In [371]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
df=pd.read_csv('/content/UberDataset1.csv')
```

In [372]:
```
df
```

Out[372]:

	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
...	...	...	...	...	...	...	...
1151	12/31/2016 13:24	12/31/2016 13:42	Business	Kar?chi	Unknown Location	3.9	Temporary Site
1152	12/31/2016 15:03	12/31/2016 15:38	Business	Unknown Location	Unknown Location	16.2	Meeting
1153	12/31/2016 21:32	12/31/2016 21:50	Business	Katunayake	Gampaha	6.4	Temporary Site
1154	12/31/2016 22:08	12/31/2016 23:51	Business	Gampaha	Ilukwatta	48.2	Temporary Site
1155	Totals	NaN	NaN	NaN	NaN	12204.7	NaN

1156 rows × 7 columns

In [373]:
```
df.head()
```

Out[373]:

	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

In [374]:
```
df.describe()
```

Out[374]:

	MILES
count	1156.000000
mean	21.115398
std	359.299007
min	0.500000
25%	2.900000
50%	6.000000
75%	10.400000
max	12204.700000

In [375]:
```
df.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  1156 non-null   object
 1   END_DATE    1155 non-null   object
 2   CATEGORY    1121 non-null   object
 3   START       1155 non-null   object
 4   STOP        1155 non-null   object
 5   MILES       1156 non-null   float64
 6   PURPOSE     652 non-null    object
dtypes: float64(1), object(6)
memory usage: 63.3+ KB
```

In [376]:
```
df[df.duplicated()]
```

Out[376]:

	START_DATE	END_DATE	CATEGORY	START	STOP	MILES	PURPOSE
492	6/28/2016 23:34	6/28/2016 23:59	Business	Durham	Cary	9.9	Meeting

In [377]:
```
# Remove duplicate rows
df = df.drop_duplicates()
```

In [378]:
```
df[df.duplicated()]
```

Out[378]:

START_DATE	END_DATE	CATEGORY	START	STOP	MILES	PURPOSE

In [379]:
```
df.isnull().sum()
```

Out[379]:
```
START_DATE      0
END_DATE        1
CATEGORY       35
START           1
STOP            1
MILES           0
PURPOSE       504
dtype: int64
```

In [380]:
```
#calculate the percentage of missing values in each column
(df.isnull().sum()/len(df))*100
```

Out[380]:
```
START_DATE      0.000000
END_DATE        0.086580
CATEGORY        3.030303
START           0.086580
STOP            0.086580
MILES           0.000000
PURPOSE        43.636364
dtype: float64
```

In [381]:
```
df['CATEGORY'].value_counts()
```

Out[381]:
```
Business    1046
Personal      74
Name: CATEGORY, dtype: int64
```

In [382]:
```
#filling in the most probable value
df.fillna(("CATEGORY":'Business'),inplace=True)
```

```
<ipython-input-182-879855af01fa>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df.fillna(("CATEGORY":'Business'),inplace=True)
```

In [383]:
```
df['PURPOSE'].value_counts()
```

Out[383]:
```
Meeting            185
Meal/Entertain     160
Errand/Supplies    128
Customer Visit     101
Temporary Site      50
Between Offices     18
Moving               4
Airport/Travel       3
Charity ($)          1
Commute              1
Name: PURPOSE, dtype: int64
```

In [384]:
```
#filling in the most probable value
df.fillna(("PURPOSE":'Meeting'),inplace=True)
```

```
<ipython-input-184-9353bd30d714>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df.fillna(("PURPOSE":'Meeting'),inplace=True)
```

In [385]:
```
df.isnull().sum()
```

Out[385]:
```
START_DATE    0
END_DATE      1
CATEGORY      0
START         1
STOP          1
MILES         0
PURPOSE       0
dtype: int64
```

In [386]:
```
df = df.dropna(subset=["END_DATE"])
df = df.dropna(subset=["START"])
df = df.dropna(subset=["STOP"])
```

In [387]:
```
df.isnull().sum()
```

Out[387]:
```
START_DATE    0
END_DATE      0
CATEGORY      0
START         0
STOP          0
MILES         0
PURPOSE       0
dtype: int64
```

In [388]:
```
unknown_locations = df[(df['START'] == 'Unknown Location') | (df['STOP'] == 'Unknown Location')]

# Print the filtered DataFrame
print(unknown_locations)
```

```
            START_DATE          END_DATE  CATEGORY           START  \
108    2/16/2016 3:21    2/16/2016 6:13  Business       Katunayaka   
109    2/16/2016 8:29    2/16/2016 9:34  Business  Unknown Location   
116    2/16/2016 17:40   2/16/2016 17:44  Business         Nugegoda   
117    2/17/2016 13:18   2/17/2016 14:04  Business  Unknown Location   
121    2/18/2016 8:19    2/18/2016 8:27  Business  Unknown Location   
...               ...               ...       ...               ...   
1141  12/29/2016 19:50  12/29/2016 20:12  Business  Unknown Location   
1143  12/29/2016 20:53  12/29/2016 21:42  Business         Kar?chi   
1144  12/29/2016 23:14  12/29/2016 23:47  Business  Unknown Location   
1151  12/31/2016 13:24  12/31/2016 13:42  Business         Kar?chi   
1152  12/31/2016 15:03  12/31/2016 15:38  Business  Unknown Location   

                STOP  MILES          PURPOSE  
108          Colombo   43.7   Customer Visit  
109          Colombo   14.1          Meeting  
116   Unknown Location    3.6  Errand/Supplies  
117          Colombo   14.7   Temporary Site  
121   Unknown Location   23.5   Temporary Site  
...                ...    ...              ...  
1141         Kar?chi    4.1   Customer Visit  
1143  Unknown Location    6.4          Meeting  
1144         Kar?chi   12.9          Meeting  
1151  Unknown Location    3.9   Temporary Site  
1152  Unknown Location   16.2          Meeting  

[211 rows x 7 columns]
```

In [389]:
```
df = df.drop(unknown_locations.index)

# Reset index after dropping rows
df = df.reset_index(drop=True)
```

In [390]:
```
df
```

Out[390]:

	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	Meeting
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
...	...	...	...	...	...	...	...
938	12/30/2016 16:45	12/30/2016 17:08	Business	Kar?chi	Kar?chi	4.6	Meeting
939	12/30/2016 23:06	12/30/2016 23:10	Business	Kar?chi	Kar?chi	0.8	Customer Visit
940	12/31/2016 1:07	12/31/2016 1:14	Business	Kar?chi	Kar?chi	0.7	Meeting
941	12/31/2016 21:32	12/31/2016 21:50	Business	Katunayake	Gampaha	6.4	Temporary Site
942	12/31/2016 22:08	12/31/2016 23:51	Business	Gampaha	Ilukwatta	48.2	Temporary Site

943 rows × 7 columns

In [391]:
```
output = []
for col in df.columns:
    unique = df[col].nunique()
    colType = str(df[col].dtype)
    categories=df[col].unique()

    output.append([col, unique, colType,categories])

output = pd.DataFrame(output)
output.columns = ['colName','unique','dtype','categories']
output
```

Out[391]:

	colName	unique	dtype	categories
0	START_DATE	943	object	[01-01-2016 21:11, 01-02-2016 01:25, 01-02-201...
1	END_DATE	943	object	[01-01-2016 21:17, 01-02-2016 01:37, 01-02-201...
2	CATEGORY	2	object	[Business, Personal]
3	START	175	object	[Fort Pierce, West Palm Beach, Cary, Jamaica, ...
4	STOP	187	object	[Fort Pierce, West Palm Beach, Palm Beach, Car...
5	MILES	229	float64	[5.1, 5.0, 4.8, 4.7, 63.7, 4.3, 7.1, 0.8, 8.3,...
6	PURPOSE	10	object	[Meal/Entertain, Meeting, Errand/Supplies, Cus...

In [392]:
```
print(np.max(df['MILES']))
print(np.min(df['MILES']))
```

```
310.3
0.5
```

In [393]:
```
df['MILES bins'] = pd.cut(x=df['MILES'], bins=[0,50,100,150,200,250,300,350])
```

In [394]:
```
df['MILES bins'].value_counts()
```

Out[394]:
```
(0, 50]       924
(50, 100]       7
(100, 150]      5
(150, 200]      5
(200, 250]      1
(200, 350]      1
(250, 300]      0
Name: MILES bins, dtype: int64
```

In [215]:
```
df = df.rename(columns={'MILES': 'distance_travelled'})
```

In [216]:
```
df
```

Out[216]:

	START_DATE	END_DATE	CATEGORY	START	STOP	distance_travelled	PURPOSE	MILES bins
0	01-01-2016 21:11	01-01-2016 21:17	Business	Fort Pierce	Fort Pierce	5.1	Meal/Entertain	(0, 50]
1	01-02-2016 01:25	01-02-2016 01:37	Business	Fort Pierce	Fort Pierce	5.0	Meeting	(0, 50]
2	01-02-2016 20:25	01-02-2016 20:38	Business	Fort Pierce	Fort Pierce	4.8	Errand/Supplies	(0, 50]
3	01-05-2016 17:31	01-05-2016 17:45	Business	Fort Pierce	Fort Pierce	4.7	Meeting	(0, 50]
4	01-06-2016 14:42	01-06-2016 15:49	Business	Fort Pierce	West Palm Beach	63.7	Customer Visit	(50, 100]
...	...	...	...	...	...	...	...	...
938	12/30/2016 16:45	12/30/2016 17:08	Business	Kar?chi	Kar?chi	4.6	Meeting	(0, 50]
939	12/30/2016 23:06	12/30/2016 23:10	Business	Kar?chi	Kar?chi	0.8	Customer Visit	(0, 50]
940	12/31/2016 1:07	12/31/2016 1:14	Business	Kar?chi	Kar?chi	0.7	Meeting	(0, 50]
941	12/31/2016 21:32	12/31/2016 21:50	Business	Katunayake	Gampaha	6.4	Temporary Site	(0, 50]
942	12/31/2016 22:08	12/31/2016 23:51	Business	Gampaha	Ilukwatta	48.2	Temporary Site	(0, 50]

943 rows × 8 columns

In [210]:
```
pd.crosstab(df['CATEGORY'], df['PURPOSE'])
```

Out[210]:

PURPOSE	Airport/Travel	Between Offices	Charity ($)	Commute	Customer Visit	Errand/Supplies	Meal/Entertain	Meeting	Moving	Temporary Site
CATEGORY										
Business	1	18	0	0	92	111	148	471	0	32
Personal	0	0	1	1	0	0	0	64	4	0

In [214]:
```

```

In [208]:
```

```

In [208]:
```

```

In [ ]:
```

```