

# Uber Ride Data Analysis

# **Key Question to Answer:**

1. In which category do people book the most Uber rides?
2. For which purpose do people book Uber rides the most?
3. At what time do people book cabs the most from Uber?
4. In which months do people book Uber rides the most?
5. On which days of the week do people book Uber rides the most?
6. How many miles do people usually book a cab for through Uber?



# Project Purpose:

The purpose of this project was to perform exploratory data analysis (EDA) on Uber ride data collected in 2016 to:

- Understand user behavior and ride trends over time
- Identify peak usage periods by time of day, day of week, and month
- Analyze the purpose behind trips (e.g., meetings, errands, customer visits)
- Clean and transform real-world, messy data for accurate analysis
- Practice feature engineering and visualization using Python, Pandas, and Seaborn

Ultimately, this project helped demonstrate how structured analysis and visualization can reveal meaningful insights from everyday data like ride logs.

# Uber Ride Data Analysis

## Importing Libraries

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

## Loading the Dataset

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

	START_DATE	END_DATE	CATEGORY	START	STOP	MILES	PURPOSE
0	01-01-2016 21:11	01-01-2016 21:17	Business	Fort Pierce	Fort Pierce	5.1	Meal/Entertain
1	01-02-2016 01:25	01-02-2016 01:37	Business	Fort Pierce	Fort Pierce	5.0	NaN
2	01-02-2016 20:25	01-02-2016 20:38	Business	Fort Pierce	Fort Pierce	4.8	Errand/Supplies
3	01-05-2016 17:31	01-05-2016 17:45	Business	Fort Pierce	Fort Pierce	4.7	Meeting
4	01-06-2016 14:42	01-06-2016 15:49	Business	Fort Pierce	West Palm Beach	63.7	Customer Visit



View Run Kernel Tabs Settings Help

\_Analysis.ipynb X +

      Code ▾

Notebook   Python 3 (ipykernel)  

## ***Data Inspection and Overview***

```
data.shape
```

```
(1156, 7)
```

```
data.columns
```

```
Index(['START_DATE', 'END_DATE', 'CATEGORY', 'START', 'STOP', 'MILES',  
      'PURPOSE'],  
      dtype='object')
```

```
data.duplicated().sum()
```

```
np.int64(1)
```

```
data.dtypes
```

```
START_DATE    object  
END_DATE      object  
CATEGORY      object  
START         object  
STOP         object  
MILES         float64  
PURPOSE       object  
dtype: object
```



View Run Kernel Tabs Settings Help

data\_Analysis.ipynb X +

Code

Notebook Python 3 (ipy

```
data["START_DATE"] = pd.to_datetime(data["START_DATE"], errors='coerce')
```

```
data["END_DATE"] = pd.to_datetime(data["END_DATE"], errors='coerce')
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1156 entries, 0 to 1155
Data columns (total 7 columns):
#   Column      Non-Null Count  Dtype
---  -
0   START_DATE  421 non-null    datetime64[ns]
1   END_DATE    420 non-null    datetime64[ns]
2   CATEGORY    1155 non-null    object
3   START       1155 non-null    object
4   STOP        1155 non-null    object
5   MILES       1156 non-null    float64
6   PURPOSE     1156 non-null    object
dtypes: datetime64[ns](2), float64(1), object(4)
memory usage: 63.3+ KB
```

```
from datetime import datetime
```

```
data['Date'] = pd.DatetimeIndex(data['START_DATE']).date
```

```
data['Time'] = pd.DatetimeIndex(data['START_DATE']).hour
```

ViewRunKernelTabsSettingsHelp

a\_Analysis.ipynb

×

+

Code

NotebookPython 3 (ipyk

data.head()

↑

↓

	START_DATE	END_DATE	CATEGORY	START	STOP	MILES	PURPOSE	Date	Time
0	2016-01-01 21:11:00	2016-01-01 21:17:00	Business	Fort Pierce	Fort Pierce	5.1	Meal/Entertain	2016-01-01	21.0
1	2016-01-02 01:25:00	2016-01-02 01:37:00	Business	Fort Pierce	Fort Pierce	5.0	NOT	2016-01-02	1.0
2	2016-01-02 20:25:00	2016-01-02 20:38:00	Business	Fort Pierce	Fort Pierce	4.8	Errand/Supplies	2016-01-02	20.0
3	2016-01-05 17:31:00	2016-01-05 17:45:00	Business	Fort Pierce	Fort Pierce	4.7	Meeting	2016-01-05	17.0
4	2016-01-06 14:42:00	2016-01-06 15:49:00	Business	Fort Pierce	West Palm Beach	63.7	Customer Visit	2016-01-06	14.0

data['Day-Night'] = pd.cut(x=data['Time'],bins = [0,10,15,19,24],labels = ['Morning','Afternoon','Evening','Night'])

data.head()

	START_DATE	END_DATE	CATEGORY	START	STOP	MILES	PURPOSE	Date	Time	Day-Night
0	2016-01-01 21:11:00	2016-01-01 21:17:00	Business	Fort Pierce	Fort Pierce	5.1	Meal/Entertain	2016-01-01	21.0	Night
1	2016-01-02 01:25:00	2016-01-02 01:37:00	Business	Fort Pierce	Fort Pierce	5.0	NOT	2016-01-02	1.0	Morning
2	2016-01-02 20:25:00	2016-01-02 20:38:00	Business	Fort Pierce	Fort Pierce	4.8	Errand/Supplies	2016-01-02	20.0	Night
3	2016-01-05 17:31:00	2016-01-05 17:45:00	Business	Fort Pierce	Fort Pierce	4.7	Meeting	2016-01-05	17.0	Evening



```
data.dropna(inplace=True)
```

```
data.shape
```

(413, 10)

```
data.describe()
```

	START_DATE	END_DATE	MILES	Time
count	413	413	413.000000	413.000000
mean	2016-07-01 16:00:42.711864320	2016-07-01 16:20:05.230024192	9.120823	15.222760
min	2016-01-01 21:11:00	2016-01-01 21:17:00	0.500000	1.000000
25%	2016-03-12 09:13:00	2016-03-12 09:22:00	2.700000	12.000000
50%	2016-07-04 18:23:00	2016-07-04 18:49:00	5.700000	16.000000
75%	2016-10-11 01:27:00	2016-10-11 02:08:00	9.900000	18.000000
max	2016-12-12 20:48:00	2016-12-12 20:57:00	174.200000	23.000000
std	NaN	NaN	16.517267	4.265322

# Data Visualization

```
: data.head()
```

	START_DATE	END_DATE	CATEGORY	START	STOP	MILES	PURPOSE	Date	Time	Day-Night
0	2016-01-01 21:11:00	2016-01-01 21:17:00	Business	Fort Pierce	Fort Pierce	5.1	Meal/Entertain	2016-01-01	21.0	Night
1	2016-01-02 01:25:00	2016-01-02 01:37:00	Business	Fort Pierce	Fort Pierce	5.0	NOT	2016-01-02	1.0	Morning
2	2016-01-02 20:25:00	2016-01-02 20:38:00	Business	Fort Pierce	Fort Pierce	4.8	Errand/Supplies	2016-01-02	20.0	Night
3	2016-01-05 17:31:00	2016-01-05 17:45:00	Business	Fort Pierce	Fort Pierce	4.7	Meeting	2016-01-05	17.0	Evening
4	2016-01-06 14:42:00	2016-01-06 15:49:00	Business	Fort Pierce	West Palm Beach	63.7	Customer Visit	2016-01-06	14.0	Afternoon

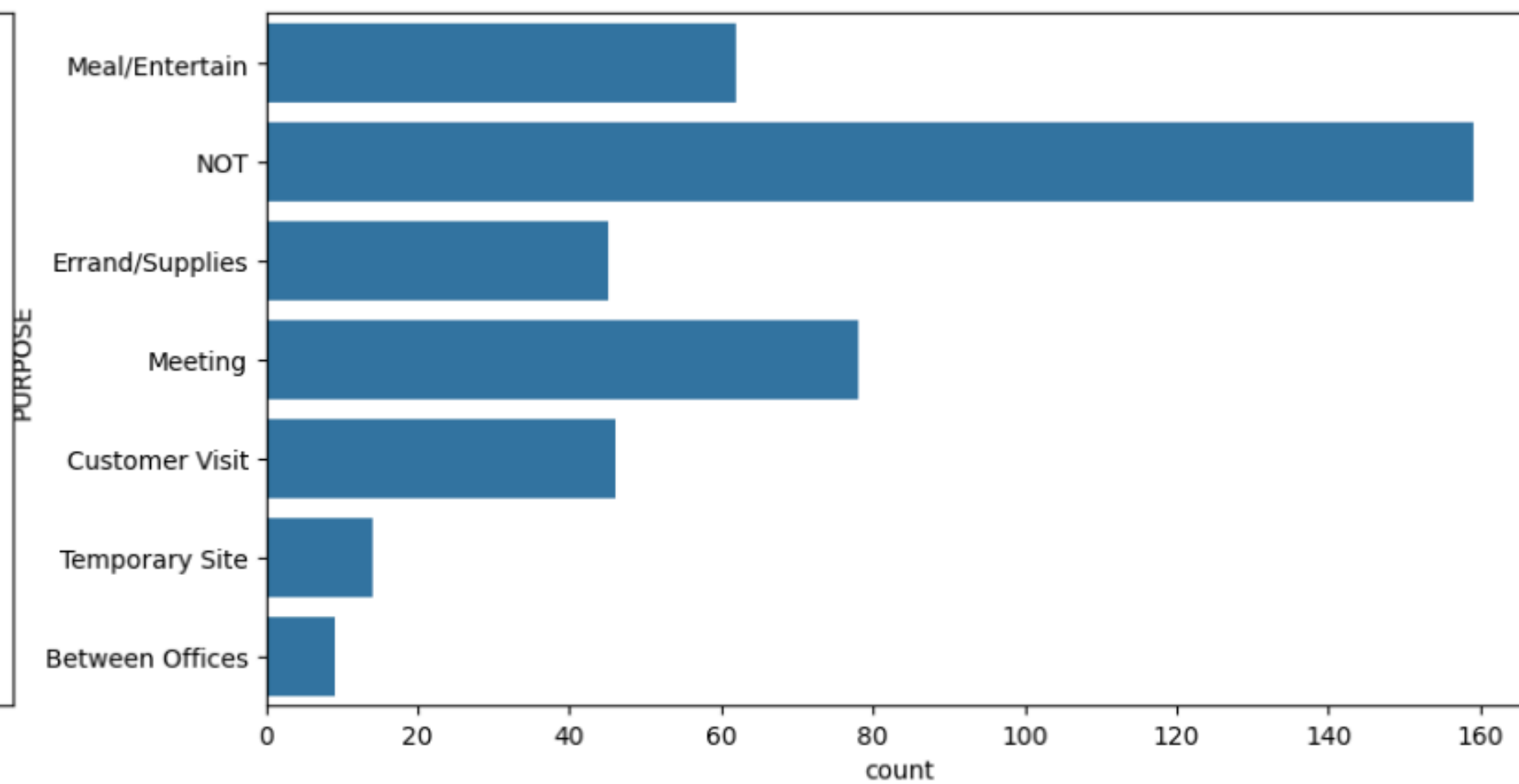
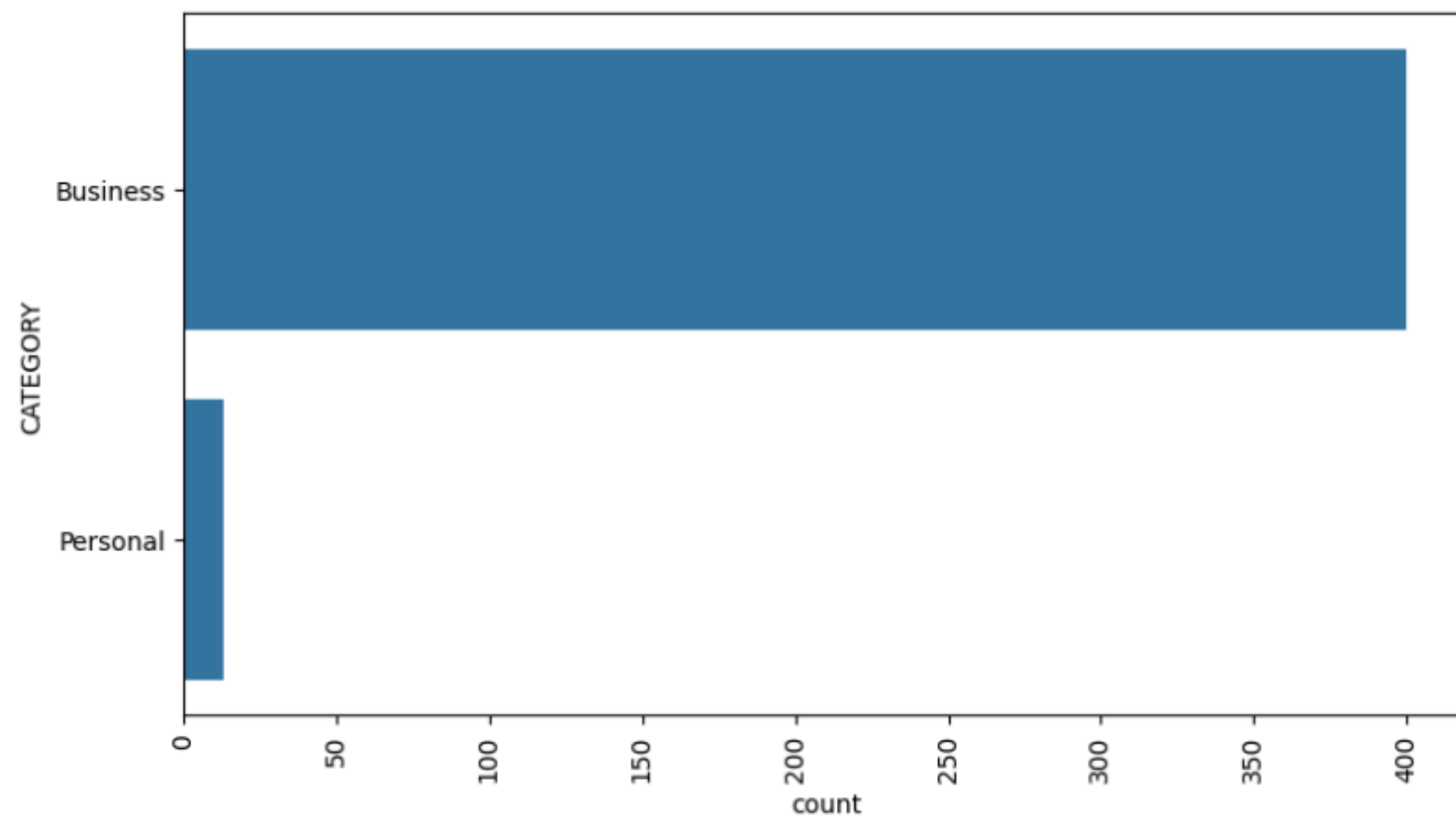
```
: plt.figure(figsize=(20,5))

plt.subplot(1,2,1)

sns.countplot(data['CATEGORY'])
plt.xticks(rotation = 90)

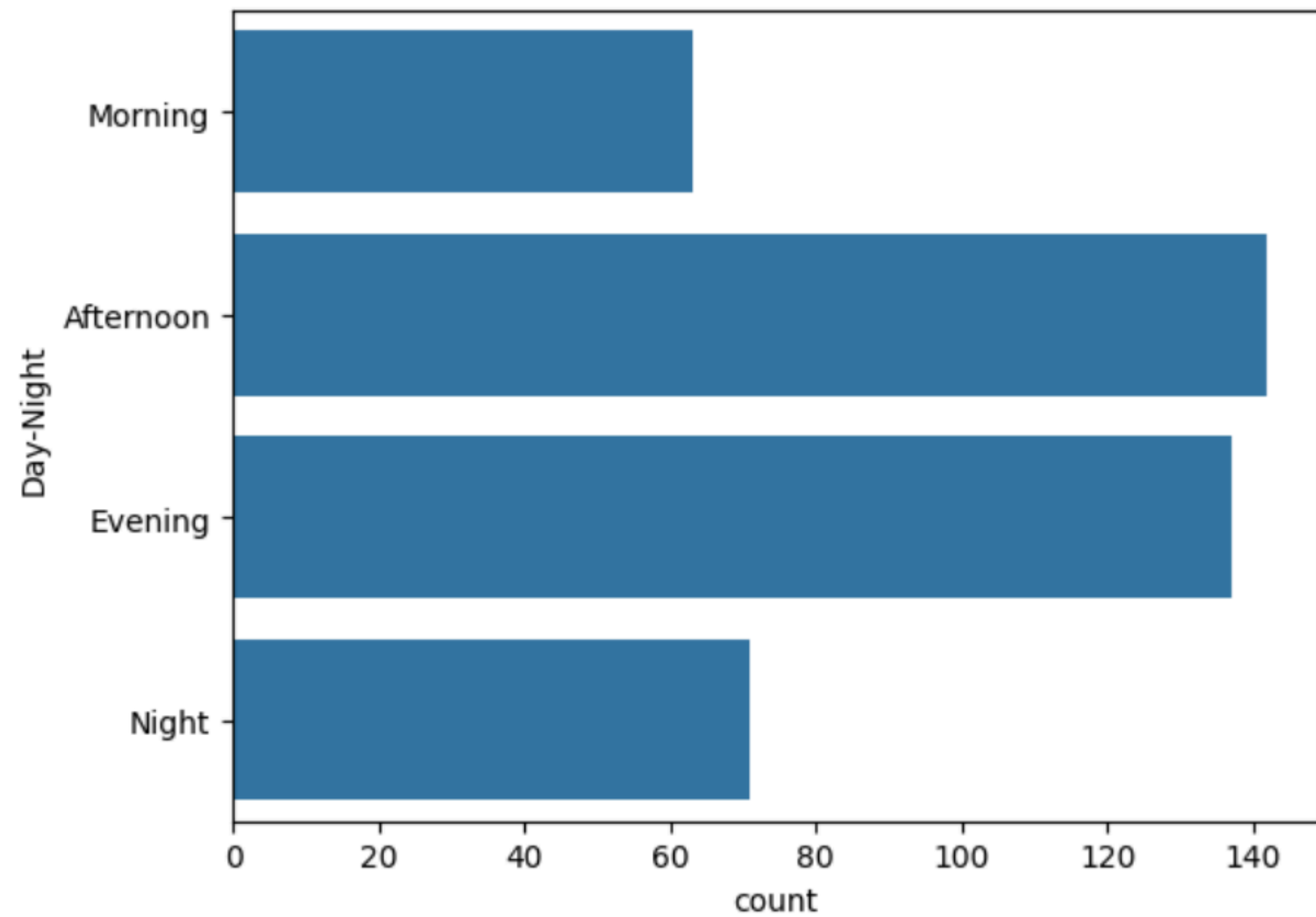
plt.subplot(1,2,2)
sns.countplot(data['PURPOSE'])
```

<Axes: xlabel='count', ylabel='PURPOSE'>



```
sns.countplot(data['Day-Night'])
```

```
: <Axes: xlabel='count', ylabel='Day-Night'>
```



```
data['Month'] = pd.DatetimeIndex(data['START_DATE']).month

month_label = {1.0: 'Jan', 2.0: 'Feb', 3.0: 'Mar', 4.0: 'APRIL',
               5.0: 'May', 6.0: 'June', 7.0: 'July', 8.0: 'Aug',
               9.0: 'Sep', 10.0: 'Oct', 11.0: 'Nov', 12.0: 'Dec'}

data['Month'] = data.Month.map(month_label)

mon = data.Month.value_counts(sort=False)
```

```
data.head()
```

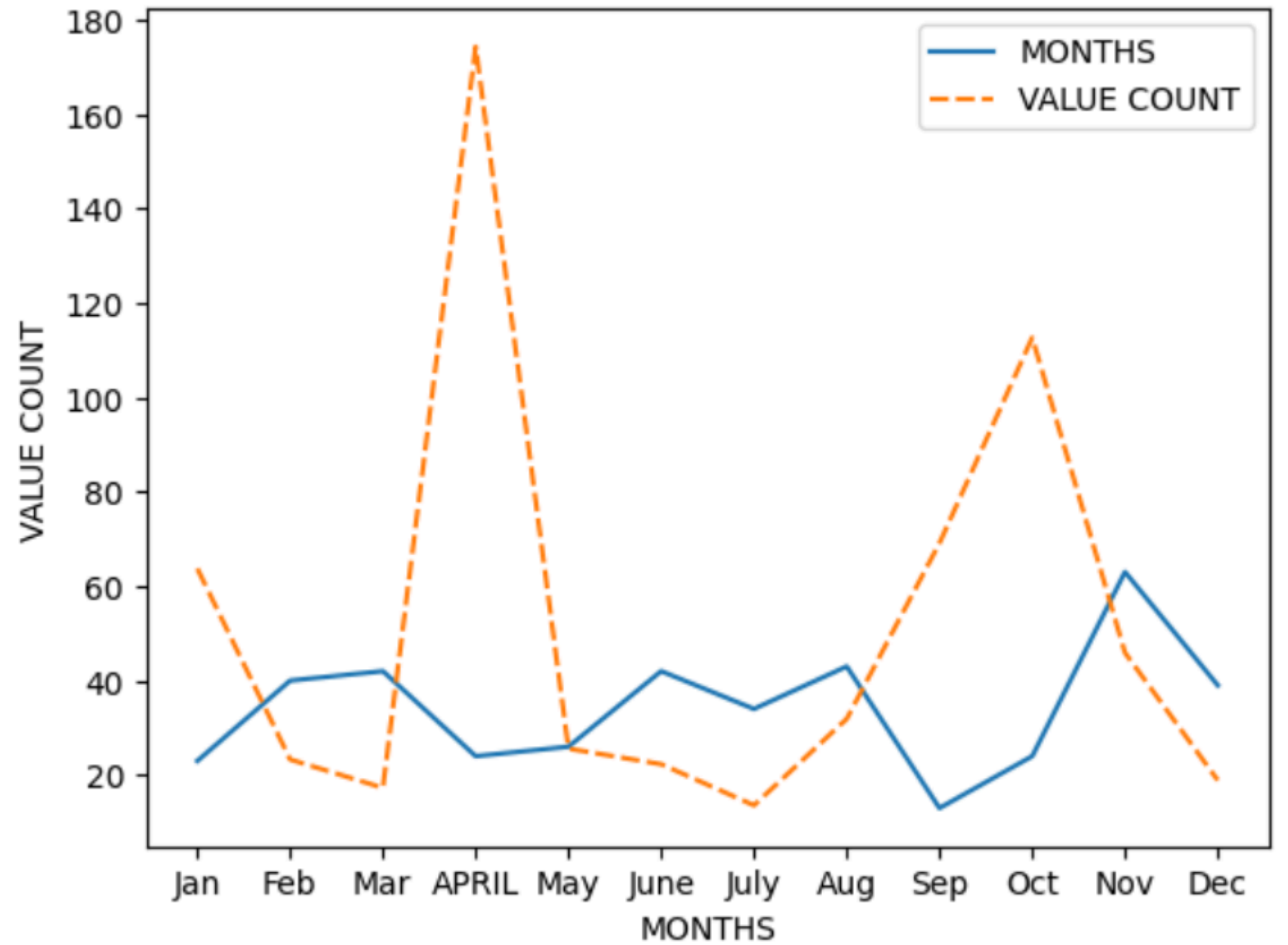
	START_DATE	END_DATE	CATEGORY	START	STOP	MILES	PURPOSE	Date	Time	Day-Night	Month
0	2016-01-01 21:11:00	2016-01-01 21:17:00	Business	Fort Pierce	Fort Pierce	5.1	Meal/Entertain	2016-01-01	21.0	Night	Jan
1	2016-01-02 01:25:00	2016-01-02 01:37:00	Business	Fort Pierce	Fort Pierce	5.0	NOT	2016-01-02	1.0	Morning	Jan
2	2016-01-02 20:25:00	2016-01-02 20:38:00	Business	Fort Pierce	Fort Pierce	4.8	Errand/Supplies	2016-01-02	20.0	Night	Jan
3	2016-01-05 17:31:00	2016-01-05 17:45:00	Business	Fort Pierce	Fort Pierce	4.7	Meeting	2016-01-05	17.0	Evening	Jan
4	2016-01-06 14:42:00	2016-01-06 15:49:00	Business	Fort Pierce	West Palm Beach	63.7	Customer Visit	2016-01-06	14.0	Afternoon	Jan



```
df = pd.DataFrame({
    "MONTHS": mon.values,
    "VALUE COUNT": data.groupby('Month', sort=False)['MILES'].max()
})

p = sns.lineplot(data=df)
p.set(xlabel="MONTHS", ylabel="VALUE COUNT")
```

```
: [Text(0.5, 0, 'MONTHS'), Text(0, 0.5, 'VALUE COUNT')]
```



```
data.head()
```

	START_DATE	END_DATE	CATEGORY	START	STOP	MILES	PURPOSE	Date	Time	Day-Night	Month
0	2016-01-01 21:11:00	2016-01-01 21:17:00	Business	Fort Pierce	Fort Pierce	5.1	Meal/Entertain	2016-01-01	21.0	Night	Jan
1	2016-01-02 01:25:00	2016-01-02 01:37:00	Business	Fort Pierce	Fort Pierce	5.0	NOT	2016-01-02	1.0	Morning	Jan
2	2016-01-02 20:25:00	2016-01-02 20:38:00	Business	Fort Pierce	Fort Pierce	4.8	Errand/Supplies	2016-01-02	20.0	Night	Jan
3	2016-01-05 17:31:00	2016-01-05 17:45:00	Business	Fort Pierce	Fort Pierce	4.7	Meeting	2016-01-05	17.0	Evening	Jan
4	2016-01-06 14:42:00	2016-01-06 15:49:00	Business	Fort Pierce	West Palm Beach	63.7	Customer Visit	2016-01-06	14.0	Afternoon	Jan

```
data['Day'] = data.START_DATE.dt.weekday
```

```
day_label = {
    0: 'Mon', 1: 'Tues', 2: 'Wed', 3: 'Thur', 4: 'Fri', 5: 'Sat', 6: 'Sun'}
```

```
data['Day'] = data['Day'].map(day_label)
```

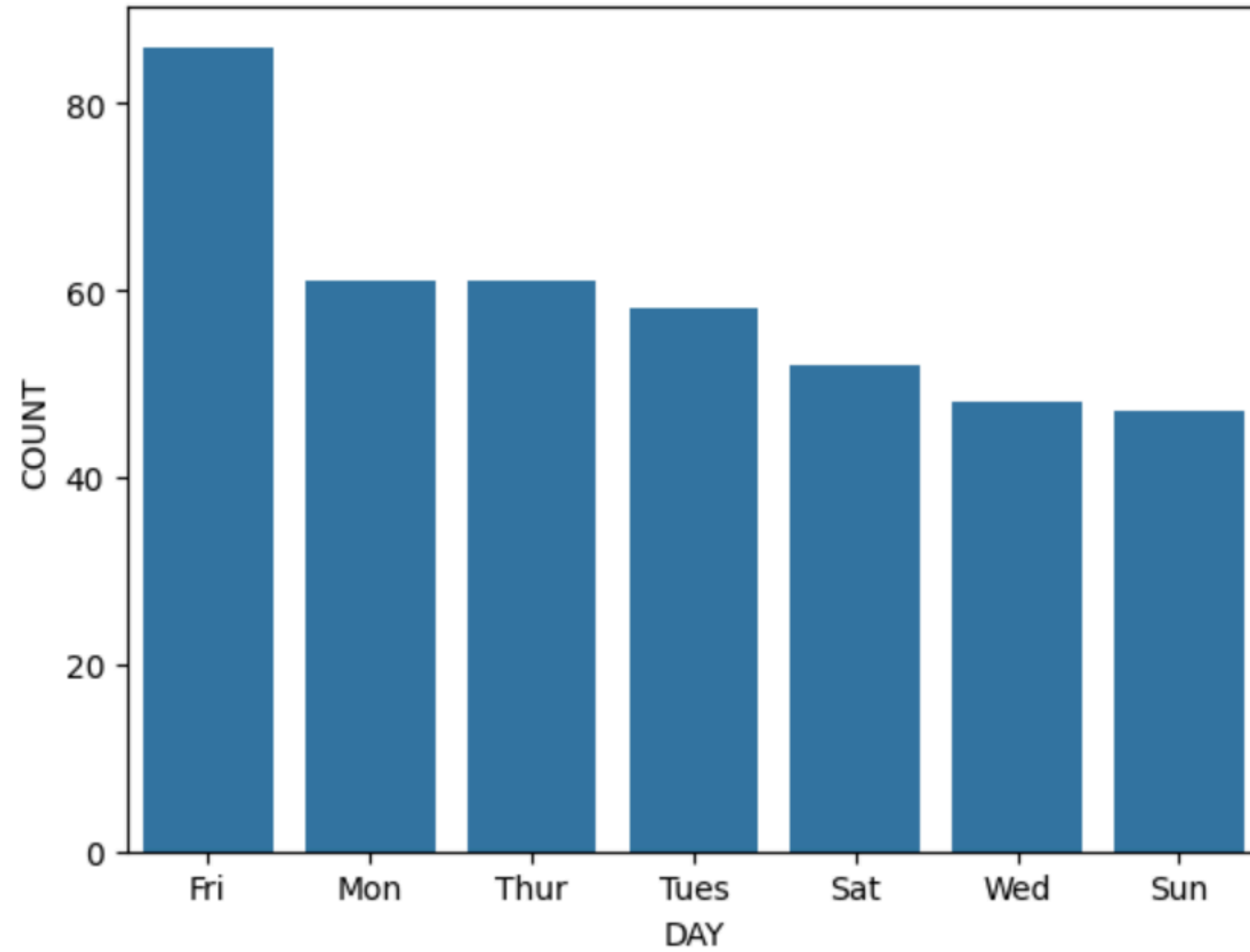
```
data.head()
```

	START_DATE	END_DATE	CATEGORY	START	STOP	MILES	PURPOSE	Date	Time	Day-Night	Month	Day
0	2016-01-01 21:11:00	2016-01-01 21:17:00	Business	Fort Pierce	Fort Pierce	5.1	Meal/Entertain	2016-01-01	21.0	Night	Jan	Fri
1	2016-01-02 01:25:00	2016-01-02 01:37:00	Business	Fort Pierce	Fort Pierce	5.0	NOT	2016-01-02	1.0	Morning	Jan	Sat
2	2016-01-02 20:25:00	2016-01-02 20:38:00	Business	Fort Pierce	Fort Pierce	4.8	Errand/Supplies	2016-01-02	20.0	Night	Jan	Sat
3	2016-01-05 17:31:00	2016-01-05 17:45:00	Business	Fort Pierce	Fort Pierce	4.7	Meeting	2016-01-05	17.0	Evening	Jan	Tues
4	2016-01-06 14:42:00	2016-01-06 15:49:00	Business	Fort Pierce	West Palm Beach	63.7	Customer Visit	2016-01-06	14.0	Afternoon	Jan	Wed

```
day_label = data.Day.value_counts()
```

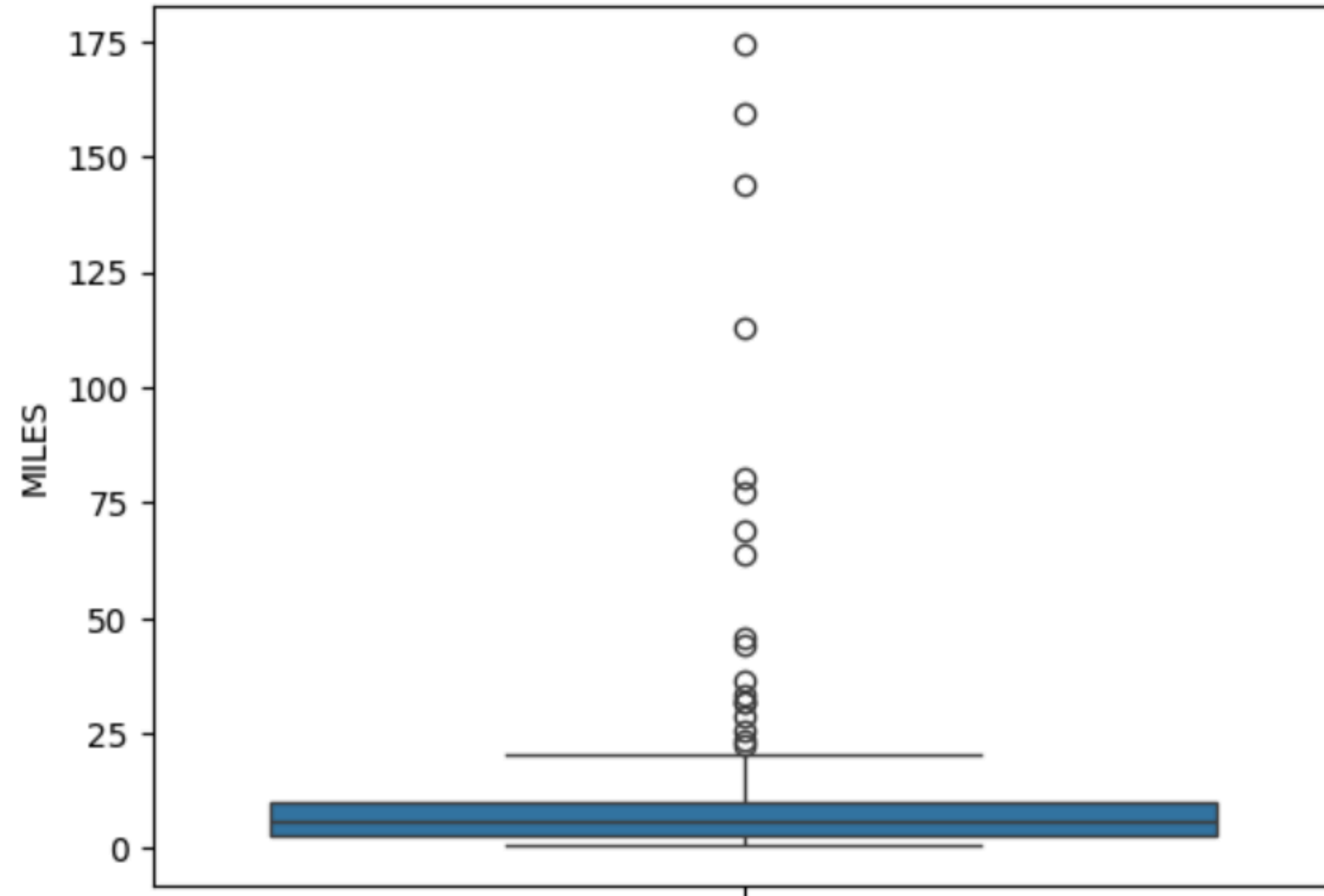
```
sns.barplot(x=day_label.index, y=day_label)
plt.xlabel('DAY')
plt.ylabel('COUNT')
```

```
: Text(0, 0.5, 'COUNT')
```



```
sns.boxplot(data['MILES'])
```

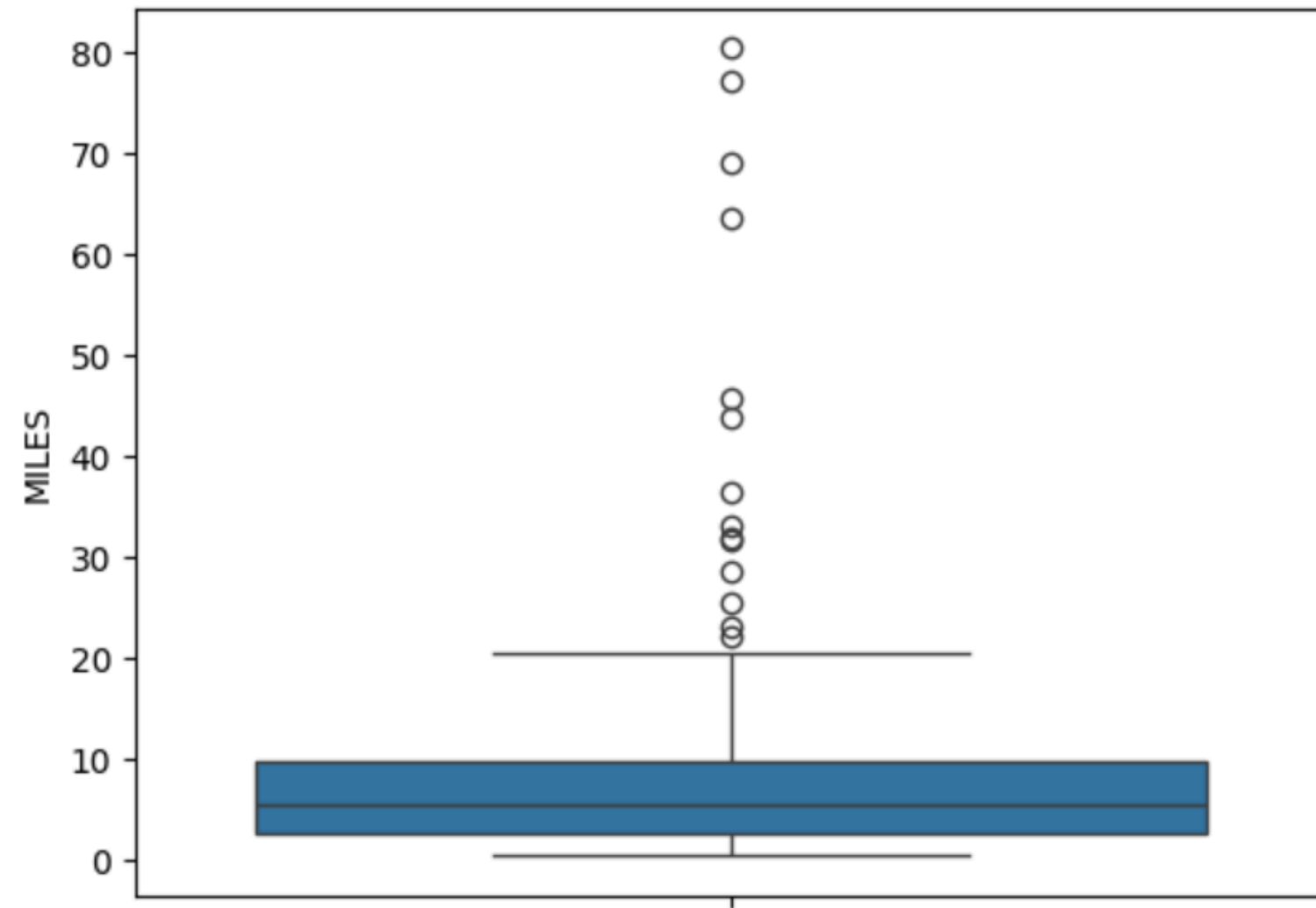
<Axes: ylabel='MILES'>





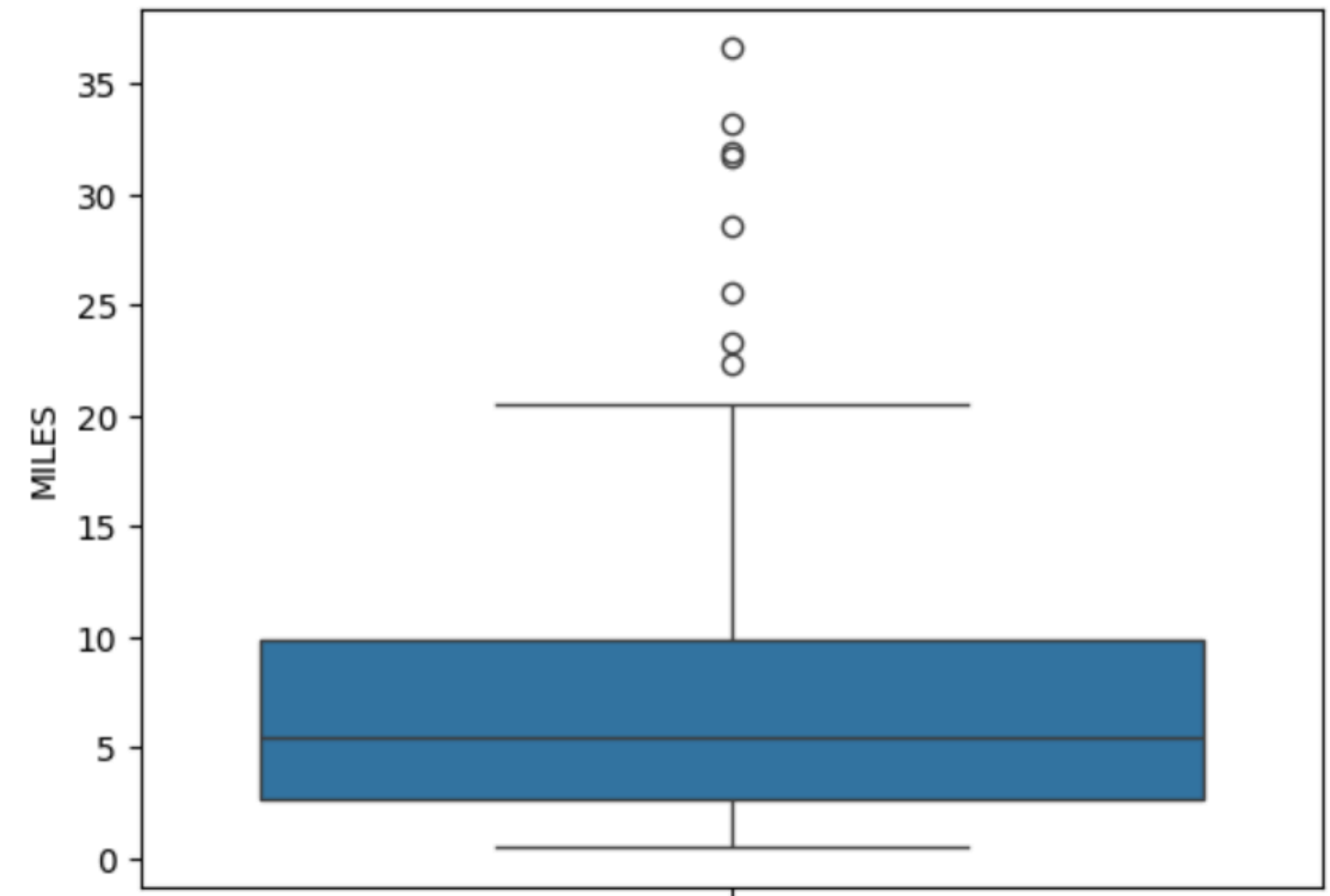
```
sns.boxplot(data[data['MILES']<100]['MILES'])
```

<Axes: ylabel='MILES'>



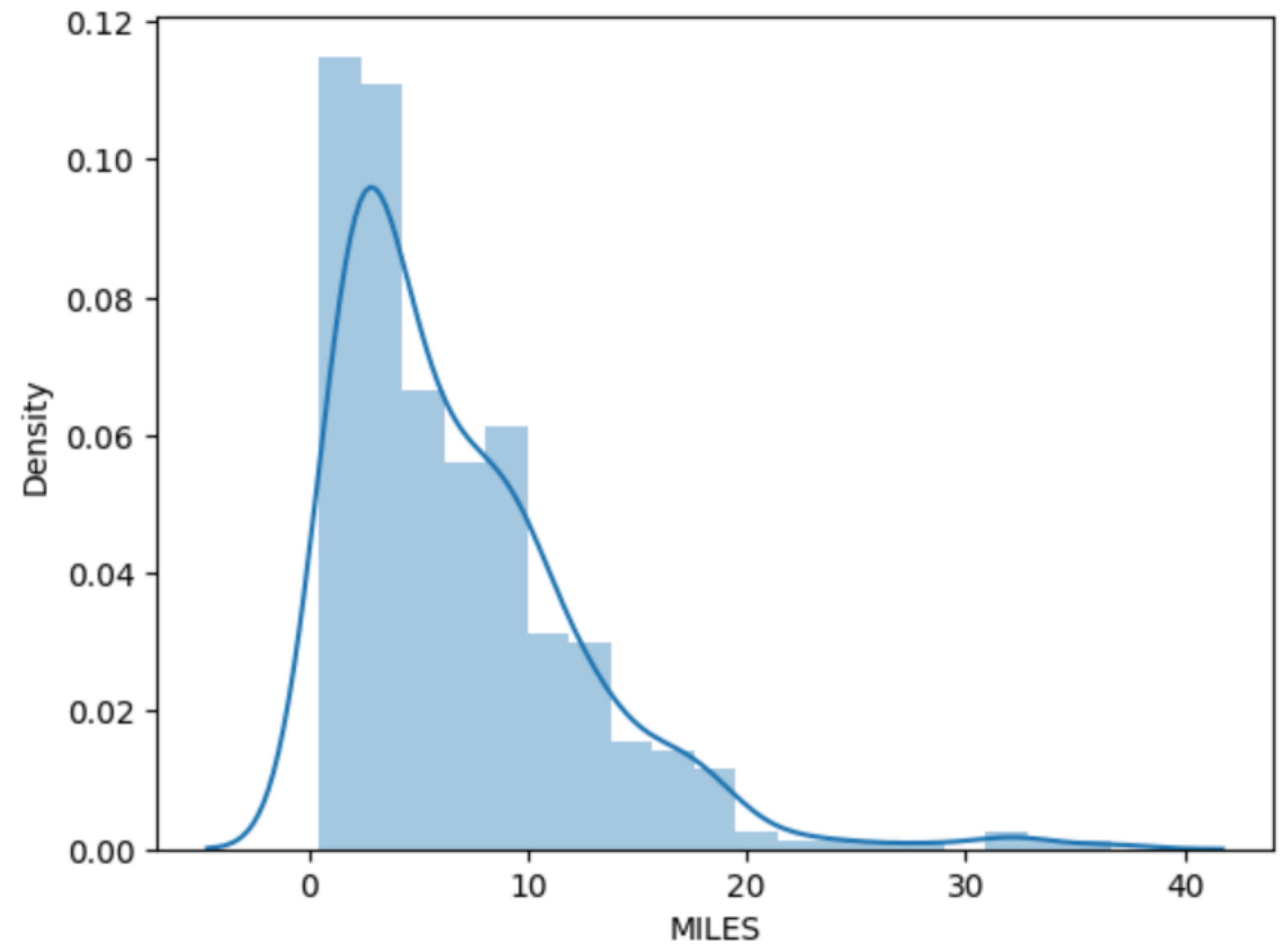
```
sns.boxplot(data[data['MILES']<40]['MILES'])
```

<Axes: ylabel='MILES'>



```
sns.boxplot(data[data['MILES']<40]['MILES'])
```

<Axes: xlabel='MILES', ylabel='Density'>





# Insights

## **Most Booked Ride Category**

People book the most Uber rides under the Business category.

## **Most Common Ride Purpose**

The most frequent purpose for Uber rides is Meeting.

## **Peak Booking Time**

Most Uber rides are booked during morning rush hours (7-9 AM) and evening hours (4-6 PM).

## **Most Active Booking Months**

Uber ride bookings peak in the months of December and January.

## **Most Active Days of the Week**

People book Uber rides the most on Fridays, followed by Thursdays.

## **Typical Ride Distance**

The majority of Uber rides are within 2 to 10 miles, indicating short to medium-distance travel.



# Conclusion

**The Uber Ride Data Analysis reveals key patterns in user behavior. Most rides are booked for business purposes, particularly for meetings, with a high volume during rush hours and on weekdays, especially Fridays. Booking activity peaks in December and January, and the majority of trips are short to medium distances (2-10 miles). These insights can help Uber better understand customer needs and optimize services during high-demand periods.**

**Thank You!**