Data Analysis/Data Visualization Report: Bike Ride Trends and Biker Types of Ford GoBike System April, 2019

Investigation overview

In this investigation, I will to look at the bike ride trends and biker type of the bay Area bike share system. The main focus was on biking duration, the time (weekday, hour), and the bike types.

Dataset Ocerview

This data set includes information about individual rides made in a bike-sharing system covering the greater San Francisco Bay area. The data consists of around 239k records for the trips

In [1]:

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

%matplotlib inline

```

In [2]:

```
df = pd.read_csv('clean_master_file.csv')
```

In [3]:

```
df.info()

```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 222660 entries, 0 to 222659
Data columns (total 27 columns):
duration_sec                 222660 non-null int64
start_time                   222660 non-null object
end_time                     222660 non-null object
start_station_id             222660 non-null float64
start_station_name           222660 non-null object
start_station_latitude       222660 non-null float64
start_station_longitude      222660 non-null float64
end_station_id               222660 non-null float64
end_station_name             222660 non-null object
end_station_latitude         222660 non-null float64
end_station_longitude        222660 non-null float64
bike_id                      222660 non-null int64
user_type                    222660 non-null object
member_birth_year            222660 non-null int64
member_gender                222660 non-null object
bike_share_for_all_trip      222660 non-null object
start_time_dayofweek         222660 non-null object
start_time_hour              222660 non-null int64
member_age                   222660 non-null int64
duration_min                 222660 non-null float64
log_duration_min             222660 non-null float64
start_month                  222660 non-null object
start_day                    222660 non-null object
start_hour                   222660 non-null int64
end_month                    222660 non-null object
end_day                      222660 non-null object
end_hour                     222660 non-null int64
dtypes: float64(8), int64(7), object(12)
memory usage: 45.9+ MB
```

In [4]:

```
# Convert the start_time_dayofweek .
weekdays = ['Mon','Tue','Wed','Thu','Fri', 'Sat', 'Sun']
ordered_weekdays = pd.api.types.CategoricalDtype(ordered = True, categories = we
df['start_time_dayofweek'] = df['start_time_dayofweek'].astype(ordered_weekdays
```

In [5]:

```python
df.info()

```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 222660 entries, 0 to 222659
Data columns (total 27 columns):
duration_sec                 222660 non-null int64
start_time                   222660 non-null object
end_time                     222660 non-null object
start_station_id             222660 non-null float64
start_station_name           222660 non-null object
start_station_latitude       222660 non-null float64
start_station_longitude      222660 non-null float64
end_station_id               222660 non-null float64
end_station_name             222660 non-null object
end_station_latitude         222660 non-null float64
end_station_longitude        222660 non-null float64
bike_id                      222660 non-null int64
user_type                    222660 non-null object
member_birth_year            222660 non-null int64
member_gender                222660 non-null object
bike_share_for_all_trip      222660 non-null object
start_time_dayofweek         222660 non-null category
start_time_hour              222660 non-null int64
member_age                   222660 non-null int64
duration_min                 222660 non-null float64
log_duration_min             222660 non-null float64
start_month                  222660 non-null object
start_day                    222660 non-null object
start_hour                   222660 non-null int64
end_month                    222660 non-null object
end_day                      222660 non-null object
end_hour                     222660 non-null int64
dtypes: category(1), float64(8), int64(7), object(11)
memory usage: 44.4+ MB
```

In [16]:

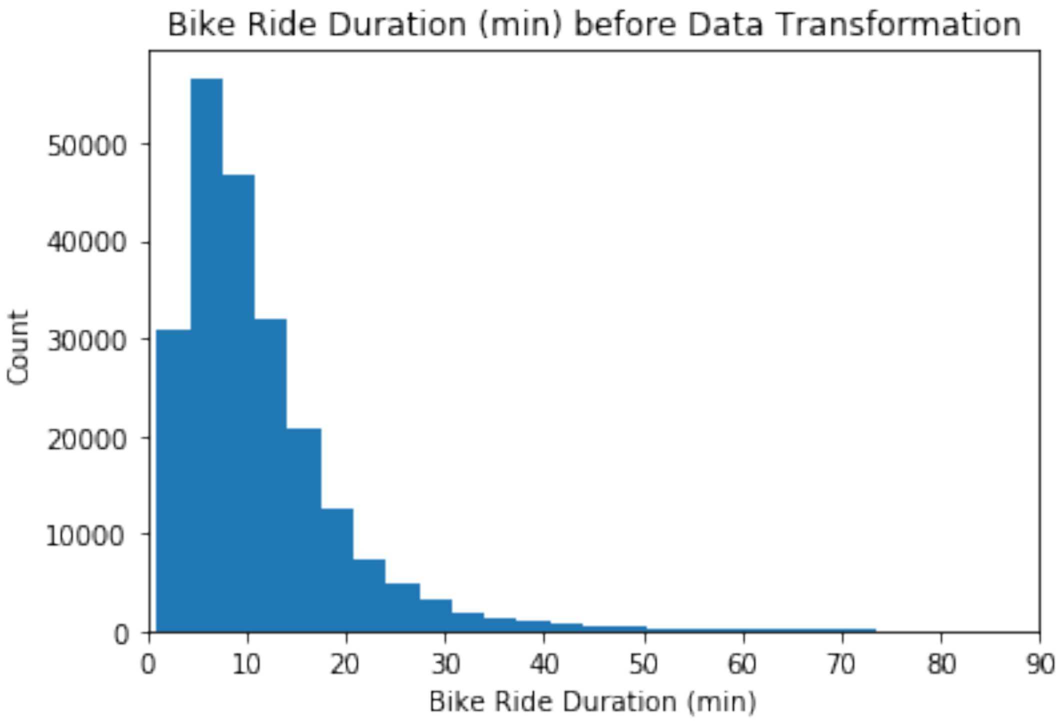
```python
default_color = sb.color_palette()[0]
```

Bike Ride Duration:

The origianl duration data has right skew issue - bike durations range from less than 1 minute to 1400+ minutes with median at around 9 min and mean at around 12 min. the following are the plots before/after data transformation

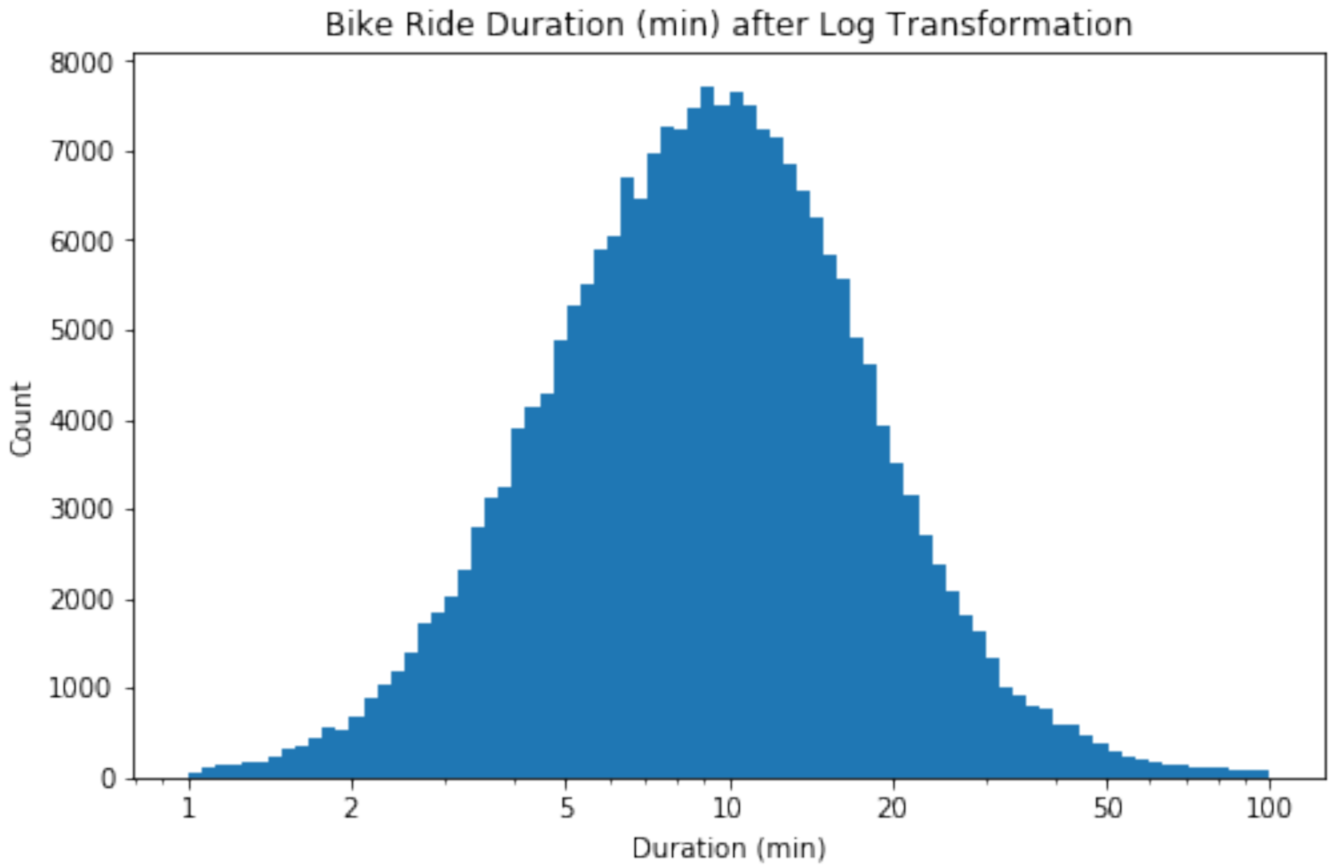
In [6]:

```
plt.hist(data = df, x = 'duration_min', bins = 30);
plt.xlim(0, 90);
plt.xlabel("Bike Ride Duration (min)");
plt.ylabel("Count");
plt.title("Bike Ride Duration (min) before Data Transformation");
```



In [7]:

```python
# there's a long tail in the distribution, so let's put it on a log scale inste
log_binsize = 0.025
bins = 10 ** np.arange(0, np.log10(df['duration_min'].max())+log_binsize, log_b

plt.figure(figsize=[8, 5]);
plt.hist(data = df, x = 'duration_min', bins = bins);
plt.xscale('log');
plt.xticks([1, 2, 5, 10, 20, 50, 100], [1, 2, 5, 10, 20, 50, 100]);
plt.xlabel('Duration (min)');
plt.ylabel('Count');
plt.title("Bike Ride Duration (min) after Log Transformation");
```



Bike ride Trends and Biker daily and weekly:

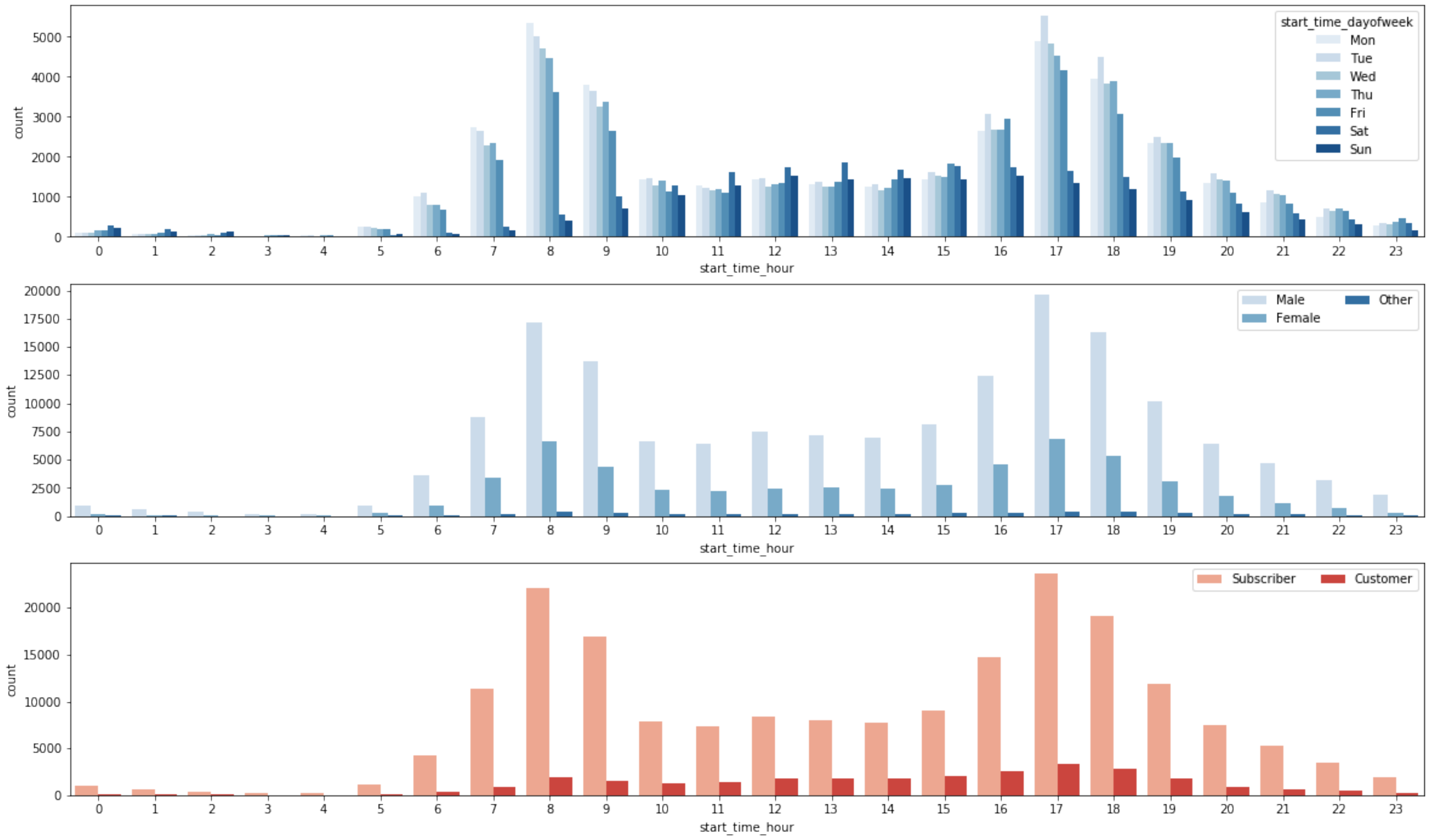
The investigating is including start time hour, start time dayofweek, member gender, and user type.

the visualizations is showing the following:

- Tuesday, 5:00 PM has the highest biker counts across 7 days, 24 hours.
- 5:00 PM has the most male bikers compared to other hours. 8:00 AM and 5:00 PM have more female bikers compared to other hours.
- 5:00 PM has the most 'Subscriber' bikers compared to other hours. It also has the most 'Customer' bikers compared to other hours.
- Tuesday has the most male bikers compared to other days. It also has the most female bikers compared to other days.
- Tuesday has the most 'Subscriber' bikers compared to other days. Saturday has the most 'Customer' bikers compared to other days.
- Most 'Subscriber' are male. Most 'Customer' bikers are also male.
- We can see that there is gradual decrease in the number of riders as the days of the week pass. We can also notice that Tuesday is the day having the highest number of riders for all the categories.

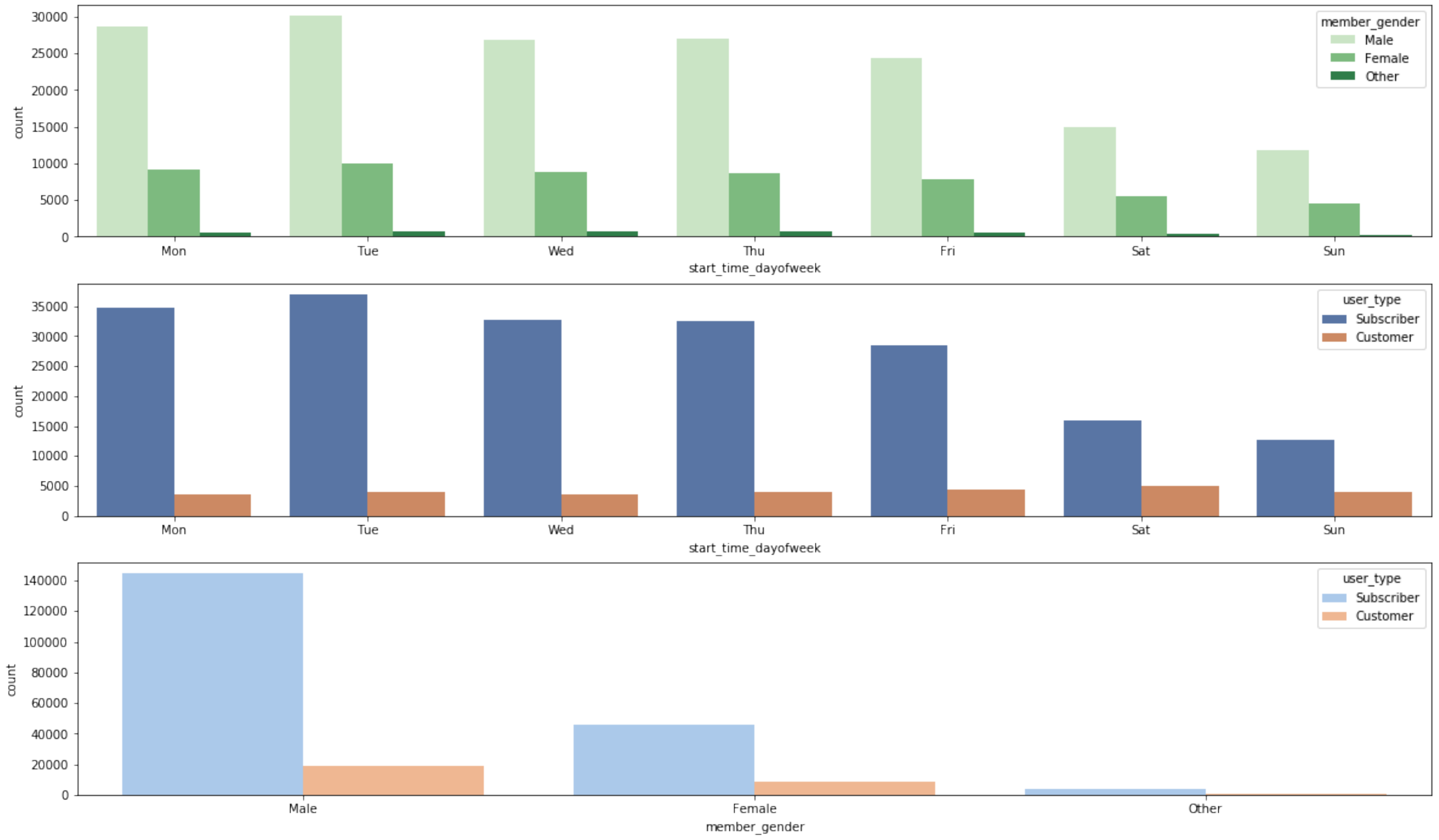
In [8]:

```
# since there's only three subplots to create, using the full data should be fi
plt.figure(figsize = [20, 12]);

plt.subplot(3, 1, 1);
sb.countplot(data = df, x = 'start_time_hour', hue = 'start_time_dayofweek', pal

ax = plt.subplot(3, 1, 2);
sb.countplot(data = df, x = 'start_time_hour', hue = 'member_gender', palette =
ax.legend(ncol = 2); # re-arrange legend to reduce overlapping


ax = plt.subplot(3, 1, 3);
sb.countplot(data = df, x = 'start_time_hour', hue = 'user_type', palette = 'Red
ax.legend(loc = 1, ncol = 2); # re-arrange legend to remove overlapping
```



In [9]:

```
plt.figure(figsize = [20, 12]);

ax = plt.subplot(3, 1, 1)
sb.countplot(data = df, x = 'start_time_dayofweek', hue = 'member_gender', palet

ax = plt.subplot(3, 1, 2);
sb.countplot(data = df, x = 'start_time_dayofweek', hue = 'user_type', palette =

ax = plt.subplot(3, 1, 3);

sb.countplot(data = df, x = 'member_gender', hue = 'user_type', palette = 'paste

```



Where most common start stations for ride

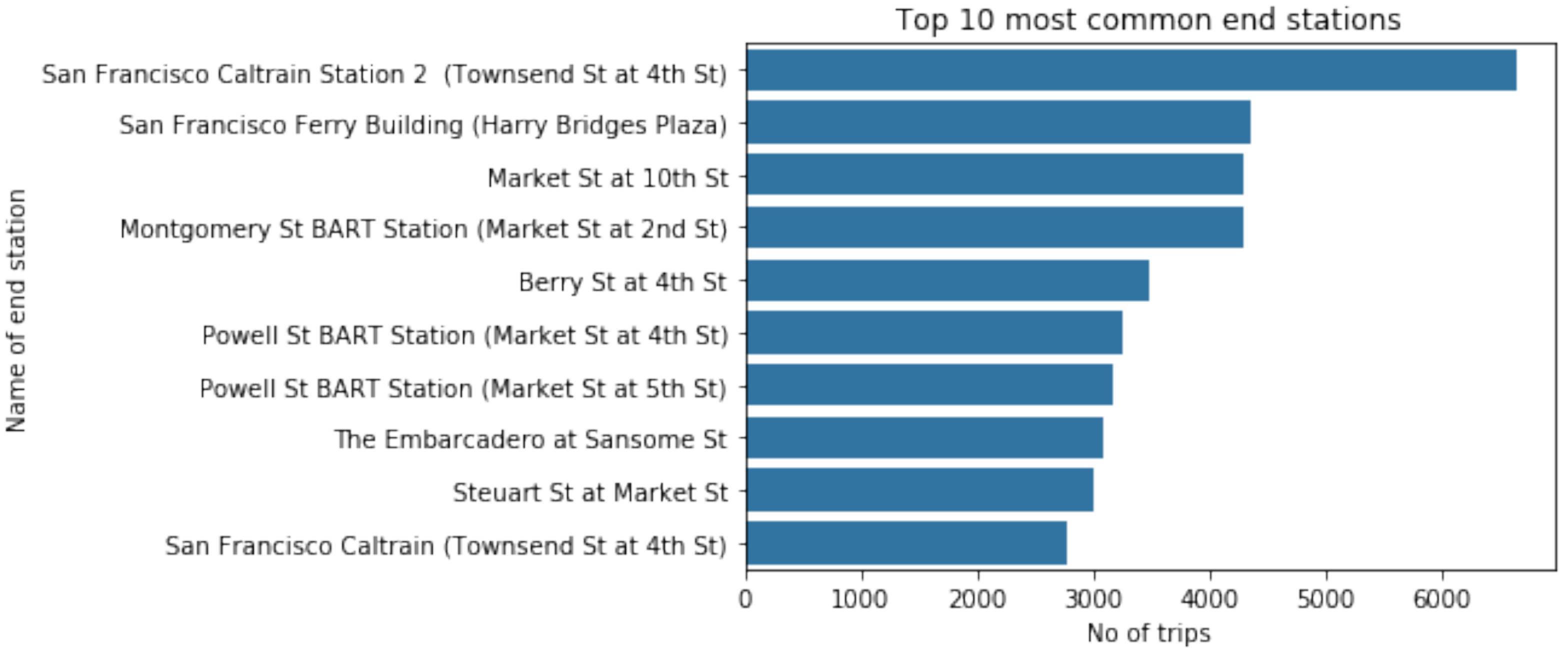
The purpose of this graph below to know where is the most trip start

Observation

we can see most of trips start from san Francisco caltrain station 2(Townsend st at 4th st)

In [17]:

```
neighbourhood_counts = df['end_station_name'].value_counts()
neighbourhood_order = neighbourhood_counts.index
sb.countplot(data = df, y = 'end_station_name', order = neighbourhood_order[:10
plt.xlabel('No of trips')
plt.ylabel('Name of end station')
plt.title('Top 10 most common end stations');
```

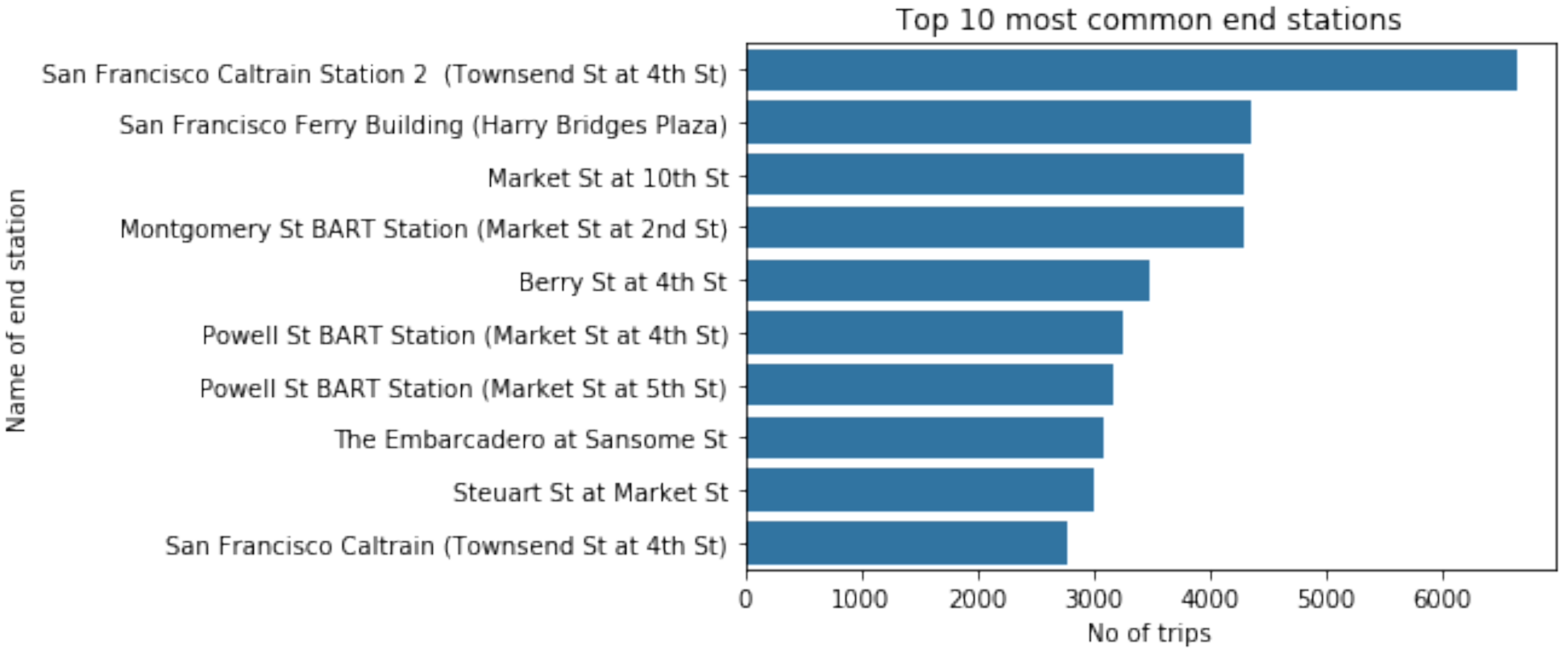


finding out where most trips end

based on the graph below we can see most end station is San Franciso caltrain Station 2(Townsend st at 4th st)

In [23]:

```
end_station = df['end_station_name'].value_counts()
neighbourhood_order = end_station.index
sb.countplot(data = df, y = 'end_station_name', order = neighbourhood_order[:10
plt.xlabel('No of trips')
plt.ylabel('Name of end station')
plt.title('Top 10 most common end stations');
```

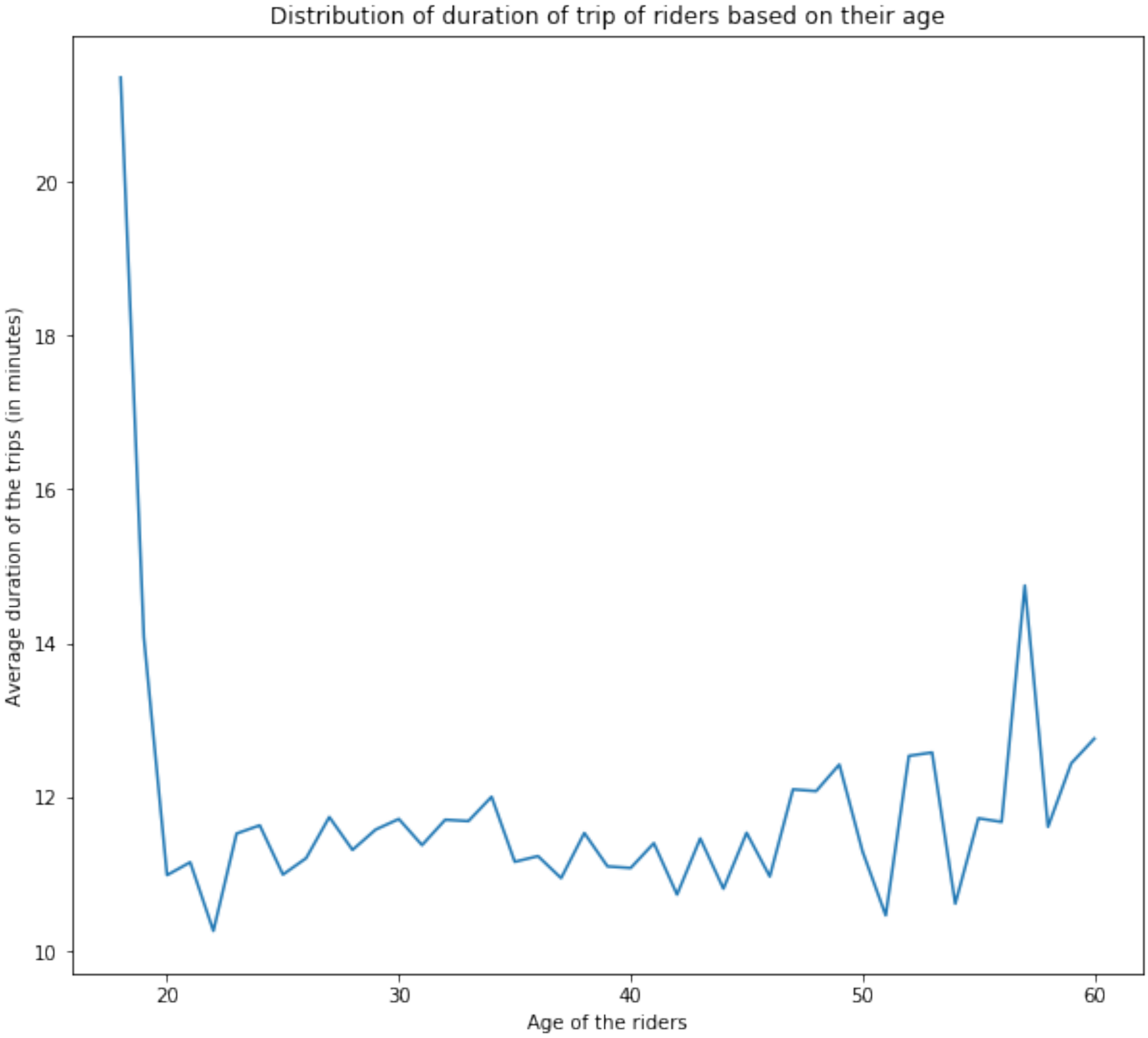


identifying the average duration of the road trip as travelled by people age

- people who ride the bike are between 20-60 ages
- most ages who ride the bike are between 18-20
- there is no one is older than 60 is using bike in Ford GoBike

In [26]:

```
plt.figure(figsize=(10,9))
sb.lineplot(data=df[df['member_age']<100], x='member_age', y='duration_min', er
plt.xlabel('Age of the riders')
plt.ylabel('Average duration of the trips (in minutes)')
plt.title('Distribution of duration of trip of riders based on their age');
```

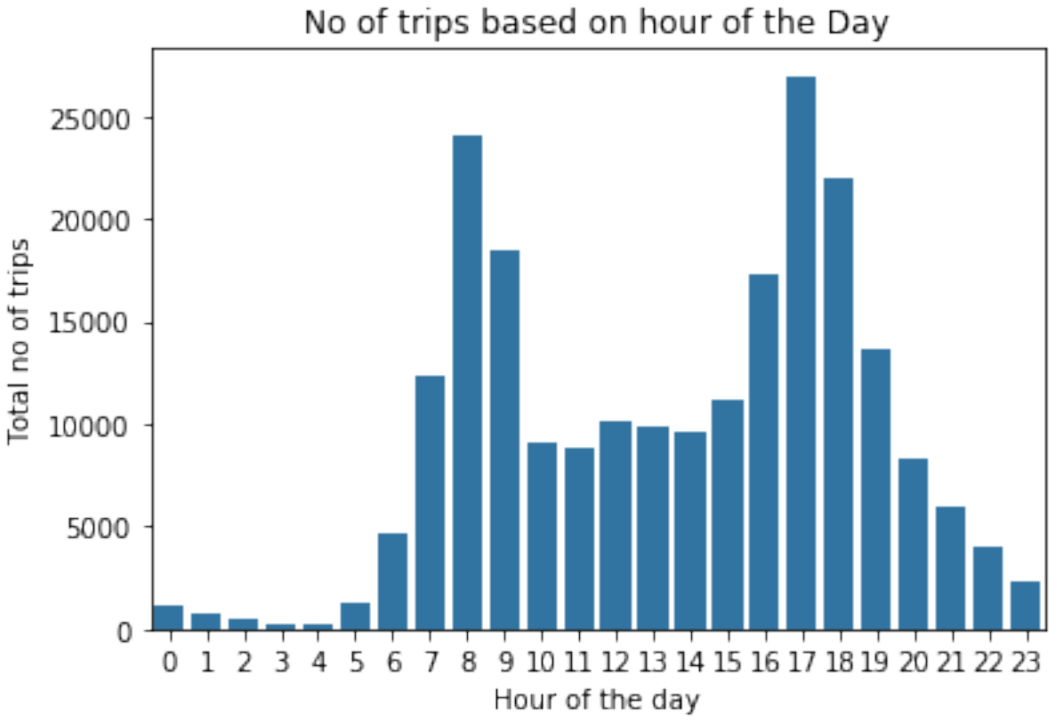


start time of trips

- as we can see below, most of the trips start at 5pm
- second most trips start at 8 am

In [27]:

```
df.groupby('start_hour').count()
sb.countplot(data=df, x='start_hour', color=default_color);
plt.xlabel('Hour of the day')
plt.ylabel('Total no of trips')
plt.title('No of trips based on hour of the Day');
```



In [29]:

```
!jupyter nbconvert 'slide.ipynb' --to slides --post serve --template output_tog

```

```
[NbConvertApp] Converting notebook slide.ipynb to slides
Traceback (most recent call last):
  File "/Users/hamedbintalib/anaconda3/bin/jupyter-nbconvert", line 11
, in <module>
    sys.exit(main())
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/jup
yter_core/application.py", line 266, in launch_instance
    return super(JupyterApp, cls).launch_instance(argv=argv, **kwargs)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/tra
itlets/config/application.py", line 658, in launch_instance
    app.start()
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/nbconvertapp.py", line 337, in start
    self.convert_notebooks()
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/nbconvertapp.py", line 507, in convert_notebooks
    self.convert_single_notebook(notebook_filename)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/nbconvertapp.py", line 478, in convert_single_notebook
    output, resources = self.export_single_notebook(notebook_filename,
resources, input_buffer=input_buffer)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/nbconvertapp.py", line 407, in export_single_notebook
```

```
    output, resources = self.exporter.from_filename(notebook_filename,
resources=resources)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/exporters/exporter.py", line 178, in from_filename
    return self.from_file(f, resources=resources, **kw)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/exporters/exporter.py", line 196, in from_file
    return self.from_notebook_node(nbformat.read(file_stream, as_versi
on=4), resources=resources, **kw)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/exporters/slides.py", line 183, in from_notebook_node
    return super(SlidesExporter, self).from_notebook_node(nb, resource
s=resources, **kw)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/exporters/html.py", line 96, in from_notebook_node
    output, resources = super(HTMLExporter, self).from_notebook_node(n
b, resources, **kw)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/exporters/templateexporter.py", line 315, in from_notebook_node
    output = self.template.render(nb=nb_copy, resources=resources)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/exporters/templateexporter.py", line 113, in template
    self._template_cached = self._load_template()
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/exporters/templateexporter.py", line 286, in _load_template
    return self.environment.get_template(template_file)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/jin
ja2/environment.py", line 830, in get_template
    return self._load_template(name, self.make_globals(globals))
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/jin
ja2/environment.py", line 804, in _load_template
    template = self.loader.load(self, name, globals)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/jin
ja2/loaders.py", line 408, in load
    raise TemplateNotFound(name)
jinja2.exceptions.TemplateNotFound: output_toggle
```

In [ ]:

```

```

Data Analysis/Data Visualization Report: Bike Ride Trends and Biker Types of Ford GoBike System April, 2019

Investigation overview

In this investigation, I will to look at the bike ride trends and biker type of the bay Area bike share system. The main focus was on biking duration, the time (weekday, hour), and the bike types.

Dataset Ocerview

This data set includes information about individual rides made in a bike-sharing system covering the greater San Francisco Bay area. The data consists of around 239k records for the trips

In [1]:

In [2]:

In [3]:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 222660 entries, 0 to 222659
Data columns (total 27 columns):
duration_sec               222660 non-null int64
start_time                 222660 non-null object
end_time                   222660 non-null object
start_station_id           222660 non-null float64
start_station_name         222660 non-null object
start_station_latitude     222660 non-null float64
start_station_longitude    222660 non-null float64
end_station_id             222660 non-null float64
end_station_name           222660 non-null object
end_station_latitude       222660 non-null float64
end_station_longitude      222660 non-null float64
bike_id                    222660 non-null int64
user_type                  222660 non-null object
member_birth_year          222660 non-null int64
member_gender              222660 non-null object
bike_share_for_all_trip    222660 non-null object
```

In [4]:

In [5]:

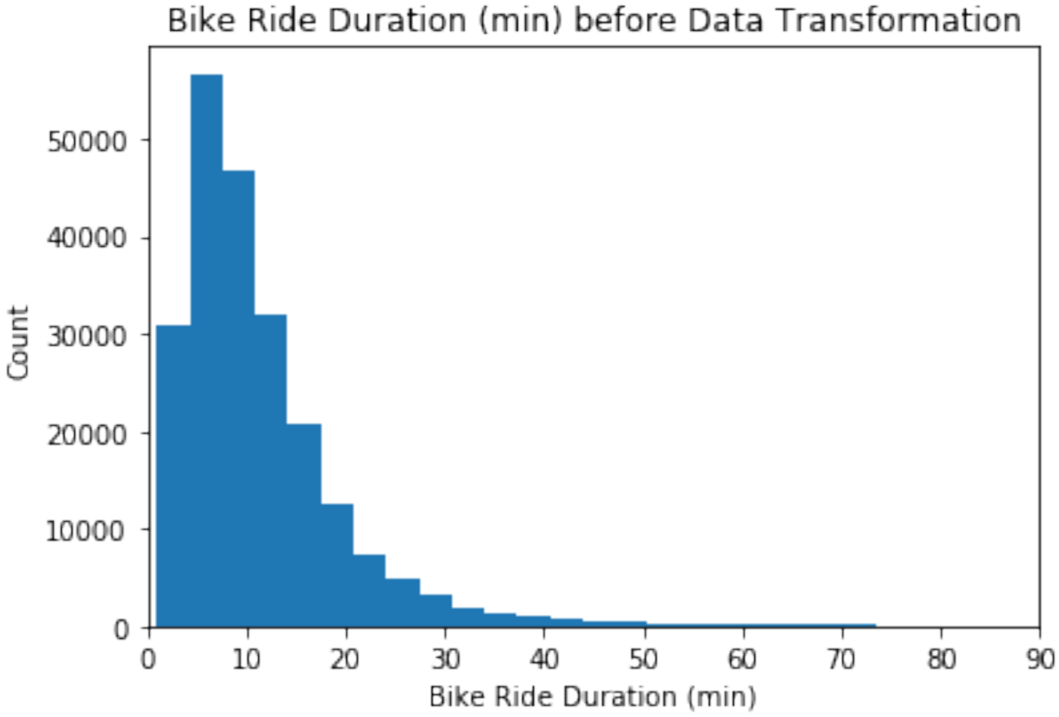
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 222660 entries, 0 to 222659
Data columns (total 27 columns):
duration_sec                 222660 non-null int64
start_time                   222660 non-null object
end_time                     222660 non-null object
start_station_id             222660 non-null float64
start_station_name           222660 non-null object
start_station_latitude       222660 non-null float64
start_station_longitude      222660 non-null float64
end_station_id               222660 non-null float64
end_station_name             222660 non-null object
end_station_latitude         222660 non-null float64
end_station_longitude        222660 non-null float64
bike_id                      222660 non-null int64
user_type                    222660 non-null object
member_birth_year            222660 non-null int64
member_gender                222660 non-null object
bike_share_for_all_trip      222660 non-null object
```

In [16]:

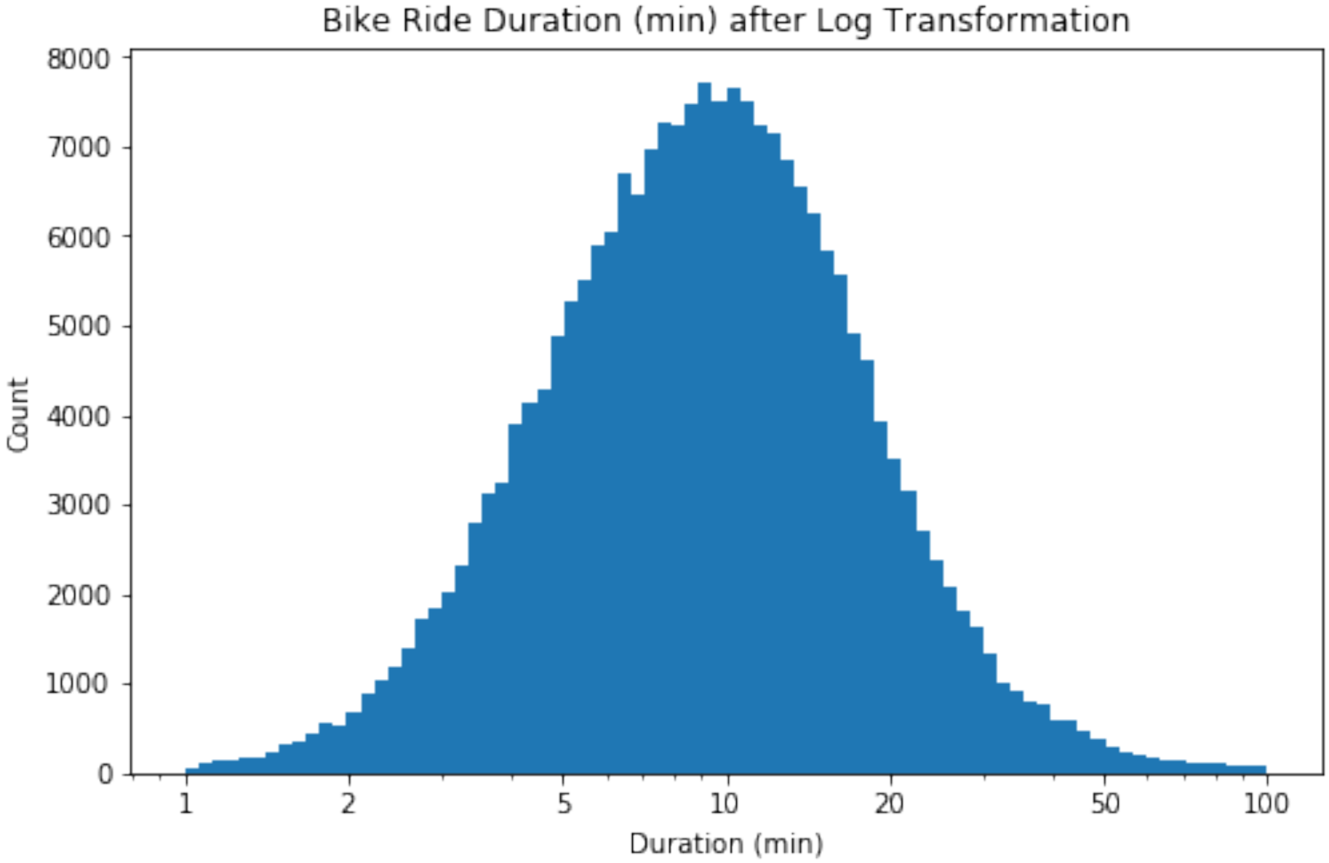
Bike Ride Duration:

The origianl duration data has right skew issue - bike durations range from less than 1 minute to 1400+ minutes with median at around 9 min and mean at around 12 min. the following are the plots before/after data transformation

In [6]:


Bike Ride Duration (min) before Data Transformation
Count
0
10000
20000
30000
40000
50000
0
10
20
30
40
50
60
70
80
90
Bike Ride Duration (min)

In [7]:


Bike Ride Duration (min) after Log Transformation
Count
0
1000
2000
3000
4000
5000
6000
7000
8000
1
2
5
10
20
50
100
Duration (min)

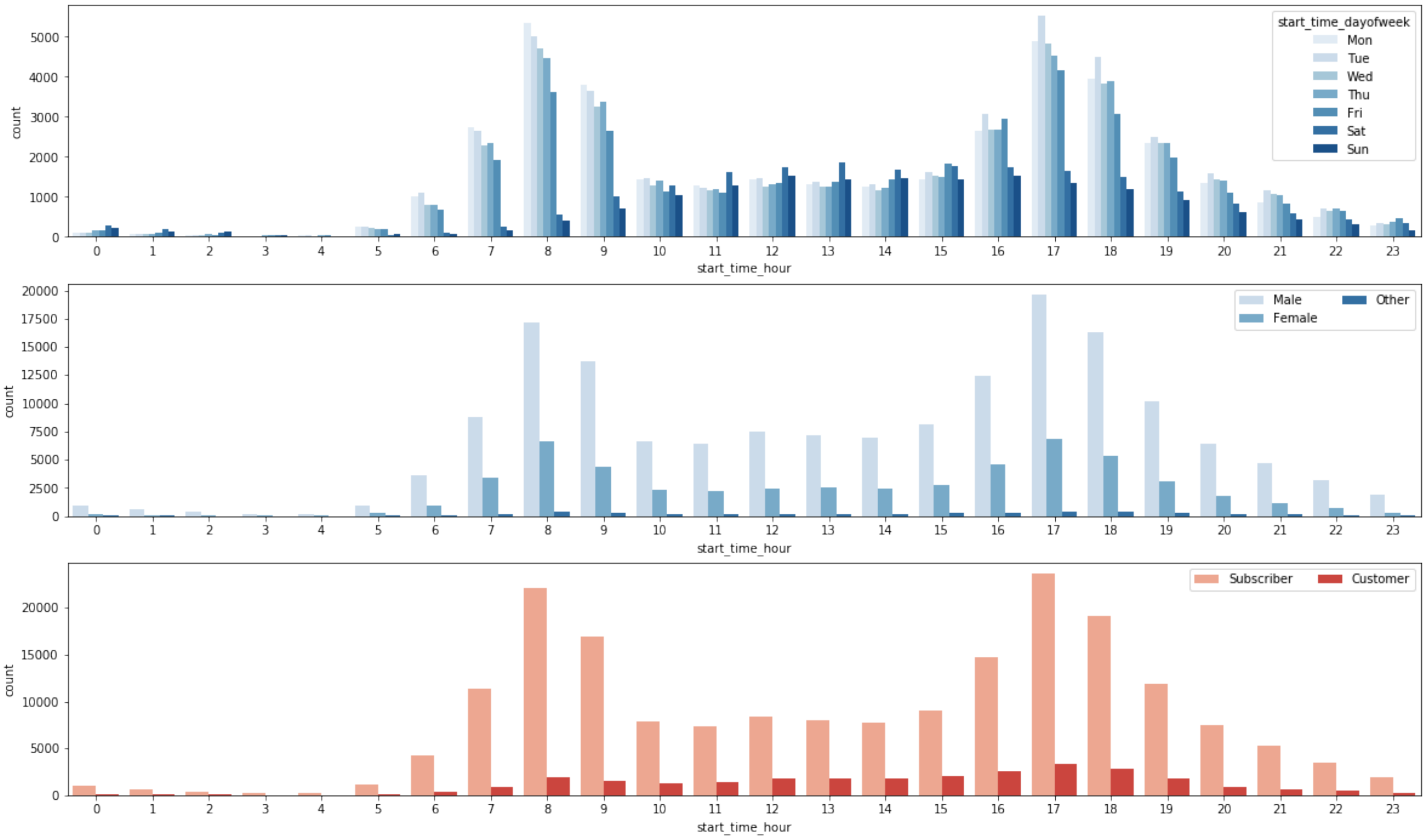
Bike ride Trends and Biker daily and weekly:

The investigating is including start time hour, start time dayofweek, member gender, and user type.

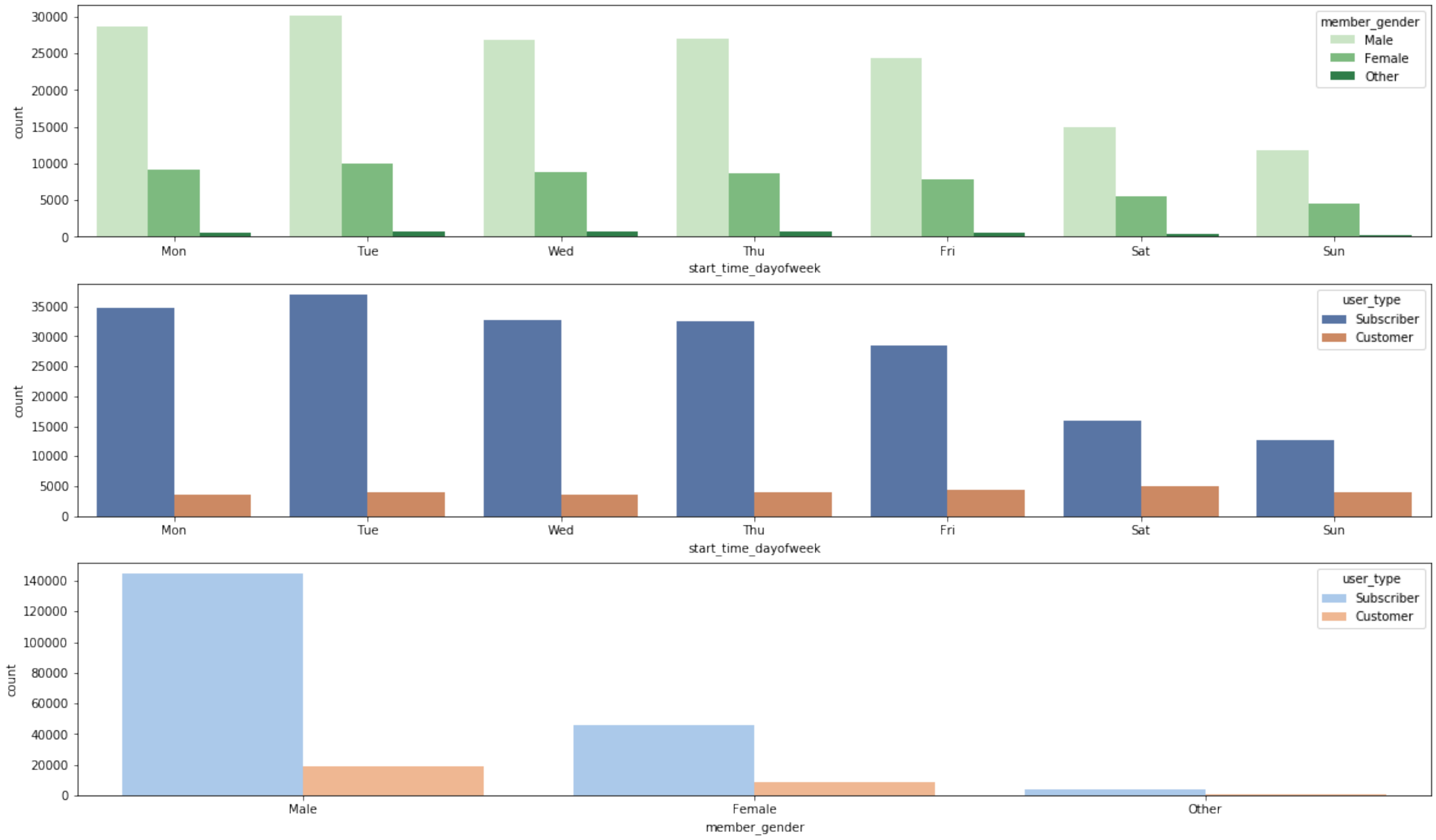
the visualizations is showing the following:

- Tuesday, 5:00 PM has the highest biker counts across 7 days, 24 hours.
- 5:00 PM has the most male bikers compared to other hours. 8:00 AM and 5:00 PM have more female bikers compared to other hours.
- 5:00 PM has the most 'Subscriber' bikers compared to other hours. It also has the most 'Customer' bikers compared to other hours.
- Tuesday has the most male bikers compared to other days. It also has the most female bikers compared to other days.
- Tuesday has the most 'Subscriber' bikers compared to other days. Saturday has the most 'Customer' bikers compared to other days.
- Most 'Subscriber' are male. Most 'Customer' bikers are also male.
- We can see that there is gradual decrease in the number of riders as the days of the week pass. We can also notice that Tuesday is the day having the highest number of riders for all the categories.

In [8]:



In [9]:



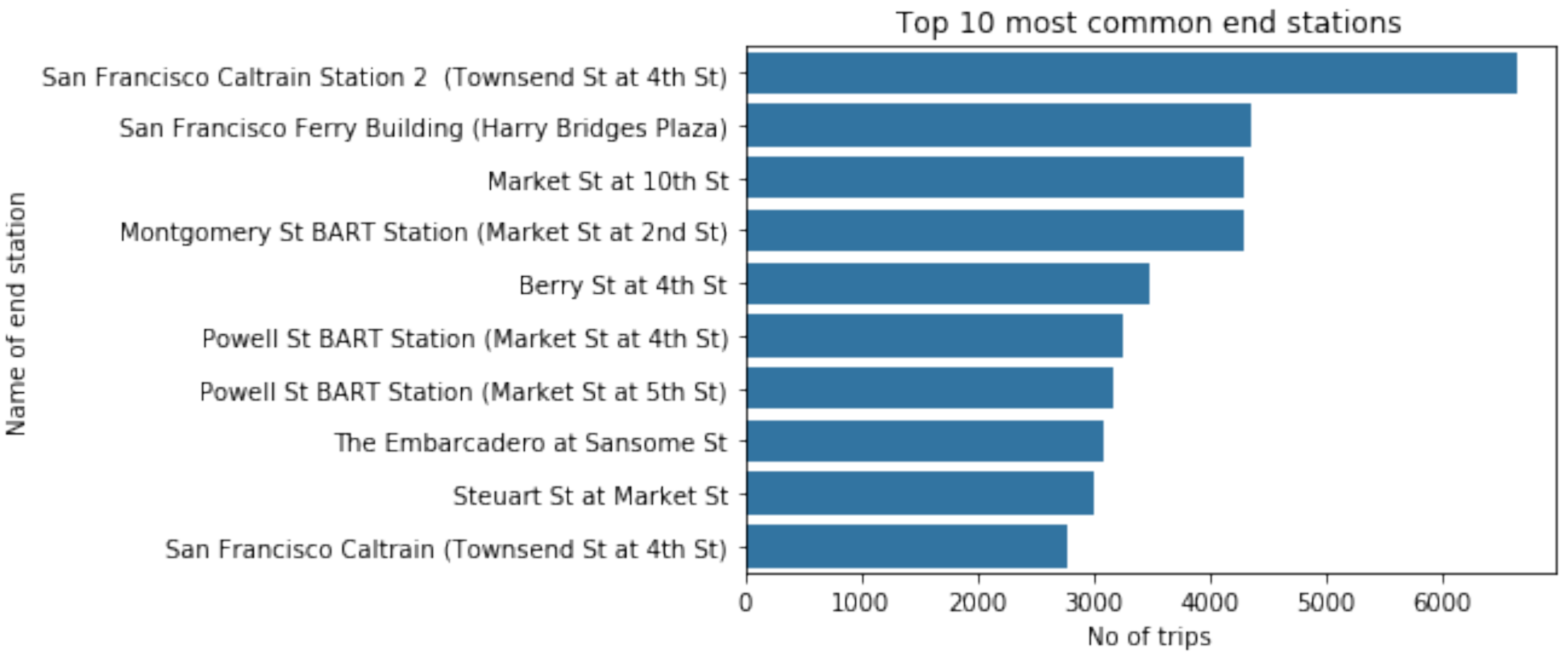
Where most common start stations for ride

The purpose of this graph below to know where is the most trip start

Observation

we can see most of trips start from san Francisco caltrain station 2(Townsend st at 4th st)

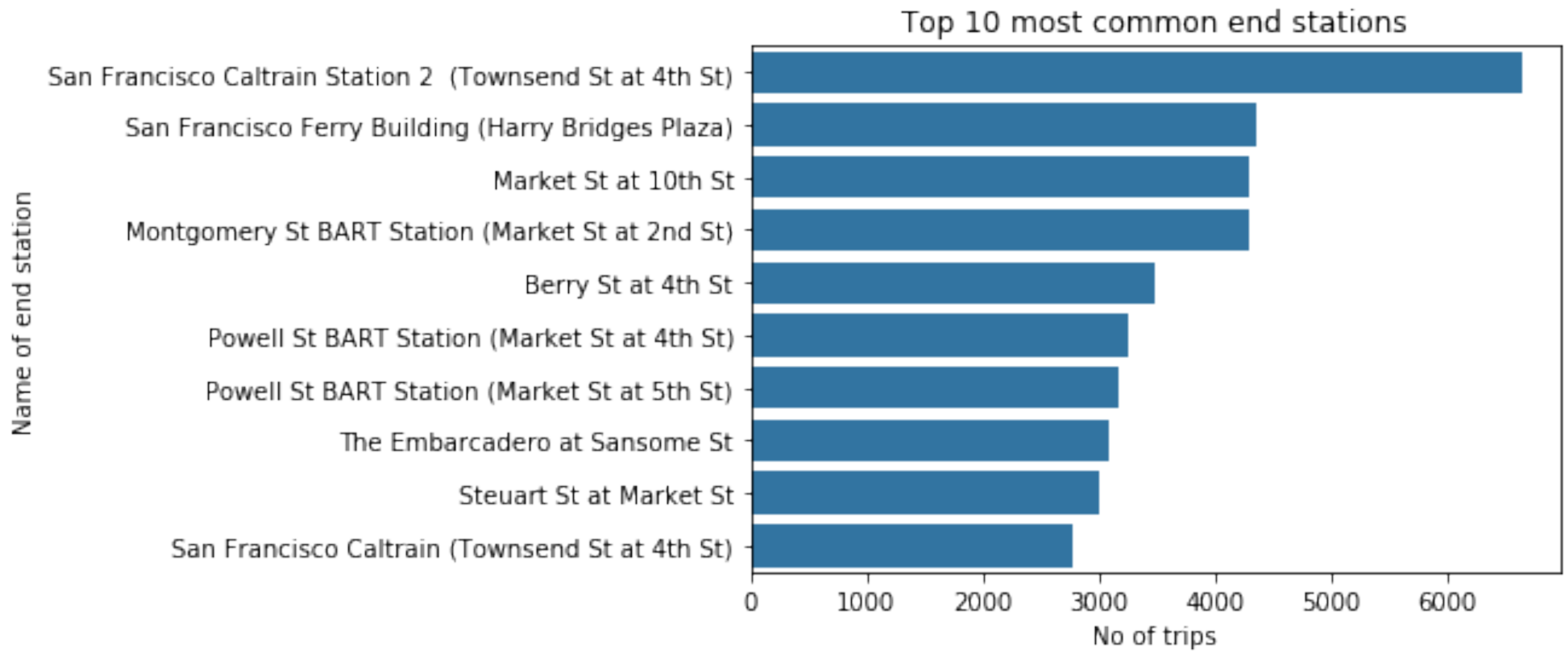
In [17]:



finding out where most trips end

based on the graph below we can see most end station is San Franciso caltrain Station 2(Townsend st at 4th st)

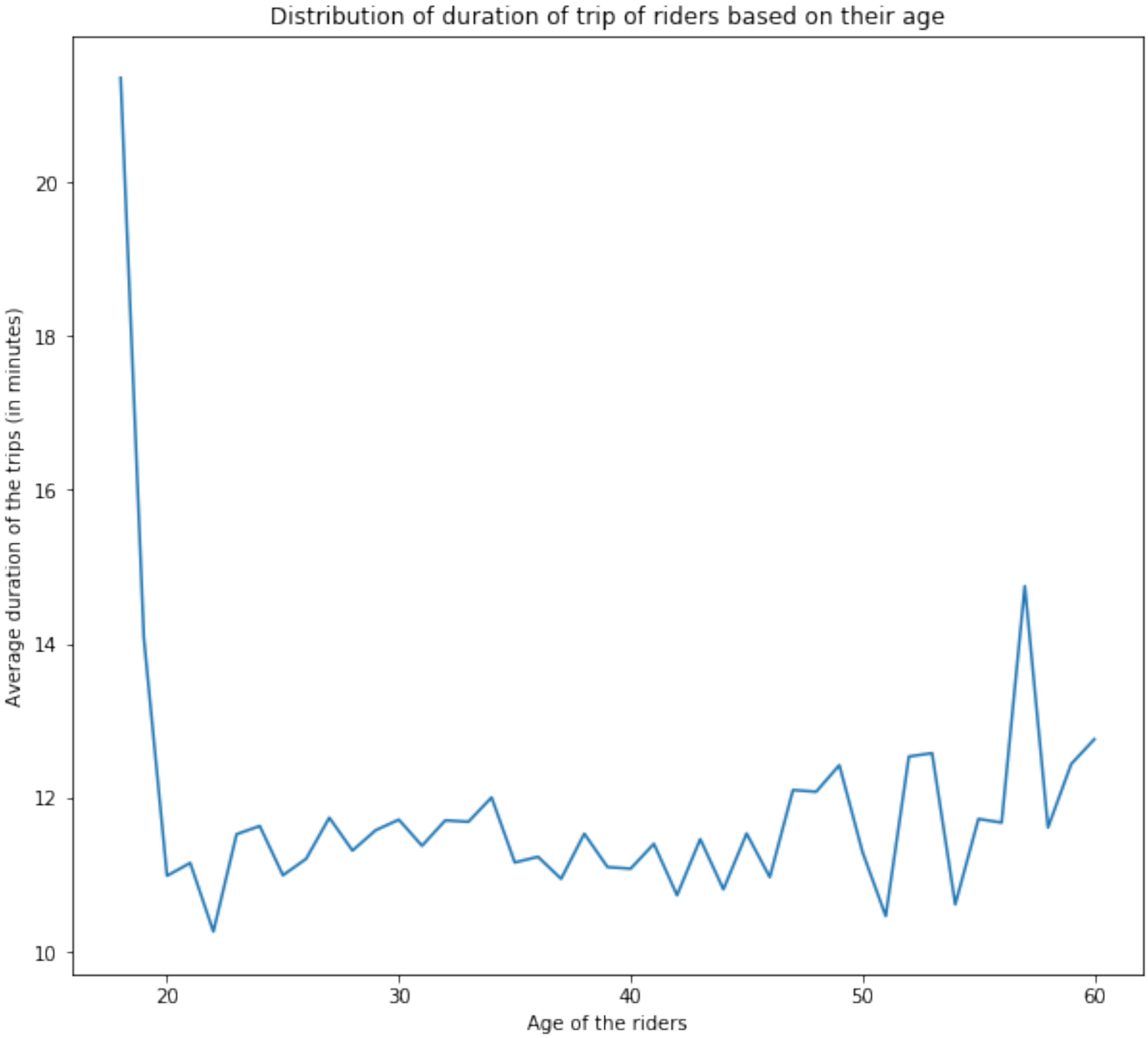
```
In [23]:
```



identifying the average duration of the road trip as travelled by people age

- people who ride the bike are between 20-60 ages
- most ages who ride the bike are between 18-20
- there is no one is older than 60 is using bike in Ford GoBike

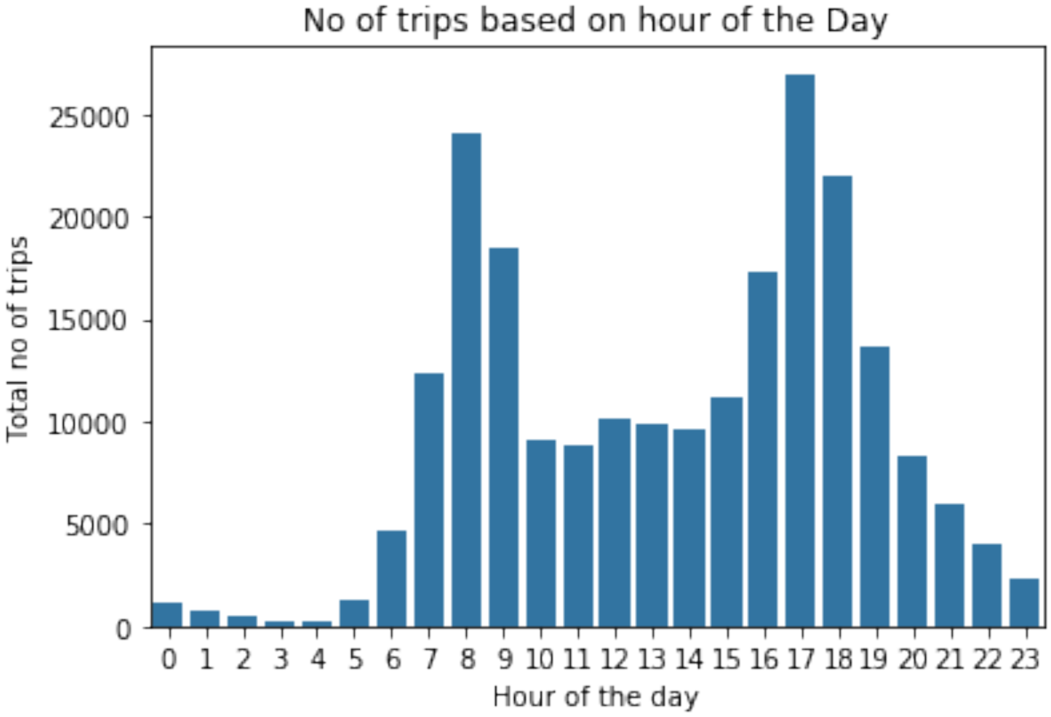
```
In [26]:
```



start time of trips

- as we can see below, most of the trips start at 5pm
- second most trips start at 8 am

In [27]:



In [29]:

```
[NbConvertApp] Converting notebook slide.ipynb to slides
Traceback (most recent call last):
  File "/Users/hamedbintalib/anaconda3/bin/jupyter-nbconvert", line 11
, in <module>
    sys.exit(main())
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/jup
yter_core/application.py", line 266, in launch_instance
    return super(JupyterApp, cls).launch_instance(argv=argv, **kwargs)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/tra
itlets/config/application.py", line 658, in launch_instance
    app.start()
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/nbconvertapp.py", line 337, in start
    self.convert_notebooks()
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/nbconvertapp.py", line 507, in convert_notebooks
    self.convert_single_notebook(notebook_filename)
  File "/Users/hamedbintalib/anaconda3/lib/python3.7/site-packages/nbc
onvert/nbconvertapp.py", line 478, in convert_single_notebook
```

In [ ]: