.weekday, bins=7,range=(-.5,6.5),rwidth=.8,color='green')
xticks(range(7),'MON TUE WED THU FRI SAT SUN'.split())
;
```

Out[47]: ''



cross analysis (hour, day of week)

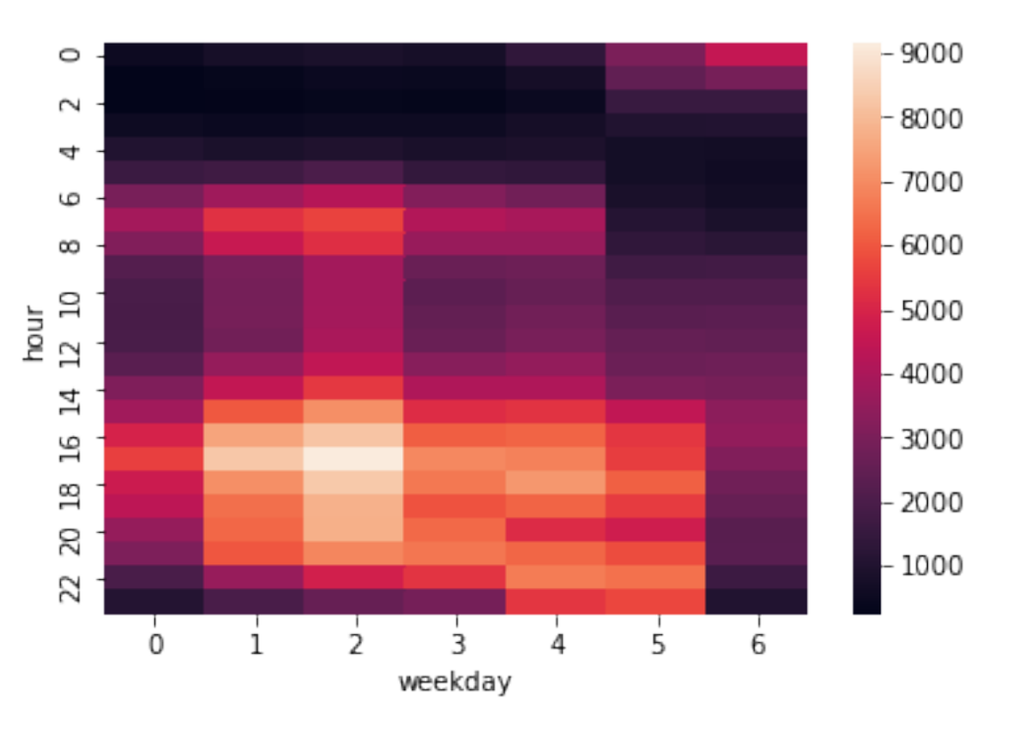
In [55]:
```python
data.groupby('hour weekday'.split()).apply(count_rows).unstack()
```

Out[55]:

weekday hour	0	1	2	3	4	5	6
0	518	765	899	792	1367	3027	4542
1	261	367	507	459	760	2479	2936
2	238	304	371	342	513	1577	1590
3	571	516	585	567	736	1013	1052
4	1021	887	1003	861	932	706	685
5	1619	1734	1990	1454	1382	704	593
6	2974	3766	4230	3179	2836	844	669
7	3888	5304	5647	4159	3943	1110	873
8	3138	4594	5242	3616	3648	1372	1233
9	2211	2962	3846	2654	2732	1764	1770
10	1953	2900	3844	2370	2599	2086	2113
11	1929	2949	3889	2516	2816	2315	2360
12	1945	2819	3988	2657	2978	2560	2478
13	2294	3556	4469	3301	3535	2685	2763
14	3117	4489	5438	4083	4087	3042	2934
15	3818	6042	7071	5182	5354	4457	3400
16	4962	7521	8213	6149	6259	5410	3489
17	5574	8297	9151	6951	6790	5558	3154
18	4725	7089	8334	6637	7258	6165	2795
19	4386	6459	7794	5929	6247	5529	2579
20	3573	6310	7783	6345	5165	4792	2276
21	3079	5993	6921	6585	6265	5811	2310
22	1976	3614	4845	5370	6708	6493	1639
23	1091	1948	2571	2909	5393	5719	1018

In [56]:
```python
data_cross=data.groupby('hour weekday'.split()).apply(count_rows).unstack()
seaborn.heatmap(data_cross)
```

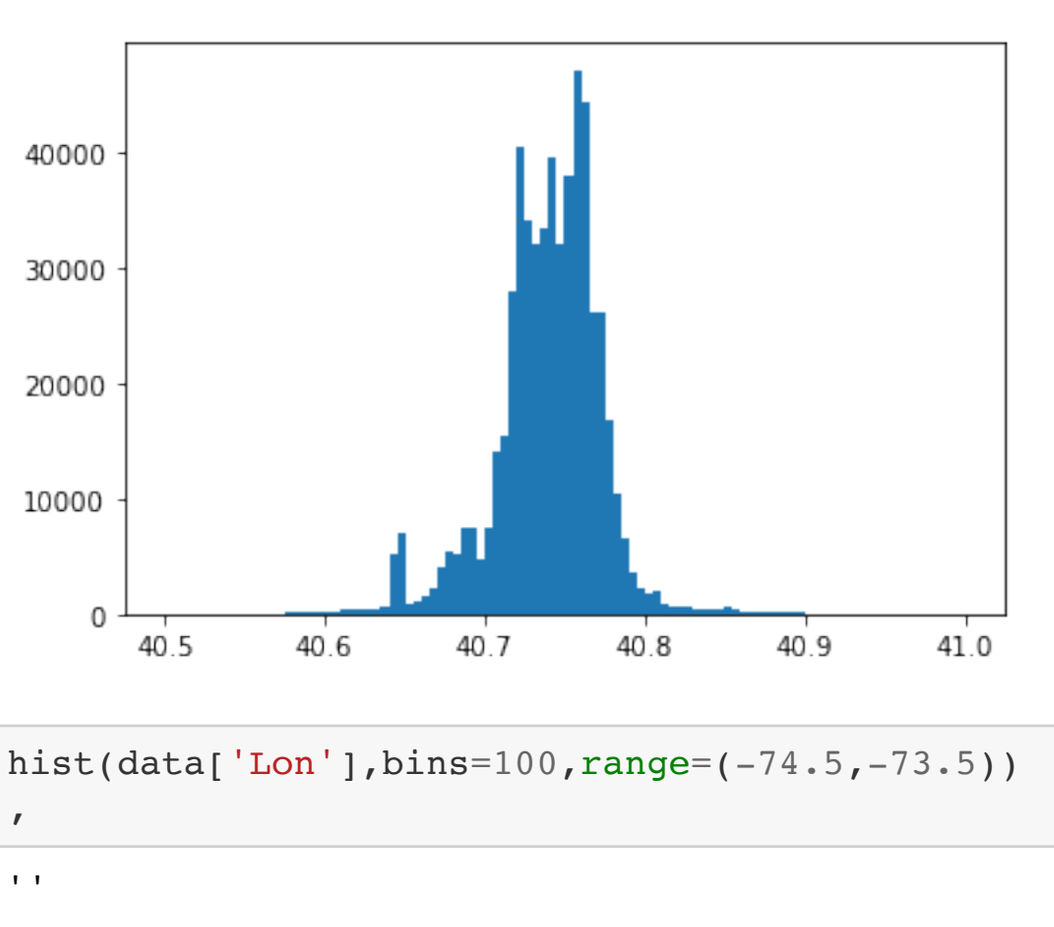
Out[56]: <matplotlib.axes._subplots.AxesSubplot at 0x7f98504ac190>



By LAT AND LON

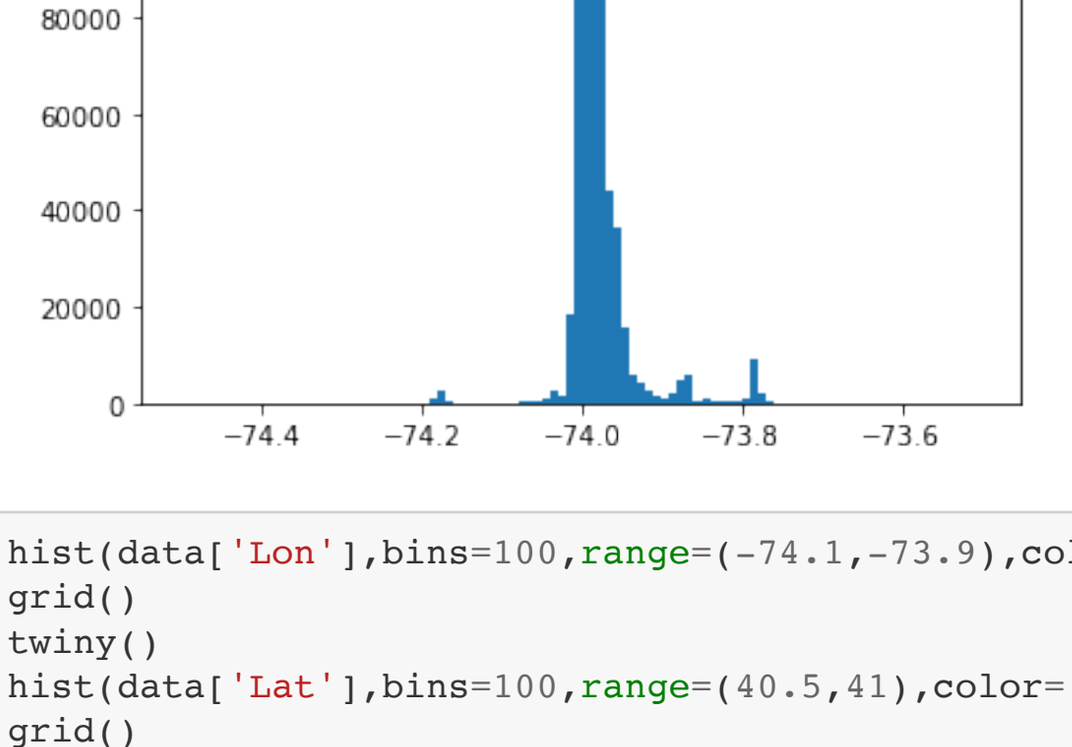
In [60]:
```python
hist(data['Lat'],bins=100,range=(40.5,41))
;
```

Out[60]: ''



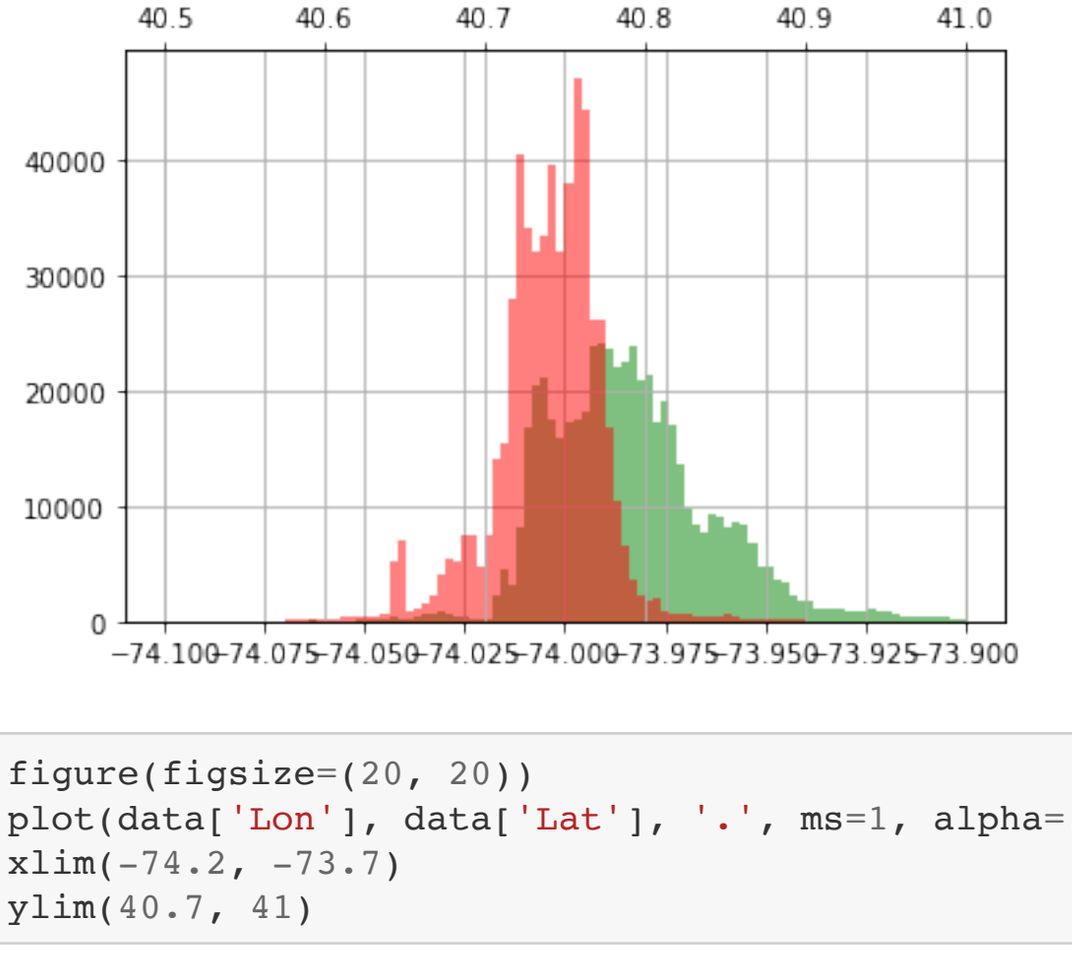
In [63]:
```python
hist(data['Lon'],bins=100,range=(-74.5,-73.5))
;
```

Out[63]: ''



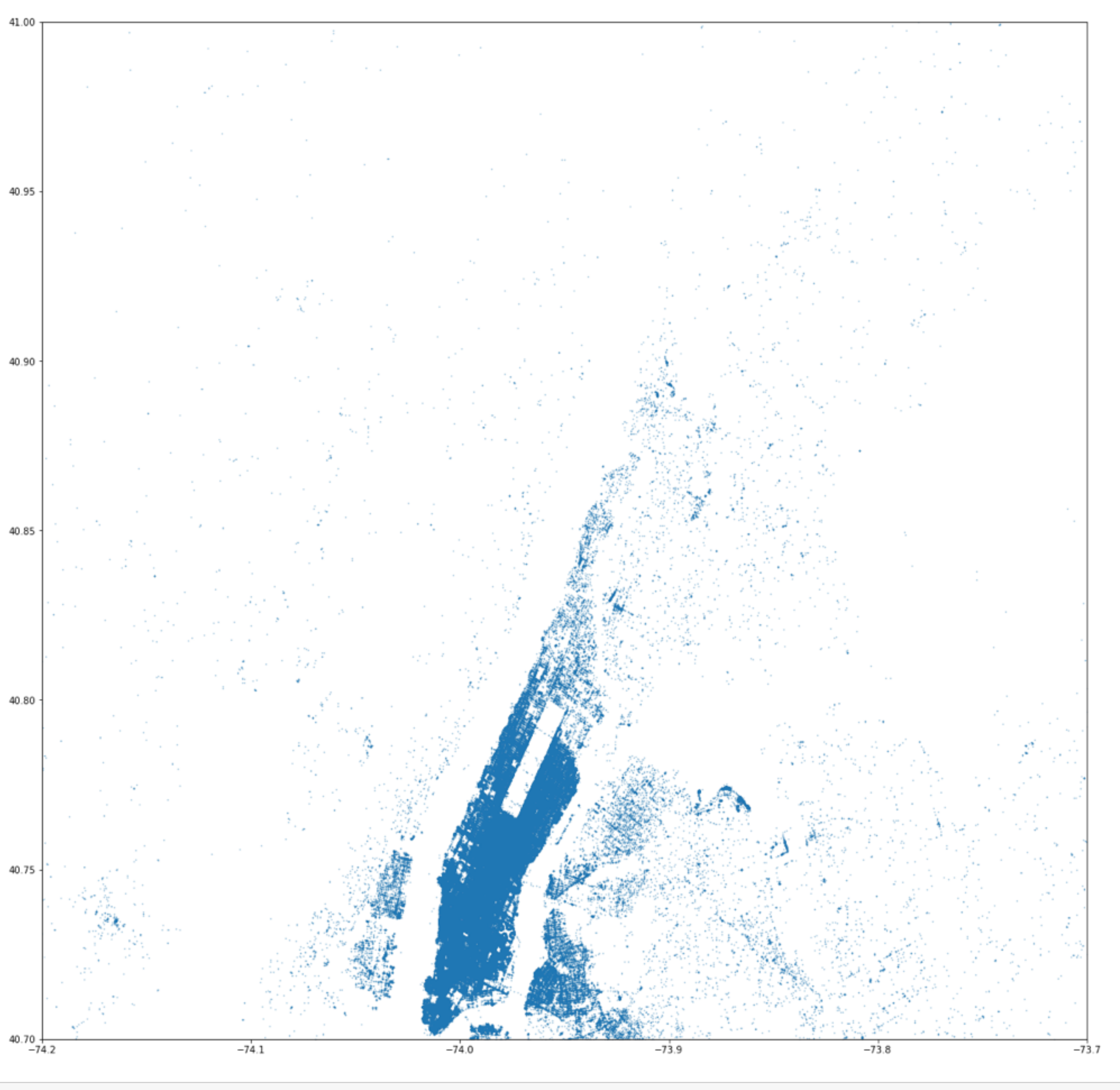
In [67]:
```python
hist(data['Lon'],bins=100,range=(-74.1,-73.9),color='g',alpha=.5)
grid()
twiny()
hist(data['Lat'],bins=100,range=(40.5,41),color='r',alpha=.5)
grid()
;
```

Out[67]: ''



In [68]:
```python
figure(figsize=(20, 20))
plot(data['Lon'], data['Lat'], '.', ms=1, alpha=.5)
xlim(-74.2, -73.7)
ylim(40.7, 41)
```

Out[68]: (40.7, 41.0)



In [ ]: