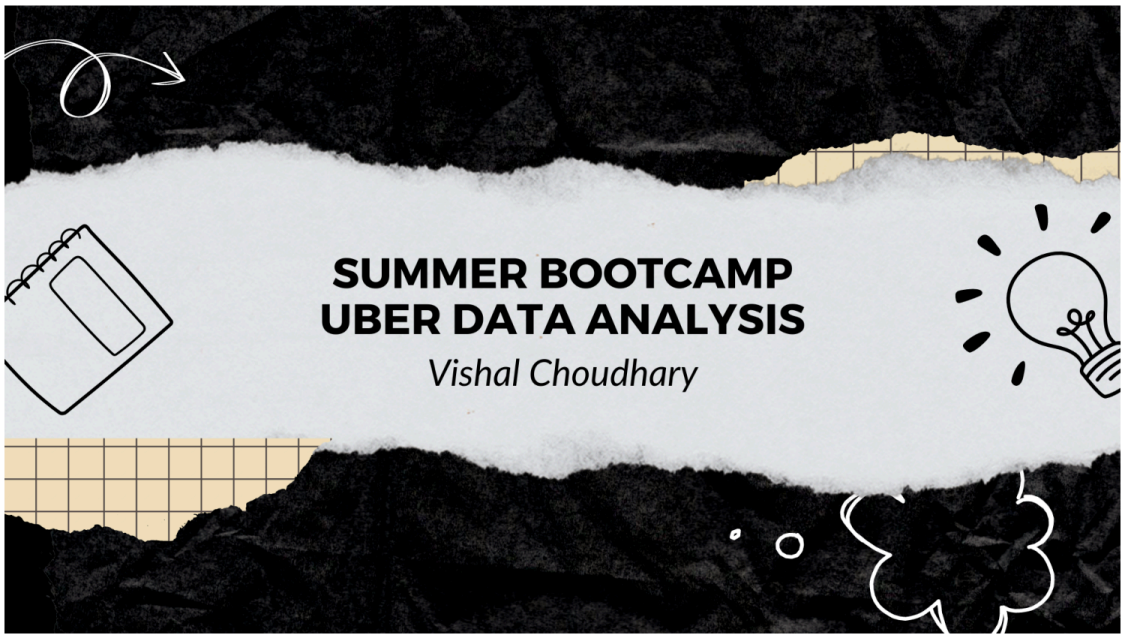
In [1]:



Uber Data Analysis

In [2]:

Out[2]: '/Users/vishal'

Importing the necessary Libraries

In [3]:

Loading the dataset

In [4]:

Basic Exploration

1- Display the top 5 rows.

In [6]:

	pickup_dt	borough	pickups	spd	vsb	temp	dewp	slp	pcp01	pcp06	pcp24	sd	hday
0	1/1/2015 1:00	Bronx	152.0	5.0	10.0	30.0	7.0	1023.5	0.0	0.0	0.0	0.0	Y
1	1/1/2015 1:00	Brooklyn	1519.0	5.0	10.0	NaN	7.0	1023.5	0.0	0.0	0.0	0.0	?
2	1/1/2015 1:00	EWR	0.0	5.0	10.0	30.0	7.0	1023.5	0.0	0.0	0.0	0.0	Y
3	1/1/2015 1:00	Manhattan	5258.0	5.0	10.0	30.0	7.0	1023.5	0.0	0.0	0.0	0.0	Y
4	1/1/2015 1:00	Queens	405.0	5.0	10.0	30.0	7.0	1023.5	0.0	0.0	0.0	0.0	Y

Obervation :- on column 'temp' and 'hday' there is missing values on 2nd row Brooklyn temp = nan and hday = wrong entry(?)

2- Display the last 5 rows

In [8]:

	pickup_dt	borough	pickups	spd	vsb	temp	dewp	slp	pcp01	pcp06	pcp24	sd	hday
29096	30-06-2015 23:00	EWR	0.0	7.0	10.0	75.0	65.0	1011.8	0.0	0.0	0.0	0.0	N
29097	30-06-2015 23:00	Manhattan	3828.0	7.0	10.0	75.0	65.0	1011.8	0.0	0.0	0.0	0.0	N
29098	30-06-2015 23:00	Queens	580.0	7.0	10.0	75.0	65.0	1011.8	0.0	0.0	0.0	0.0	N
29099	30-06-2015 23:00	Staten Island	0.0	7.0	10.0	75.0	65.0	1011.8	0.0	0.0	0.0	0.0	N
29100	30-06-2015 23:00	NaN	3.0	7.0	10.0	75.0	65.0	1011.8	0.0	0.0	0.0	0.0	N

Obervation :- In column 'borough' and row 29100 value = NaN

3- Check the shape of dataset.

In [158]:

```
(29101, 13)
```

Observation:- number of Rows = 29101 and number of column = 13

4- Check the info of dataset

In [36]:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 29101 entries, 0 to 29100
Data columns (total 13 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   pickup_dt  17559 non-null  object 
 1   borough    26058 non-null  object 
 2   pickups    29099 non-null  float64
 3   spd        29101 non-null  float64
 4   vsb        29101 non-null  float64
 5   temp       28742 non-null  float64
 6   dewp       29101 non-null  float64
 7   slp        29101 non-null  float64
 8   pcp01      29101 non-null  float64
 9   pcp06      29101 non-null  float64
 10  pcp24      29101 non-null  float64
 11  sd         29101 non-null  float64
 12  hday       29099 non-null  object 
dtypes: float64(10), object(3)
memory usage: 2.9+ MB
```

5- Check the datatypes of each feature.

Obeservation:- We seen that pickup_dt is in wrong data type we have to convert into date time format, and also we seen their is two more columns in our data set which is not showing here where names are weekday and is_weekend

In [13]:

```
pickup_dt     object
borough       object
pickups      float64
spd          float64
vsb          float64
temp         float64
dewp         float64
slp          float64
pcp01        float64
pcp06        float64
pcp24        float64
sd           float64
hday          object
dtype: object
```

Converting pickup_dt in date time format

After Converting pickup_dt in date time format

In [38]:

```
pickup_dt    datetime64[ns]
borough              object
pickups             float64
spd                 float64
vsb                 float64
temp                float64
dewp                float64
slp                 float64
pcp01               float64
pcp06               float64
pcp24               float64
sd                  float64
hday                 object
dtype: object
```

6- Check the Statistical summary

obervation:- minimum dew point = -16 and maximum sea level pressure = 1015200.00

In [39]:

	count	mean	min	25%	50%	75%	max	std
pickup_dt	17559	2015-04-08 05:01:44.151717120	2015-01-13 00:00:00	2015-02-21 15:00:00	2015-04-14 00:00:00	2015-05-23 03:00:00	2015-06-30 23:00:00	NaN
pickups	29099.0	490.236022	0.0	1.0	54.0	449.0	7883.0	995.680628
spd	29101.0	5.984924	0.0	3.0	6.0	8.0	21.0	3.699007
vsb	29101.0	8.818125	0.0	9.1	10.0	10.0	10.0	2.442897
temp	28742.0	47.900262	0.0	32.0	46.5	65.0	89.0	19.800541
dewp	29101.0	30.823065	-16.0	14.0	30.0	50.0	73.0	21.283444
slp	29101.0	1052.633123	1.0	1012.5	1018.2	1022.9	1015200.0	5945.147362
pcp01	29101.0	0.00383	0.0	0.0	0.0	0.0	0.28	0.018933
pcp06	29101.0	0.026129	0.0	0.0	0.0	0.0	1.24	0.093125
pcp24	29101.0	0.090464	0.0	0.0	0.0	0.05	2.1	0.219402
sd	29101.0	2.529169	0.0	0.0	0.0	2.958333	19.0	4.520325

7- Check the null values

Observation:- column borough having "3043" null values, pickups having "2" null values and temp having "359" null values and pickup_dt having "11542" null values

In [20]:

```
pickup_dt    11542
borough       3043
pickups          2
spd              0
vsb              0
temp           359
dewp             0
slp              0
pcp01            0
pcp06            0
pcp24            0
sd               0
hday             0
dtype: int64
```

Null values in percentage

Observation :- Here hday is showing zero null value but by checking data set we seen there is two wrong value, we must have to replace wrong values from Null

In [22]:

```
pickup_dt    39.661867
borough      10.456685
pickups       0.006873
spd           0.000000
vsb           0.000000
temp          1.233635
dewp          0.000000
slp           0.000000
pcp01         0.000000
pcp06         0.000000
pcp24         0.000000
sd            0.000000
hday          0.000000
dtype: float64
```

8. Check the duplicate values

In [9]:

Out[9]: 0

9- Check the anomalies or wrong entries.

Identifying Wrong Value

In [17]:

Out[17]:

	pickup_dt	borough	pickups	spd	vsb	temp	dewp	slp	pcp01	pcp06	pcp24	sd	hday
1	2015-01-01 01:00:00	Brooklyn	1519.0	5.0	10.0	NaN	7.0	1023.5	0.0	0.0	0.0	0.0	?
123	2015-01-01 19:00:00	Queens	238.0	7.0	10.0	37.0	7.0	1016.2	0.0	0.0	0.0	0.0	?

In [20]:

pickup_dt	borough	pickups	spd	vsb	temp	dewp	slp	pcp01	pcp06	pcp24	sd	hday

Observation in above we remove all wrong entries using Null values

After making some changes we are repeating step 7 again and again we check for NULL values

Top ten null values

In [22]:

	pickup_dt	borough	pickups	spd	vsb	temp	dewp	slp	pcp01	pcp06	pcp24	sd	hday
1	1/1/2015 1:00	Brooklyn	1519.0	5.0	10.0	NaN	7.0	1023.5	0.0	0.0	0.0	0.0	NaN
6	1/1/2015 1:00	NaN	4.0	5.0	10.0	30.0	7.0	1023.5	0.0	0.0	0.0	0.0	Y
8	1/1/2015 2:00	Brooklyn	1229.0	3.0	10.0	NaN	6.0	1023.0	0.0	0.0	0.0	0.0	Y
13	1/1/2015 2:00	NaN	11.0	3.0	10.0	30.0	6.0	1023.0	0.0	0.0	0.0	0.0	Y
15	1/1/2015 3:00	Brooklyn	1601.0	5.0	10.0	NaN	8.0	1022.3	0.0	0.0	0.0	0.0	Y
20	1/1/2015 3:00	NaN	1.0	5.0	10.0	30.0	8.0	1022.3	0.0	0.0	0.0	0.0	Y
21	1/1/2015 4:00	Bronx	NaN	5.0	10.0	29.0	9.0	1022.0	0.0	0.0	0.0	0.0	Y
22	1/1/2015 4:00	Brooklyn	1390.0	5.0	10.0	NaN	9.0	1022.0	0.0	0.0	0.0	0.0	Y
27	1/1/2015 4:00	NaN	2.0	5.0	10.0	29.0	9.0	1022.0	0.0	0.0	0.0	0.0	Y
29	1/1/2015 5:00	Brooklyn	759.0	5.0	10.0	NaN	9.0	1021.8	0.0	0.0	0.0	0.0	Y

NULL Values in percentage

In [24]:

```
pickup_dt    39.661867
borough      10.456685
pickups       0.006873
spd           0.000000
vsb           0.000000
temp          1.233635
dewp          0.000000
slp           0.000000
pcp01         0.000000
pcp06         0.000000
pcp24         0.000000
sd            0.000000
hday          0.006873
dtype: float64
```

Observation :- Now we can see in our data "hday" also having null values and after converting pickup_dt into date time fromat we can also see null values in pickup_dt

BEFORE Replacing all null values using median and mode because we also have objective type data type

In [26]:

	pickup_dt	borough	pickups	spd	vsb	temp	dewp	slp	pcp01	pcp06	pcp24	sd	hday
1	1/1/2015 1:00	Brooklyn	1519.0	5.0	10.0	NaN	7.0	1023.5	0.0	0.0	0.0	0.0	NaN
6	1/1/2015 1:00	NaN	4.0	5.0	10.0	30.0	7.0	1023.5	0.0	0.0	0.0	0.0	Y
8	1/1/2015 2:00	Brooklyn	1229.0	3.0	10.0	NaN	6.0	1023.0	0.0	0.0	0.0	0.0	Y
13	1/1/2015 2:00	NaN	11.0	3.0	10.0	30.0	6.0	1023.0	0.0	0.0	0.0	0.0	Y
15	1/1/2015 3:00	Brooklyn	1601.0	5.0	10.0	NaN	8.0	1022.3	0.0	0.0	0.0	0.0	Y
20	1/1/2015 3:00	NaN	1.0	5.0	10.0	30.0	8.0	1022.3	0.0	0.0	0.0	0.0	Y
21	1/1/2015 4:00	Bronx	NaN	5.0	10.0	29.0	9.0	1022.0	0.0	0.0	0.0	0.0	Y
22	1/1/2015 4:00	Brooklyn	1390.0	5.0	10.0	NaN	9.0	1022.0	0.0	0.0	0.0	0.0	Y
27	1/1/2015 4:00	NaN	2.0	5.0	10.0	29.0	9.0	1022.0	0.0	0.0	0.0	0.0	Y
29	1/1/2015 5:00	Brooklyn	759.0	5.0	10.0	NaN	9.0	1021.8	0.0	0.0	0.0	0.0	Y

After Removing all NULL Values showing in percentage

In [33]:

```
pickup_dt      0.0
borough        0.0
pickups        0.0
spd            0.0
vsb            0.0
temp           0.0
dewp           0.0
slp            0.0
pcp01          0.0
pcp06          0.0
pcp24          0.0
sd             0.0
hday           0.0
weekday        0.0
is_weekend     0.0
dtype: float64
```

Solving pickup_dt issues

In [37]:

```
# Forward fill
df['pickup_dt'] = df['pickup_dt'].fillna(method='ffill')
```

```
/var/folders/cz/49clhtds01v79zh558p1fb3m0000gn/T/ipykernel_6205/2513
405293.py:2: FutureWarning: Series.fillna with 'method' is deprecate
d and will raise in a future version. Use obj.ffill() or obj.bfill()
instead.
  df['pickup_dt'] = df['pickup_dt'].fillna(method='ffill')
```

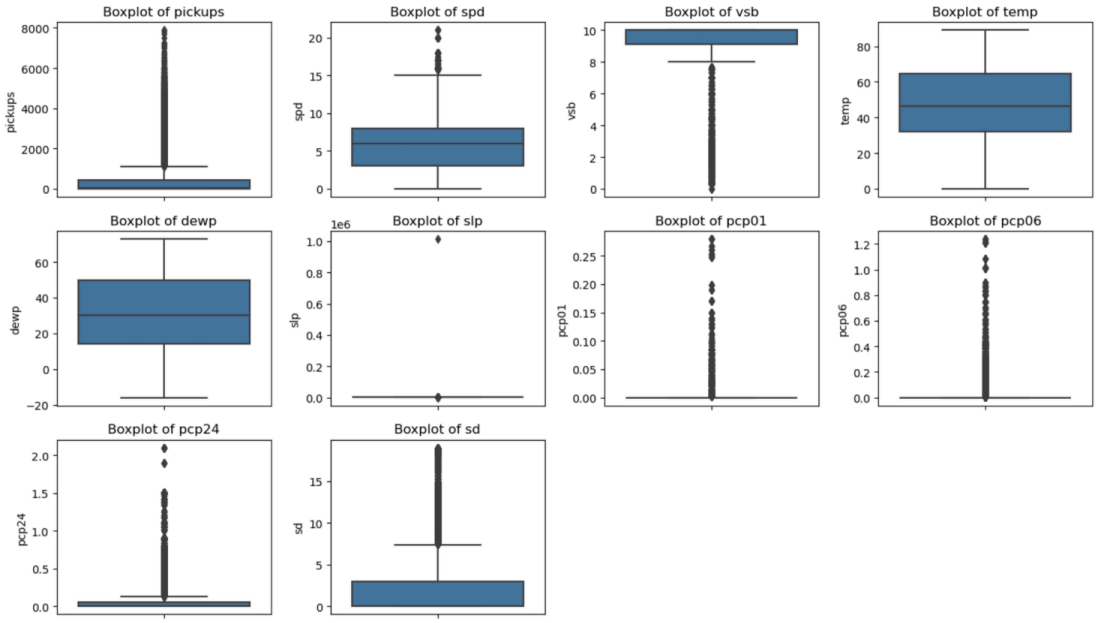
10- Do the necessary data cleaning steps like dropping duplicates, unnecessary columns, null value imputation, outliers treatment etc.

A. DATA Cleaning Steps

In [38]:

B .Check for Outliers: Identify outliers in numerical columns.

In [87]:



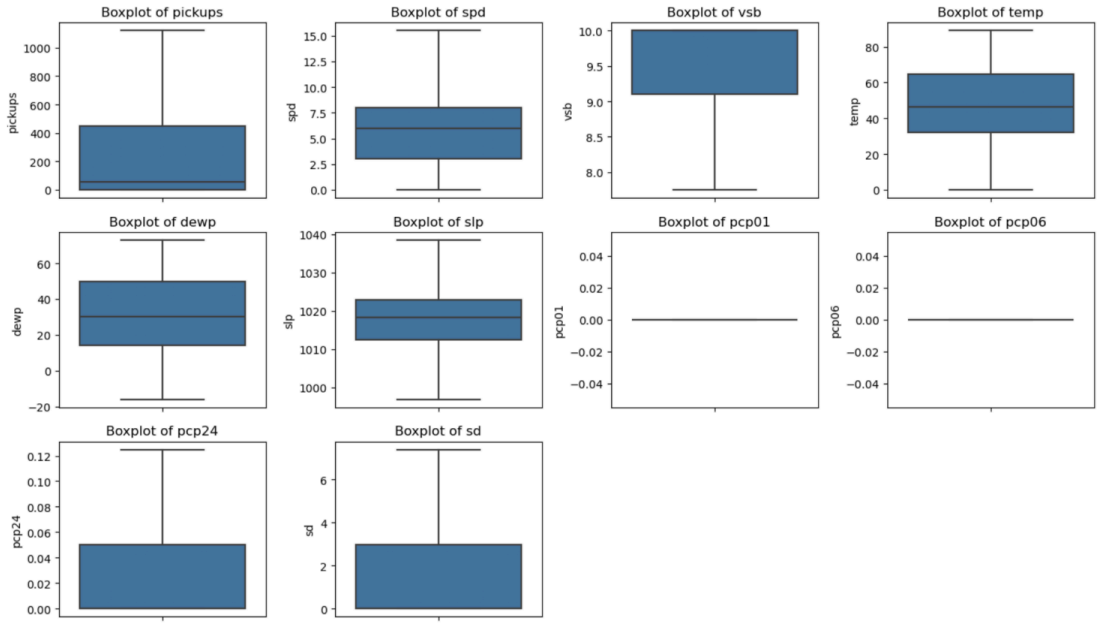
Treating Outliers

In [32]:
```
#Treating Outliers
def remove_outlier(col):
    sorted(col)
    Q1,Q3=col.quantile([0.25,0.75])
    IQR=Q3-Q1
    lower_range= Q1-(1.5 * IQR)
    upper_range= Q3+(1.5 * IQR)
    return lower_range, upper_range

for i in df.columns:
    if df[i].dtype !='object':
        lr,ur=remove_outlier(df[i])
        df[i]=np.where(df[i]>ur,ur,df[i])
        df[i]=np.where(df[i]<lr,lr,df[i])
```

After treating outliers

In [35]:



1. Pickup Analysis

1. What is the total number of Uber pickups across all boroughs?

Output

In [155]:

```
Total number of Uber pickups across all boroughs :  14265486.0
```

2. Which borough has the highest average number of hourly pickups?

Output

In [152]:

```
Highest average is:- Manhattan
```

3. How do the number of pickups vary across different hours of the day?

Output

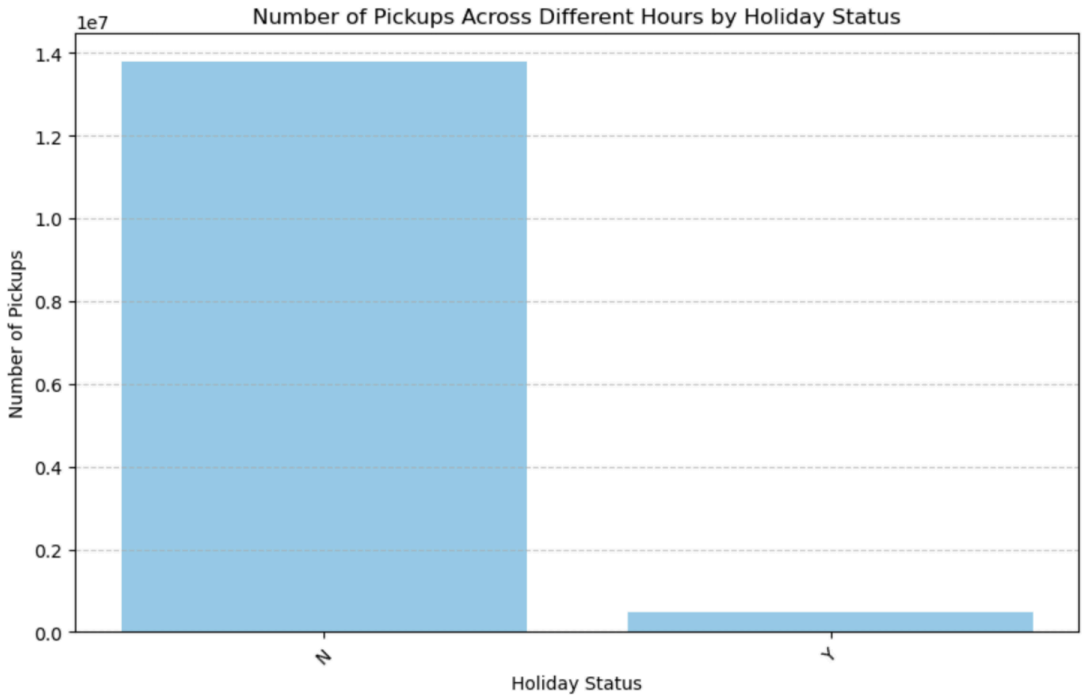
Numerical representation

In [150]:

```
Number of pickups across different hours of the day:
hday
N    13777429.0
Y      488057.0
Name: pickups, dtype: float64
```

Graphical Represenatation

In [148]:



4. Which day of the week has the highest number of pickups?

Output

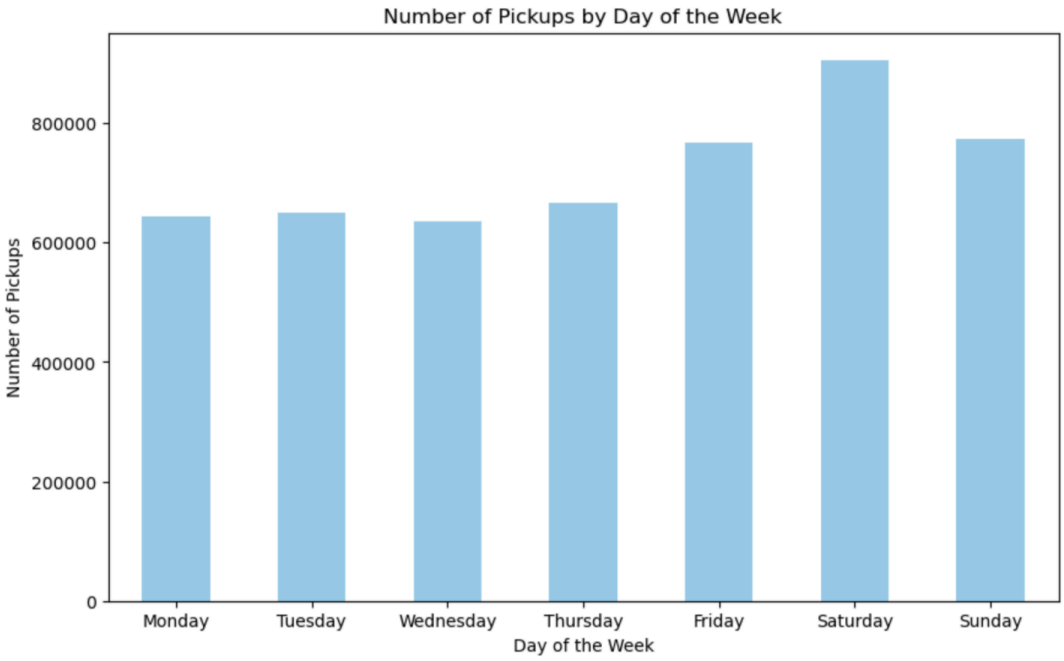
Numerical data represenation

In [145]:

```
season
Fall       700499.0
Spring    7507697.0
Summer    1700758.0
Winter    4356532.0
Name: pickups, dtype: float64
```

Graphical Representation

In [48]:



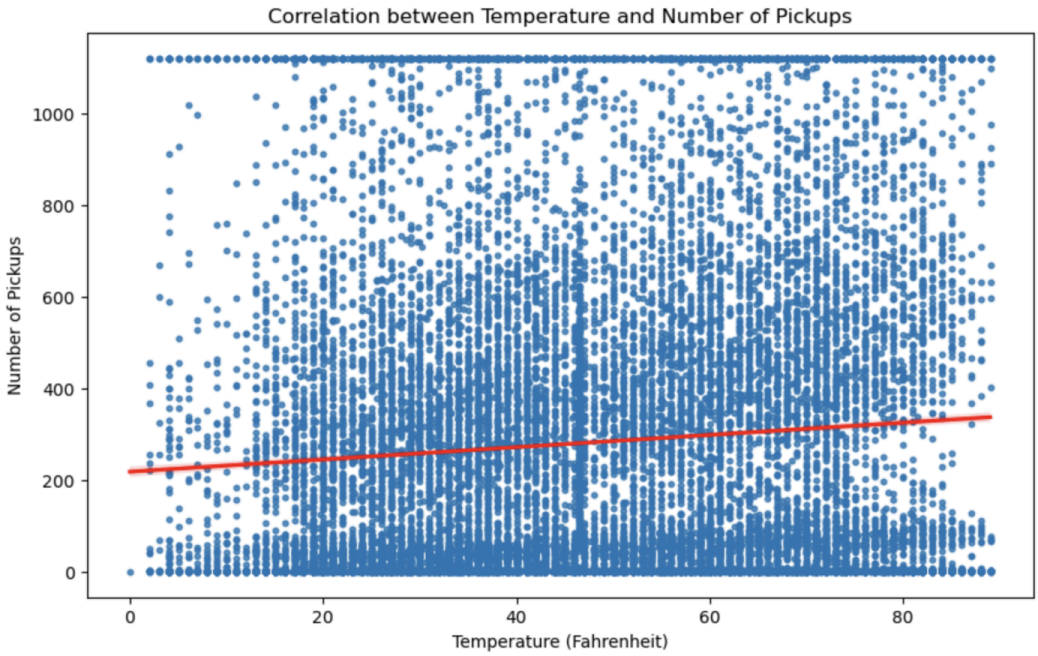
2. Weather Impact

1. What is the correlation between temperature and the number of pickups?

Output

In [50]:

```
Correlation between temperature and number of pickups: 0.06813480303893267
```



2. How does visibility impact the number of pickups?

Output

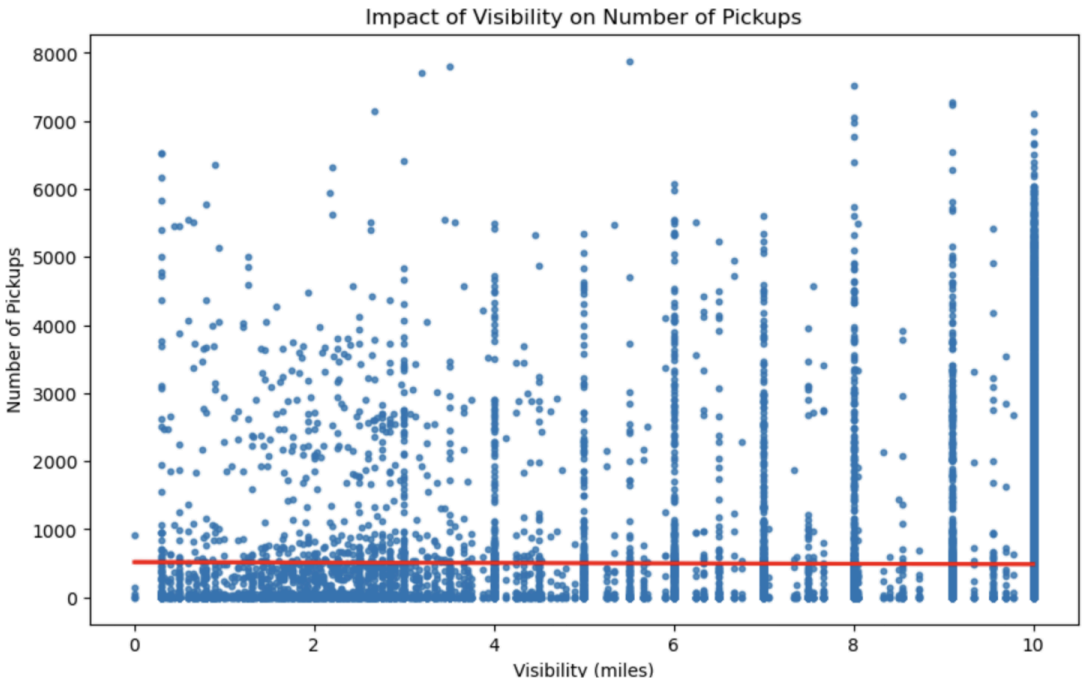
Numerical representation

In [143]:

```
borough
Bronx            0.200889
Brooklyn         0.291973
EWR              0.045346
Manhattan        0.159015
Queens           0.401791
Staten Island    0.243382
dtype: float64
```

Graphical representation

In [141]:



3. Is there a relationship between wind speed and the number of pickups?

Output

In [139]:

```
Correlation between wind speed and number of pickups: 0.011075499985935629
```

4. How does precipitation (1-hour, 6-hour, 24-hour) affect the number of pickups?

Output

In [140]:

```
Correlation between 1-hour precipitation and number of pickups: 0.004398191771173067
```

3. Seasonal Trends

1. How do the number of pickups vary across different seasons (winter, spring, summer, fall)?

Output

In [137]:

```
season
Fall       700499.0
Spring    7507697.0
Summer    1700758.0
Winter    4356532.0
Name: pickups, dtype: float64
```

2. What is the average number of pickups during holidays compared to non-holidays?

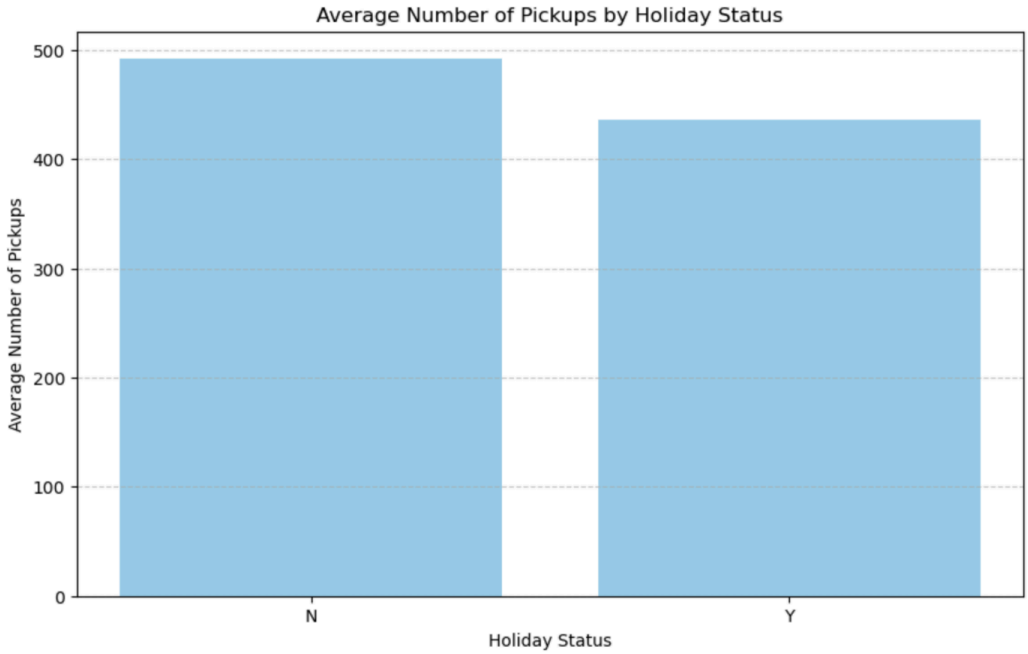
Numerical representation

In [135]:

```
hday
N    492.402752
Y    436.544723
Name: pickups, dtype: float64
```

Graphical representation

In [132]:



4. Hourly Trends

1. What are the peak hours for Uber pickups in each borough?

In [3]:

```
('Manhattan', 23.0)
```

2. How do the number of pickups change during rush hours (e.g., 7-9 AM, 5-7 PM)?

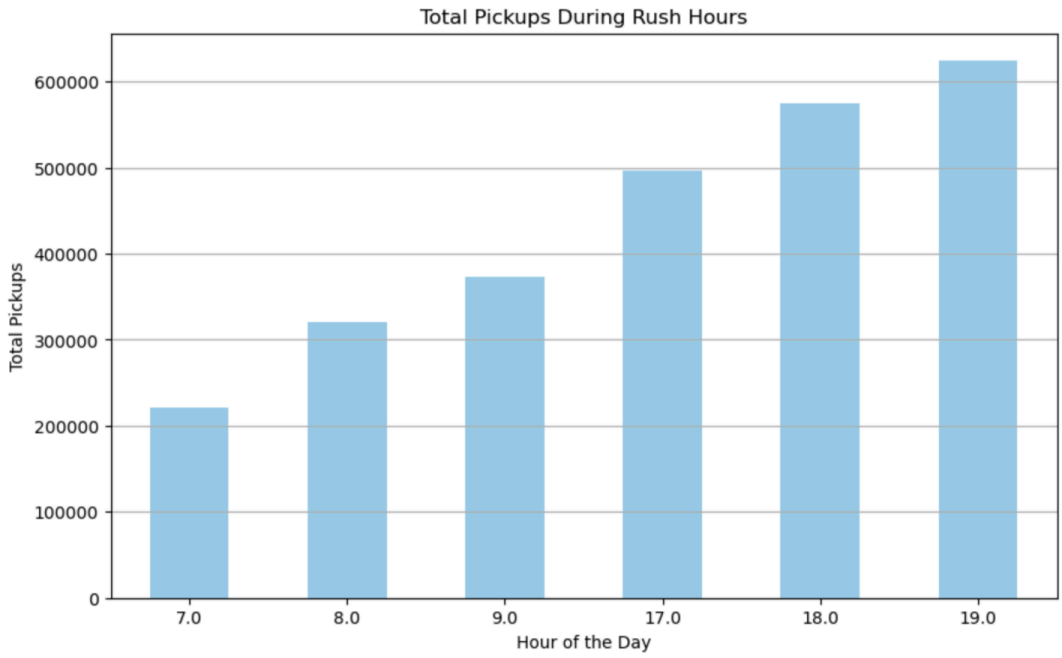
Output

In [124]:

```
pickup_dt
7.0     220406.0
8.0     320601.0
9.0     372725.0
17.0    496190.0
18.0    575359.0
19.0    624742.0
```

Graphical Representation

In [126]:



3. What is the average number of pickups during late-night hours (e.g., 12 AM - 4 AM)?

Output

In [123]:

```
337.9056603773585
```

5. Borough Comparison

1. How do pickup trends differ between boroughs during different weather conditions?

Output

Numerical represenatation

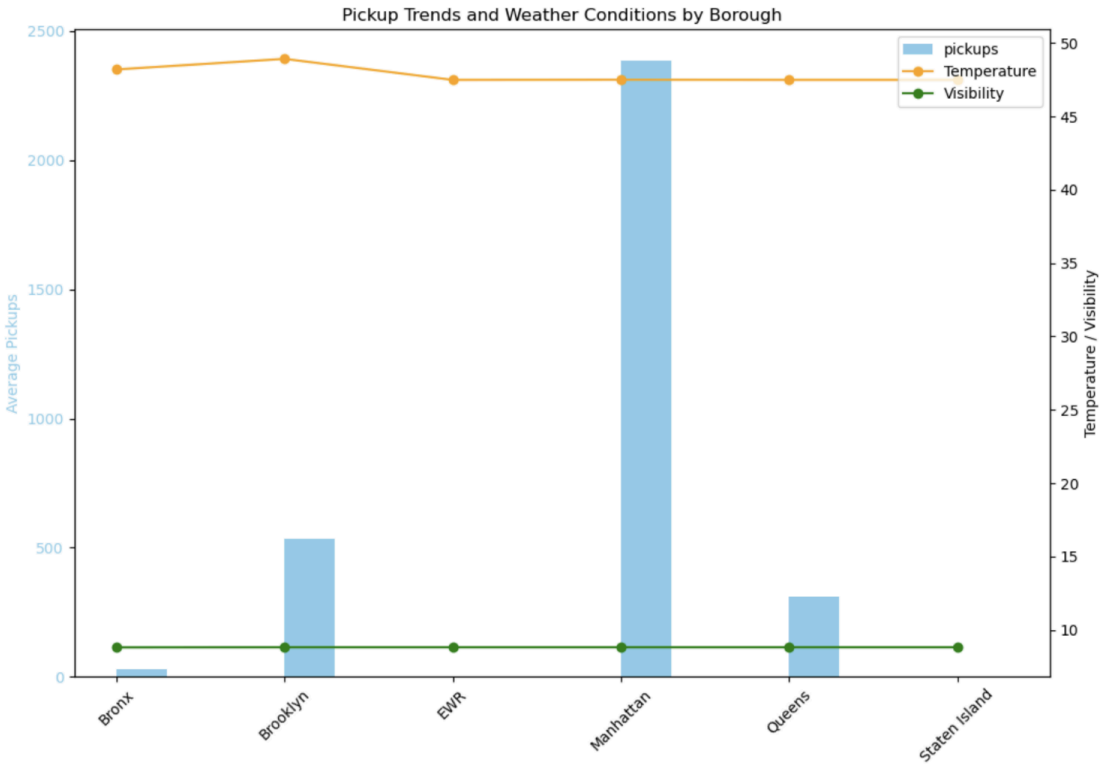
In [161]:

borough	temp	vsb	pickups
Bronx	48.198356	8.812534	30.629976
Brooklyn	48.920975	8.820027	534.431269
EWR	47.488421	8.819902	0.024194
Manhattan	47.498215	8.820027	2387.253281
Queens	47.489005	8.820027	309.305779
Staten Island	47.489005	8.820027	1.601888

Graphical representation

Output

In [120]:



2. Which borough shows the highest increase in pickups during holidays?

Output

In [118]:

```
Queens
```

3. How does the number of pickups compare between weekdays and weekends for each borough?

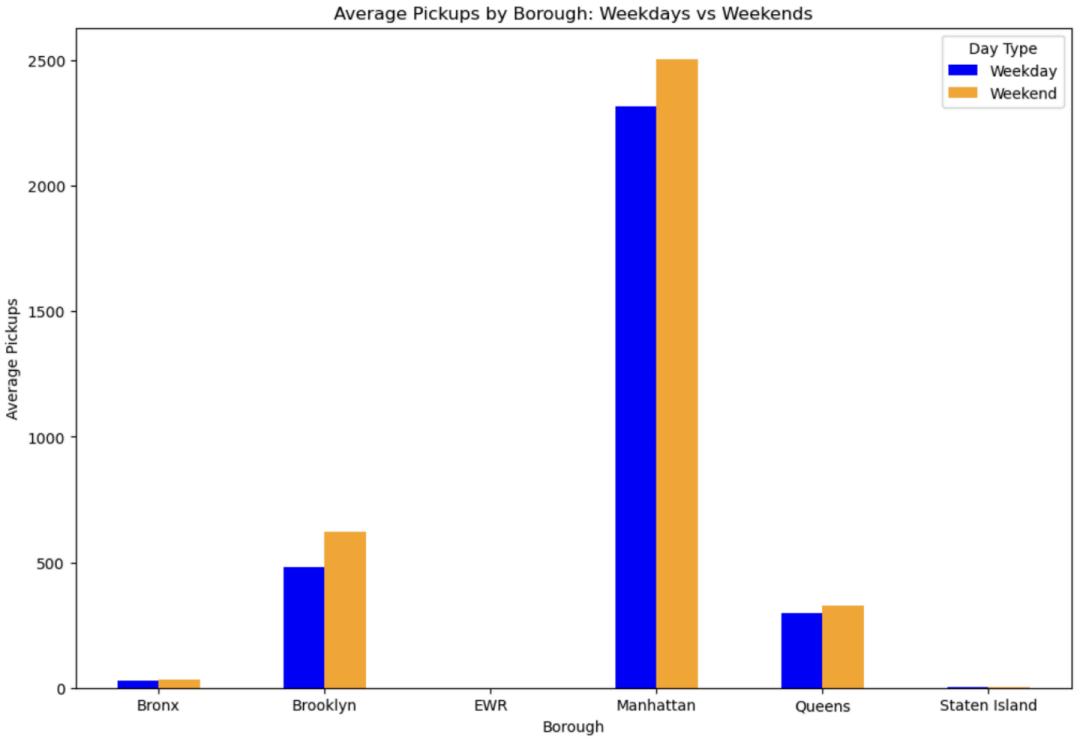
Output

In [115]:

```
borough        is_weekend
Bronx          0              28.649704
               1              33.823488
Brooklyn       0             481.049797
               1             623.106005
EWR            0               0.024363
               1               0.023912
Manhattan      0            2316.959425
               1            2504.022059
Queens         0             298.128366
               1             327.873162
Staten Island  0               1.512726
               1               1.750000
Name: pickups, dtype: float64
```

visual representation using bar graph

In [113]:

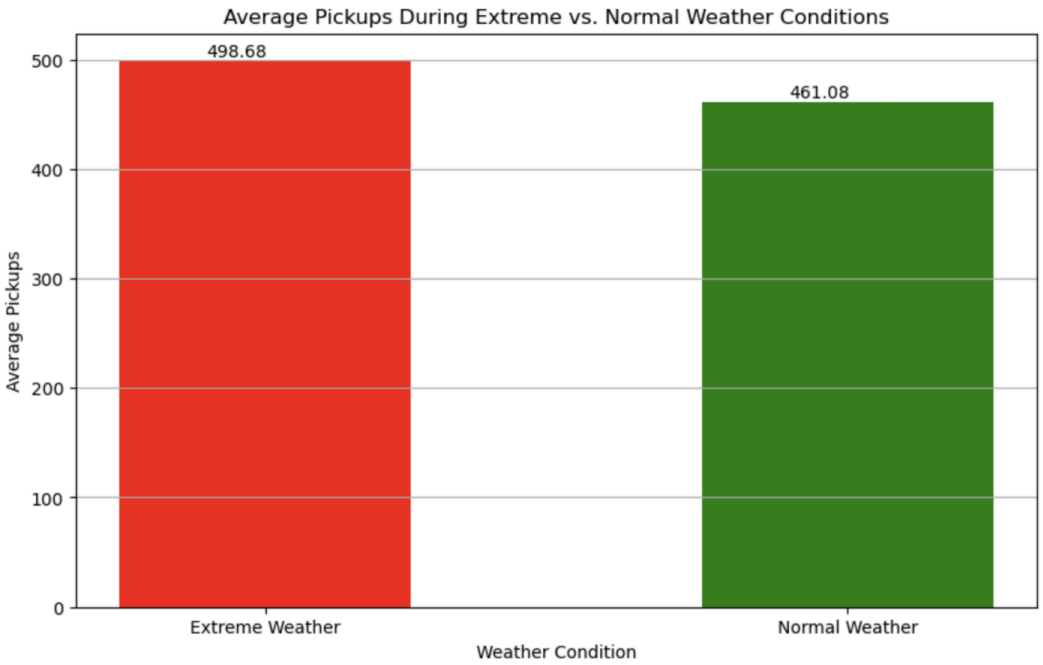


6. Weather Extremes

1. How do extreme weather conditions (e.g., very high or very low temperatures, heavy rainfall, snowstorms) affect the number of pickups?

Output

In [112]:



2. What is the impact of visibility less than 1 mile on the number of pickups?

Output

In [110]:

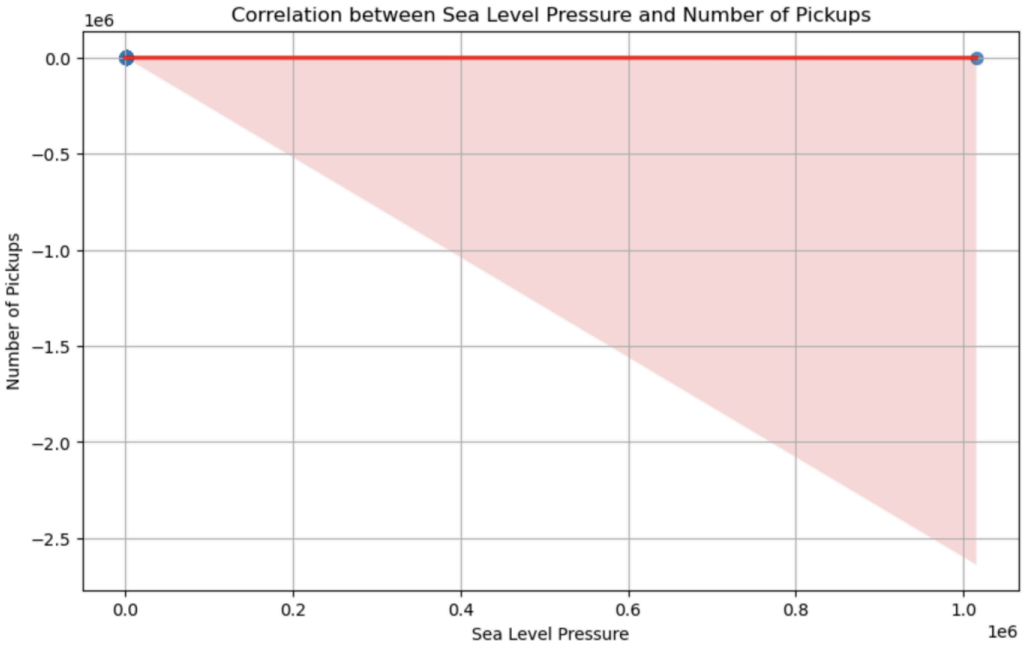
```
567.004555808656
```

7. Data Correlations

1. Is there a correlation between sea level pressure and the number of pickups?

Output

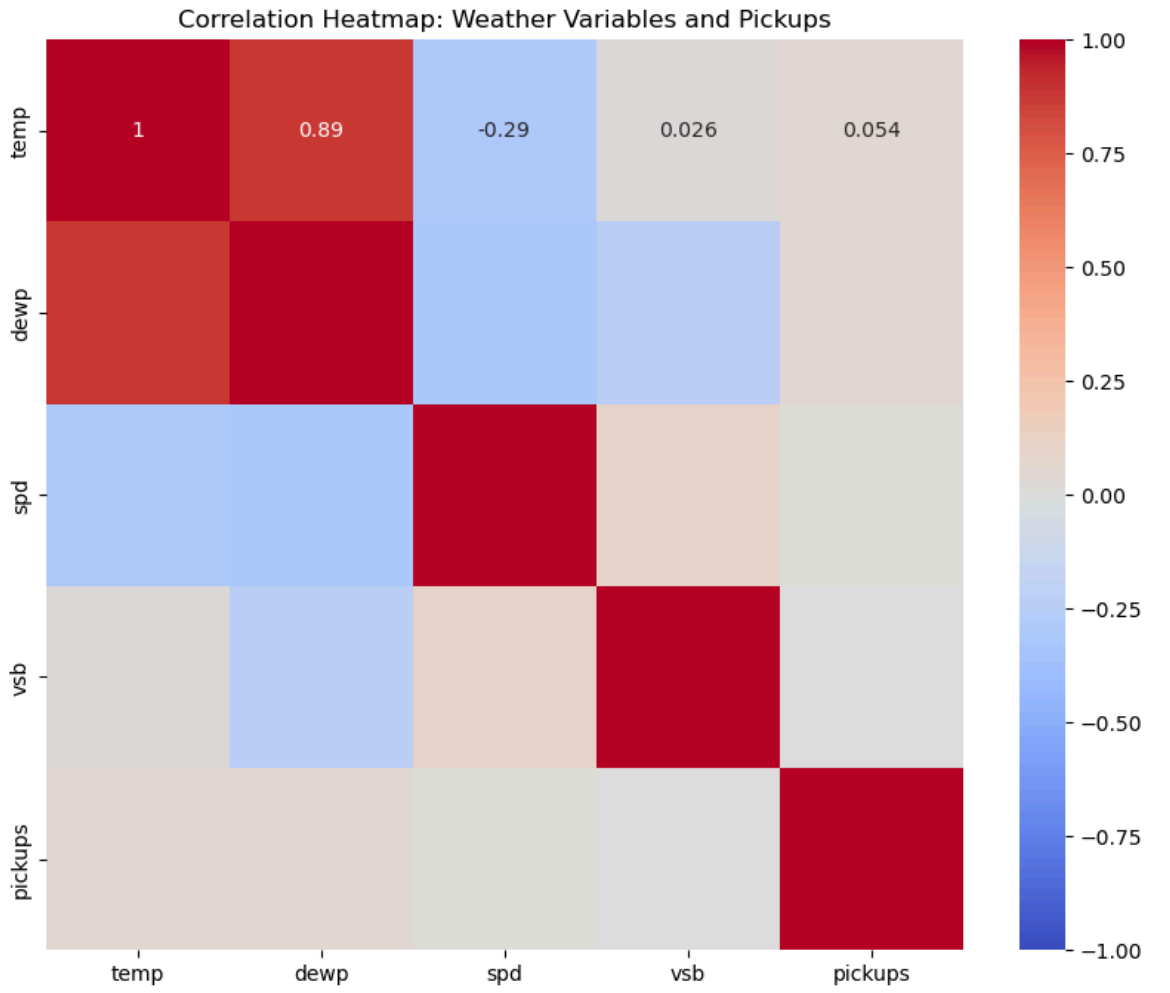
In [106]:



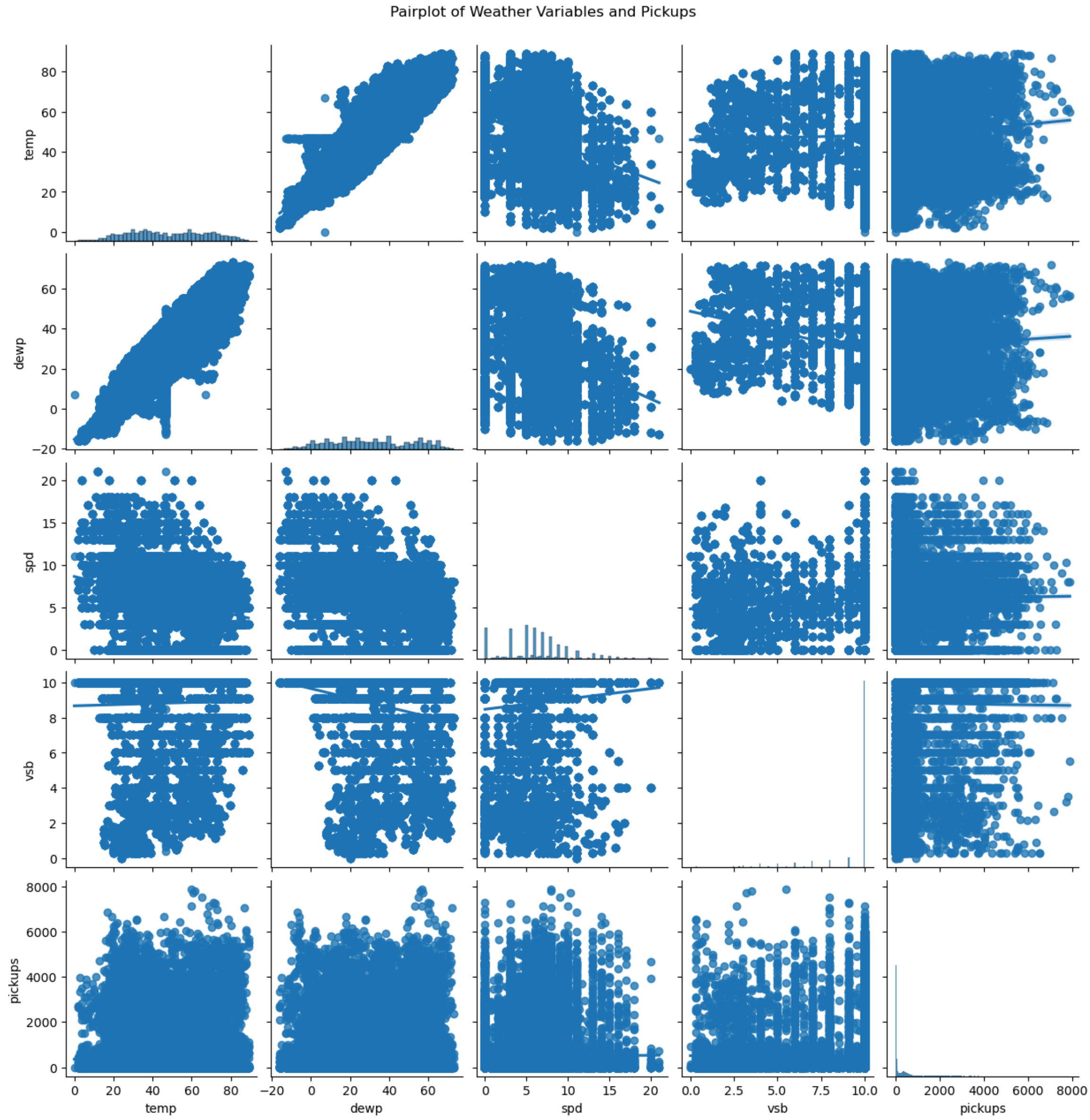
2. How do different weather variables (temperature, dew point, wind speed, visibility) collectively impact the number of pickups?

Output

In [104]:



```
/opt/anaconda3/lib/python3.11/site-packages/seaborn/_oldcore.py:111
9: FutureWarning: use_inf_as_na option is deprecated and will be rem
oved in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
/opt/anaconda3/lib/python3.11/site-packages/seaborn/_oldcore.py:111
9: FutureWarning: use_inf_as_na option is deprecated and will be rem
oved in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
/opt/anaconda3/lib/python3.11/site-packages/seaborn/_oldcore.py:111
9: FutureWarning: use_inf_as_na option is deprecated and will be rem
oved in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
/opt/anaconda3/lib/python3.11/site-packages/seaborn/_oldcore.py:111
9: FutureWarning: use_inf_as_na option is deprecated and will be rem
oved in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
/opt/anaconda3/lib/python3.11/site-packages/seaborn/_oldcore.py:111
9: FutureWarning: use_inf_as_na option is deprecated and will be rem
oved in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

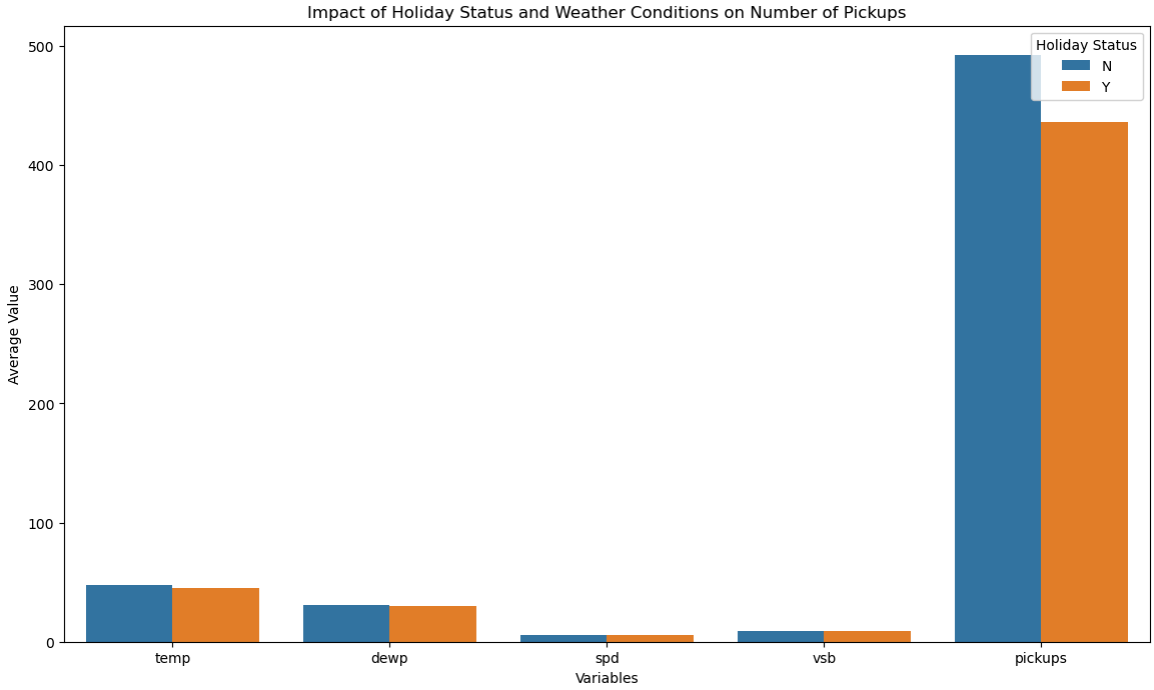


3. What is the relationship between holiday status and weather conditions on the number of pickups?

Output

In [98]:

	hday	temp	dewp	spd	vsb	pickups
0	N	47.987781	30.855647	5.990062	8.807282	492.402752
1	Y	45.288670	30.025060	5.859869	9.088994	436.544723

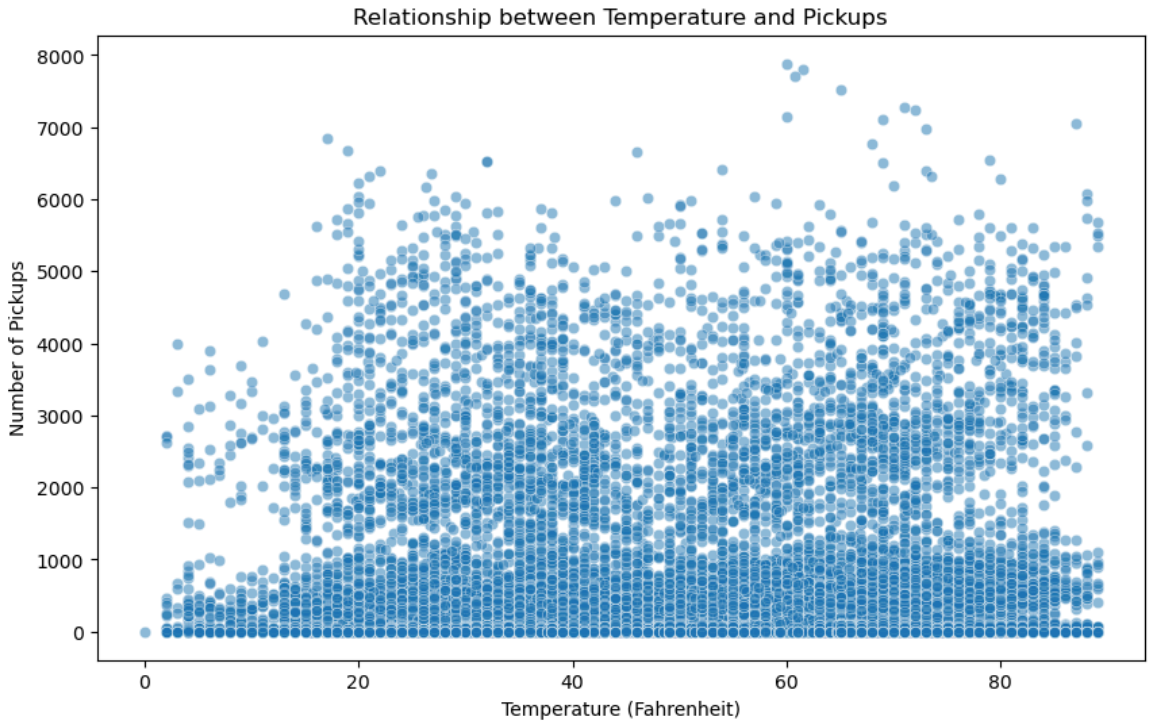


8. Growth Insights

1. Which weather conditions are most favorable for Uber pickups, and how can this information be used to optimize driver availability?

Output

In [99]:



2. Based on the data, what recommendations can be made to Uber to increase pickups during low-demand periods?

Output

In [100]:

```
/opt/anaconda3/lib/python3.11/site-packages/seaborn/categorical.py:641: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
  grouped_vals = vals.groupby(grouper)
```