In [1]:

```
import pandas as pd
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
import os
```

In [2]:

```
files=os.listdir(r'C:\Users\PRANITA THAKARE\Favorites\Desktop\Data science project\newyork\
files
```

Out[2]:

```
['uber-raw-data-apr14.csv',
 'uber-raw-data-aug14.csv',
 'uber-raw-data-janjune-15.csv',
 'uber-raw-data-jul14.csv',
 'uber-raw-data-jun14.csv',
 'uber-raw-data-may14.csv',
 'uber-raw-data-sep14.csv']
```

In [3]:

```
files.remove('uber-raw-data-janjune-15.csv')
```

In [4]:

```
files
```

Out[4]:

```
['uber-raw-data-apr14.csv',
 'uber-raw-data-aug14.csv',
 'uber-raw-data-jul14.csv',
 'uber-raw-data-jun14.csv',
 'uber-raw-data-may14.csv',
 'uber-raw-data-sep14.csv']
```

In [5]:

```
path=r'C:\Users\PRANITA THAKARE\Favorites\Desktop\Data science project\newyork\uber-pickups

#blank dataframe
final=pd.DataFrame()

for file in files:
    df=pd.read_csv(path+"/"+file,encoding='utf-8')
    final=pd.concat([df,final])
```

In [6]:

```
final.shape
```

Out[6]:

```
(4534327, 4)
```

In [7]:

```
df=final.copy()
```

In [8]:

```
df.head()
```

Out[8]:

	Date/Time	Lat	Lon	Base
0	9/1/2014 0:01:00	40.2201	-74.0021	B02512
1	9/1/2014 0:01:00	40.7500	-74.0027	B02512
2	9/1/2014 0:03:00	40.7559	-73.9864	B02512
3	9/1/2014 0:06:00	40.7450	-73.9889	B02512
4	9/1/2014 0:11:00	40.8145	-73.9444	B02512

In [9]:

```
df.shape
```

Out[9]:

```
(4534327, 4)
```

In [10]:

```
df.dtypes
```

Out[10]:

```
Date/Time     object
Lat          float64
Lon          float64
Base          object
dtype: object
```

In [11]:

```
df['Date/Time'] = pd.to_datetime(df['Date/Time'], format="%m/%d/%Y %H:%M:%S")
```

In [12]:

```
df.dtypes
```

Out[12]:

```
Date/Time    datetime64[ns]
Lat                 float64
Lon                 float64
Base                 object
dtype: object
```

In [13]:

```
df['weekday']=df['Date/Time'].dt.day_name()
df['day']=df['Date/Time'].dt.day
df['minute']=df['Date/Time'].dt.minute
df['month']=df['Date/Time'].dt.month
df['hour']=df['Date/Time'].dt.hour
```

In [14]:

```
df.dtypes
```

Out[14]:

```
Date/Time    datetime64[ns]
Lat                 float64
Lon                 float64
Base                 object
weekday              object
day                   int64
minute                int64
month                 int64
hour                  int64
dtype: object
```

In [15]:

```
df.head()
```

Out[15]:

	Date/Time	Lat	Lon	Base	weekday	day	minute	month	hour
0	2014-09-01 00:01:00	40.2201	-74.0021	B02512	Monday	1	1	9	0
1	2014-09-01 00:01:00	40.7500	-74.0027	B02512	Monday	1	1	9	0
2	2014-09-01 00:03:00	40.7559	-73.9864	B02512	Monday	1	3	9	0
3	2014-09-01 00:06:00	40.7450	-73.9889	B02512	Monday	1	6	9	0
4	2014-09-01 00:11:00	40.8145	-73.9444	B02512	Monday	1	11	9	0

In [16]:

```
df['Base'].unique()
```

Out[16]:

```
array(['B02512', 'B02598', 'B02617', 'B02682', 'B02764'], dtype=object)
```

In [17]:

```
df['day'].unique()
```

Out[17]:

```
array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16, 17,
       18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31],
      dtype=int64)
```

In [18]:

```
df['weekday'].unique()
```

Out[18]:

```
array(['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday',
       'Sunday'], dtype=object)
```

In [19]:

```
pip install plotly
```

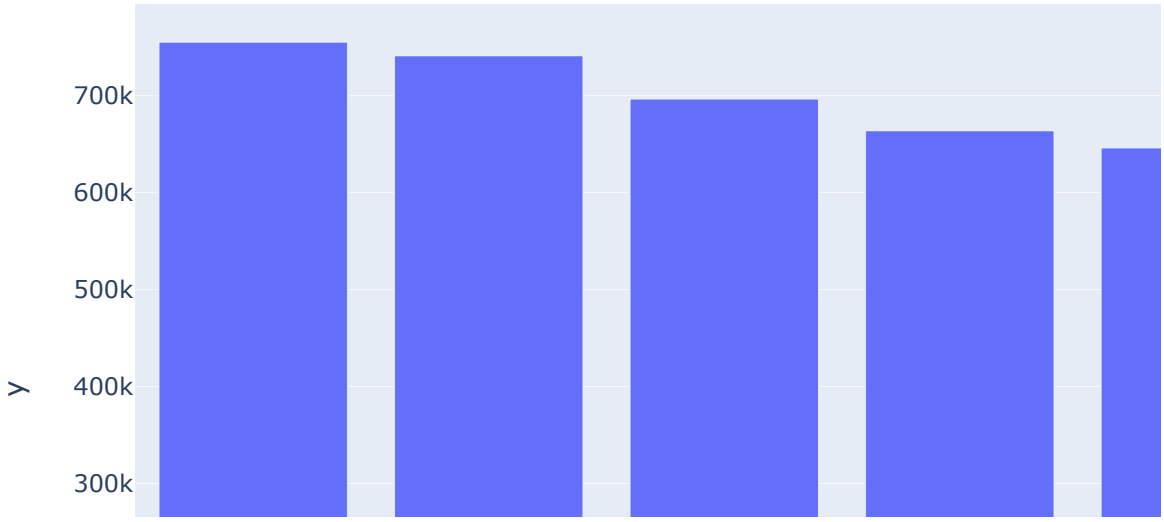
```
Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: plotly in c:\python\python38\lib\site-package
s (5.1.0)

WARNING: You are using pip version 21.1.1; however, version 21.2.4 is availa
ble.
You should consider upgrading via the 'c:\python\python38\python.exe -m pip
install --upgrade pip' command.


Requirement already satisfied: tenacity>=6.2.0 in c:\python\python38\lib\sit
e-packages (from plotly) (8.0.1)
Requirement already satisfied: six in c:\python\python38\lib\site-packages
(from plotly) (1.16.0)
```

In [20]:

```
import plotly.express as px
```

In [21]:

```
px.bar(x=df['weekday'].value_counts().index,
      y=df['weekday'].value_counts().values
      )
```

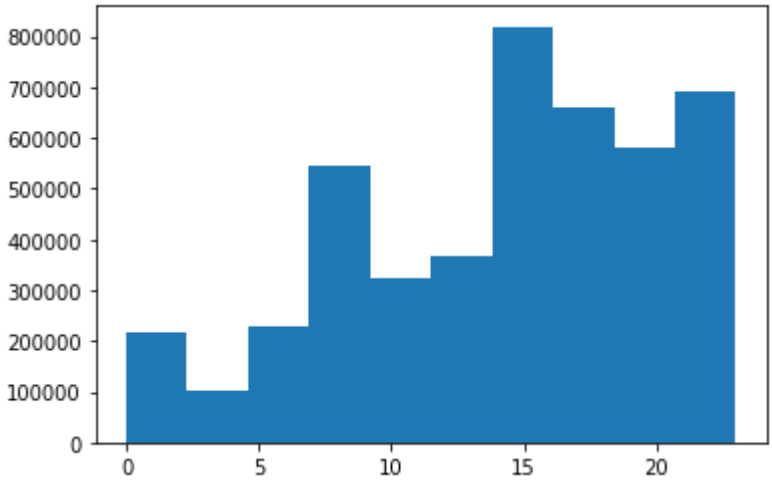


In [22]:

```
plt.hist(df['hour'])
```

Out[22]:

```
(array([216928., 103517., 227152., 543565., 324851., 366329., 819491.,
        660869., 579117., 692508.]),
 array([ 0. ,  2.3,  4.6,  6.9,  9.2, 11.5, 13.8, 16.1, 18.4, 20.7, 23. ]),
 <BarContainer object of 10 artists>)
```



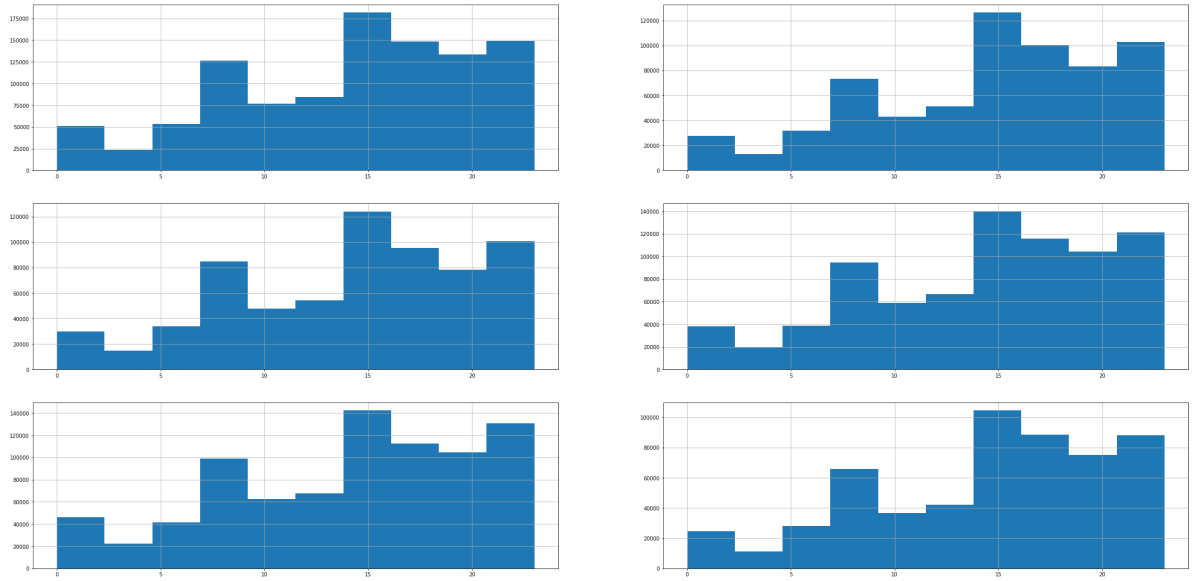
In [23]:

```
for i,month in enumerate(df['month'].unique()):
    print(month)
```

```
9
5
6
7
8
4
```

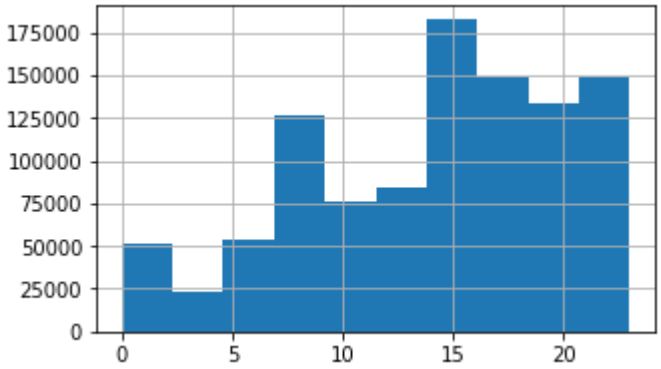
In [24]:

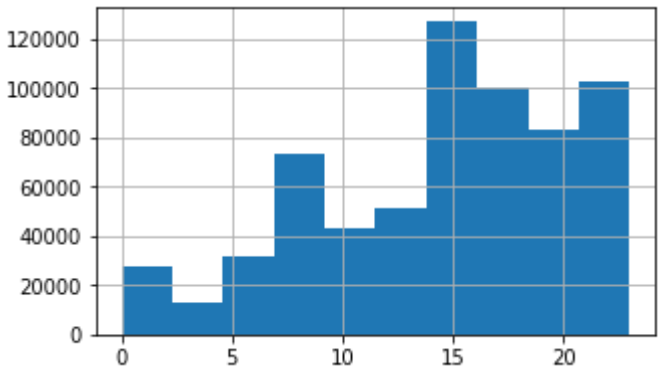
```
plt.figure(figsize=(40,20))
for i,month in enumerate(df['month'].unique()):
    plt.subplot(3,2,i+1)
    df[df['month']==month]['hour'].hist()
```

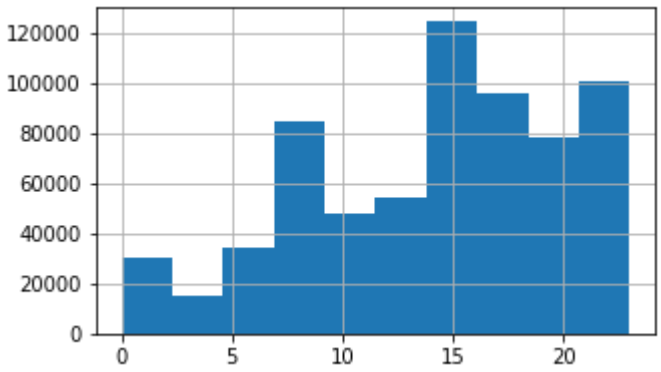


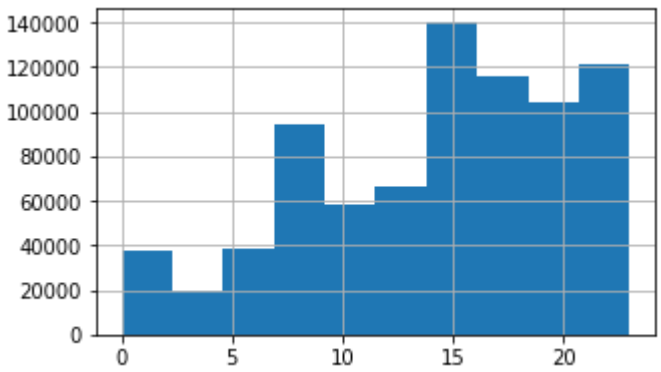
In [25]:

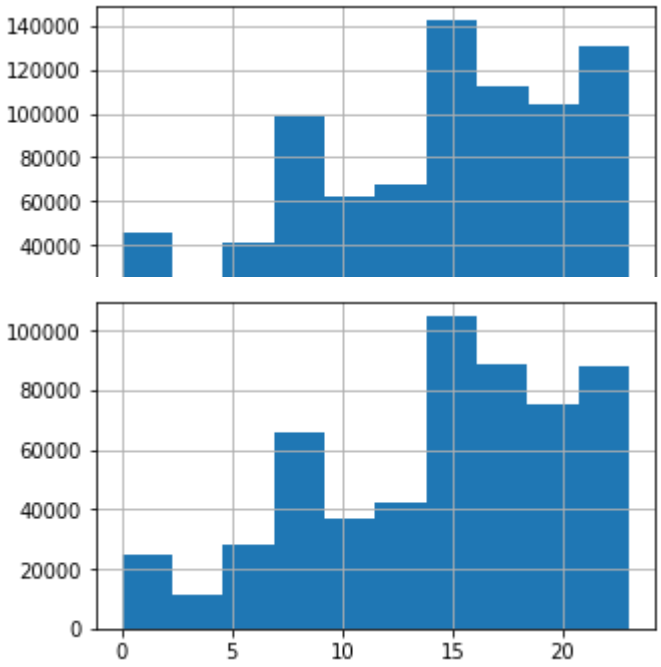
```
for i in df['month'].unique():
    plt.figure(figsize=(5,3))
    df[df['month']==i]['hour'].hist()
```











In [26]:

```
import chart_studio.plotly as py
import plotly.graph_objs as go
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot
```

In [27]:

```
pip install chart_studio
```

```
Requirement already satisfied: chart_studio in c:\python\python38\lib\site-p
ackages (1.1.0)Note: you may need to restart the kernel to use updated packa
ges.
Requirement already satisfied: requests in c:\python\python38\lib\site-packa
ges (from chart_studio) (2.25.1)
Requirement already satisfied: retrying>=1.3.3 in c:\python\python38\lib\sit
e-packages (from chart_studio) (1.3.3)
Requirement already satisfied: six in c:\python\python38\lib\site-packages
(from chart_studio) (1.16.0)
Requirement already satisfied: plotly in c:\python\python38\lib\site-package
s (from chart_studio) (5.1.0)
Requirement already satisfied: tenacity>=6.2.0 in c:\python\python38\lib\sit
e-packages (from plotly->chart_studio) (8.0.1)
Requirement already satisfied: idna<3,>=2.5 in c:\python\python38\lib\site-p
ackages (from requests->chart_studio) (2.10)
Requirement already satisfied: certifi>=2017.4.17 in c:\python\python38\lib
\site-packages (from requests->chart_studio) (2020.12.5)
Requirement already satisfied: chardet<5,>=3.0.2 in c:\python\python38\lib\s
ite-packages (from requests->chart_studio) (4.0.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\python\python38\l
ib\site-packages (from requests->chart_studio) (1.26.4)


WARNING: You are using pip version 21.1.1; however, version 21.2.4 is availa
ble.
You should consider upgrading via the 'c:\python\python38\python.exe -m pip
install --upgrade pip' command.
```

In [28]:

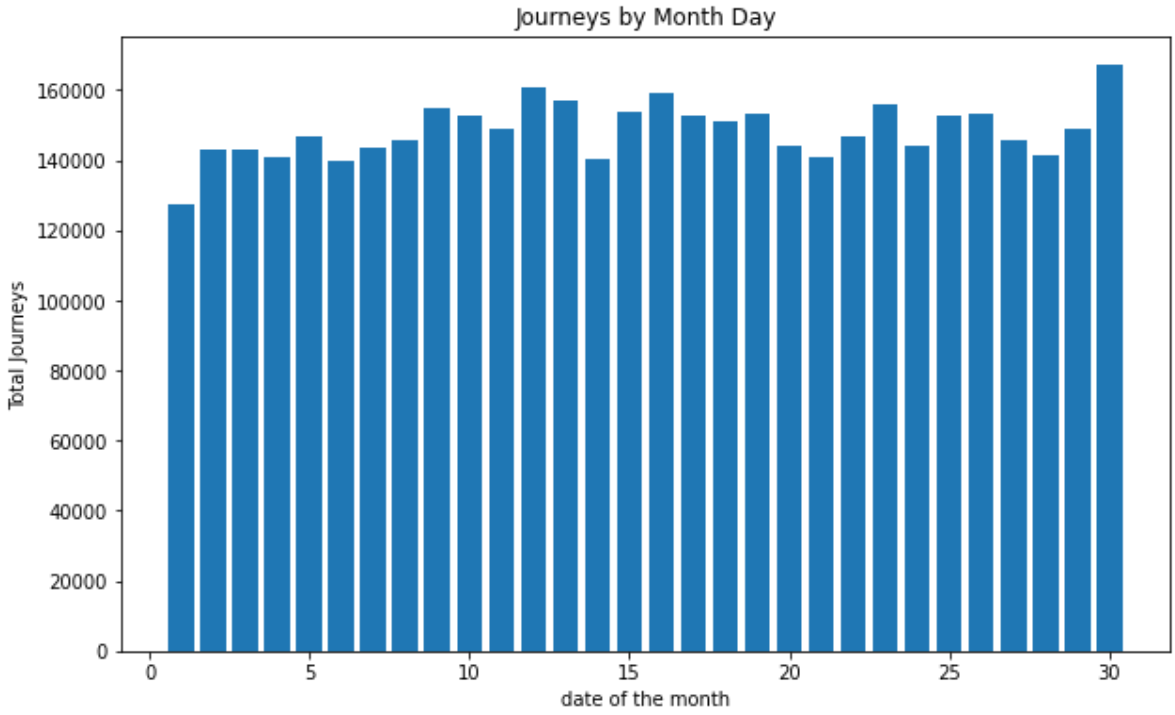
```
trace1 = go.Bar(
        x = df.groupby('month')['hour'].sum().index,
        y = df.groupby('month')['hour'].sum(),
        name= 'Priority')
iplot([trace1])
```

In [29]:

```
plt.figure(figsize=(10,6))
plt.hist(df['day'], bins=30, rwidth=.8, range=(0.5, 30.5))
plt.xlabel('date of the month')
plt.ylabel('Total Journeys')
plt.title('Journeys by Month Day')
```

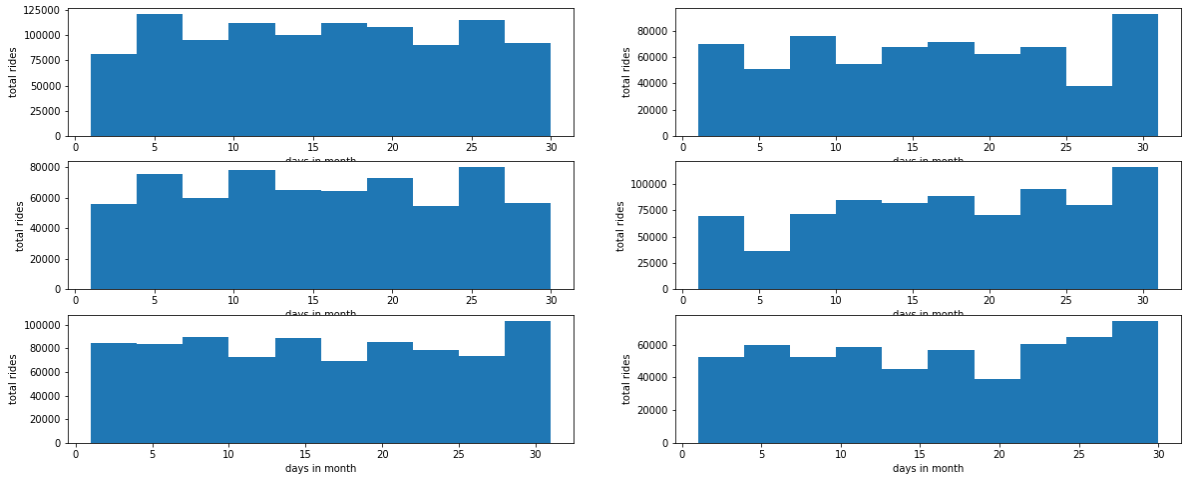
Out[29]:

```
Text(0.5, 1.0, 'Journeys by Month Day')
```



In [30]:

```
plt.figure(figsize=(20,8))
for i,month in enumerate(df['month'].unique(),1):
    plt.subplot(3,2,i)
    df_out=df[df['month']==month]
    plt.hist(df_out['day'])
    plt.xlabel('days in month'.format(i))
    plt.ylabel('total rides')
```

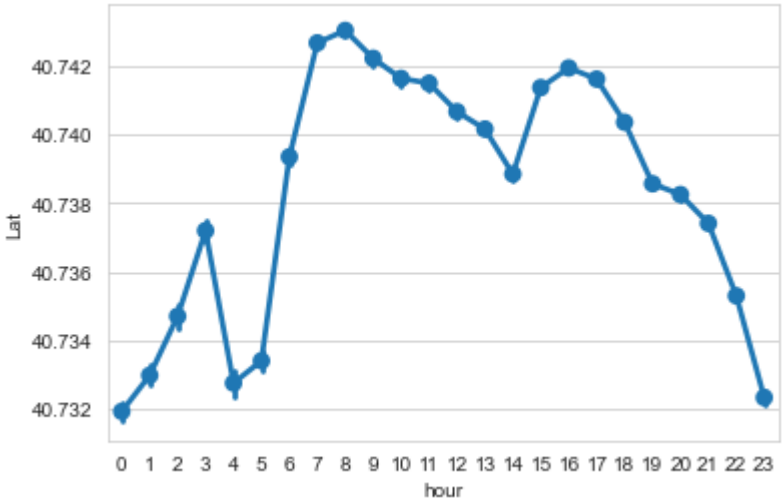


In [31]:

```
sns.set_style(style='whitegrid')
sns.pointplot(x="hour",y="Lat",data=df)
```

Out[31]:

```
<AxesSubplot:xlabel='hour', ylabel='Lat'>
```

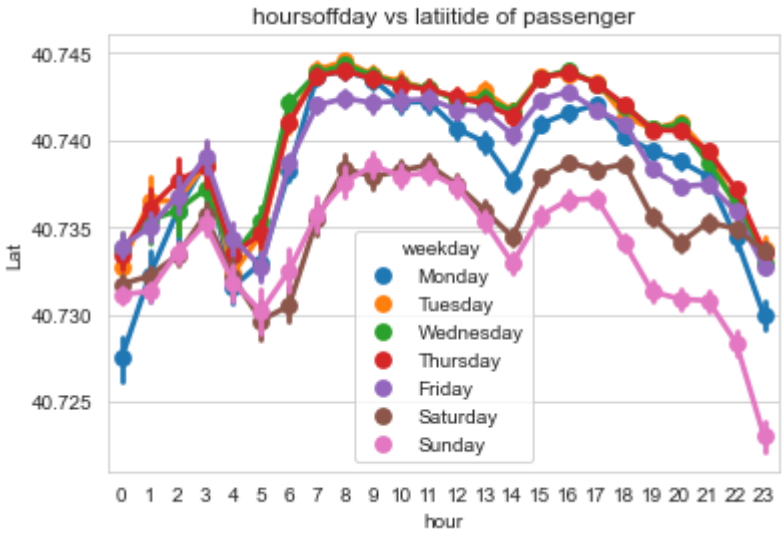


In [32]:

```
ax=sns.pointplot(x="hour",y="Lat", hue="weekday",data=df)
ax.set_title('hoursoffday vs latiitide of passenger')
```

Out[32]:

```
Text(0.5, 1.0, 'hoursoffday vs latiitide of passenger')
```



In [33]:

```
df.head()
```

Out[33]:

	Date/Time	Lat	Lon	Base	weekday	day	minute	month	hour
0	2014-09-01 00:01:00	40.2201	-74.0021	B02512	Monday	1	1	9	0
1	2014-09-01 00:01:00	40.7500	-74.0027	B02512	Monday	1	1	9	0
2	2014-09-01 00:03:00	40.7559	-73.9864	B02512	Monday	1	3	9	0
3	2014-09-01 00:06:00	40.7450	-73.9889	B02512	Monday	1	6	9	0
4	2014-09-01 00:11:00	40.8145	-73.9444	B02512	Monday	1	11	9	0

In [34]:

```
df['Base'].head()
```

Out[34]:

```
0    B02512
1    B02512
2    B02512
3    B02512
4    B02512
Name: Base, dtype: object
```

In [35]:

```
df.groupby(['Base','month'])['Date/Time'].count()
```

Out[35]:

```
Base    month
B02512  4         35536
        5         36765
        6         32509
        7         35021
        8         31472
        9         34370
B02598  4        183263
        5        260549
        6        242975
        7        245597
        8        220129
        9        240600
B02617  4        108001
        5        122734
        6        184460
        7        310160
        8        355803
        9        377695
B02682  4        227808
        5        222883
        6        194926
        7        196754
        8        173280
        9        197138
B02764  4          9908
        5          9504
        6          8974
        7          8589
        8         48591
        9        178333
Name: Date/Time, dtype: int64
```

In [36]:

```
base=df.groupby(['Base','month'])['Date/Time'].count().reset_index()
base
```

Out[36]:

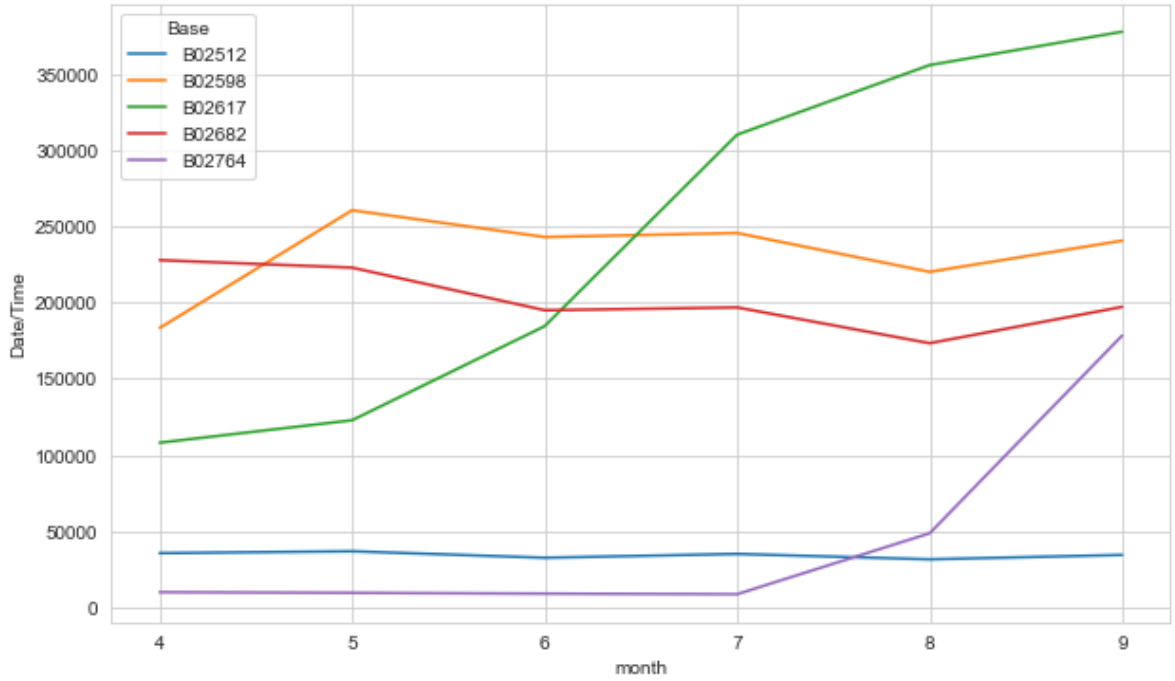
	Base	month	Date/Time
0	B02512	4	35536
1	B02512	5	36765
2	B02512	6	32509
3	B02512	7	35021
4	B02512	8	31472
5	B02512	9	34370
6	B02598	4	183263
7	B02598	5	260549
8	B02598	6	242975
9	B02598	7	245597
10	B02598	8	220129
11	B02598	9	240600
12	B02617	4	108001
13	B02617	5	122734
14	B02617	6	184460
15	B02617	7	310160
16	B02617	8	355803
17	B02617	9	377695
18	B02682	4	227808
19	B02682	5	222883
20	B02682	6	194926
21	B02682	7	196754
22	B02682	8	173280
23	B02682	9	197138
24	B02764	4	9908
25	B02764	5	9504
26	B02764	6	8974
27	B02764	7	8589
28	B02764	8	48591
29	B02764	9	178333

In [37]:

```
plt.figure(figsize=(10,6))
sns.lineplot(x='month',y='Date/Time',hue='Base',data=base)
```

Out[37]:

```
<AxesSubplot:xlabel='month', ylabel='Date/Time'>
```



In [38]:

```
def count_rows(rows):
    return len(rows)
```

In [39]:

```
by_cross = df.groupby(['weekday','hour']).apply(count_rows)
by_cross
```

Out[39]:

```
weekday    hour
Friday     0       13716
           1        8163
           2        5350
           3        6930
           4        8806
                   ...
Wednesday  19      47017
           20      47772
           21      44553
           22      32868
           23      18146
Length: 168, dtype: int64
```

In [40]:

```
pivot=by_cross.unstack()
pivot
```

Out[40]:

hour	0	1	2	3	4	5	6	7	8	9	...	14
weekday												
Friday	13716	8163	5350	6930	8806	13450	23412	32061	31509	25230	...	36206
Monday	6436	3737	2938	6232	9640	15032	23746	31159	29265	22197	...	28157
Saturday	27633	19189	12710	9542	6846	7084	8579	11014	14411	17669	...	31418
Sunday	32877	23015	15436	10597	6374	6169	6596	8728	12128	16401	...	28151
Thursday	9293	5290	3719	5637	8505	14169	27065	37038	35431	27812	...	36699
Tuesday	6237	3509	2571	4494	7548	14241	26872	36599	33934	25023	...	34846
Wednesday	7644	4324	3141	4855	7511	13794	26943	36495	33826	25635	...	35148

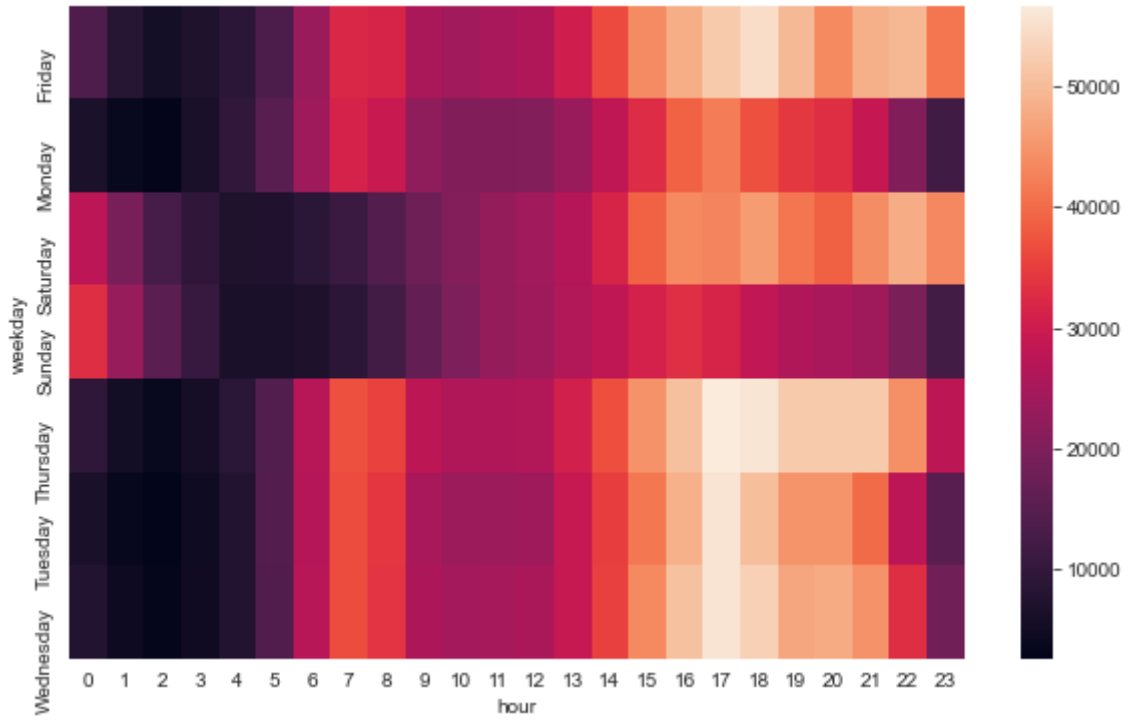
7 rows × 24 columns

In [41]:

```
plt.figure(figsize=(10,6))
sns.heatmap(pivot, annot=False)
```

Out[41]:

```
<AxesSubplot:xlabel='hour', ylabel='weekday'>
```



In [42]:

```
df.head()
```

Out[42]:

	Date/Time	Lat	Lon	Base	weekday	day	minute	month	hour
0	2014-09-01 00:01:00	40.2201	-74.0021	B02512	Monday	1	1	9	0
1	2014-09-01 00:01:00	40.7500	-74.0027	B02512	Monday	1	1	9	0
2	2014-09-01 00:03:00	40.7559	-73.9864	B02512	Monday	1	3	9	0
3	2014-09-01 00:06:00	40.7450	-73.9889	B02512	Monday	1	6	9	0
4	2014-09-01 00:11:00	40.8145	-73.9444	B02512	Monday	1	11	9	0

In [43]:

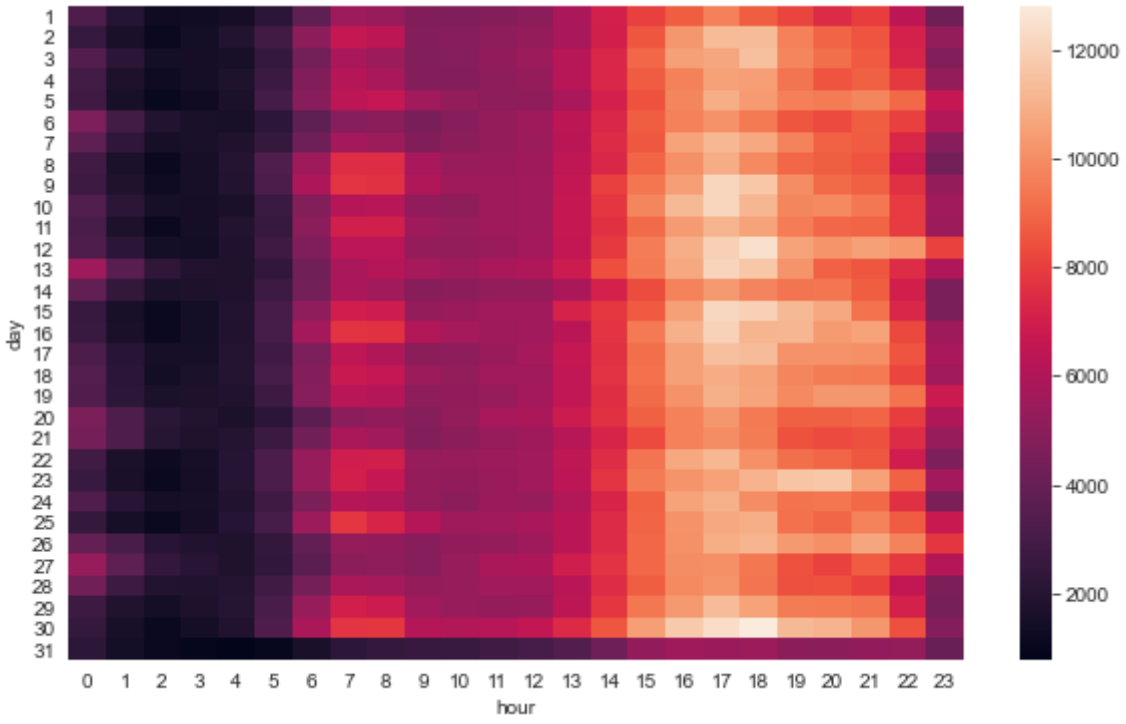
```
def heatmap(col1,col2):
    by_cross = df.groupby([col1,col2]).apply(lambda x:len(x))
    pivot=by_cross.unstack()
    plt.figure(figsize=(10,6))
    return sns.heatmap(pivot,annot=False)
```

In [44]:

```
## validating above Analysis through Heatmap
heatmap('day','hour')
```

Out[44]:

```
<AxesSubplot:xlabel='hour', ylabel='day'>
```

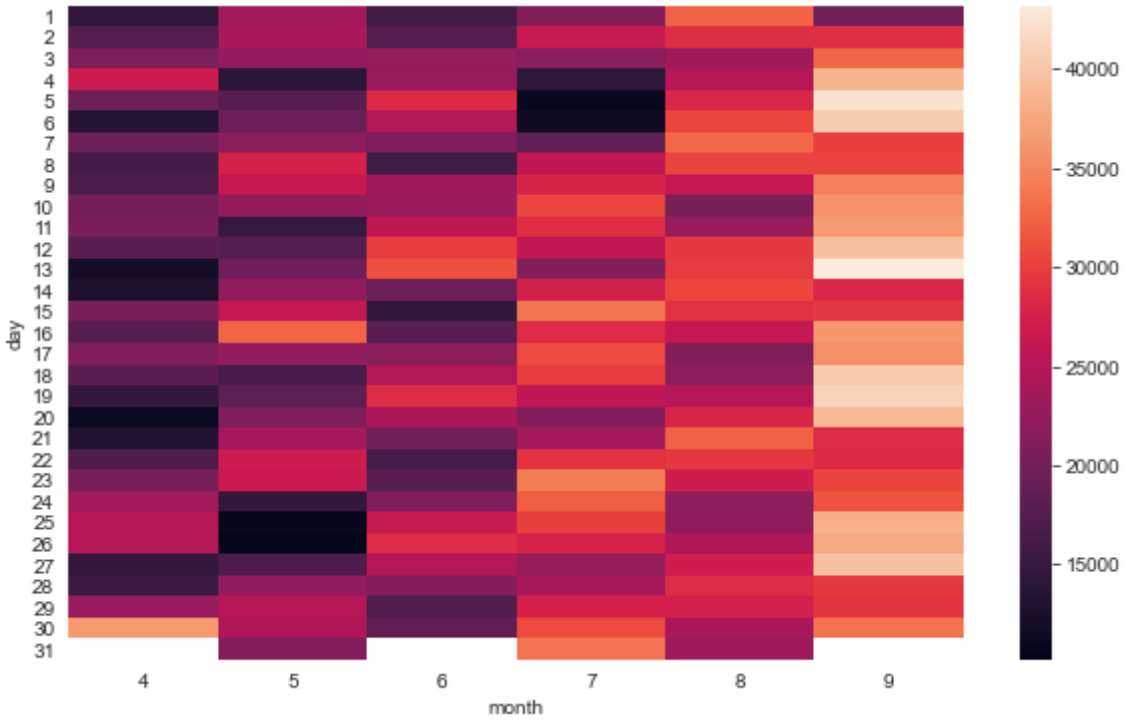


In [45]:

```
heatmap('day','month')
```

Out[45]:

```
<AxesSubplot:xlabel='month', ylabel='day'>
```



In [46]:

```
df[df['month']==4]
```

Out[46]:

	Date/Time	Lat	Lon	Base	weekday	day	minute	month	hour
0	2014-04-01 00:11:00	40.7690	-73.9549	B02512	Tuesday	1	11	4	0
1	2014-04-01 00:17:00	40.7267	-74.0345	B02512	Tuesday	1	17	4	0
2	2014-04-01 00:21:00	40.7316	-73.9873	B02512	Tuesday	1	21	4	0
3	2014-04-01 00:28:00	40.7588	-73.9776	B02512	Tuesday	1	28	4	0
4	2014-04-01 00:33:00	40.7594	-73.9722	B02512	Tuesday	1	33	4	0
...	...	...	...	...	...	...	...	...	...
564511	2014-04-30 23:22:00	40.7640	-73.9744	B02764	Wednesday	30	22	4	23
564512	2014-04-30 23:26:00	40.7629	-73.9672	B02764	Wednesday	30	26	4	23
564513	2014-04-30 23:31:00	40.7443	-73.9889	B02764	Wednesday	30	31	4	23
564514	2014-04-30 23:32:00	40.6756	-73.9405	B02764	Wednesday	30	32	4	23
564515	2014-04-30 23:48:00	40.6880	-73.9608	B02764	Wednesday	30	48	4	23

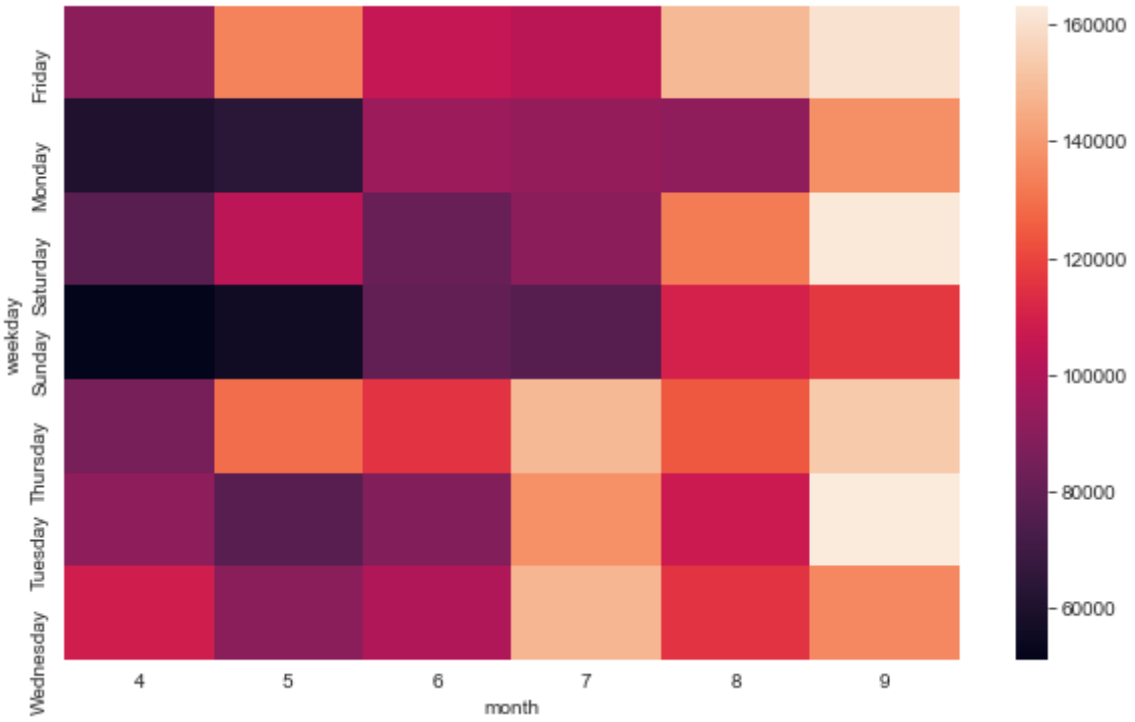
564516 rows × 9 columns

In [47]:

```
heatmap('weekday','month')
```

Out[47]:

```
<AxesSubplot:xlabel='month', ylabel='weekday'>
```

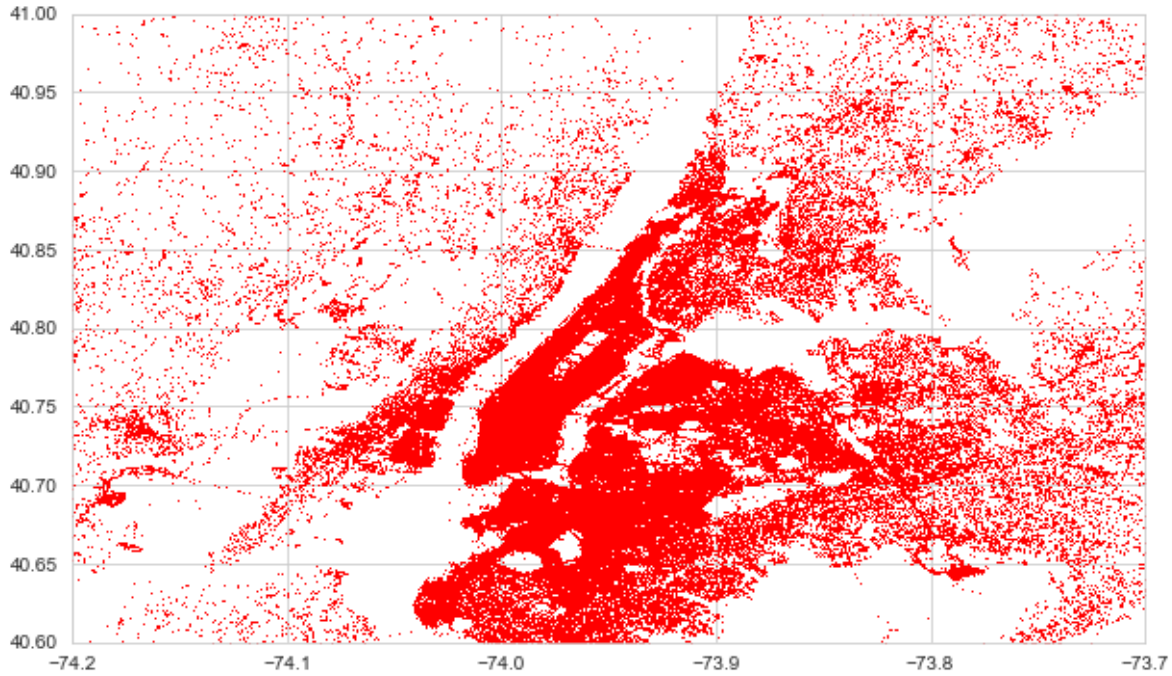


In [48]:

```
plt.figure(figsize=(10,6))

plt.plot(df['Lon'], df['Lat'],'r+', ms=0.5)
plt.xlim(-74.2, -73.7)
plt.ylim(40.6,41)
```

Out[48]:

```
(40.6, 41.0)
```



In [49]:

```
df.head()
```

Out[49]:

	Date/Time	Lat	Lon	Base	weekday	day	minute	month	hour
0	2014-09-01 00:01:00	40.2201	-74.0021	B02512	Monday	1	1	9	0
1	2014-09-01 00:01:00	40.7500	-74.0027	B02512	Monday	1	1	9	0
2	2014-09-01 00:03:00	40.7559	-73.9864	B02512	Monday	1	3	9	0
3	2014-09-01 00:06:00	40.7450	-73.9889	B02512	Monday	1	6	9	0
4	2014-09-01 00:11:00	40.8145	-73.9444	B02512	Monday	1	11	9	0

In [50]:

```
df_out=df[df['weekday']=='Sunday']
df_out.head()
```

Out[50]:

	Date/Time	Lat	Lon	Base	weekday	day	minute	month	hour
8011	2014-09-07 00:00:00	40.7341	-74.0005	B02512	Sunday	7	0	9	0
8012	2014-09-07 00:00:00	40.7344	-73.9900	B02512	Sunday	7	0	9	0
8013	2014-09-07 00:00:00	40.7806	-73.9582	B02512	Sunday	7	0	9	0
8014	2014-09-07 00:01:00	40.7293	-73.9859	B02512	Sunday	7	1	9	0
8015	2014-09-07 00:01:00	40.7713	-74.0133	B02512	Sunday	7	1	9	0

In [51]:

```
df_out.groupby(['Lat','Lon'])['weekday'].count().reset_index()
```

Out[51]:

	Lat	Lon	weekday
0	39.9374	-74.0722	1
1	39.9378	-74.0721	1
2	39.9384	-74.0742	1
3	39.9385	-74.0734	1
4	39.9415	-74.0736	1
...	...	...	...
209225	41.3141	-74.1249	1
209226	41.3180	-74.1298	1
209227	41.3195	-73.6905	1
209228	41.3197	-73.6903	1
209229	42.1166	-72.0666	1

209230 rows × 3 columns

In [52]:

```
from folium.plugins import HeatMap
```

In [53]:

```
pip install folium
```

```
Requirement already satisfied: folium in c:\python\python38\lib\site-package
s (0.12.1)Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: branca>=0.3.0 in c:\python\python38\lib\site-
packages (from folium) (0.4.2)
Requirement already satisfied: jinja2>=2.9 in c:\python\python38\lib\site-pa
ckages (from folium) (3.0.1)
Requirement already satisfied: requests in c:\python\python38\lib\site-packa
ges (from folium) (2.25.1)
Requirement already satisfied: numpy in c:\python\python38\lib\site-packages
(from folium) (1.20.3)
Requirement already satisfied: MarkupSafe>=2.0 in c:\python\python38\lib\sit
e-packages (from jinja2>=2.9->folium) (2.0.1)
Requirement already satisfied: certifi>=2017.4.17 in c:\python\python38\lib
\site-packages (from requests->folium) (2020.12.5)
Requirement already satisfied: idna<3,>=2.5 in c:\python\python38\lib\site-p
ackages (from requests->folium) (2.10)
Requirement already satisfied: chardet<5,>=3.0.2 in c:\python\python38\lib\s
ite-packages (from requests->folium) (4.0.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\python\python38\l
ib\site-packages (from requests->folium) (1.26.4)


WARNING: You are using pip version 21.1.1; however, version 21.2.4 is availa
ble.
You should consider upgrading via the 'c:\python\python38\python.exe -m pip
install --upgrade pip' command.
```

In [54]:

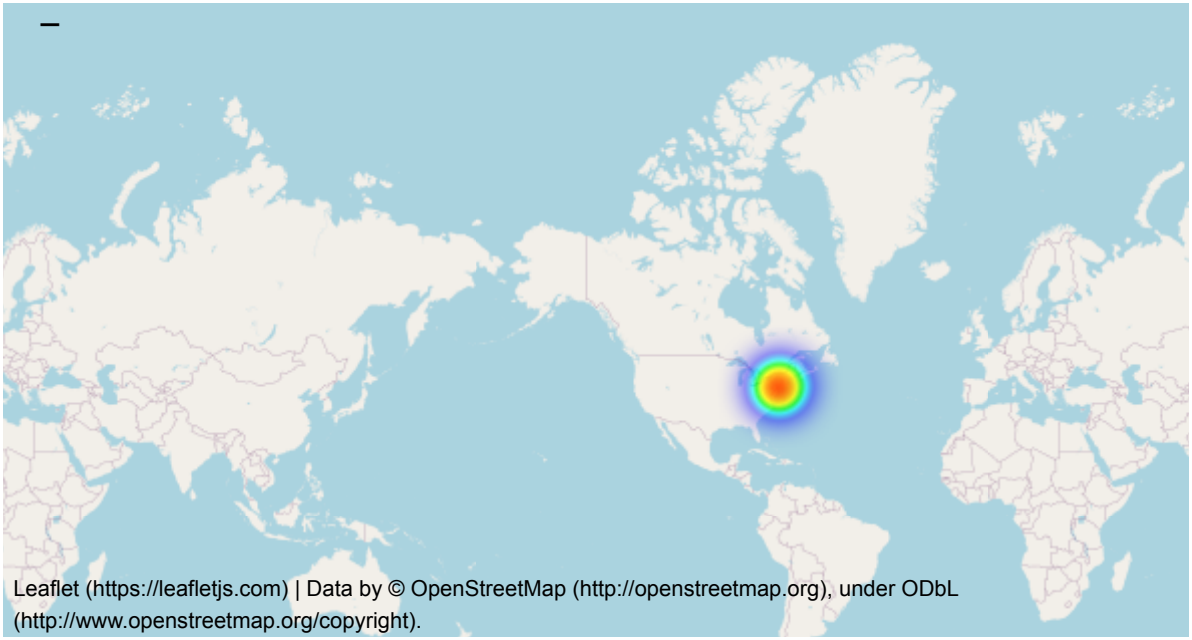
```
import folium
from folium.plugins import HeatMap
basemap=folium.Map()
```

In [55]:

```
HeatMap(df_out.groupby(['Lat','Lon'])['weekday'].count().reset_index(),zoom=20,radius=15).a
basemap
```

Out[55]:

Make this Notebook Trusted to load map: File -> Trust Notebook



In [56]:

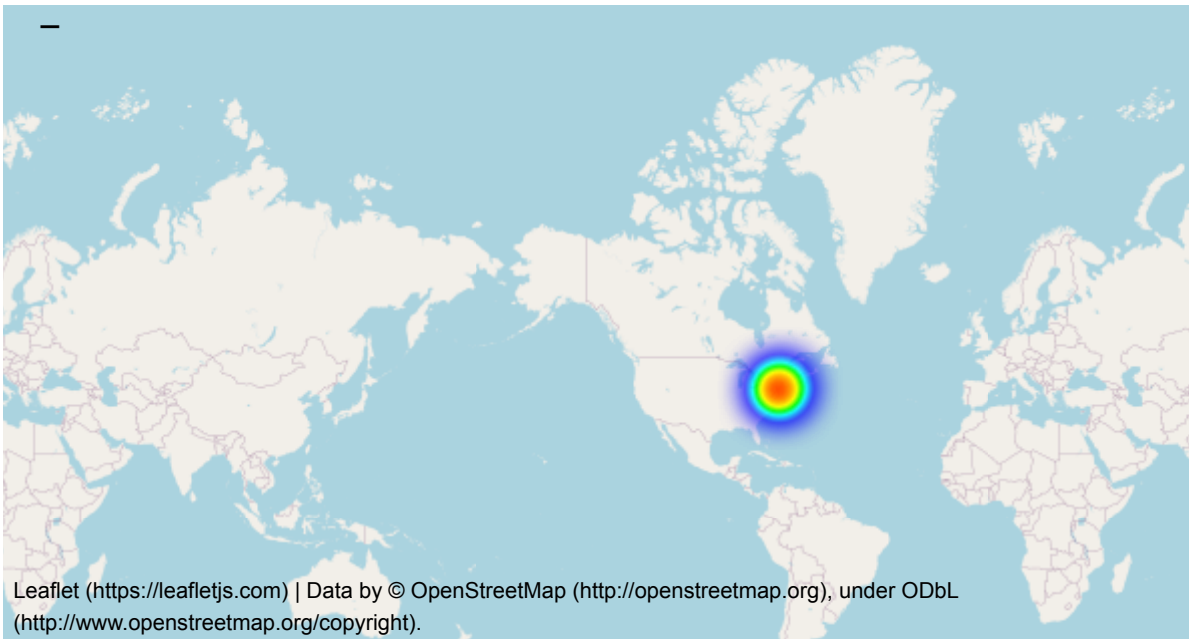
```
def plot(df,day):
    df_out=df[df['weekday']==day]
    df_out.groupby(['Lat','Lon'])['weekday'].count().reset_index()
    HeatMap(df_out.groupby(['Lat','Lon'])['weekday'].count().reset_index(),zoom=20,radius=1
    return basemap
```

In [57]:

```
plot(df,'Sunday')
```

Out[57]:

Make this Notebook Trusted to load map: File -> Trust Notebook



In [58]:

```
uber_15 = pd.read_csv(r'C:\Users\PRANITA THAKARE\Favorites\Desktop\Data science project\new
uber_15.head()
```

Out[58]:

	Dispatching_base_num	Pickup_date	Affiliated_base_num	locationID
0	B02617	2015-05-17 09:47:00	B02617	141
1	B02617	2015-05-17 09:47:00	B02617	65
2	B02617	2015-05-17 09:47:00	B02617	100
3	B02617	2015-05-17 09:47:00	B02774	80
4	B02617	2015-05-17 09:47:00	B02617	90

In [59]:

```
uber_15.shape
```

Out[59]:

```
(14270479, 4)
```

In [60]:

```
#Checking the minimum date in the uber_15
uber_15['Pickup_date'].min()
```

Out[60]:

```
'2015-01-01 00:00:05'
```

In [61]:

```
#Checking the maximum date in the uber_15
uber_15['Pickup_date'].max()
```

Out[61]:

```
'2015-06-30 23:59:00'
```

In [62]:

```
uber_15['Pickup_date'] =  pd.to_datetime(uber_15['Pickup_date'], format='%Y-%m-%d %H:%M:%S'
```

In [63]:

```
uber_15['weekday']=uber_15['Pickup_date'].dt.day_name()
uber_15['day']=uber_15['Pickup_date'].dt.day
uber_15['minute']=uber_15['Pickup_date'].dt.minute
uber_15['month']=uber_15['Pickup_date'].dt.month
uber_15['hour']=uber_15['Pickup_date'].dt.hour
```

In [64]:

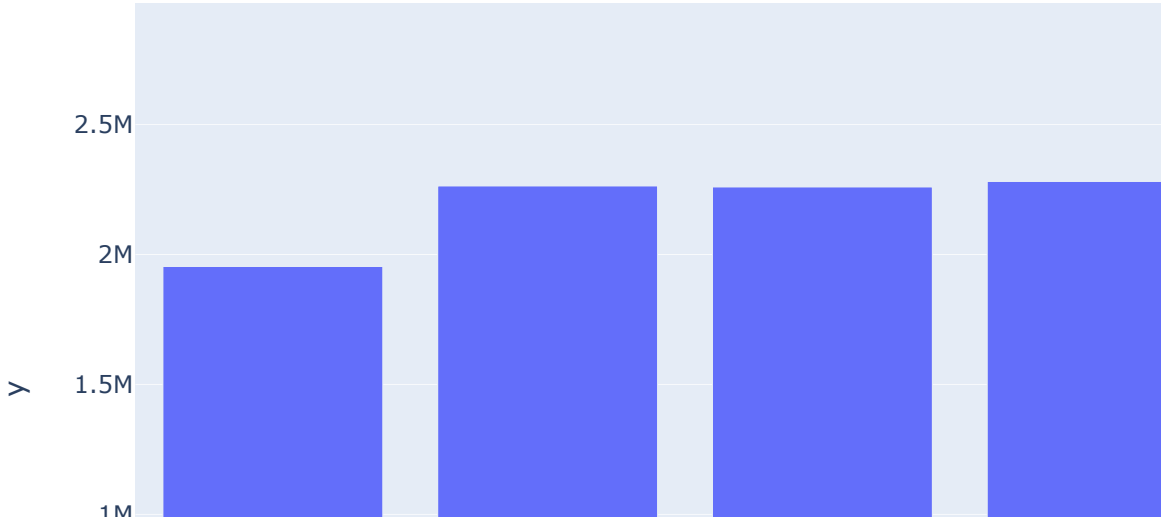
```
uber_15.head()
```

Out[64]:

	Dispatching_base_num	Pickup_date	Affiliated_base_num	locationID	weekday	day	minute
0	B02617	2015-05-17 09:47:00	B02617	141	Sunday	17	47
1	B02617	2015-05-17 09:47:00	B02617	65	Sunday	17	47
2	B02617	2015-05-17 09:47:00	B02617	100	Sunday	17	47
3	B02617	2015-05-17 09:47:00	B02774	80	Sunday	17	47
4	B02617	2015-05-17 09:47:00	B02617	90	Sunday	17	47

In [65]:

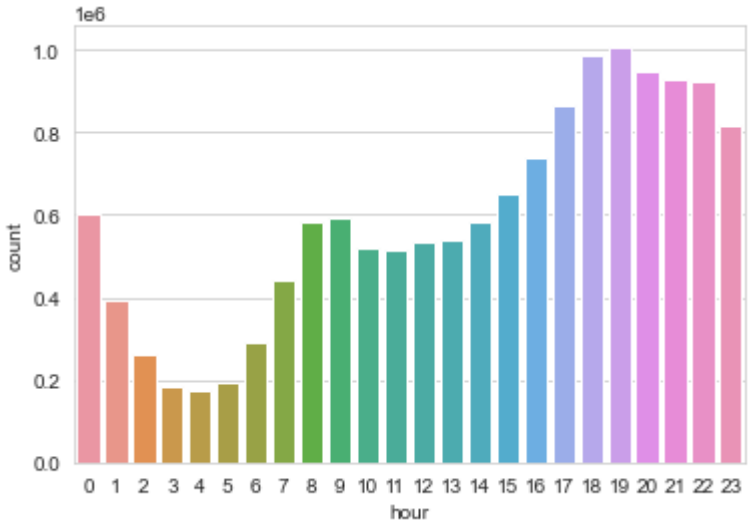
```
px.bar(x=uber_15['month'].value_counts().index,
           y=uber_15['month'].value_counts().values)
```



In [66]:

```
ax=sns.countplot(uber_15['hour'])
ax.yaxis.set_major_formatter(tick.FormatStrFormatter('%.0f'))
```

```
c:\python\python38\lib\site-packages\seaborn\_decorators.py:36: FutureWarnin
g:

Pass the following variable as a keyword arg: x. From version 0.12, the only
valid positional argument will be `data`, and passing other arguments withou
t an explicit keyword will result in an error or misinterpretation.


---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-66-bf8c2b36a6cd> in <module>
      1 ax=sns.countplot(uber_15['hour'])
----> 2 ax.yaxis.set_major_formatter(tick.FormatStrFormatter('%.0f'))

NameError: name 'tick' is not defined
```



In [ ]:

```
uber_15.groupby(['weekday', 'hour'])['Pickup_date'].count()
```

In [ ]:

```
uber_15.groupby(['weekday', 'hour'])['Pickup_date'].count().reset_index()
```

In [ ]:

```
summary=uber_15.groupby(['weekday', 'hour'])['Pickup_date'].count().reset_index()
```

In [ ]:

```
summary=summary.rename(columns = {'Pickup_date':'Counts'})
summary
```

In [ ]:

```
plt.figure(figsize=(10,6))
sns.pointplot(x="hour", y="Counts", hue="weekday", data=summary)
```

In [ ]:

```
uber_foil=pd.read_csv(r'C:\Users\PRANITA THAKARE\Favorites\Desktop\Data science project\new
```

In [ ]:

```
uber_foil.head()
```

In [ ]:

```
uber_foil['dispatching_base_number'].unique()
```

In [ ]:

```
sns.boxplot(x = 'dispatching_base_number', y = 'active_vehicles', data = uber_foil)
```

In [ ]:

```
sns.boxplot(x = 'dispatching_base_number', y = 'trips', data = uber_foil)
```

In [ ]:

```
# Finding the ratio of trips/active_vehicles
uber_foil['trips/vehicle'] = uber_foil['trips']/uber_foil['active_vehicles']
```

In [ ]:

```
uber_foil.head()
```

In [ ]:

```
uber_foil.set_index('date')
```

In [ ]:

```
plt.figure(figsize=(10,6))
uber_foil.set_index('date').groupby(['dispatching_base_number'])['trips/vehicle'].plot()
plt.ylabel('Average trips/vehicle')
plt.title('Demand vs Supply chart (Date-wise)')
plt.legend()
```

In [ ]:

```

```