New York Taxi Data: Load Data from CSV files & Work with Categorical Data

In [4]:
```
from os import path
import bz2
import pandas as pd
```

In [3]:
```
! pip install pandas
```

```
Collecting pandas
  Downloading pandas-1.1.0-cp37-cp37m-win_amd64.whl (9.4 MB)
Requirement already satisfied: numpy>=1.15.4 in c:\users\danal\anaconda3\envs\t
rack\lib\site-packages (from pandas) (1.19.1)
Collecting pytz>=2017.2
  Downloading pytz-2020.1-py2.py3-none-any.whl (510 kB)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\users\danal\anacond
a3\envs\track\lib\site-packages (from pandas) (2.8.1)
Requirement already satisfied: six>=1.5 in c:\users\danal\anaconda3\envs\track
\lib\site-packages (from python-dateutil>=2.7.3->pandas) (1.15.0)
Installing collected packages: pytz, pandas
Successfully installed pandas-1.1.0 pytz-2020.1
```

In [15]:
```
fname = 'C:/Users/danal/Desktop/Ex_Files_Data_Science_Python/Exercise Files/Ch07/
```

In [16]:
```
path.getsize(fname) / (1<<20)
```

Out[16]:
```
2.7408742904663086
```

In [17]:
```
with bz2.open(fname) as fp:
    print(sum(1 for line in fp))
```

```
100001
```

In [18]:
```
with bz2.open(fname, 'rt') as fp:
    for lnum, line in enumerate(fp):
        print(line[:-1])
        if lnum > 4:
            break
```

```
VendorID,lpep_pickup_datetime,Lpep_dropoff_datetime,Store_and_fwd_flag,RateCode
ID,Pickup_longitude,Pickup_latitude,Dropoff_longitude,Dropoff_latitude,Passenge
r_count,Trip_distance,Fare_amount,Extra,MTA_tax,Tip_amount,Tolls_amount,Ehail_f
ee,improvement_surcharge,Total_amount,Payment_type,Trip_type
2,2015-03-04 15:39:16,2015-03-04 15:42:30,N,1,-73.992240905761719,40.6901206970
21484,-73.999664306640625,40.684993743896484,2,.71,4.5,0,0.5,0,0,,0.3,5.3,2,1,,
2,2015-03-22 17:36:49,2015-03-22 17:45:39,N,5,-73.930038452148438,40.8195762634
27734,-73.907173156738281,40.811305999755859,2,1.41,12,0,0,0,0,,0,12,2,2,,
2,2015-03-25 22:08:45,2015-03-25 22:53:29,N,1,-73.961082458496094,40.8070220947
26563,-73.984642028808594,40.66314697265625,1,14.36,45,0.5,0.5,9.26,0,,0.3,55.5
6,1,1,,
2,2015-03-16 13:45:20,2015-03-16 13:52:04,N,1,-73.913200378417969,40.7779617309
57031,-73.926994323730469,40.772743225097656,2,1.05,6.5,0,0.5,0,0,,0.3,7.3,2,
1,,
2,2015-03-19 18:53:50,2015-03-19 18:59:04,N,1,-73.925888061523438,40.8276023864
74609,-73.916351318359375,40.824966430664063,1,.92,5.5,1,0.5,0,0,,0.3,7.3,2,1,,
```

In [19]:
```
df = pd.read_csv(fname)
```

In [20]:
```
len(df)
```

Out[20]:
```
100000
```

In [21]:
```
df.iloc[0]
```

Out[21]:
```
VendorID                  2015-03-04 15:42:30
lpep_pickup_datetime                        N
Lpep_dropoff_datetime                       1
Store_and_fwd_flag                    -73.9922
RateCodeID                             40.6901
Pickup_longitude                      -73.9997
Pickup_latitude                         40.685
Dropoff_longitude                            2
Dropoff_latitude                          0.71
Passenger_count                            4.5
Trip_distance                                0
Fare_amount                                0.5
Extra                                        0
MTA_tax                                      0
Tip_amount                                 NaN
Tolls_amount                               0.3
Ehail_fee                                  5.3
improvement_surcharge                        2
Total_amount                                 1
Payment_type                               NaN
Trip_type                                  NaN
Name: (2, 2015-03-04 15:39:16), dtype: object
```

In [22]:
```
with bz2.open(fname, 'rt') as fp:
    header = fp.readline()
    data = fp.readline()

print(header)
print(data)
```

```
VendorID,lpep_pickup_datetime,Lpep_dropoff_datetime,Store_and_fwd_flag,RateCode
ID,Pickup_longitude,Pickup_latitude,Dropoff_longitude,Dropoff_latitude,Passenge
r_count,Trip_distance,Fare_amount,Extra,MTA_tax,Tip_amount,Tolls_amount,Ehail_f
ee,improvement_surcharge,Total_amount,Payment_type,Trip_type

2,2015-03-04 15:39:16,2015-03-04 15:42:30,N,1,-73.992240905761719,40.6901206970
21484,-73.999664306640625,40.684993743896484,2,.71,4.5,0,0.5,0,0,,0.3,5.3,2,1,,
```

In [23]:
```
len(header.split(','))
```

Out[23]: 21

In [24]:
```
len(data.split(','))
```

Out[24]: 23

In [25]:
```
import numpy as np
df = pd.read_csv(fname, usecols=np.arange(21))
```

In [26]:
```
df.iloc[0]
```

Out[26]:
```
VendorID                              2
lpep_pickup_datetime     2015-03-04 15:39:16
Lpep_dropoff_datetime    2015-03-04 15:42:30
Store_and_fwd_flag                       N
RateCodeID                               1
Pickup_longitude                   -73.9922
Pickup_latitude                     40.6901
Dropoff_longitude                  -73.9997
Dropoff_latitude                     40.685
Passenger_count                          2
Trip_distance                         0.71
Fare_amount                            4.5
Extra                                    0
MTA_tax                                0.5
Tip_amount                               0
Tolls_amount                             0
Ehail_fee                              NaN
improvement_surcharge                  0.3
Total_amount                           5.3
Payment_type                             2
Trip_type                                1
Name: 0, dtype: object
```

In [27]:
```
df.dtypes
```

Out[27]:
```
VendorID                  int64
lpep_pickup_datetime     object
Lpep_dropoff_datetime    object
Store_and_fwd_flag       object
RateCodeID                int64
Pickup_longitude        float64
Pickup_latitude         float64
Dropoff_longitude       float64
Dropoff_latitude        float64
Passenger_count           int64
Trip_distance           float64
Fare_amount             float64
Extra                   float64
MTA_tax                 float64
Tip_amount              float64
Tolls_amount            float64
Ehail_fee               float64
improvement_surcharge   float64
Total_amount            float64
Payment_type              int64
Trip_type                 int64
dtype: object
```

In [31]:
```
df = pd.read_csv(fname, usecols=np.arange(21), parse_dates=['lpep_pickup_datetime
```

In [32]:
```
df.dtypes
```

Out[32]:
```
VendorID                         int64
lpep_pickup_datetime    datetime64[ns]
Lpep_dropoff_datetime   datetime64[ns]
Store_and_fwd_flag              object
RateCodeID                       int64
Pickup_longitude               float64
Pickup_latitude                float64
Dropoff_longitude              float64
Dropoff_latitude               float64
Passenger_count                  int64
Trip_distance                  float64
Fare_amount                    float64
Extra                          float64
MTA_tax                        float64
Tip_amount                     float64
Tolls_amount                   float64
Ehail_fee                      float64
improvement_surcharge          float64
Total_amount                   float64
Payment_type                     int64
Trip_type                        int64
dtype: object
```

In [33]:
```
df['VendorID'].unique()
```

Out[33]:
```
array([2, 1], dtype=int64)
```

In [34]:
```
df['Vendor'] = df['VendorID'].apply({1: 'Creative', 2: 'VeriFone'}.get)
df['Vendor'].head()
```

Out[34]:
```
0    VeriFone
1    VeriFone
2    VeriFone
3    VeriFone
4    VeriFone
Name: Vendor, dtype: object
```

In [35]:
```
df['Vendor'].memory_usage() / (1<<20)
```

Out[35]:
```
0.7630615234375
```

In [36]:
```
df['Vendor'] = df['VendorID'].apply({1: 'Creative', 2: 'VeriFone'}.get).astype('c
```

In [37]:
```
df['Vendor'].memory_usage() / (1<<20)
```

Out[37]:
```
0.0955810546875
```

In [38]:
```
df['Vendor'].head().cat.codes
```

Out[38]:
```
0    1
1    1
2    1
3    1
4    1
dtype: int8
```

In [39]:
```
len(df[df['Vendor'] == 'VeriFone'])
```

Out[39]:
```
77946
```

In [40]:
```
df['lpep_pickup_datetime'].head().dt.round('H')
```

Out[40]:
```
0   2015-03-04 16:00:00
1   2015-03-22 18:00:00
2   2015-03-25 22:00:00
3   2015-03-16 14:00:00
4   2015-03-19 19:00:00
Name: lpep_pickup_datetime, dtype: datetime64[ns]
```

In [41]:
```
keys = df['lpep_pickup_datetime'].dt.round('H')
df.groupby(keys)
```

Out[41]:
```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x000002CA1B9D8748>
```

In [42]:
```
df.groupby(keys).count().head()
```

Out[42]:

lpep_pickup_datetime	VendorID	lpep_pickup_datetime	Lpep_dropoff_datetime	Store_and_fwd_flag
2015-03-01 00:00:00	153	153	153	153
2015-03-01 01:00:00	266	266	266	266
2015-03-01 02:00:00	241	241	241	241
2015-03-01 03:00:00	180	180	180	180
2015-03-01 04:00:00	172	172	172	172

5 rows × 22 columns

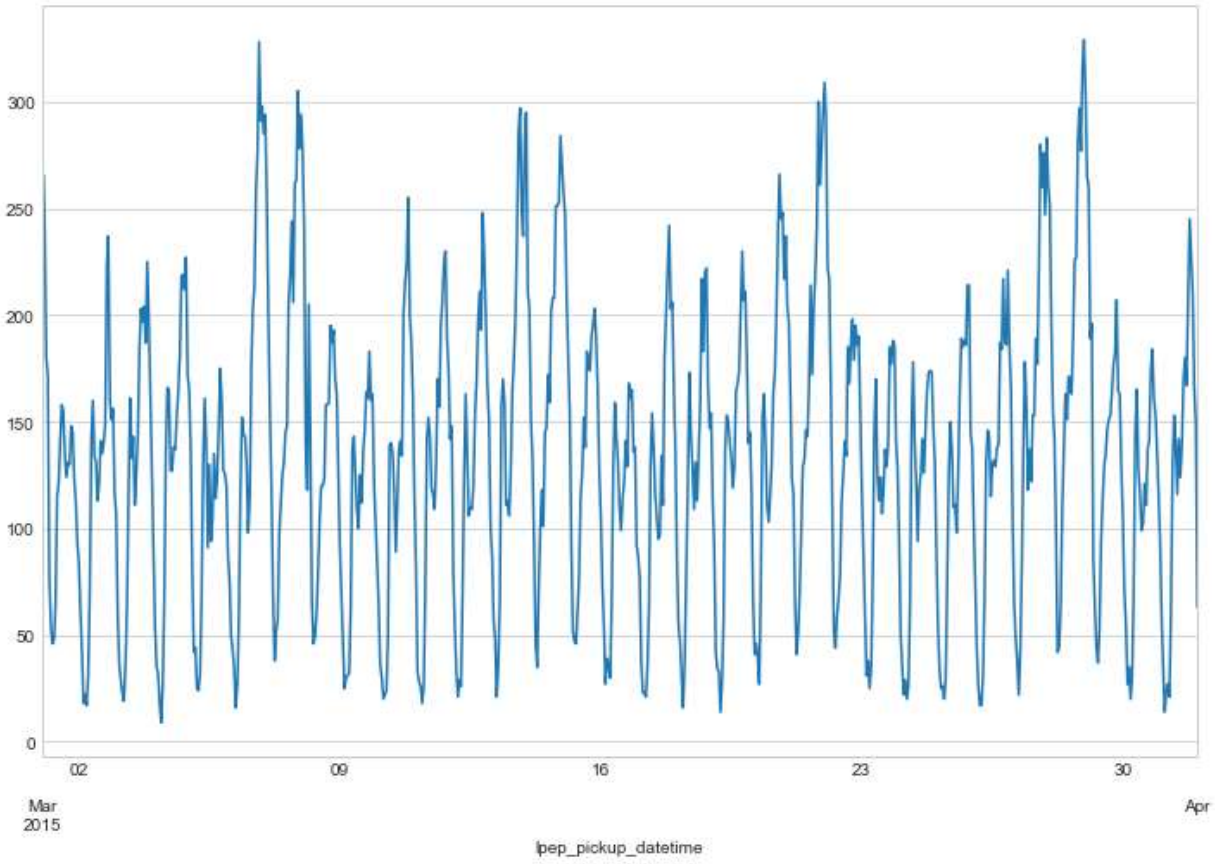
In [45]:
```
!pip install matplotlib
```

```
Collecting matplotlib
  Downloading matplotlib-3.3.0-cp37-cp37m-win_amd64.whl (8.8 MB)
Collecting pillow>=6.2.0
  Downloading Pillow-7.2.0-cp37-cp37m-win_amd64.whl (2.1 MB)
Collecting cycler>=0.10
  Downloading cycler-0.10.0-py2.py3-none-any.whl (6.5 kB)
Collecting kiwisolver>=1.0.1
  Downloading kiwisolver-1.2.0-cp37-none-win_amd64.whl (57 kB)
Requirement already satisfied: numpy>=1.15 in c:\users\danal\anaconda3\envs\tra
ck\lib\site-packages (from matplotlib) (1.19.1)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\danal\anaconda3
\envs\track\lib\site-packages (from matplotlib) (2.8.1)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.3 in c:\u
sers\danal\anaconda3\envs\track\lib\site-packages (from matplotlib) (2.4.7)
Requirement already satisfied: six in c:\users\danal\anaconda3\envs\track\lib\s
ite-packages (from cycler>=0.10->matplotlib) (1.15.0)
Installing collected packages: pillow, cycler, kiwisolver, matplotlib
Successfully installed cycler-0.10.0 kiwisolver-1.2.0 matplotlib-3.3.0 pillow-
7.2.0
```

In [50]:
```
%matplotlib inline
import matplotlib.pyplot as plt
plt.style.use('seaborn-whitegrid')
plt.rcParams['figure.figsize'] = [12,8]
```

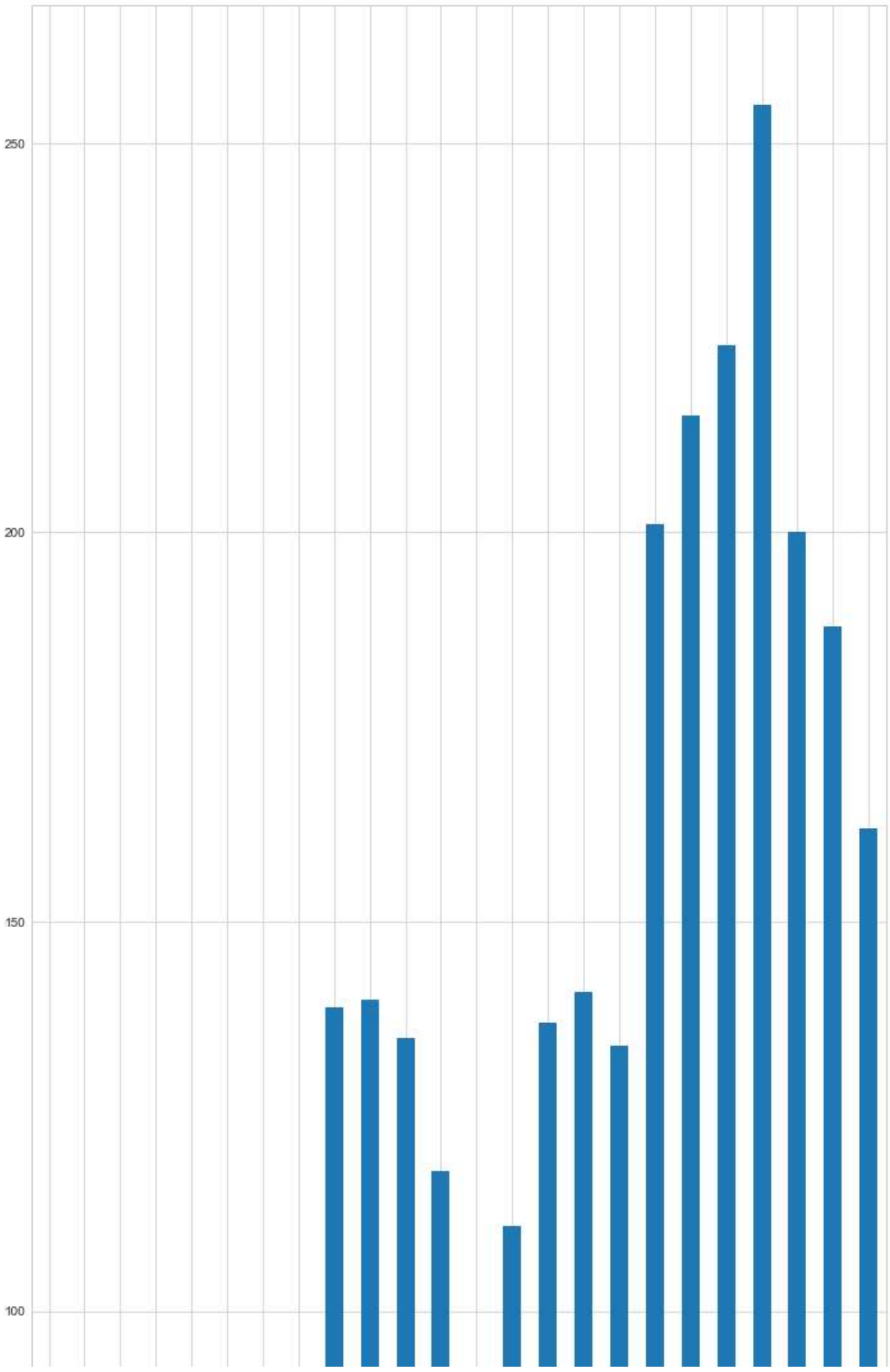
In [47]:
```
df.groupby(keys).count()['Vendor'].plot()
```

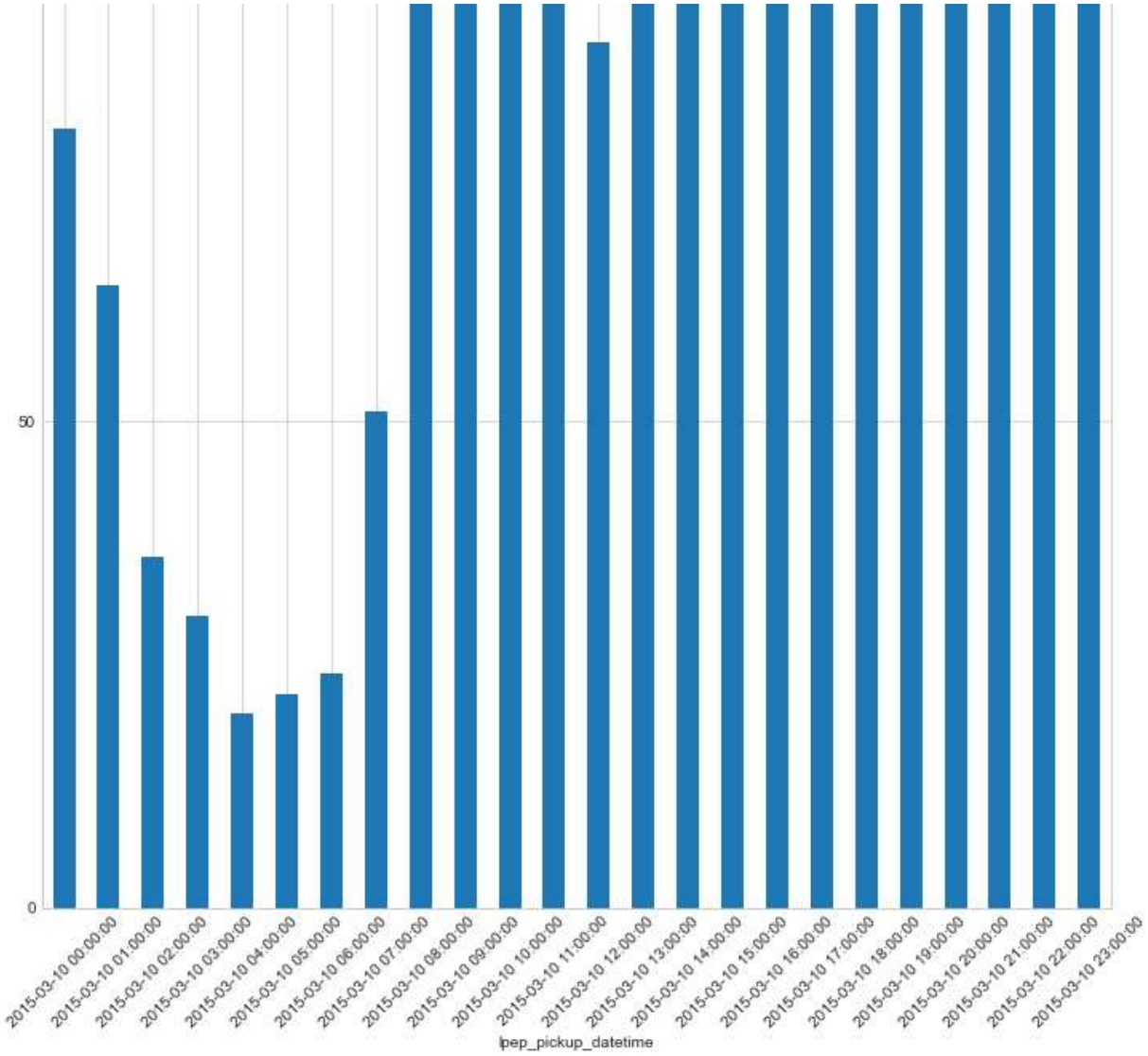
Out[47]:
```
<AxesSubplot:xlabel='lpep_pickup_datetime'>
```



In [51]:
```
df.groupby(keys).count()['Vendor'].loc['2015-03-10'].plot.bar(rot=45)
```

Out[51]:
```
<AxesSubplot:xlabel='lpep_pickup_datetime'>
```





In [52]:
```
df['hour'] = df['lpep_pickup_datetime'].dt.hour
df['day'] = df['lpep_pickup_datetime'].dt.date
```

In [53]:
```
df[['hour', 'day']].head()
```

Out[53]:

	hour	day
0	15	2015-03-04
1	17	2015-03-22
2	22	2015-03-25
3	13	2015-03-16
4	18	2015-03-19

In [54]:
```
df.groupby(['Vendor', 'day', 'hour']).count().head()
```

Out[54]:

			VendorID	lpep_pickup_datetime	Lpep_dropoff_datetime	Store_and_fwd_flag
Vendor	**day**	**hour**				
Creative	**2015-03-01**	**0**	60.0	60.0	60.0	60.0
		1	60.0	60.0	60.0	60.0
		2	51.0	51.0	51.0	51.0
		3	41.0	41.0	41.0	41.0
		4	28.0	28.0	28.0	28.0

5 rows × 21 columns

In [55]:
```
df.groupby(['Vendor', 'day', 'hour']).count().index
```

Out[55]:
```
MultiIndex([('Creative', 2015-03-01,  0),
            ('Creative', 2015-03-01,  1),
            ('Creative', 2015-03-01,  2),
            ('Creative', 2015-03-01,  3),
            ('Creative', 2015-03-01,  4),
            ('Creative', 2015-03-01,  5),
            ('Creative', 2015-03-01,  6),
            ('Creative', 2015-03-01,  7),
            ('Creative', 2015-03-01,  8),
            ('Creative', 2015-03-01,  9),
            ...
            ('VeriFone', 2015-03-31, 14),
            ('VeriFone', 2015-03-31, 15),
            ('VeriFone', 2015-03-31, 16),
            ('VeriFone', 2015-03-31, 17),
            ('VeriFone', 2015-03-31, 18),
            ('VeriFone', 2015-03-31, 19),
            ('VeriFone', 2015-03-31, 20),
            ('VeriFone', 2015-03-31, 21),
            ('VeriFone', 2015-03-31, 22),
            ('VeriFone', 2015-03-31, 23)],
           names=['Vendor', 'day', 'hour'], length=1488)
```

In [56]:
```
ddf = df.groupby(['Vendor', 'day', 'hour'], as_index=False).count()
```

In [57]:
```
ddf.head()
```

Out[57]:

	Vendor	day	hour	VendorID	lpep_pickup_datetime	Lpep_dropoff_datetime	Store_and_fwd_fla
0	Creative	2015-03-01	0	60.0	60.0	60.0	60
1	Creative	2015-03-01	1	60.0	60.0	60.0	60
2	Creative	2015-03-01	2	51.0	51.0	51.0	51
3	Creative	2015-03-01	3	41.0	41.0	41.0	41
4	Creative	2015-03-01	4	28.0	28.0	28.0	28

5 rows × 24 columns

In [58]:
```
hdf = ddf.groupby(['Vendor', 'hour'], as_index=False).median()
hdf.head()
```

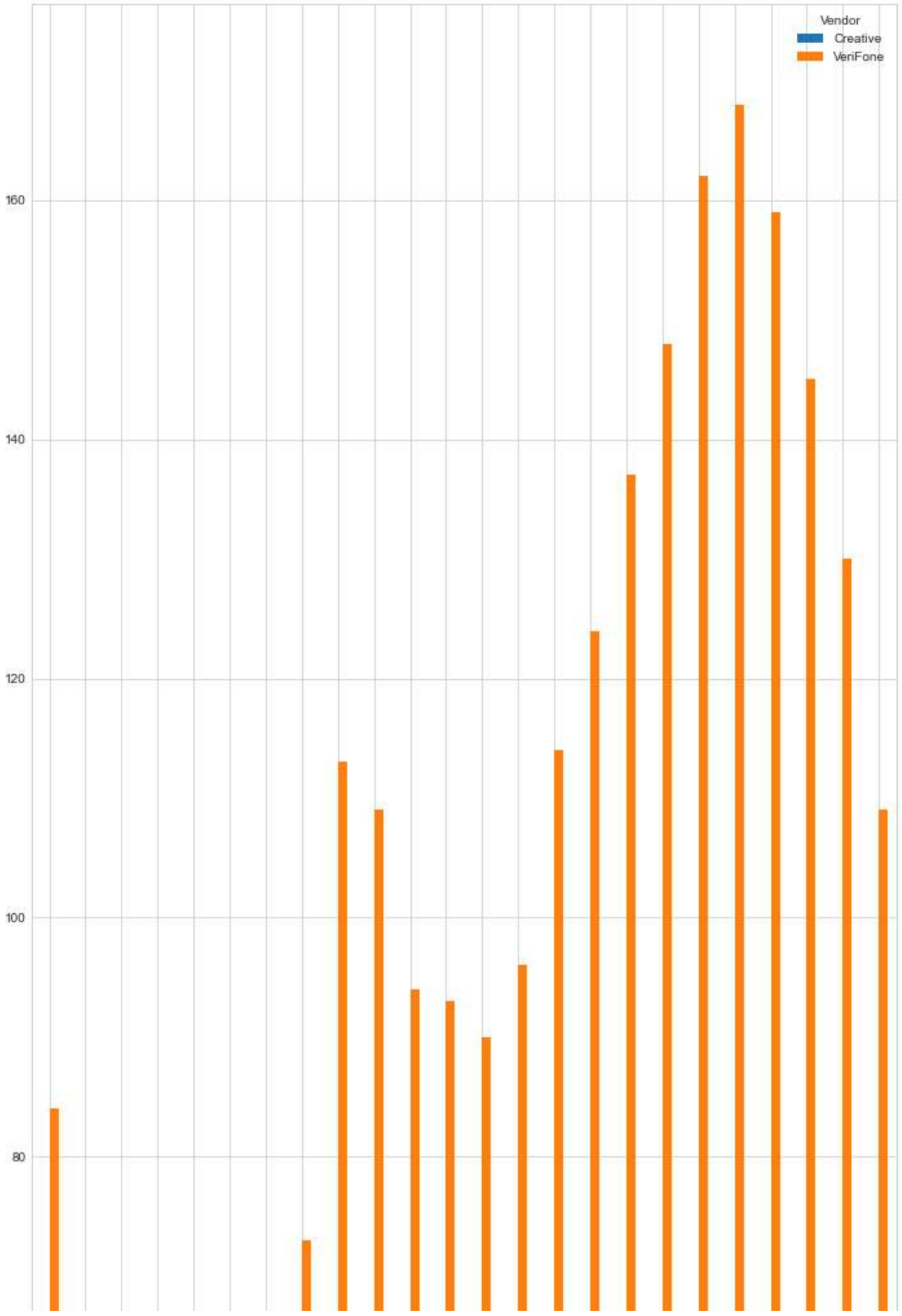
Out[58]:

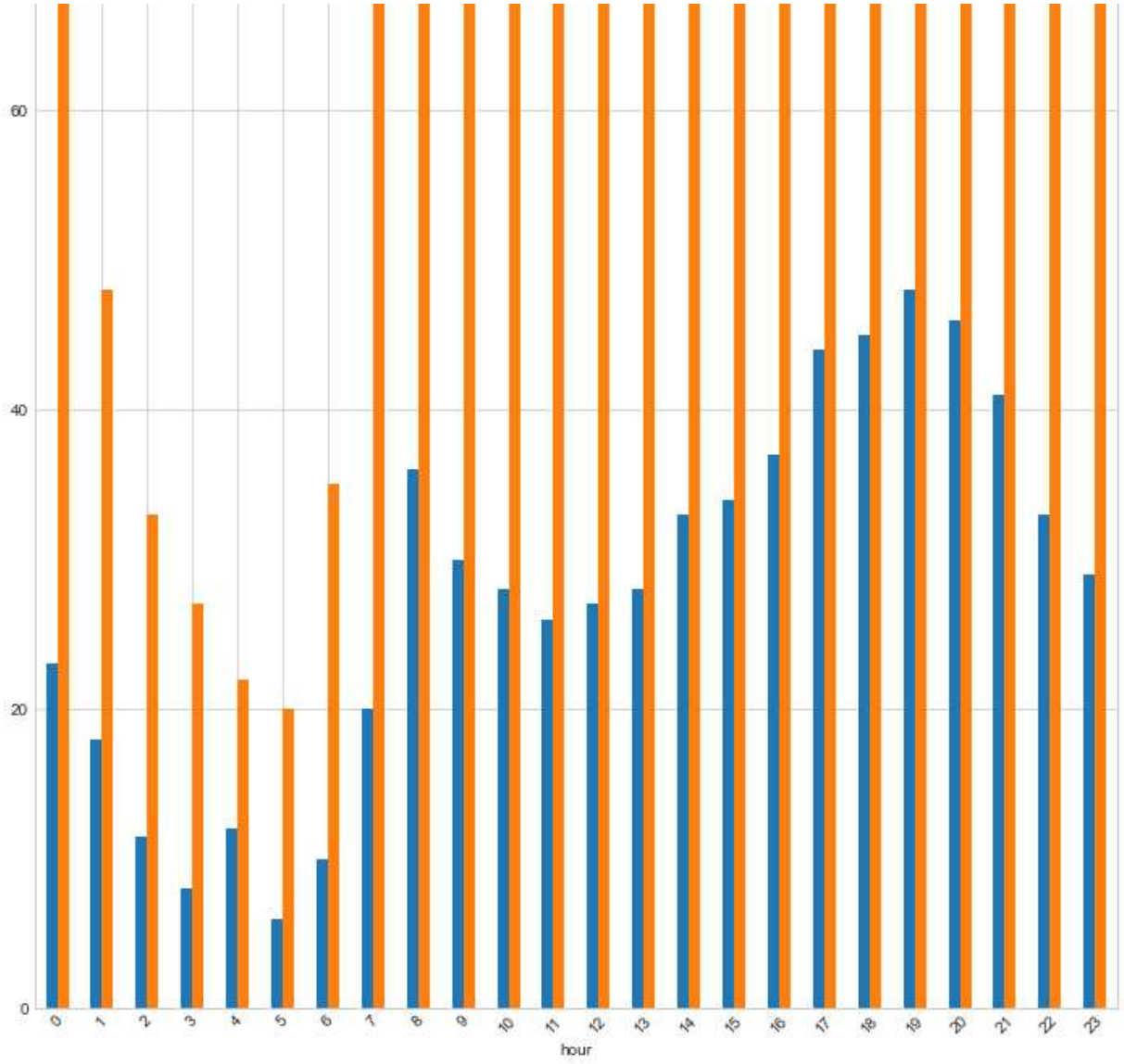
	Vendor	hour	VendorID	lpep_pickup_datetime	Lpep_dropoff_datetime	Store_and_fwd_flag	Rat
0	Creative	0	23.0	23.0	23.0	23.0	
1	Creative	1	18.0	18.0	18.0	18.0	
2	Creative	2	11.5	11.5	11.5	11.5	
3	Creative	3	8.0	8.0	8.0	8.0	
4	Creative	4	12.0	12.0	12.0	12.0	

5 rows × 23 columns

In [60]:
```
vdf = hdf.pivot(columns='Vendor', index='hour', values='Extra')
vdf.plot.bar(rot=45)
```

Out[60]: <AxesSubplot:xlabel='hour'>





In [62]:
```
import sqlite3
conn = sqlite3.connect('C:/Users/danal/Desktop/Ex_Files_Data_Science_Python/Exerc
```

In [63]:
```
wdf = pd.read_sql('SELECT * FROM weather', conn)
wdf.columns
```

Out[63]:
```
Index(['STATION', 'DATE', 'PRCP', 'SNOW', 'TMAX', 'TMIN'], dtype='object')
```

In [64]:
```
wdf = pd.read_sql("SELECT * FROM weather", conn, parse_dates=['DATE'], index_col=
wdf.dtypes
```

Out[64]:
```
STATION     object
PRCP       float64
SNOW       float64
TMAX         int64
TMIN         int64
dtype: object
```

In [65]:
```
wdf.index
```

Out[65]:
```
DatetimeIndex(['2015-03-01', '2015-03-02', '2015-03-03', '2015-03-04',
               '2015-03-05', '2015-03-06', '2015-03-07', '2015-03-08',
               '2015-03-09', '2015-03-10',
               ...
               '2016-03-23', '2016-03-24', '2016-03-25', '2016-03-26',
               '2016-03-27', '2016-03-28', '2016-03-29', '2016-03-30',
               '2016-03-31', '2016-04-01'],
              dtype='datetime64[ns]', name='DATE', length=398, freq=None)
```

In [66]:
```
wdf.describe()
```

Out[66]:

	PRCP	SNOW	TMAX	TMIN
count	398.000000	398.000000	398.000000	398.000000
mean	0.109799	0.129146	65.017588	50.035176
std	0.310245	1.463461	17.556593	16.336782
min	0.000000	0.000000	15.000000	-1.000000
25%	0.000000	0.000000	52.000000	38.000000
50%	0.000000	0.000000	65.000000	50.000000
75%	0.020000	0.000000	81.750000	65.000000
max	2.310000	27.300000	97.000000	82.000000

In [69]:
```
! pip install scipy
```

```
Collecting scipy
  Downloading scipy-1.5.2-cp37-cp37m-win_amd64.whl (31.2 MB)
Requirement already satisfied: numpy>=1.14.5 in c:\users\danal\anaconda3\envs\t
rack\lib\site-packages (from scipy) (1.19.1)
Installing collected packages: scipy
Successfully installed scipy-1.5.2
```

In [70]:
```
from scipy.constants import convert_temperature
wdf['tempF']= convert_temperature(wdf['TMAX']/10, 'C', 'F')
wdf.head()
```

Out[70]:

DATE	STATION	PRCP	SNOW	TMAX	TMIN	tempF
2015-03-01	GHCND:USW00094728	0.52	4.8	31	24	37.58
2015-03-02	GHCND:USW00094728	0.00	0.0	39	27	39.02
2015-03-03	GHCND:USW00094728	0.67	1.8	37	22	38.66
2015-03-04	GHCND:USW00094728	0.25	0.0	45	35	40.10
2015-03-05	GHCND:USW00094728	0.76	7.5	40	19	39.20

In [71]:
```
ddf = df.groupby(df['lpep_pickup_datetime'].dt.date).count()
```

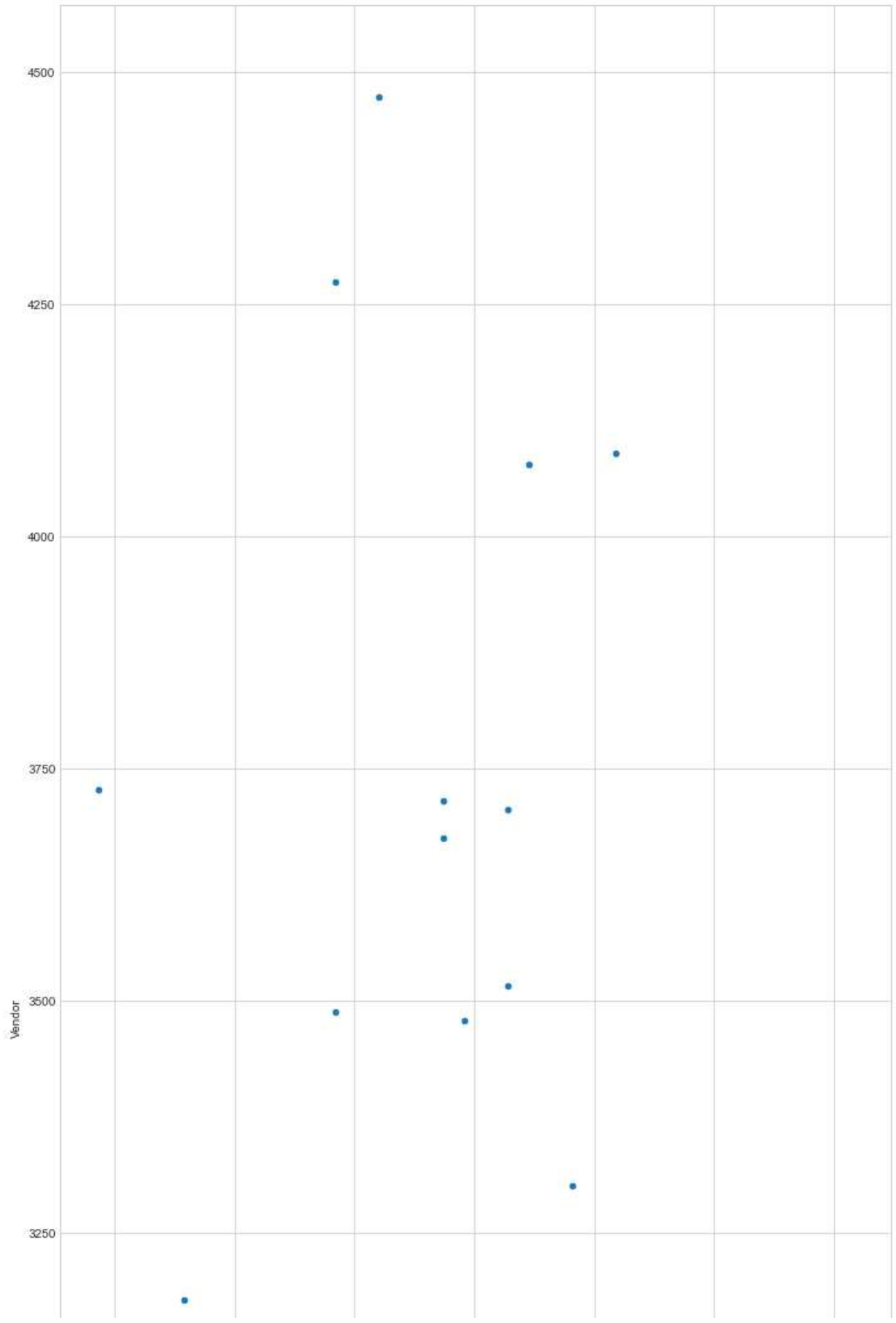
In [72]:
```
jdf = ddf.join(wdf)
jdf.head()
```

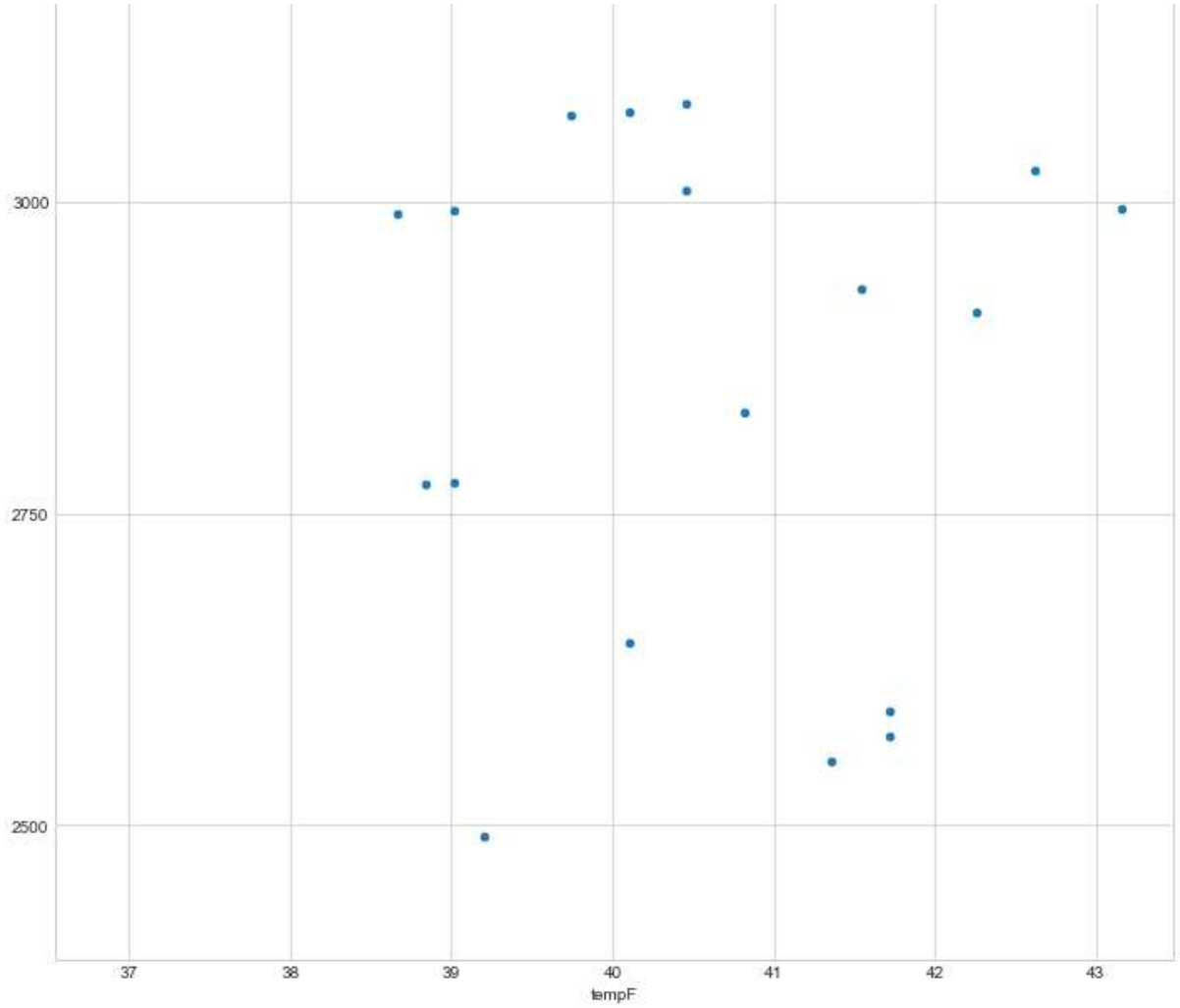
Out[72]:

kup_latitude	Dropoff_longitude	Dropoff_latitude	Passenger_count	...	Trip_type	Vendor	hour	day
3177	3177	3177	3177	...	3177	3177	3177	3177
2775	2775	2775	2775	...	2775	2775	2775	2775
2990	2990	2990	2990	...	2990	2990	2990	2990
3072	3072	3072	3072	...	3072	3072	3072	3072
2491	2491	2491	2491	...	2491	2491	2491	2491

In [73]:
```
jdf.plot.scatter(x='tempF', y='Vendor')
```

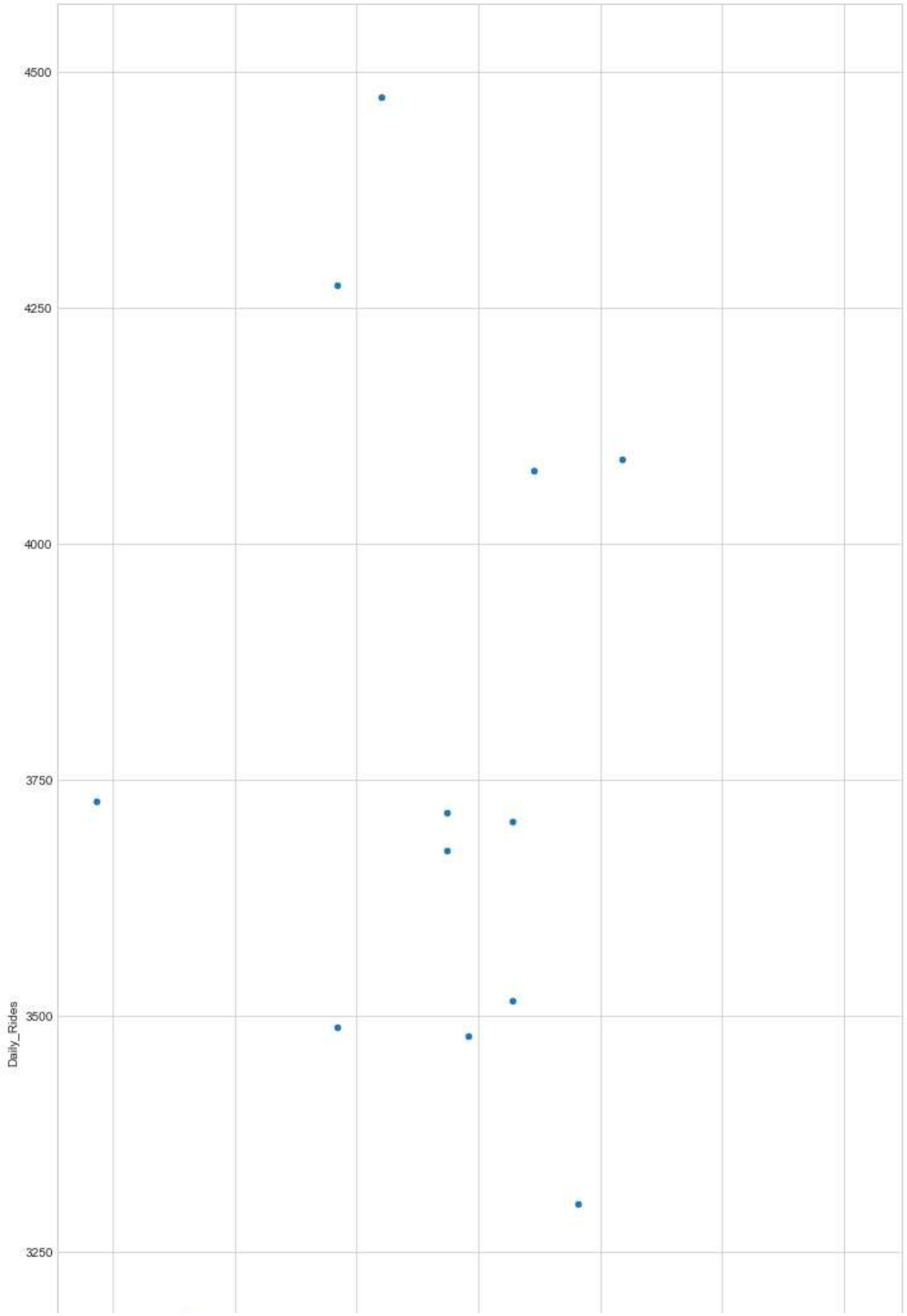
Out[73]:
```
<AxesSubplot:xlabel='tempF', ylabel='Vendor'>
```

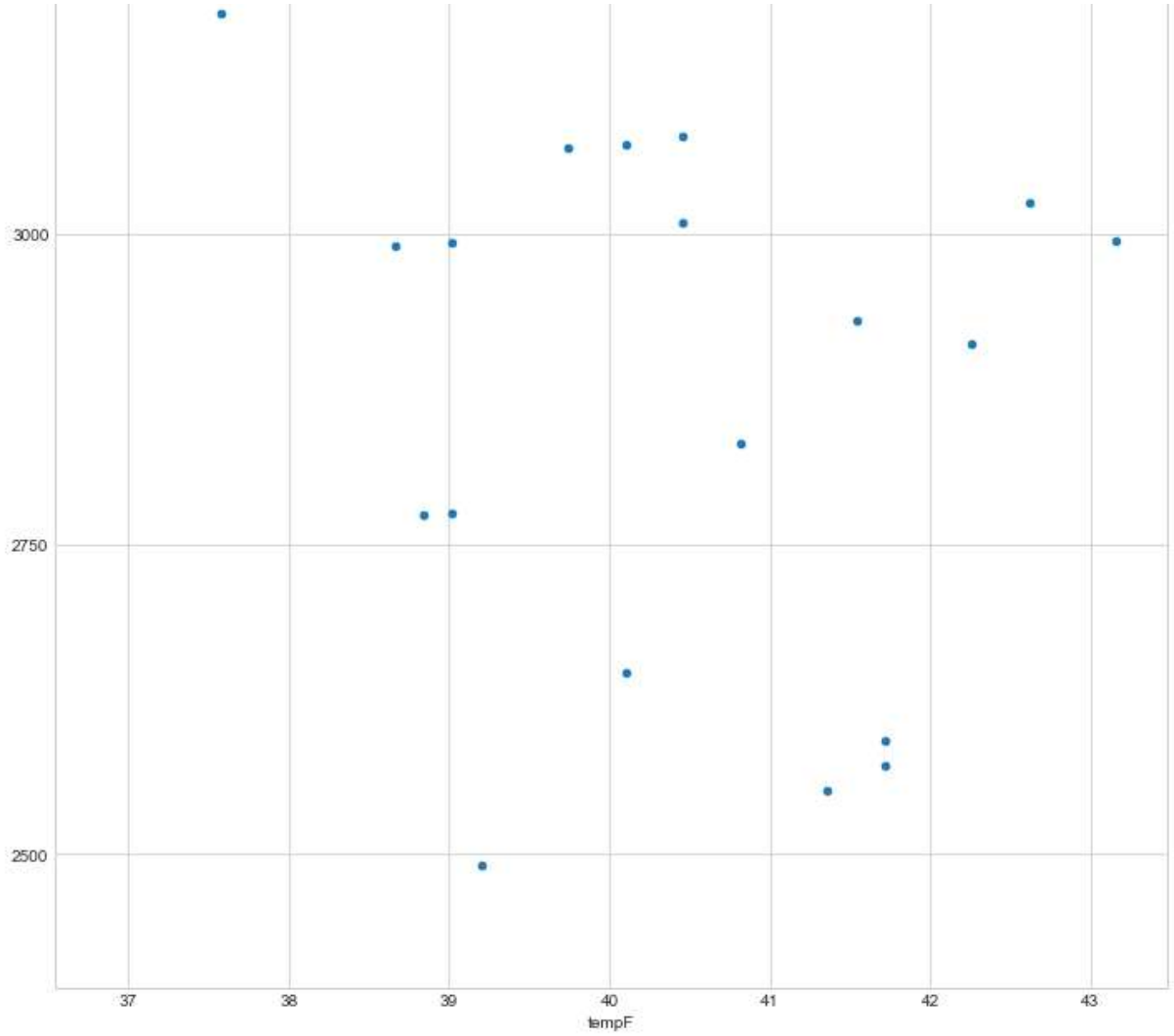



3000
2750
2500
37
38
39
40
41
42
43
tempF

In [74]:
```
ax = jdf.plot.scatter(x = 'tempF', y = 'Vendor')
ax.set_ylabel('Daily_Rides')
```

Out[74]:
```
Text(0, 0.5, 'Daily_Rides')
```





In [ ]: