

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import os
```

```
In [2]: os.listdir(r"C:\Users\naila.iram\Downloads\Datasets")
```

```
Out[2]: ['other-American_B01362.csv',
'other-Carmel_B00256.csv',
'other-Dial7_B00887.csv',
'other-Diplo_B01196.csv',
'other-Federal_02216.csv',
'other-FHV-services_jan-aug-2015.csv',
'other-Firstclass_B01536.csv',
'other-Highclass_B01717.csv',
'other-Lyft_B02510.csv',
'other-Prestige_B01338.csv',
'other-Skyline_B00111.csv',
'Uber-Jan-Feb-FOIL.csv',
'uber-raw-data-apr14.csv',
'uber-raw-data-aug14.csv',
'uber-raw-data-janjune-15.csv',
'uber-raw-data-janjune-15_sample.csv',
'uber-raw-data-jul14.csv',
'uber-raw-data-jun14.csv',
'uber-raw-data-may14.csv',
'uber-raw-data-sep14.csv']
```

```
In [3]: uber_15=pd.read_csv(r"C:\Users\naila.iram\Downloads\Datasets\uber-raw-data-janjune-uber_15.head(5)
```

```
Out[3]:
```

	Dispatching_base_num	Pickup_date	Affiliated_base_num	locationID
0	B02617	2015-05-02 21:43:00	B02764	237
1	B02682	2015-01-20 19:52:59	B02682	231
2	B02617	2015-03-19 20:26:00	B02617	161
3	B02764	2015-04-10 17:38:00	B02764	107
4	B02764	2015-03-23 07:03:00	B00111	140

```
In [4]: uber_15.describe()
```

Out[4]:

	locationID
count	100000.000000
mean	151.922810
std	71.560596
min	2.000000
25%	92.000000
50%	158.000000
75%	230.000000
max	265.000000

In [5]: `uber_15.shape`

Out[5]: (100000, 4)

Data Cleaning

In [6]: `## Removing Duplicates`
`uber_15.duplicated()`

Out[6]:

0	False
1	False
2	False
3	False
4	False
...	
99995	False
99996	False
99997	False
99998	False
99999	False

Length: 100000, dtype: bool

In [7]: `uber_15.duplicated().sum()`

Out[7]: 54

In [8]: `uber_15=uber_15.drop_duplicates()`

In [9]: `uber_15.duplicated().sum()`

Out[9]: 0

In [10]: `uber_15.shape`

Out[10]: (99946, 4)

```
In [11]: ## Check Missing Values & data types
uber_15.isnull().sum()
```

```
Out[11]: Dispatching_base_num      0
Pickup_date      0
Affiliated_base_num    1116
locationID      0
dtype: int64
```

```
In [12]: uber_15.dtypes
```

```
Out[12]: Dispatching_base_num    object
Pickup_date      object
Affiliated_base_num    object
locationID      int64
dtype: object
```

```
In [14]: ## convert Pickup_date data type to date_time
uber_15['Pickup_date'] = pd.to_datetime(uber_15['Pickup_date'])
```

```
In [15]: uber_15.dtypes
```

```
Out[15]: Dispatching_base_num      object
Pickup_date      datetime64[ns]
Affiliated_base_num      object
locationID      int64
dtype: object
```

Exploratory Data Analysis

```
In [21]: ### Which month has the highest pickups
uber_15['Month'] = uber_15['Pickup_date'].dt.month_name()
```

```
In [17]: uber_15.head(1)
```

```
Out[17]:
```

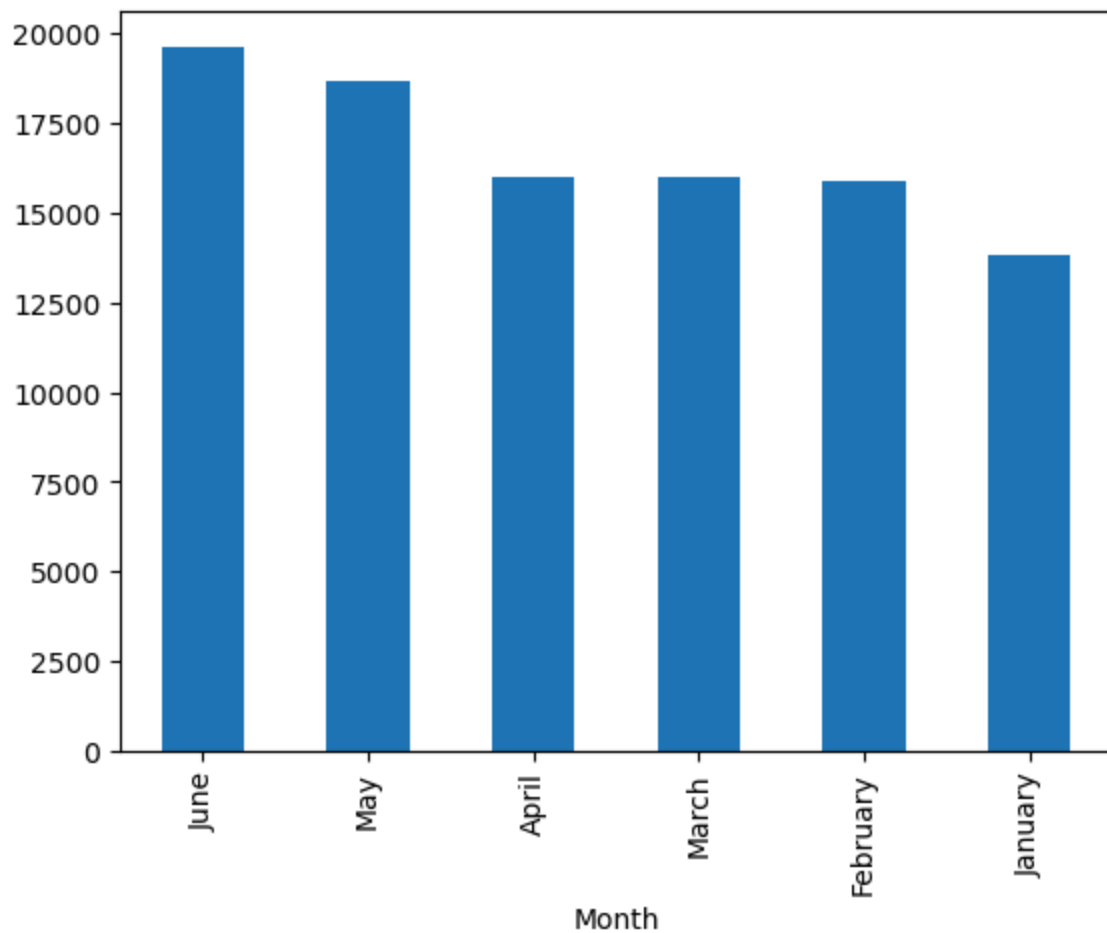
	Dispatching_base_num	Pickup_date	Affiliated_base_num	locationID	Month
0	B02617	2015-05-02 21:43:00	B02764	237	May

```
In [18]: uber_15['Month'].value_counts()
```

```
Out[18]: Month
June      19620
May       18660
April     15982
March     15969
February  15896
January   13819
Name: count, dtype: int64
```

```
In [20]: uber_15['Month'].value_counts().plot(kind='bar')
```

Out[20]: <Axes: xlabel='Month'>



```
In [23]: ### which weekday of the month has the highest pickups
uber_15['Day Name'] = uber_15['Pickup_date'].dt.day_name()
uber_15['Day Number'] = uber_15['Pickup_date'].dt.day
uber_15['Hour'] = uber_15['Pickup_date'].dt.hour
uber_15['Minute'] = uber_15['Pickup_date'].dt.minute
```

```
In [24]: uber_15.head(1)
```

```
Out[24]:
```

	Dispatching_base_num	Pickup_date	Affiliated_base_num	locationID	Month	Day Name
0	B02617	2015-05-02 21:43:00	B02764	237	May	Saturday

```
In [26]: pivot = pd.crosstab(index=uber_15['Month'], columns=uber_15['Day Name'])
```

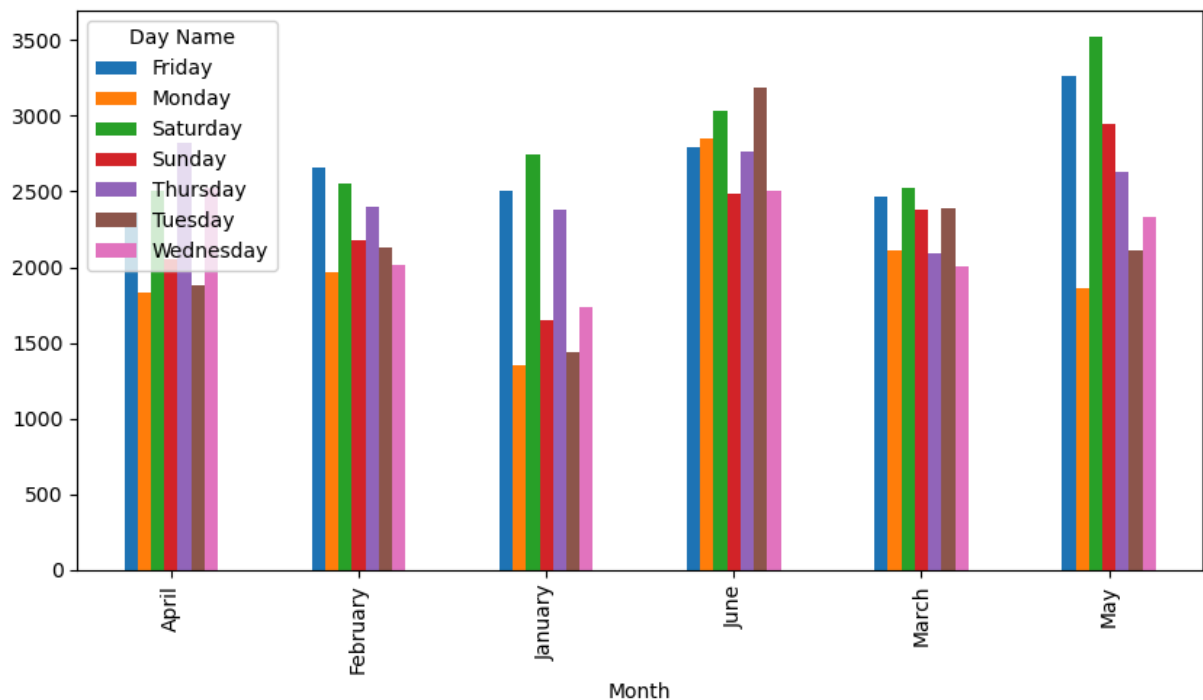
```
In [27]: pivot
```

Out[27]: **Day Name** **Friday** **Monday** **Saturday** **Sunday** **Thursday** **Tuesday** **Wednesday**

Month							
April	2365	1833	2508	2052	2823	1880	2521
February	2655	1970	2550	2183	2396	2129	2013
January	2508	1353	2745	1651	2378	1444	1740
June	2793	2848	3037	2485	2767	3187	2503
March	2465	2115	2522	2379	2093	2388	2007
May	3262	1865	3519	2944	2627	2115	2328

In [31]: `pivot.plot(kind='bar' , figsize=(10,5))`

Out[31]: `<Axes: xlabel='Month'>`



In []: `## Find the hourly rush in the city on all days`

In [33]: `summary = uber_15.groupby(['Day Name', 'Hour'], as_index=False).size()`

In [35]: `summary`

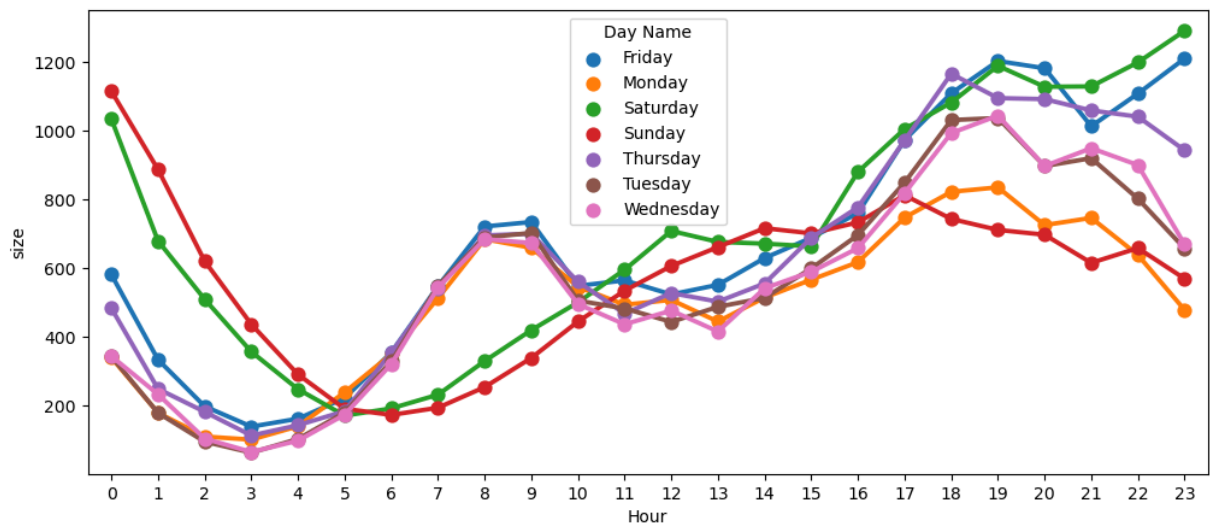
Out[35]:

	Day Name	Hour	size
0	Friday	0	581
1	Friday	1	333
2	Friday	2	197
3	Friday	3	138
4	Friday	4	161
...
163	Wednesday	19	1044
164	Wednesday	20	897
165	Wednesday	21	949
166	Wednesday	22	900
167	Wednesday	23	669

168 rows × 3 columns

```
In [39]: plt.figure(figsize=(12,5))
sns.pointplot(x="Hour",y="size" , hue="Day Name",data=summary)
```

Out[39]: <Axes: xlabel='Hour', ylabel='size'>

In [40]: *## Analyzing which Base_number has most active vehicles*

```
uber_foil= pd.read_csv(r"C:\Users\naila.iram\Downloads\Datasets\Uber-Jan-Feb-FOIL.csv")
```

In [41]: `uber_foil.head(2)`

Out[41]:

	dispatching_base_number	date	active_vehicles	trips
0	B02512	1/1/2015	190	1132
1	B02765	1/1/2015	225	1765

In [42]: !pip install chart_studio
!pip install plotly

Collecting chart_studio

Downloading chart_studio-1.1.0-py3-none-any.whl.metadata (1.3 kB)

Requirement already satisfied: plotly in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from chart_studio) (5.9.0)

Requirement already satisfied: requests in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from chart_studio) (2.31.0)

Collecting retrying>=1.3.3 (from chart_studio)

Downloading retrying-1.4.2-py3-none-any.whl.metadata (5.5 kB)

Requirement already satisfied: six in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from chart_studio) (1.16.0)

Requirement already satisfied: tenacity>=6.2.0 in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from plotly->chart_studio) (8.2.2)

Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from requests->chart_studio) (2.0.4)

Requirement already satisfied: idna<4,>=2.5 in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from requests->chart_studio) (3.4)

Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from requests->chart_studio) (2.0.7)

Requirement already satisfied: certifi>=2017.4.17 in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from requests->chart_studio) (2024.2.2)

Downloading chart_studio-1.1.0-py3-none-any.whl (64 kB)

----- 0.0/64.4 kB ? eta -:--:--

----- 0.0/64.4 kB ? eta -:--:--

----- 10.2/64.4 kB ? eta -:--:--

----- 61.4/64.4 kB 656.4 kB/s eta 0:00:01

----- 64.4/64.4 kB 576.3 kB/s eta 0:00:00

Downloading retrying-1.4.2-py3-none-any.whl (10 kB)

Installing collected packages: retrying, chart_studio

Successfully installed chart_studio-1.1.0 retrying-1.4.2

Requirement already satisfied: plotly in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (5.9.0)

Requirement already satisfied: tenacity>=6.2.0 in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from plotly) (8.2.2)

In [45]: import chart_studio.plotly as py
import plotly.graph_objs as go
import plotly.express as px
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot

In [46]: uber_foil.columns

Out[46]: Index(['dispatching_base_number', 'date', 'active_vehicles', 'trips'], dtype='object')

In [50]: px.box(x='dispatching_base_number',y='active_vehicles',data_frame=uber_foil)

```
In [52]: files= os.listdir(r"C:\Users\naila.iram\Downloads\Datasets")[-8:]
```

```
In [53]: files
```

```
Out[53]: ['uber-raw-data-apr14.csv',  
          'uber-raw-data-aug14.csv',  
          'uber-raw-data-janjune-15.csv',  
          'uber-raw-data-janjune-15_sample.csv',  
          'uber-raw-data-jul14.csv',  
          'uber-raw-data-jun14.csv',  
          'uber-raw-data-may14.csv',  
          'uber-raw-data-sep14.csv']
```

```
In [54]: files.remove('uber-raw-data-janjune-15.csv')
```

```
In [55]: files
```



```
Out[55]: ['uber-raw-data-apr14.csv',  
         'uber-raw-data-aug14.csv',  
         'uber-raw-data-janjune-15_sample.csv',  
         'uber-raw-data-jul14.csv',  
         'uber-raw-data-jun14.csv',  
         'uber-raw-data-may14.csv',  
         'uber-raw-data-sep14.csv']
```

```
In [56]: files.remove('uber-raw-data-janjune-15_sample.csv')
```

```
In [57]: files
```

```
Out[57]: ['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 [59]: ### collecting the entire datasets in the above files into one file  
final = pd.DataFrame()  
  
path=r"C:\Users\naila.iram\Downloads\Datasets"  
for file in files :  
    current_df= pd.read_csv(path+'/' +file)  
    final= pd.concat([current_df,final])
```

```
In [60]: final.shape
```

```
Out[60]: (4534327, 4)
```

```
In [61]: final.duplicated().sum()
```

```
Out[61]: 82581
```

```
In [62]: final.drop_duplicates(inplace=True)
```

```
In [63]: final.duplicated().sum()
```

```
Out[63]: 0
```

```
In [64]: final.columns
```

```
Out[64]: Index(['Date/Time', 'Lat', 'Lon', 'Base'], dtype='object')
```

```
In [65]: ### At What Locations of the city we are getting rush?  
rush_uber=final.groupby(['Lat', 'Lon'],as_index=False).size()
```

```
In [66]: rush_uber.head(2)
```

```
Out[66]:
```

	Lat	Lon	size
0	39.6569	-74.2258	1
1	39.6686	-74.1607	1

```
In [67]: !pip install folium
```

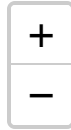
```
Collecting folium
  Downloading folium-0.20.0-py2.py3-none-any.whl.metadata (4.2 kB)
Collecting branca>=0.6.0 (from folium)
  Downloading branca-0.8.2-py3-none-any.whl.metadata (1.7 kB)
Requirement already satisfied: Jinja2>=2.9 in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from folium) (3.1.3)
Requirement already satisfied: Numpy in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from folium) (1.26.4)
Requirement already satisfied: requests in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from folium) (2.31.0)
Requirement already satisfied: xyzservices in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from folium) (2022.9.0)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from Jinja2>=2.9->folium) (2.1.3)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from requests->folium) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from requests->folium) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from requests->folium) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\naila.iram\appdata\local\anaconda3\lib\site-packages (from requests->folium) (2024.2.2)
Downloading folium-0.20.0-py2.py3-none-any.whl (113 kB)
----- 0.0/113.4 kB ? eta -:-:--
--- ----- 10.2/113.4 kB ? eta -:-:--
--- ----- 10.2/113.4 kB ? eta -:-:--
----- 41.0/113.4 kB 393.8 kB/s eta 0:00:01
----- 112.6/113.4 kB 731.4 kB/s eta 0:00:01
----- 113.4/113.4 kB 661.2 kB/s eta 0:00:00
Downloading branca-0.8.2-py3-none-any.whl (26 kB)
Installing collected packages: branca, folium
Successfully installed branca-0.8.2 folium-0.20.0
```

```
In [68]: import folium
```

```
In [71]: basemap=folium.Map()
```

```
In [72]: basemap
```

Out[72]:



```
In [73]: from folium.plugins import HeatMap
```

```
In [74]: HeatMap(rush_uber).add_to(basemap)
```

```
Out[74]: <folium.plugins.heat_map.HeatMap at 0x18bd14748d0>
```

```
In [75]: basemap
```

Out[75]:



```
In [76]: ### Rush on Hour and Weekday (Pair Wise Analysis)  
final.dtypes
```

```
Out[76]: Date/Time    object
        Lat         float64
        Lon         float64
        Base         object
        dtype: object
```

```
In [77]: final['Date/Time'] = pd.to_datetime(final['Date/Time'], format="%m/%d/%Y %H:%M:%S")
```

```
In [78]: final.dtypes
```

```
Out[78]: Date/Time    datetime64[ns]
        Lat         float64
        Lon         float64
        Base         object
        dtype: object
```

```
In [79]: final['Day'] = final['Date/Time'].dt.day
        final['Hour'] = final['Date/Time'].dt.hour
```

```
In [80]: final.head(2)
```

```
Out[80]:
```

	Date/Time	Lat	Lon	Base	Day	Hour
0	2014-09-01 00:01:00	40.2201	-74.0021	B02512	1	0
1	2014-09-01 00:01:00	40.7500	-74.0027	B02512	1	0

```
In [83]: pivot = final.groupby(['Day', 'Hour']).size().unstack()
```

```
In [85]: pivot.style.background_gradient()
```

Out[85]:

Hour	0	1	2	3	4	5	6	7	8	9	10	11	12	
Day														
1	3178	1944	1256	1308	1429	2126	3664	5380	5292	4617	4607	4729	4930	57
2	2435	1569	1087	1414	1876	2812	4920	6544	6310	4712	4797	4975	5188	56
3	3354	2142	1407	1467	1550	2387	4241	5663	5386	4657	4788	5065	5384	60
4	2897	1688	1199	1424	1696	2581	4592	6029	5704	4744	4743	4975	5193	61
5	2733	1541	1030	1253	1617	2900	4814	6261	6469	5530	5141	5011	5047	56
6	4537	2864	1864	1555	1551	2162	3642	4766	4942	4401	4801	5174	5426	62
7	3645	2296	1507	1597	1763	2422	4102	5575	5376	4639	4905	5166	5364	62
8	2830	1646	1123	1483	1889	3224	5431	7361	7357	5703	5288	5350	5483	63
9	2657	1724	1222	1480	1871	3168	5802	7592	7519	5895	5406	5443	5496	64
10	3296	2126	1464	1434	1591	2594	4664	6046	6158	5072	4976	5415	5506	65
11	3036	1665	1095	1424	1842	2520	4954	6876	6871	5396	5215	5423	5513	64
12	3227	2147	1393	1362	1757	2710	4576	6250	6231	5177	5157	5319	5570	64
13	5408	3509	2262	1832	1705	2327	4196	5685	6060	5631	5442	5720	5914	66
14	3748	2349	1605	1656	1756	2629	4257	5781	5520	4824	4911	5118	5153	57
15	2497	1515	1087	1381	1862	2980	5050	6837	6729	5201	5347	5517	5503	69
16	2547	1585	1119	1395	1818	2966	5558	7517	7495	5958	5626	5480	5525	61
17	3155	2048	1500	1488	1897	2741	4562	6315	5882	4934	5004	5306	5634	65
18	3390	2135	1332	1626	1892	2959	4688	6618	6451	5377	5150	5487	5490	63
19	3217	2188	1604	1675	1810	2639	4733	6159	6014	5006	5092	5240	5590	63
20	4475	3190	2100	1858	1618	2143	3584	4900	5083	4765	5135	5650	5745	66
21	4294	3194	1972	1727	1926	2615	4185	5727	5529	4707	4911	5212	5465	60
22	2787	1637	1175	1468	1934	3151	5204	6872	6850	5198	5277	5352	5512	63
23	2546	1580	1136	1429	1957	3132	5204	6890	6436	5177	5066	5304	5504	62
24	3200	2055	1438	1493	1798	2754	4484	6013	5913	5146	4947	5311	5229	59
25	2405	1499	1072	1439	1943	2973	5356	7627	7078	5994	5432	5504	5694	62
26	3810	3065	2046	1806	1730	2337	3776	5172	5071	4808	5061	5179	5381	61
27	5196	3635	2352	2055	1723	2336	3539	4937	5053	4771	5198	5732	5839	68
28	4123	2646	1843	1802	1883	2793	4290	5715	5671	5206	5247	5500	5486	61
29	2678	1827	1409	1678	1948	3056	5213	6852	6695	5481	5234	5163	5220	63

Hour	0	1	2	3	4	5	6	7	8	9	10	11	12	
Day														
30	2401	1510	1112	1403	1841	3216	5757	7596	7611	6064	5987	6090	6423	72
31	2174	1394	1087	919	773	997	1561	2169	2410	2525	2564	2777	2954	32

Insights

June has the highest number of Pickups

Jan(sat),Feb(Fri),Mar(sat),Apr(Thur),May(sat),June(Tues) has the highest pickups

Mostly Rush is on Saturday and friday at 23:00 hrs (also through out the evening),weekdays has the most demand during evenings

Dispatching Batch number 'B02764' has the most active vehicles.

Midtown Manhattan is clearly a huge bright spot. midtown to lower manhattan and the hrights of Brooklyn

In []: