```
import numpy as np
```

```
import pandas as pd
```

```
import seaborn as sns
```

```
import matplotlib.pyplot as plt
```

```
%matplotlib inline
```

```
from google.colab import drive        # Importing data from google drive.
drive.mount("/gdrive")                # mount is used when you have added external device as SSD, Hard

%cd /gdrive/My Drive/IMARTICUS/DV
```

```
     Mounted at /gdrive
     /gdrive/My Drive/IMARTICUS/DV
```

```
data = pd.read_csv("Uber_Data.csv")
```

```
df = data.copy()
```

```
df.head()
```

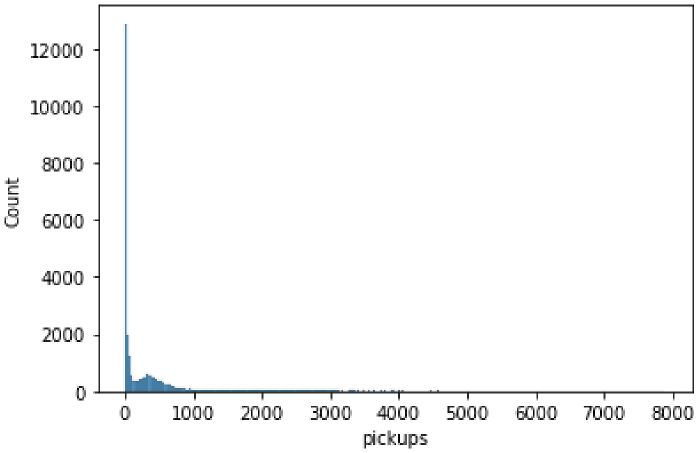
	pickup_dt	borough	pickups	spd	vsb	temp	dewp	slp	pcp01	pcp06	pcp24
0	01-01-2015 01:00	Bronx	152	5.0	10.0	30.0	7.0	1023.5	0.0	0.0	0.0
1	01-01-2015 01:00	Brooklyn	1519	5.0	10.0	NaN	7.0	1023.5	0.0	0.0	0.0

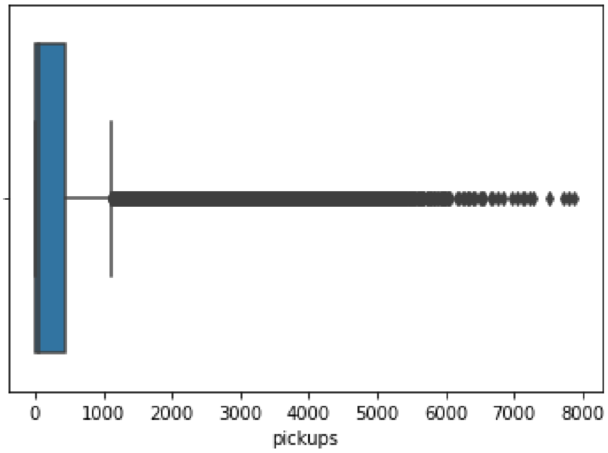
```
df.describe()
```

	pickups	spd	vsb	temp	dewp	s
count	29101.000000	29101.000000	29101.000000	28742.000000	29101.000000	29101.0000
mean	490.215903	5.984924	8.818125	47.900019	30.823065	1017.8179
std	995.649536	3.699007	2.442897	19.798783	21.283444	7.7687
min	0.000000	0.000000	0.000000	2.000000	-16.000000	991.4000
25%	1.000000	3.000000	9.100000	32.000000	14.000000	1012.5000
50%	54.000000	6.000000	10.000000	46.500000	30.000000	1018.2000
75%	449.000000	8.000000	10.000000	65.000000	50.000000	1022.9000
max	7883.000000	21.000000	10.000000	89.000000	73.000000	1043.4000

OBSERVATION ON PICKUPS

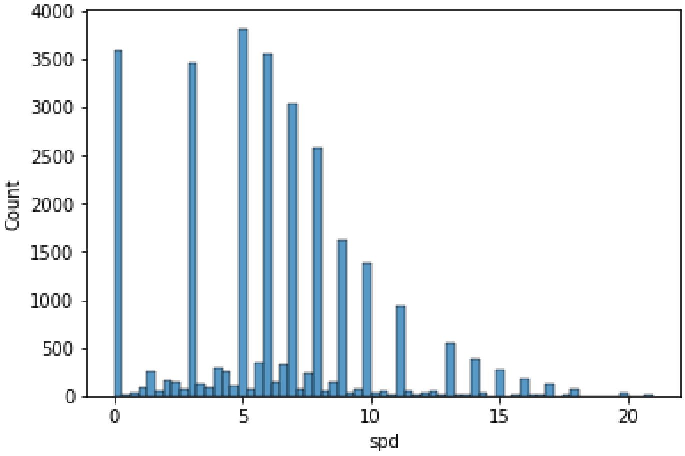
```
sns.histplot(data=df, x="pickups")
plt.show()
sns.boxplot(data=df, x="pickups")
plt.show()
```

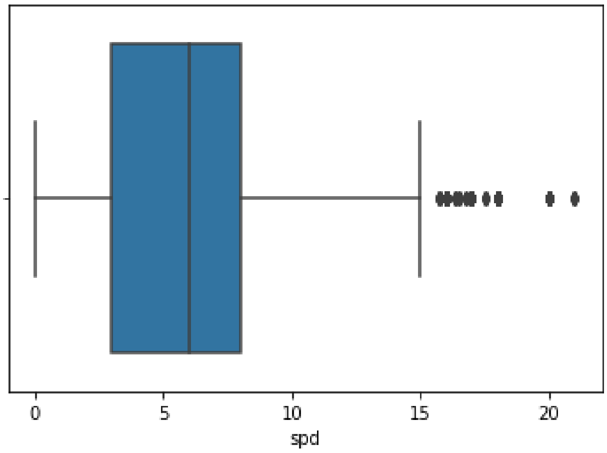




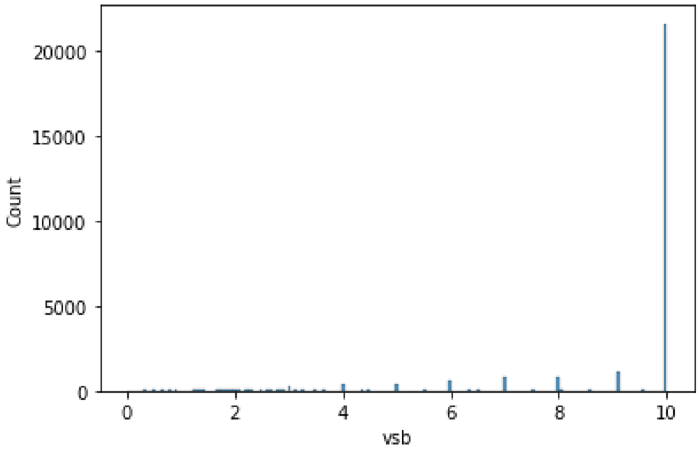
Double-click (or enter) to edit

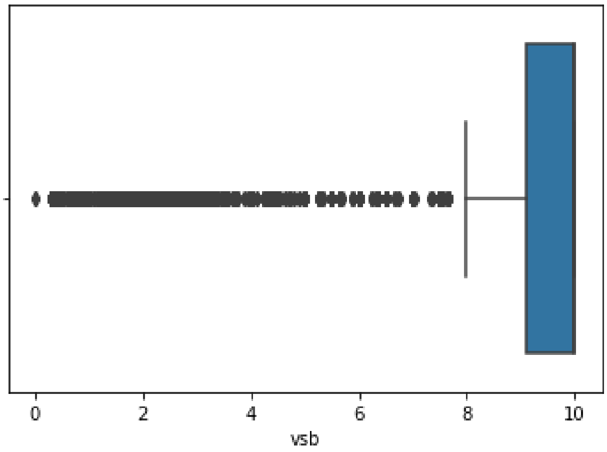
```
sns.histplot(data=df, x="spd")
plt.show()
sns.boxplot(data=df, x="spd")
plt.show()
```



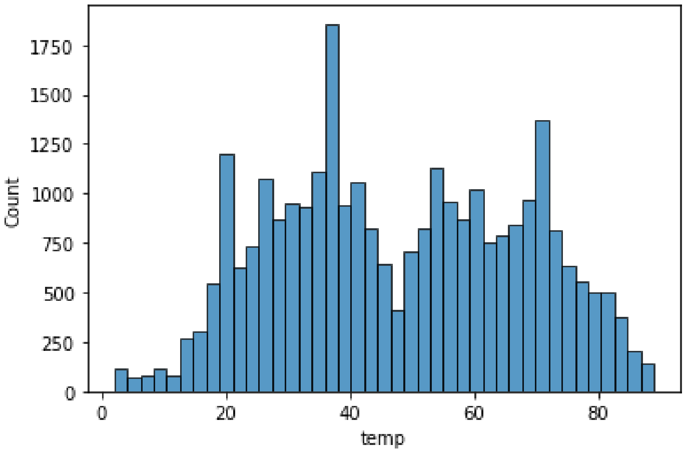


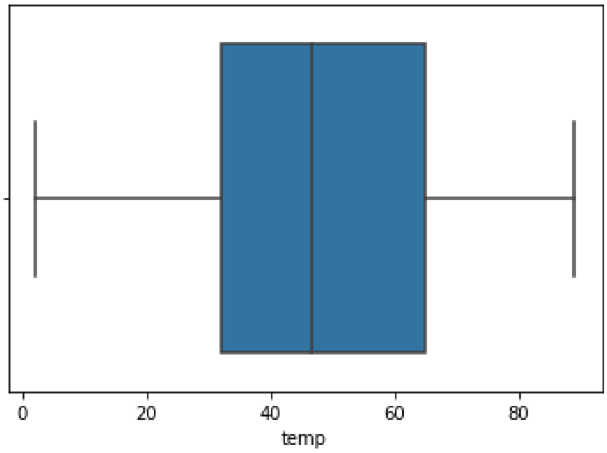
```
sns.histplot(data=df, x="vsb")
plt.show()
sns.boxplot(data=df, x="vsb")
plt.show()
```



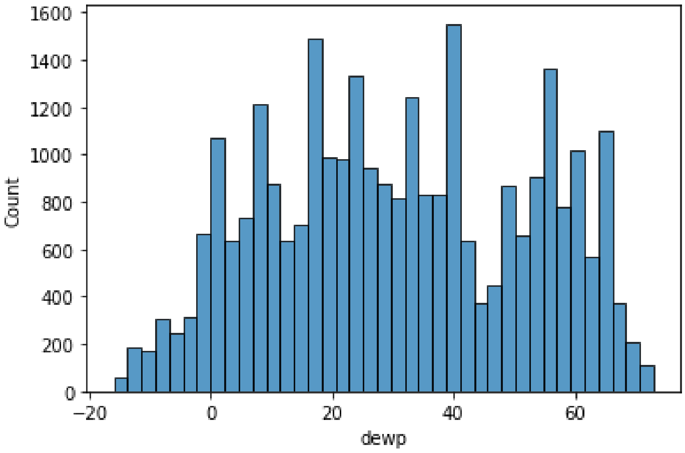


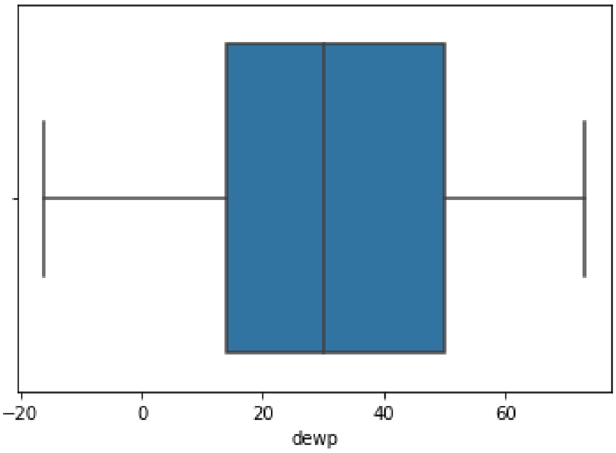
```
sns.histplot(data=df, x="temp")
plt.show()
sns.boxplot(data=df, x="temp")
plt.show()
```



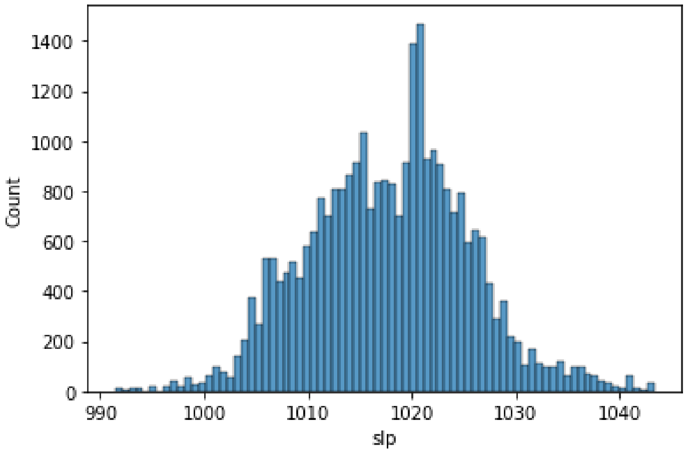


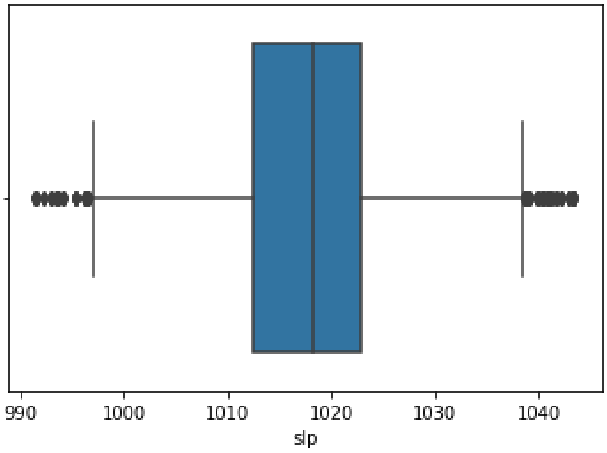
```
sns.histplot(data=df, x="dewp")
plt.show()
sns.boxplot(data=df, x="dewp")
plt.show()
```



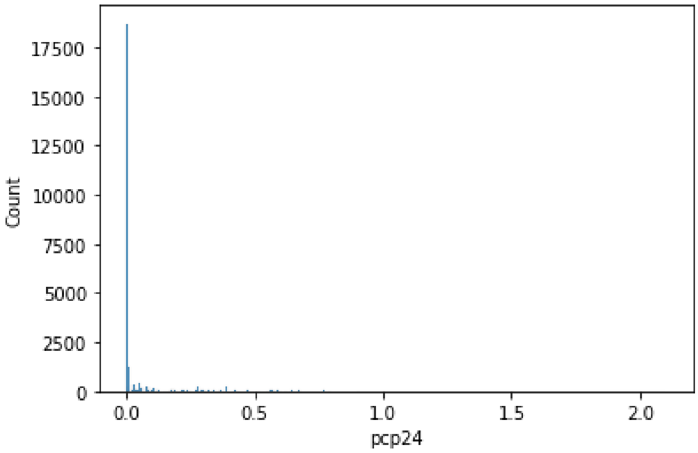


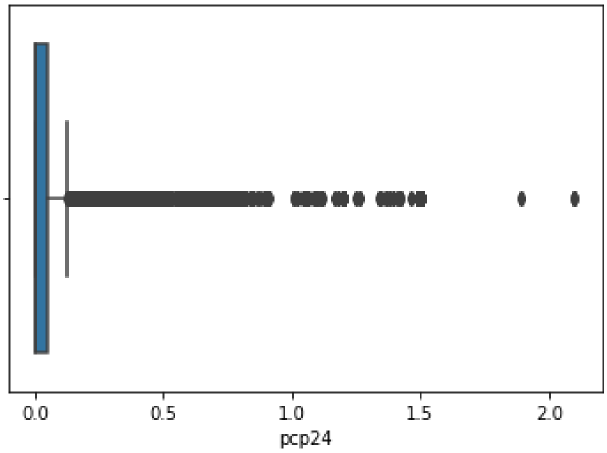
```
sns.histplot(data=df, x="slp")
plt.show()
sns.boxplot(data=df, x="slp")
plt.show()
```



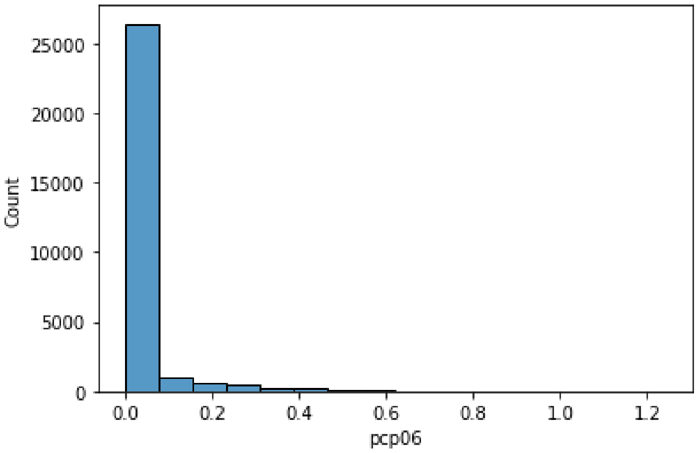


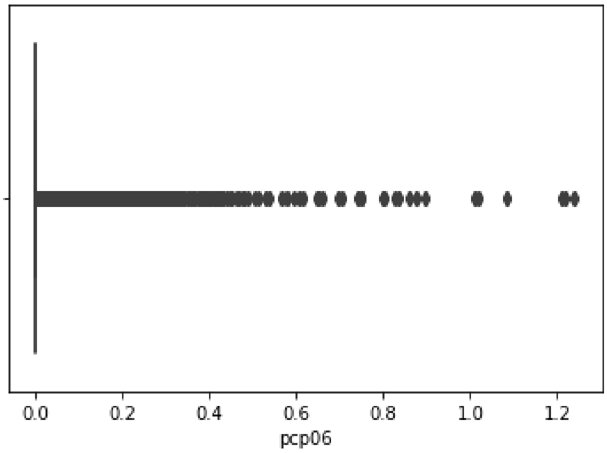
```
sns.histplot(data=df, x="pcp24")
plt.show()
sns.boxplot(data=df, x="pcp24")
plt.show()
```



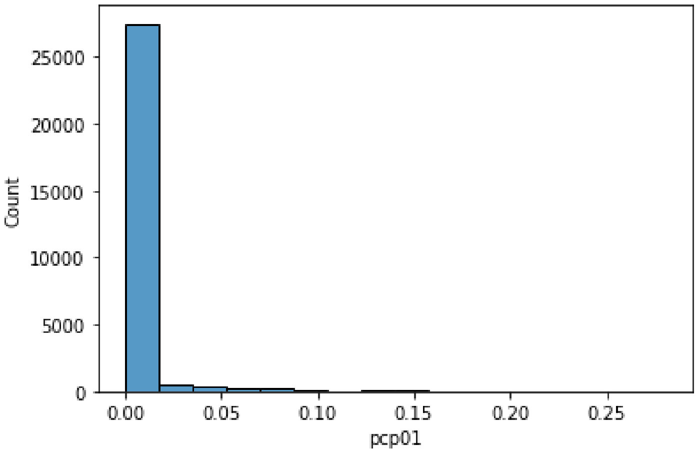


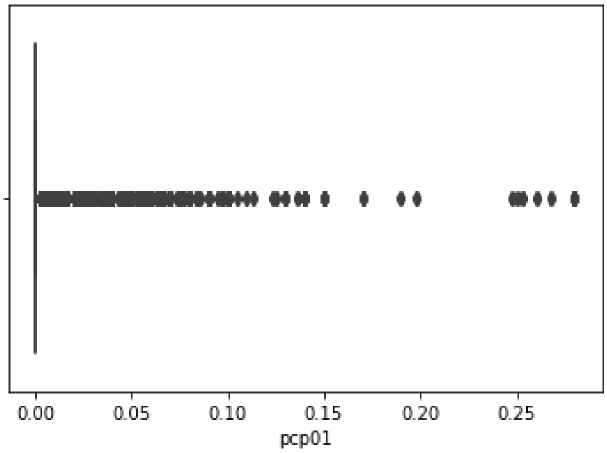
```
sns.histplot(data=df, x="pcp06")
plt.show()
sns.boxplot(data=df, x="pcp06")
plt.show()
```



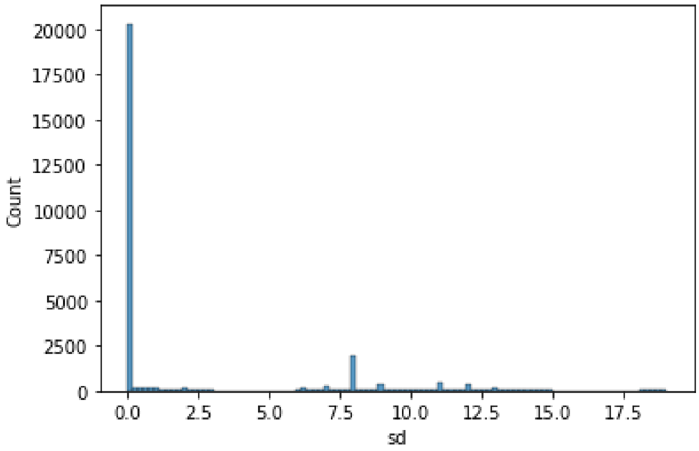


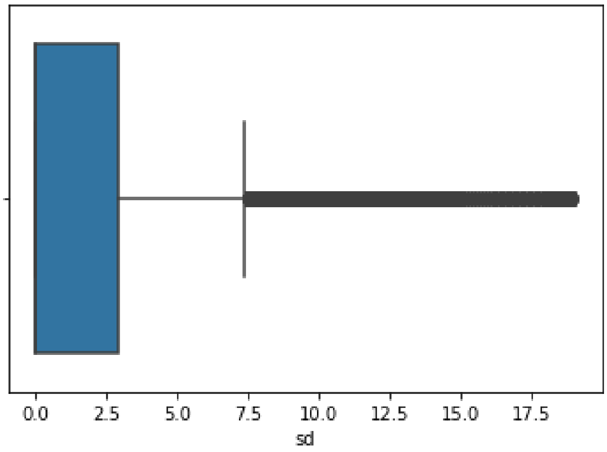
```
sns.histplot(data=df, x="pcp01")
plt.show()
sns.boxplot(data=df, x="pcp01")
plt.show()
```





```
sns.histplot(data=df, x="sd")
plt.show()
sns.boxplot(data=df, x="sd")
plt.show()
```

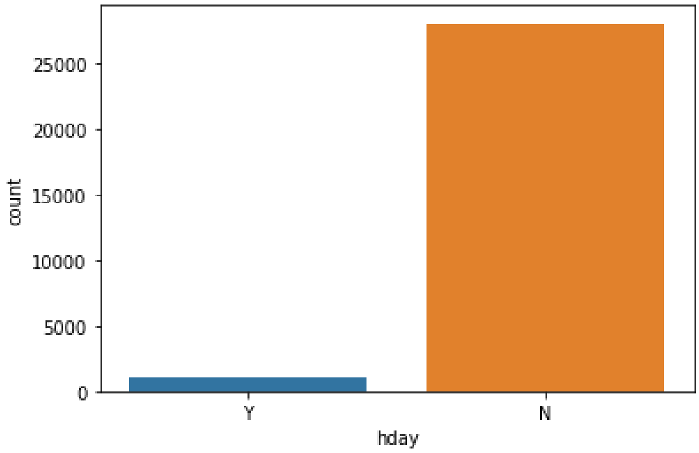




OBSERVATIONS ON HOLIDAY

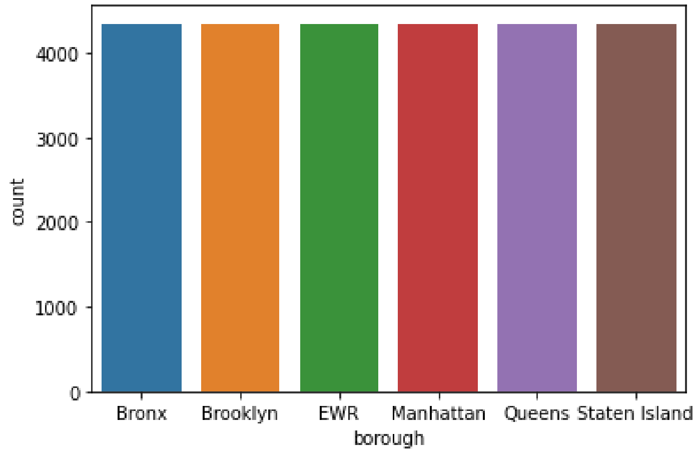
```
sns.countplot(data=df, x = "hday")
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe87fc193d0>
```

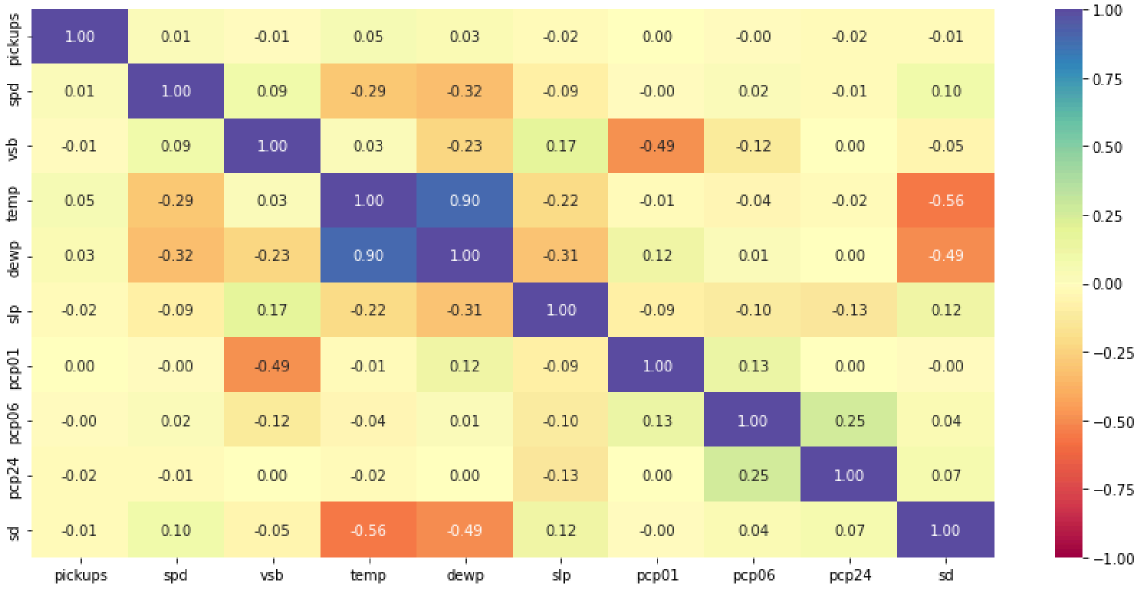


```
sns.countplot(data=df, x = "borough")
```

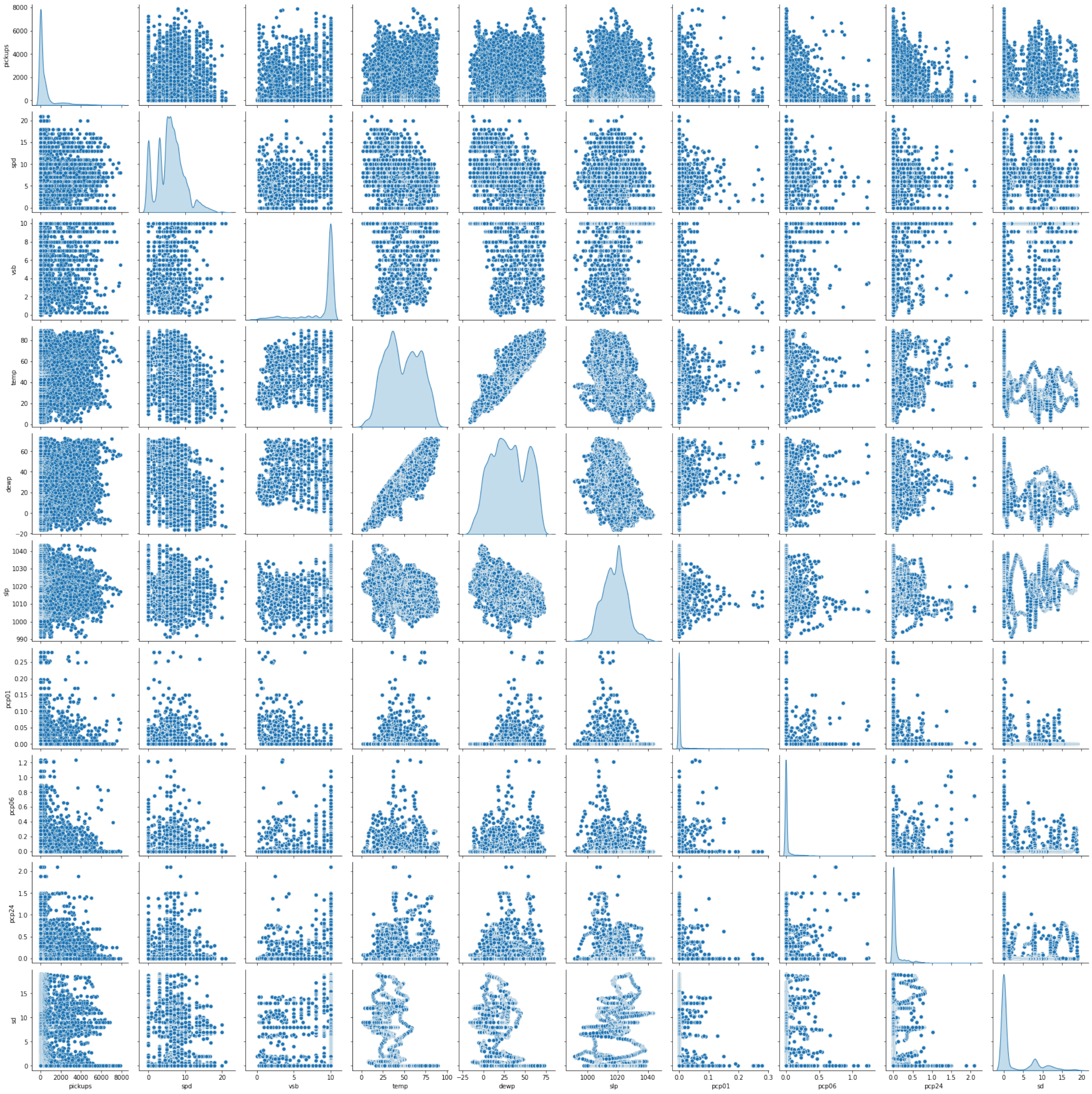
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe88062bfd0>
```



```
#Check for correlation among numerical variables
num_var = ["pickups","spd", "vsb", "temp", "dewp", "slp", "pcp01", "pcp06", "pcp24", "sd"]

corr = df[num_var].corr()

#plot the heatmap

plt.figure(figsize= (15,7))
sns.heatmap(corr, annot = True, vmin=-1, vmax=1, fmt=".2f", cmap = "Spectral")
plt.show()
```



```
sns.pairplot(data=df[num_var], diag_kind = "kde")
plt.show()
```



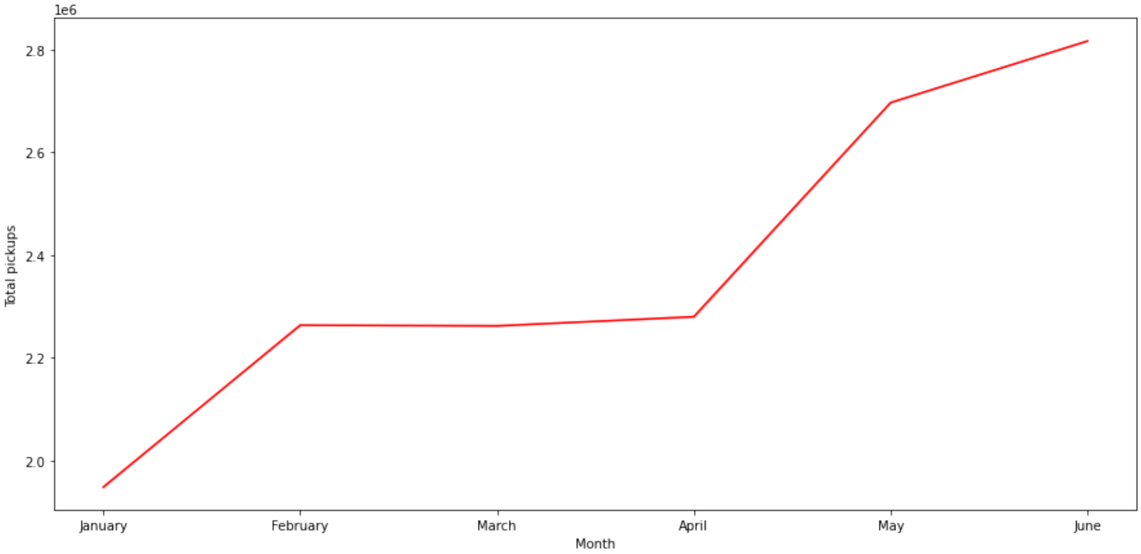
```
df["pickup_dt"] = pd.to_datetime(df["pickup_dt"], format = "%d-%m-%Y %H:%M")
```

```
df["start_year"] = df.pickup_dt.dt.year # EXTRACT YEAR FROM THE DATA
df["start_month"] = df.pickup_dt.dt.month_name()
df["start_hour"] = df.pickup_dt.dt.hour
df["start_day"] = df.pickup_dt.dt.day
df["week_day"] = df.pickup_dt.dt.day_name()
```

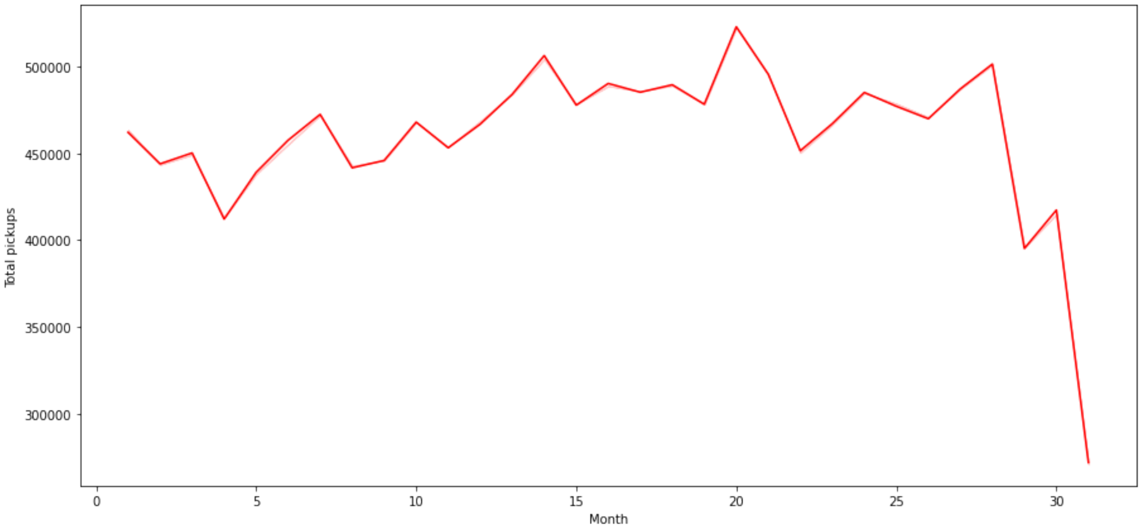
PICKUPS ACROSS MONTHS

```
cats = df.start_month.unique().tolist()
df.start_month = pd.Categorical(df.start_month, ordered=True , categories = cats)

plt.figure(figsize=(15,7))
```

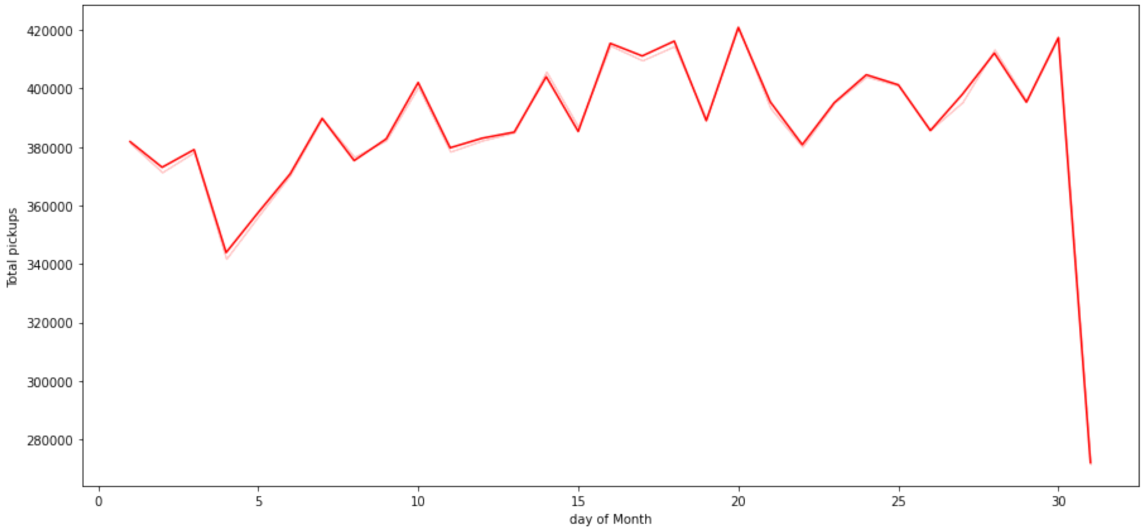
```
sns.lineplot(data=df , x= "start_month", y="pickups", ci = False, color= "red", estimator = "sum" )
plt.ylabel("Total pickups")
plt.xlabel("Month")
plt.show()
```



```
cats = df.start_month.unique().tolist()
df.start_month = pd.Categorical(df.start_month, ordered=True , categories = cats)

plt.figure(figsize=(15,7))
sns.lineplot(data=df , x= "start_day", y="pickups", ci = False, color= "red", estimator = "sum" )
plt.ylabel("Total pickups")
plt.xlabel("Month")
plt.show()
```



```
# Let us drop
df_not_feb = df[df["start_month"]!= "February"]

plt.figure(figsize=(15,7))
sns.lineplot(data=df_not_feb , x= "start_day", y="pickups", ci = False, color= "red", estimator = "sum
sns.lineplot(data=df_not_feb", x= "start_day", y= "pickups", ci = False, color= "red", estimator = "sum
plt.ylabel("Total pickups")
plt.xlabel("day of Month")
plt.show()
```



PICKUPS ACROSS HOURS OF THE DAY