UBER DATA ANALYSIS

Uber Technologies, Inc., commonly known as Uber, is an American technology company. Its services include ride-hailing, food delivery, package delivery, couriers, freight transportation, and, through a partnership with Lime, electric bicycle and motorized scooter rental.

In this tutorial, I will use Python to analyze data from Uber.

I will use Python to:

- Check how long do people travel with Uber?
- What Hour Do Most People Take Uber To Their Destination?
- Check The Purpose Of Trips
- Which Day Has The Highest Number Of Trips
- What Are The Number Of Trips Per Each Day?
- What Are The Trips In The Month
- The starting points of trips. Where Do People Start Boarding Their Trip From Most?

Import Libraries

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import calendar
import datetime
```

Loading Dataset

In [2]:
```
df = pd.read_csv('uber.csv')
df.head()
```

Out[2]:

	START_DATE*	END_DATE*	CATEGORY*	START*	STOP*	MILES*	PURPOSE*
0	1/1/2016 21:11	1/1/2016 21:17	Business	Fort Pierce	Fort Pierce	5.1	Meal/Entertain
1	1/2/2016 1:25	1/2/2016 1:37	Business	Fort Pierce	Fort Pierce	5.0	NaN
2	1/2/2016 20:25	1/2/2016 20:38	Business	Fort Pierce	Fort Pierce	4.8	Errand/Supplies
3	1/5/2016 17:31	1/5/2016 17:45	Business	Fort Pierce	Fort Pierce	4.7	Meeting
4	1/6/2016 14:42	1/6/2016 15:49	Business	Fort Pierce	West Palm Beach	63.7	Customer Visit

Check for missing values

If data is not available, Python uses NaN to represent it. Let's check below if datapoints missing in our dataset.

In [3]:
```
df.isnull().sum()
```

Out[3]:
```
START_DATE*      0
END_DATE*        1
CATEGORY*        1
START*           1
STOP*            1
MILES*           0
PURPOSE*       503
dtype: int64
```

Let's drop null values

In [6]:
```
df = df.dropna()
```

In [7]:
```
df.isnull().sum()
```

Out[7]:
```
START_DATE*    0
END_DATE*      0
CATEGORY*      0
START*         0
STOP*          0
MILES*         0
PURPOSE*       0
dtype: int64
```

Let's check datatypes of columns

In [10]:
```
df.dtypes
```

Out[10]:
```
START_DATE*     object
END_DATE*       object
CATEGORY*       object
START*          object
STOP*           object
MILES*         float64
PURPOSE*        object
dtype: object
```

"START_DATE AND END_DATE" column should have date type. Let's convert them

In [13]:
```
df['START_DATE*'] = pd.to_datetime(df['START_DATE*'],format='%m/%d/%Y %H:%M')
df['END_DATE*'] = pd.to_datetime(df['END_DATE*'],format='%m/%d/%Y %H:%M')
```

In [14]:
```
df.head()
```

Out[14]:

	START_DATE*	END_DATE*	CATEGORY*	START*	STOP*	MILES*	PURPOSE*
0	2016-01-01 21:11:00	2016-01-01 21:17:00	Business	Fort Pierce	Fort Pierce	5.1	Meal/Entertain
2	2016-01-02 20:25:00	2016-01-02 20:38:00	Business	Fort Pierce	Fort Pierce	4.8	Errand/Supplies
3	2016-01-05 17:31:00	2016-01-05 17:45:00	Business	Fort Pierce	Fort Pierce	4.7	Meeting
4	2016-01-06 14:42:00	2016-01-06 15:49:00	Business	Fort Pierce	West Palm Beach	63.7	Customer Visit
5	2016-01-06 17:15:00	2016-01-06 17:19:00	Business	West Palm Beach	West Palm Beach	4.3	Meal/Entertain

Now let's split START_DATE* column into:

- Hour
- Month
- Day
- Weekday

In [22]:
```
df['Hour'] = df['START_DATE*'].dt.hour
df['DAY'] = df['START_DATE*'].dt.day
df['MONTH'] = df['START_DATE*'].dt.month
df['WEEKDAY'] = df['START_DATE*'].dt.day_name()
df['YEAR'] = df['START_DATE*'].dt.year
```

In [23]:
```
df.head()
```

Out[23]:

	START_DATE*	END_DATE*	CATEGORY*	START*	STOP*	MILES*	PURPOSE*	Hour	DAY	MONTH	WEEKDAY	YEAR
0	2016-01-01 21:11:00	2016-01-01 21:17:00	Business	Fort Pierce	Fort Pierce	5.1	Meal/Entertain	21	1	1	Friday	2016
2	2016-01-02 20:25:00	2016-01-02 20:38:00	Business	Fort Pierce	Fort Pierce	4.8	Errand/Supplies	20	2	1	Saturday	2016
3	2016-01-05 17:31:00	2016-01-05 17:45:00	Business	Fort Pierce	Fort Pierce	4.7	Meeting	17	5	1	Tuesday	2016
4	2016-01-06 14:42:00	2016-01-06 15:49:00	Business	Fort Pierce	West Palm Beach	63.7	Customer Visit	14	6	1	Wednesday	2016
5	2016-01-06 17:15:00	2016-01-06 17:19:00	Business	West Palm Beach	West Palm Beach	4.3	Meal/Entertain	17	6	1	Wednesday	2016

Let's see how many CATEGORIES are there in dataset

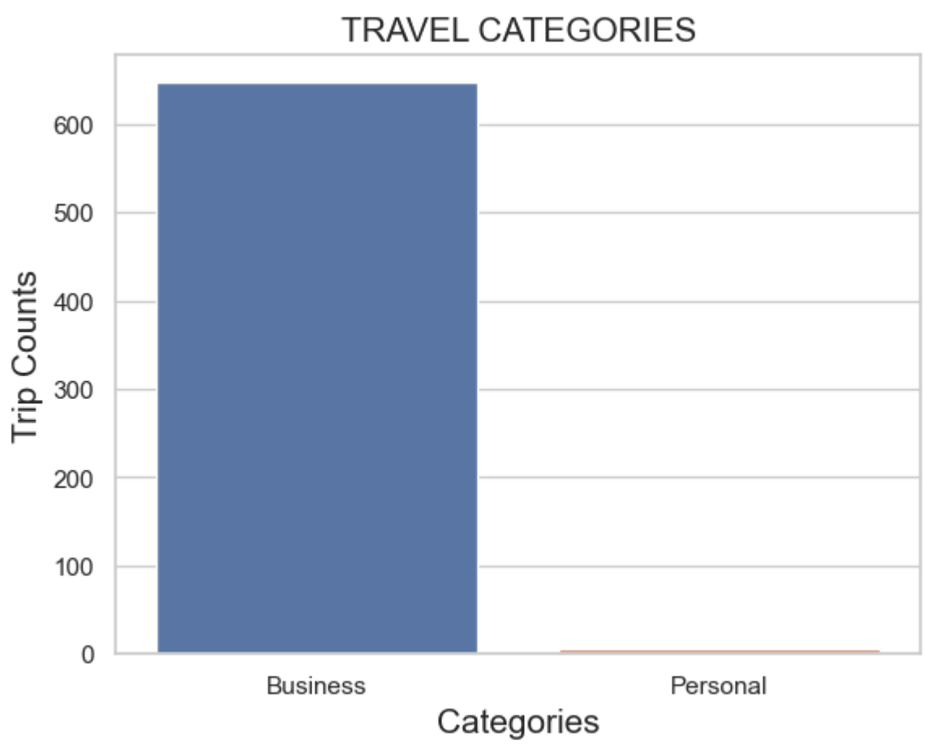
In [24]:
```
df['CATEGORY*'].value_counts()
```

Out[24]:
```
Business    647
Personal      6
Name: CATEGORY*, dtype: int64
```

Now let's create countplot to show Category distribution

In [91]:
```
ax = sns.countplot(x=df['CATEGORY*'])
ax.set_title('TRAVEL CATEGORIES', fontsize=15)
ax.set_xlabel('Categories', fontsize=15)
ax.set_ylabel('Trip Counts', fontsize=15)
```

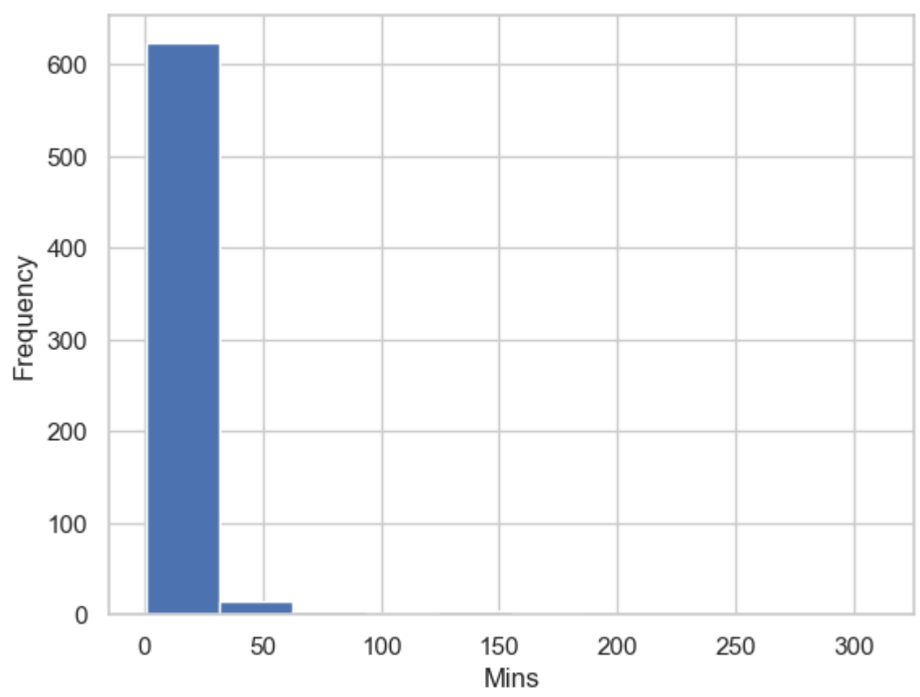
Out[91]:
```
Text(0, 0.5, 'Trip Counts')
```



Let's find out how long people travel with uber

In [39]:
```
ax = df['MILES*'].plot.hist()
ax.set_xlabel("Mins")
```

Out[39]:
```
Text(0.5, 0, 'Mins')
```

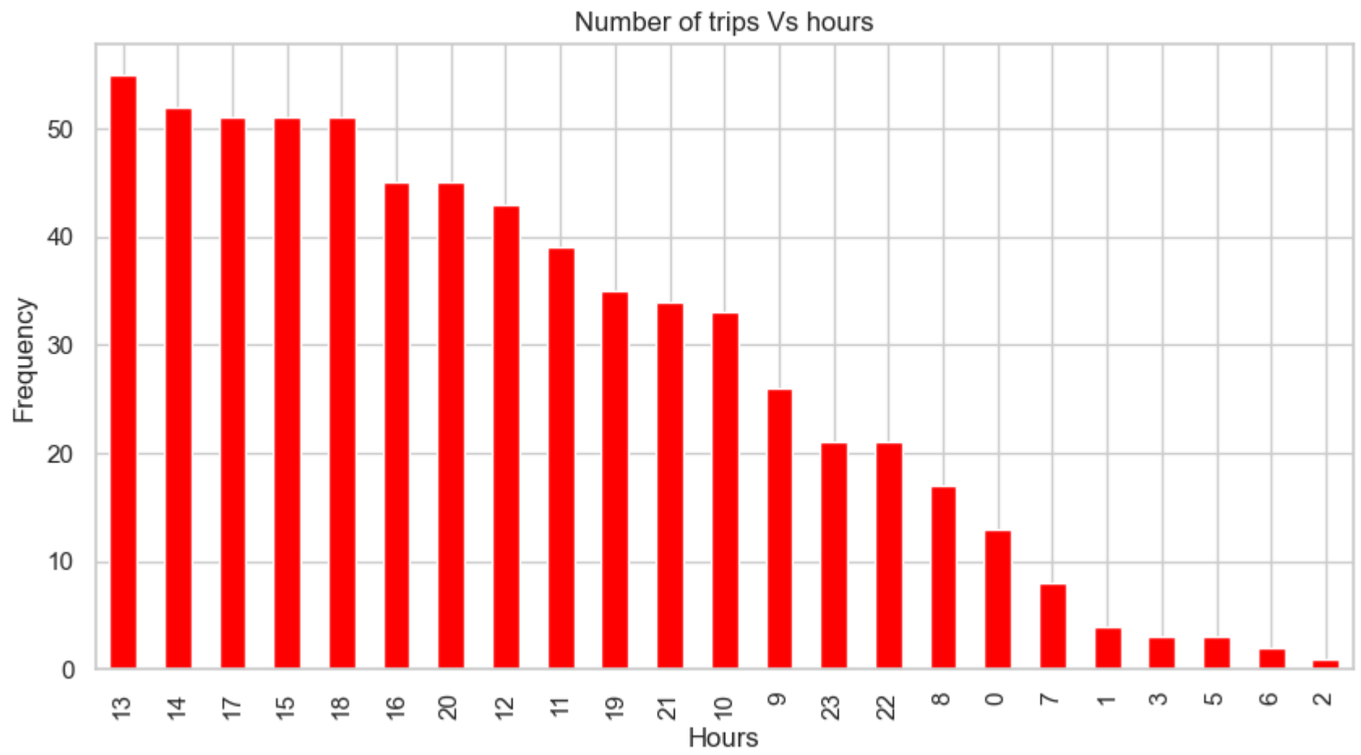


Above graph illustrate that most people travel for short period of time

Now let's see what hour do most people take Uber trip to Destination?

In [55]:
```
hours = df['Hour'].value_counts()
hours.plot(kind='bar',color='red',figsize=(10,5))
plt.xlabel('Hours')
plt.ylabel('Frequency')
plt.title('Number of trips Vs hours')
```

Out[55]: Text(0.5, 1.0, 'Number of trips Vs hours')

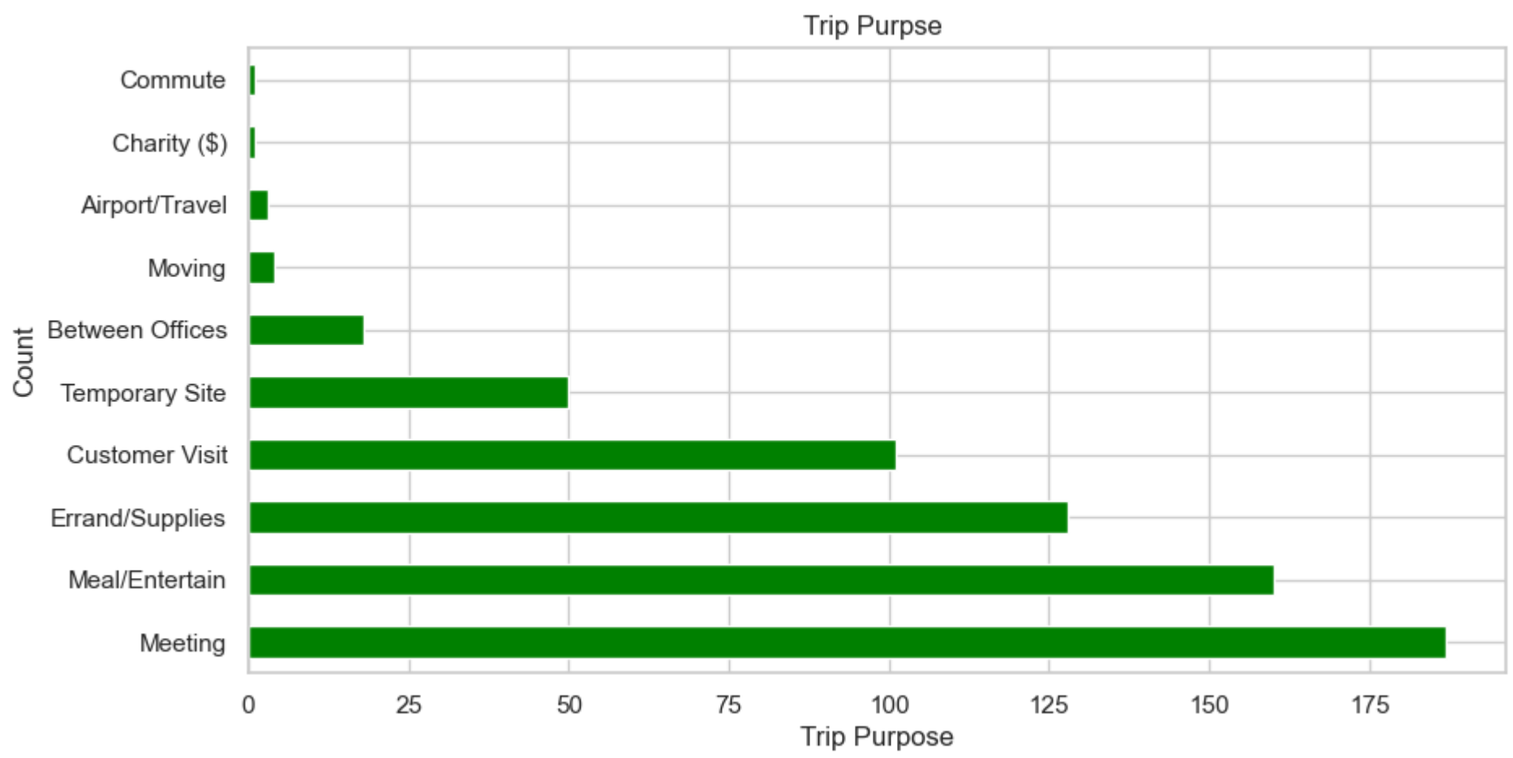


Above graph shows that at 1300 hrs (1pm) was the busiest hour and 0200 hrs (2am) is the quiet hour.

Now let's see the purpose of trip.

In [69]:
```
df['PURPOSE*'].value_counts().plot(kind='barh', color='green', figsize=(10,5))
plt.xlabel('Trip Purpose')
plt.ylabel('Count')
plt.title("Trip Purpse ")
```

Out[69]: Text(0.5, 1.0, 'Trip Purpse ')

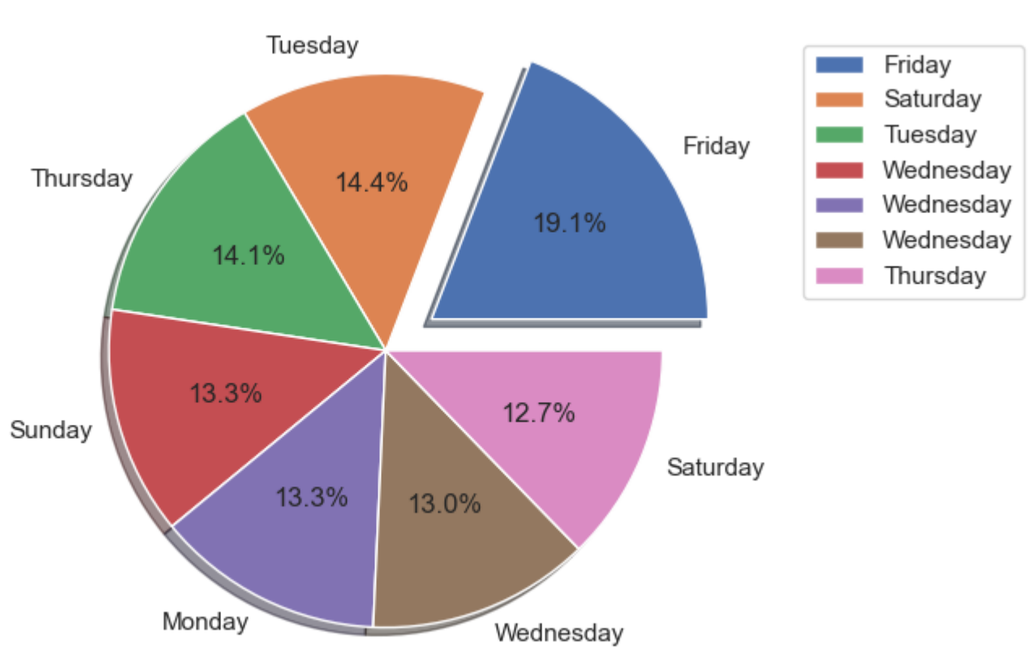


Above graph shows that most people use Meeting and Meal/Entertain as their purpose of trip and commute is used for very less number of people

Now let's find out which day has the highest number of trips.

In [80]:
```
myexplode = [0.2, 0, 0, 0,0,0,0]
df['WEEKDAY'].value_counts().plot(kind='pie',autopct='%1.1f%%' ,shadow = True, explode = myexplode,figsize=(10,5))
plt.ylabel(' ')
plt.legend(df['WEEKDAY'], loc='best')
plt.axis('equal')
plt.title('Heighest Number of Trips Each Day')
```

Out[80]:
```
(-1.118436304293846,
 1.273642330005422,
 -1.1077749827068821,
 1.1487587343614798)
```

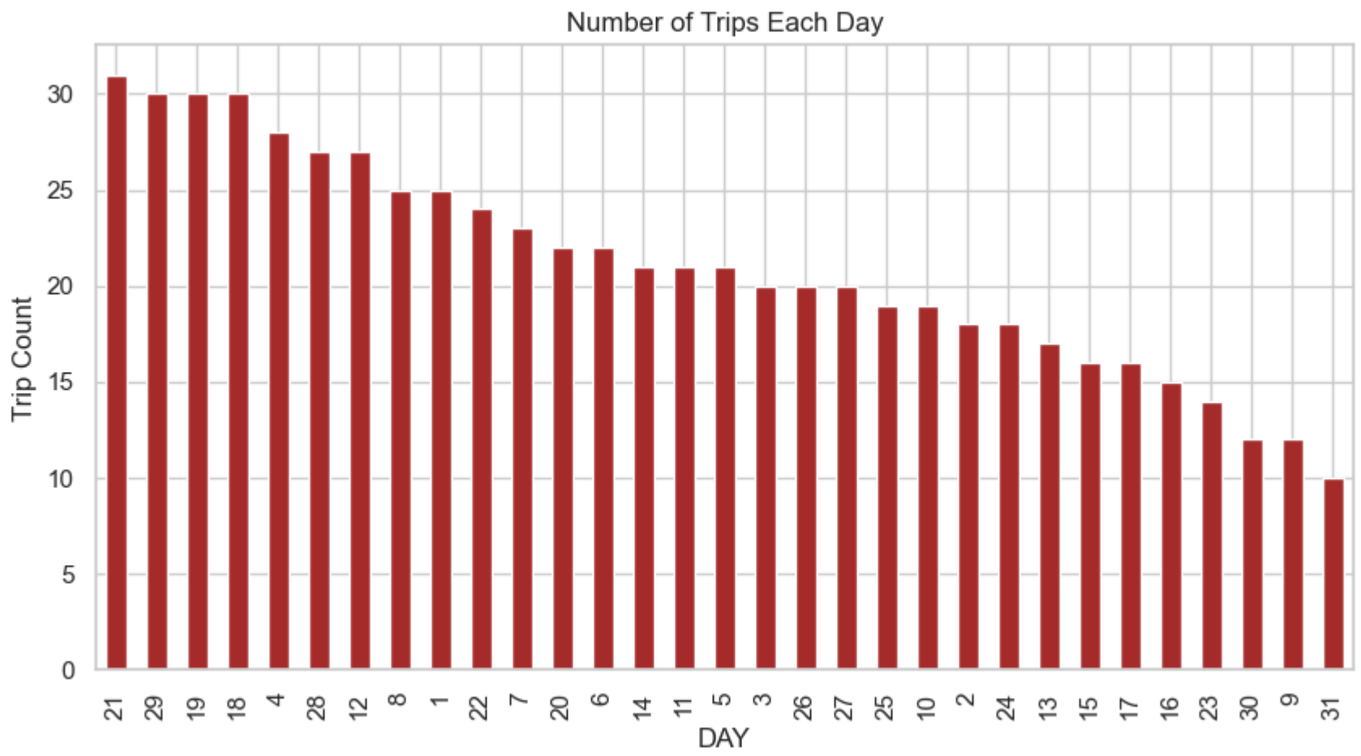


Above graph shows that Friday has the highest number of trips while Saturday has the lowest

Now let's find out number of trips each day

In [84]:
```
df['DAY'].value_counts().plot(kind='bar', color = 'brown', figsize = (10,5))
plt.xlabel('DAY')
plt.ylabel('Trip Count')
plt.title('Number of Trips Each Day')
```

Out[84]:
```
Text(0.5, 1.0, 'Number of Trips Each Day')
```

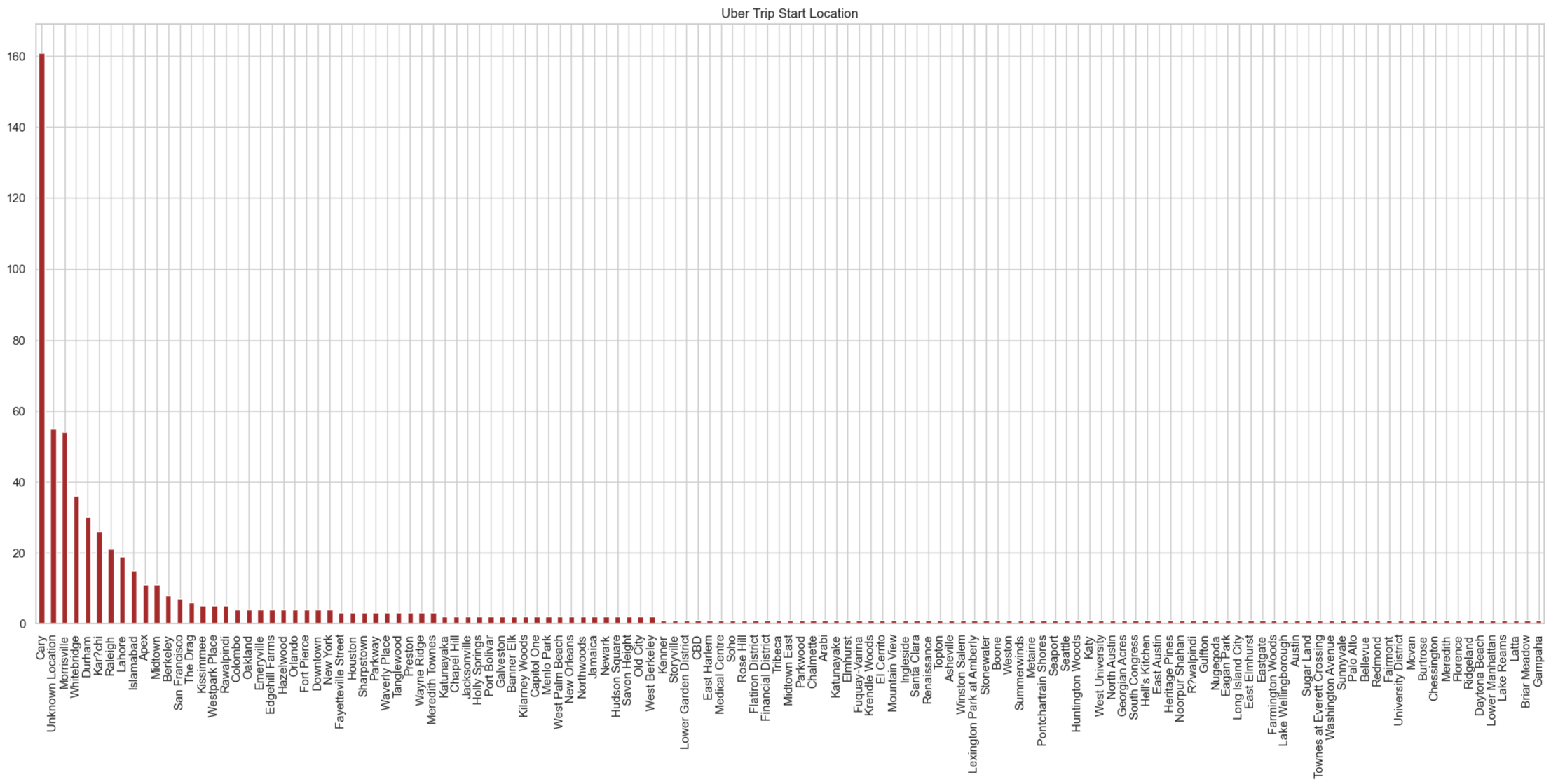


Above graph shows that on 21st of every month people book uber ride the most while end of the month (31st) people least book uber ride

Now let's find out where do people start boarding their trips

In [90]:
```
df['START*'].value_counts().plot(kind='bar', color='brown', figsize=(25,10))
plt.title('Uber Trip Start Location')
```

Out[90]:
```
Text(0.5, 1.0, 'Uber Trip Start Location')
```



Most people in this dataset starts their journey from Cary followed by some unknown location and then Morrisville.