Exploratory Data Analysis on NYC Taxi Trip Duration Dataset

Importing necessary libraries

In [ ]:

```
import pandas as pd       #data processing
import numpy as np        #linear algebra
```

In [3]:

```
#data visualisation
import seaborn as sns
sns.set()
import matplotlib.pyplot as plt
%matplotlib inline
```

In [4]:

```
import datetime as dt
```

In [5]:

```
import warnings; warnings.simplefilter('ignore')
```

Importing the Dataset

In [6]:

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

Exploring the dataset

In [7]:

```
data.shape
```

Out[7]:

```
(729322, 11)
```

In [8]:

```
data.columns
```

Out[8]:

```
Index(['id', 'vendor_id', 'pickup_datetime', 'dropoff_datetime',
       'passenger_count', 'pickup_longitude', 'pickup_latitude',
       'dropoff_longitude', 'dropoff_latitude', 'store_and_fwd_flag',
       'trip_duration'],
      dtype='object')
```

In [9]:

```
data.dtypes
```

Out[9]:

```
id                     object
vendor_id               int64
pickup_datetime        object
dropoff_datetime       object
passenger_count         int64
pickup_longitude      float64
pickup_latitude       float64
dropoff_longitude     float64
dropoff_latitude      float64
store_and_fwd_flag     object
trip_duration           int64
dtype: object
```

In [10]:

```
data.head()
```

Out[10]:

	id	vendor_id	pickup_datetime	dropoff_datetime	passenger_count	pickup_longitude
0	id1080784	2	2016-02-29 16:40:21	2016-02-29 16:47:01	1	-73.953918
1	id0889885	1	2016-03-11 23:35:37	2016-03-11 23:53:57	2	-73.988312
2	id0857912	2	2016-02-21 17:59:33	2016-02-21 18:26:48	2	-73.997314
3	id3744273	2	2016-01-05 09:44:31	2016-01-05 10:03:32	6	-73.961670
4	id0232939	1	2016-02-17 06:42:23	2016-02-17 06:56:31	1	-74.017120

In [11]:

```
data.isnull().sum()
```

Out[11]:

```
id                    0
vendor_id             0
pickup_datetime       0
dropoff_datetime      0
passenger_count       0
pickup_longitude      0
pickup_latitude       0
dropoff_longitude     0
dropoff_latitude      0
store_and_fwd_flag    0
trip_duration         0
dtype: int64
```

In [12]:

```
data.nunique()
```

Out[12]:

```
id                    729322
vendor_id                  2
pickup_datetime       709359
dropoff_datetime      709308
passenger_count            9
pickup_longitude       19729
pickup_latitude        39776
dropoff_longitude      27892
dropoff_latitude       53579
store_and_fwd_flag         2
trip_duration           6296
dtype: int64
```

In [13]:

```
data.describe()
```

Out[13]:

	vendor_id	passenger_count	pickup_longitude	pickup_latitude	dropoff_longitude	d
count	729322.000000	729322.000000	729322.000000	729322.000000	729322.000000	
mean	1.535403	1.662055	-73.973513	40.750919	-73.973422	
std	0.498745	1.312446	0.069754	0.033594	0.069588	
min	1.000000	0.000000	-121.933342	34.712234	-121.933304	
25%	1.000000	1.000000	-73.991859	40.737335	-73.991318	
50%	2.000000	1.000000	-73.981758	40.754070	-73.979759	
75%	2.000000	2.000000	-73.967361	40.768314	-73.963036	
max	2.000000	9.000000	-65.897385	51.881084	-65.897385	

Feature Creation

In [14]:

```
data['pickup_datetime']=pd.to_datetime(data['pickup_datetime'])
data['dropoff_datetime']=pd.to_datetime(data['dropoff_datetime'])
```

In [15]:

```
data['pickup_day']=data['pickup_datetime'].dt.day_name()
data['dropoff_day']=data['dropoff_datetime'].dt.day_name()
```

In [16]:

```
data['pickup_day_no']=data['pickup_datetime'].dt.weekday
data['dropoff_day_no']=data['dropoff_datetime'].dt.weekday
```

In [17]:

```
data['pickup_hour']=data['pickup_datetime'].dt.hour
data['dropoff_hour']=data['dropoff_datetime'].dt.hour
```

In [18]:

```
data['pickup_month']=data['pickup_datetime'].dt.month
data['dropoff_month']=data['dropoff_datetime'].dt.month
```

In [19]:

```
def time_of_day(x):
    if x in range(6,12):
        return 'Morning'
    elif x in range(12,16):
        return 'Afternoon'
    elif x in range(16,22):
        return 'Evening'
    else:
        return 'Late night'
```

In [20]:

```
data['pickup_timeofday']=data['pickup_hour'].apply(time_of_day)
data['dropoff_timeofday']=data['dropoff_hour'].apply(time_of_day)
```

In [21]:

```
from geopy.distance import great_circle
```

In [22]:

```
def cal_distance(pickup_lat,pickup_long,dropoff_lat,dropoff_long):
    start_coordinates=(pickup_lat,pickup_long)
    stop_coordinates=(dropoff_lat,dropoff_long)
    return great_circle(start_coordinates,stop_coordinates).km
```

In [23]:

```
data['distance'] = data.apply(lambda x: cal_distance(x['pickup_latitude'],x['pickup_longitu
```

Univariate Analysis

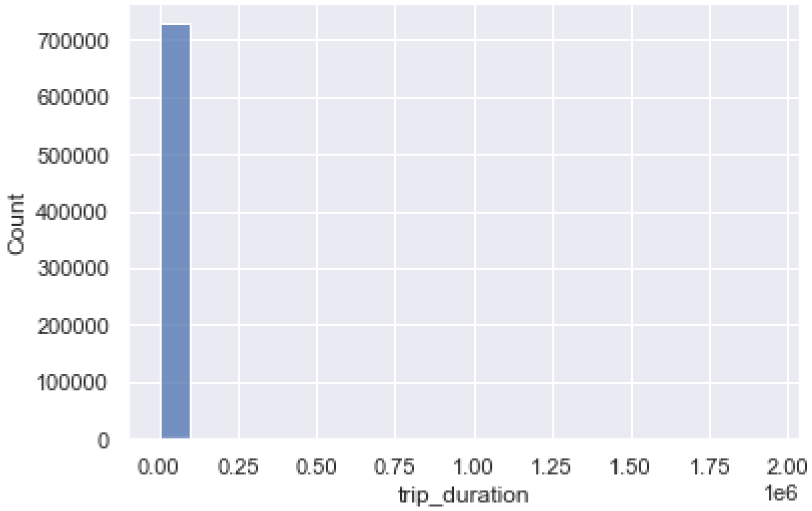
Target Variable

In [24]:

```
sns.histplot(data['trip_duration'],kde=False,bins=20)
```

Out[24]:

```
<AxesSubplot:xlabel='trip_duration', ylabel='Count'>
```

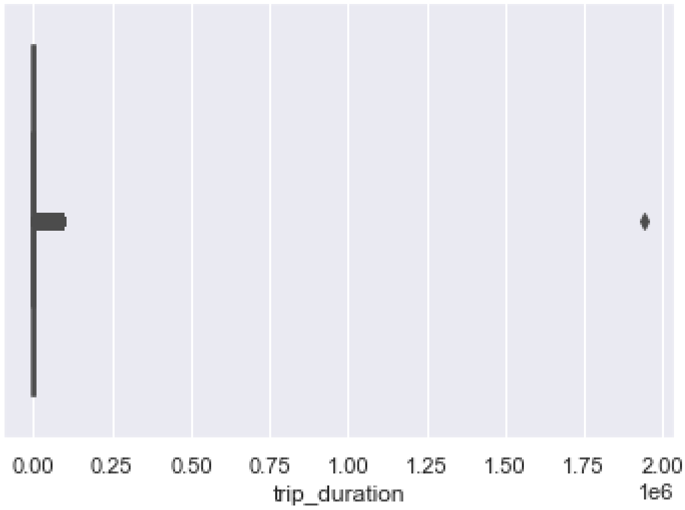


In [25]:

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

Out[25]:

```
<AxesSubplot:xlabel='trip_duration'>
```



In [26]:

```
data['trip_duration'].sort_values(ascending=False)
```

Out[26]:

```
21813     1939736
259437      86391
119185      86387
177225      86378
496391      86377
           ...   
672240          1
102646          1
533760          1
512833          1
622664          1
Name: trip_duration, Length: 729322, dtype: int64
```

In [27]:

```
data.drop(data[data['trip_duration'] == 1939736].index, inplace = True)
```

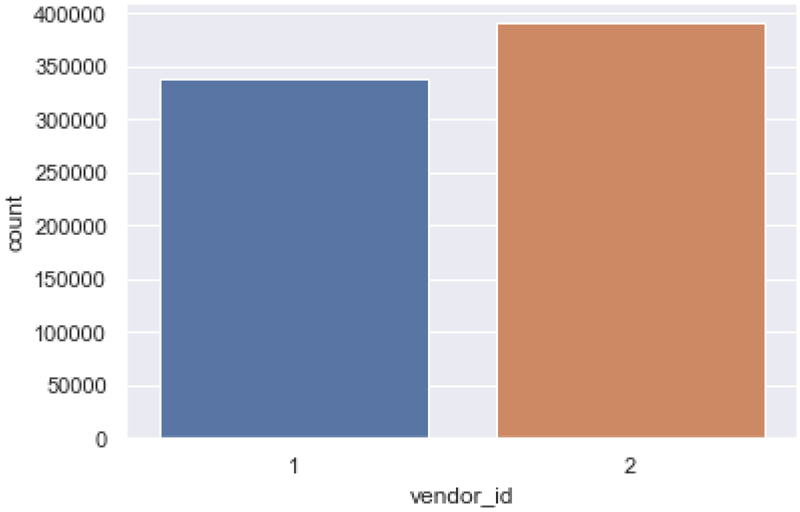
Vendor id

In [28]:

```
sns.countplot(x='vendor_id',data=data)
```

Out[28]:

```
<AxesSubplot:xlabel='vendor_id', ylabel='count'>
```



Passenger Count

In [29]:

```
data.passenger_count.value_counts()
```

Out[29]:

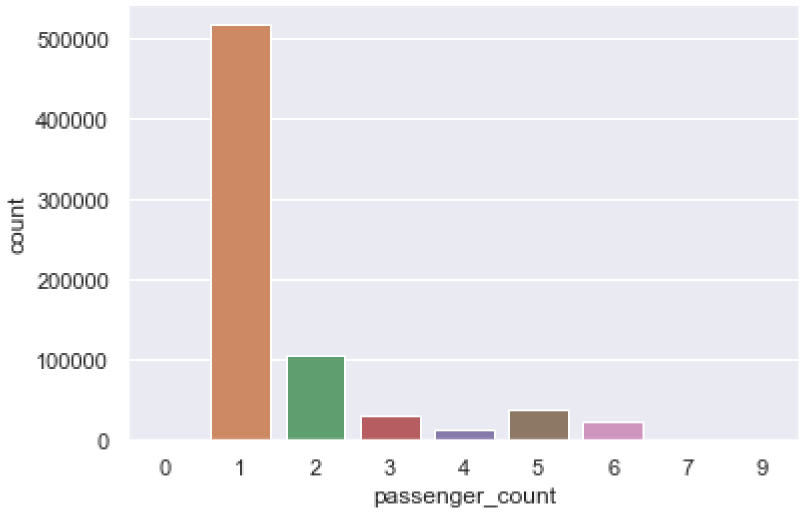
```
1    517414
2    105097
5     38926
3     29692
6     24107
4     14050
0        33
7         1
9         1
Name: passenger_count, dtype: int64
```

In [30]:

```
sns.countplot(x='passenger_count',data=data)
```

Out[30]:

```
<AxesSubplot:xlabel='passenger_count', ylabel='count'>
```



In [31]:

```
data=data[data['passenger_count']!=0]
data=data[data['passenger_count']<=6]
```

Store and Forward Flag

In [32]:

```
data['store_and_fwd_flag'].value_counts(normalize=True)
```

Out[32]:

```
N    0.994463
Y    0.005537
Name: store_and_fwd_flag, dtype: float64
```

Distance

In [33]:

```
data['distance'].value_counts()
```

Out[33]:

```
0.000000    2893
0.000424      20
0.000424      19
0.000424      16
0.000424      11
            ... 
0.643029       1
1.804800       1
0.358108       1
0.809034       1
2.246576       1
Name: distance, Length: 726217, dtype: int64
```

In [ ]:

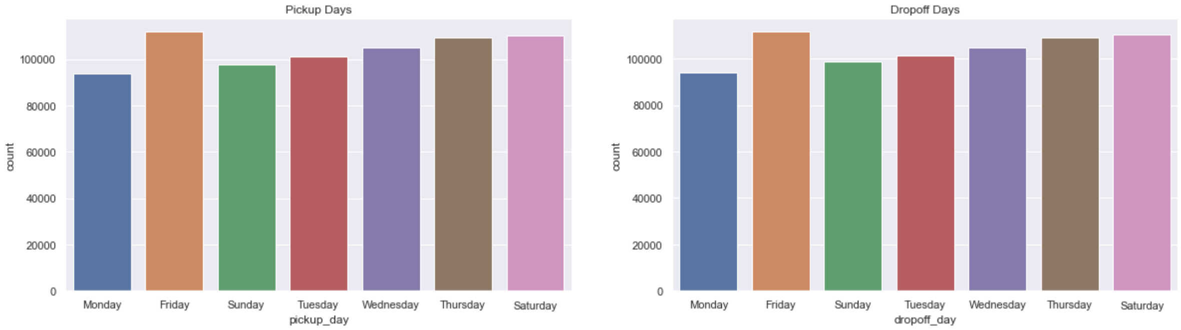
```

```

Trips per Day

In [37]:

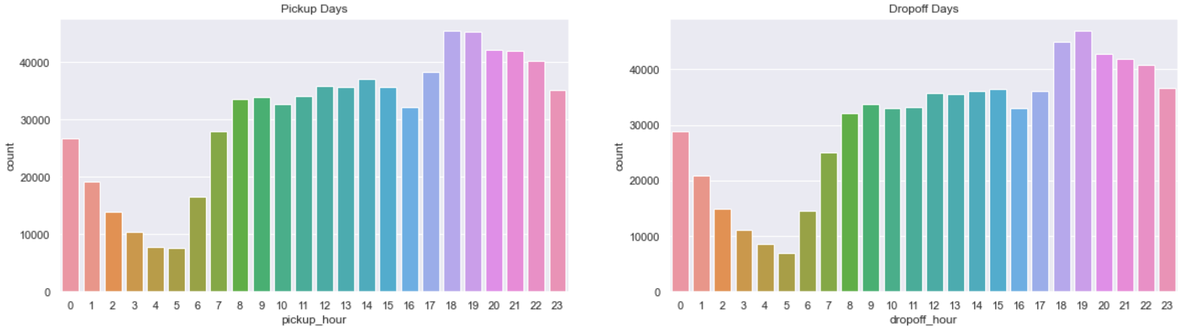
```
figure,(ax1,ax2)=plt.subplots(ncols=2,figsize=(20,5))
ax1.set_title('Pickup Days')
ax=sns.countplot(x="pickup_day",data=data,ax=ax1)
ax2.set_title('Dropoff Days')
ax=sns.countplot(x="dropoff_day",data=data,ax=ax2)
```



Trips per Hour

In [38]:

```
figure,(ax9,ax10)=plt.subplots(ncols=2,figsize=(20,5))
ax9.set_title('Pickup Days')
ax=sns.countplot(x="pickup_hour",data=data,ax=ax9)
ax10.set_title('Dropoff Days')
ax=sns.countplot(x="dropoff_hour",data=data,ax=ax10)
```



In [ ]:

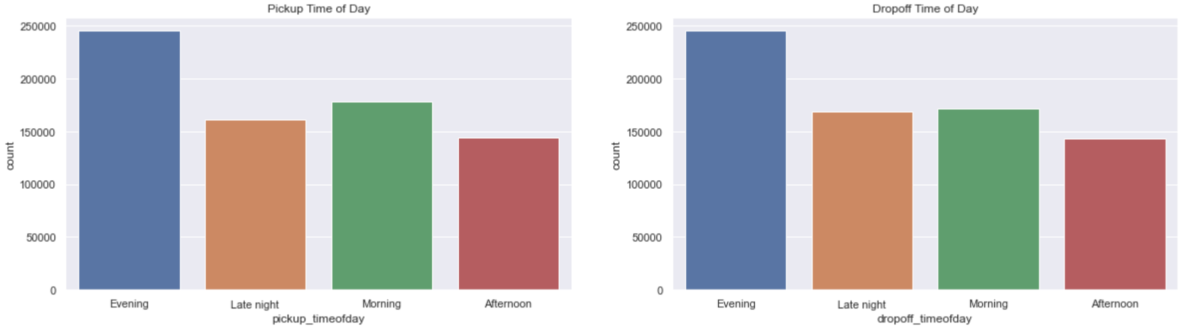
```

```

Trips per Time of Day

In [39]:

```
figure,(ax3,ax4)=plt.subplots(ncols=2,figsize=(20,5))
ax3.set_title('Pickup Time of Day')
ax=sns.countplot(x="pickup_timeofday",data=data,ax=ax3)
ax4.set_title('Dropoff Time of Day')
ax=sns.countplot(x="dropoff_timeofday",data=data,ax=ax4)
```



In [41]:

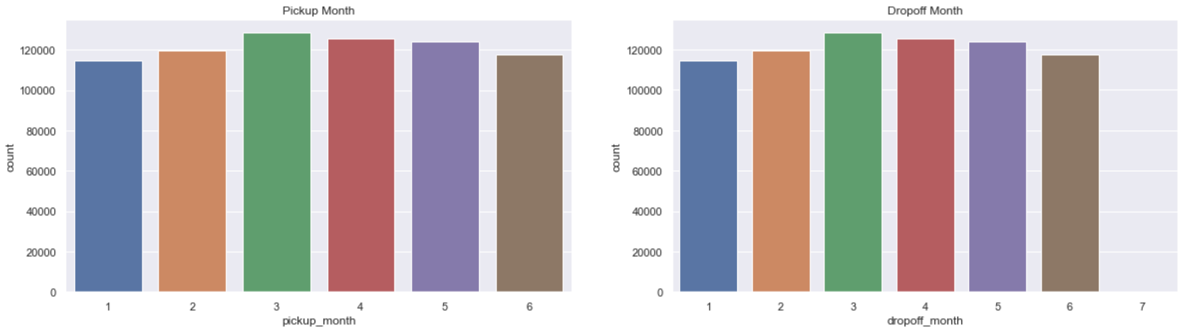
```

```

Trips per month

In [40]:

```
figure,(ax11,ax12)=plt.subplots(ncols=2,figsize=(20,5))
ax11.set_title('Pickup Month')
ax=sns.countplot(x="pickup_month",data=data,ax=ax11)
ax12.set_title('Dropoff Month')
ax=sns.countplot(x="dropoff_month",data=data,ax=ax12)
```



Bivariate Analysis

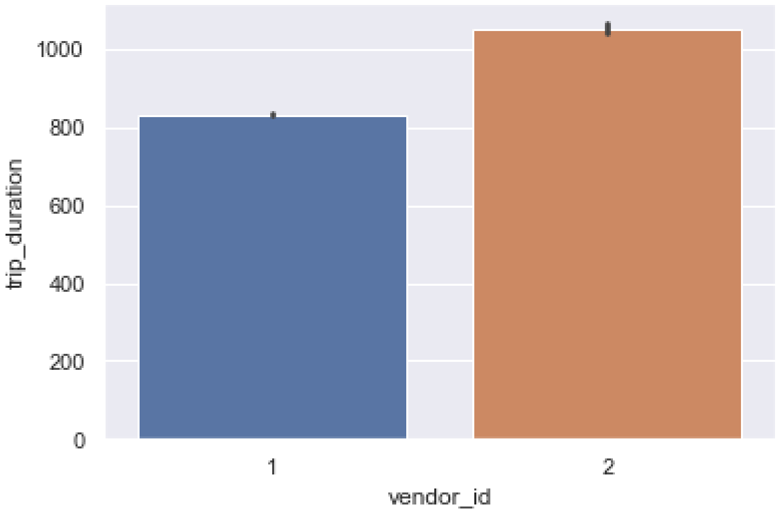
Trip Duration per Vendor

In [46]:

```
sns.barplot(y='trip_duration',x='vendor_id',data=data,estimator=np.mean)
```

Out[46]:

```
<AxesSubplot:xlabel='vendor_id', ylabel='trip_duration'>
```



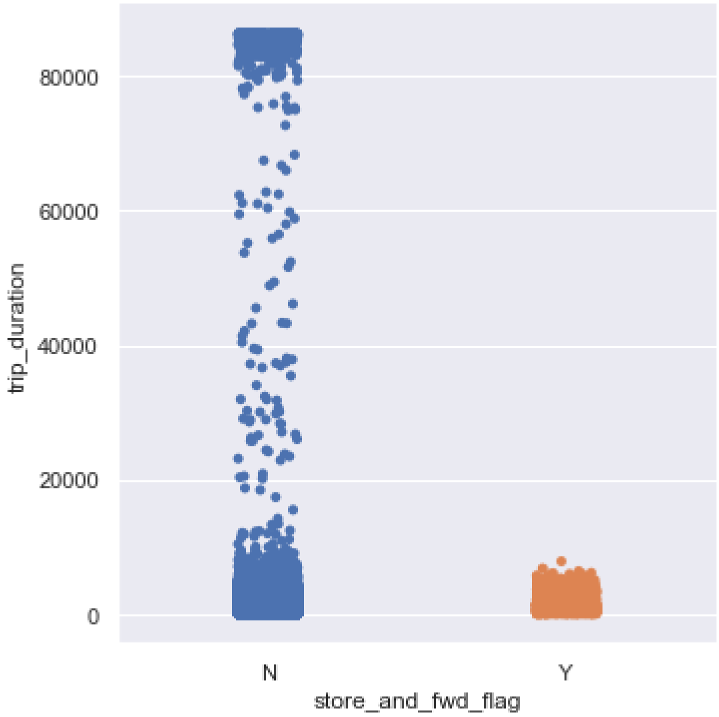
Trip Duration per Store and Forward Flag

In [47]:

```
sns.catplot(y='trip_duration',x='store_and_fwd_flag',data=data,kind="strip")
```

Out[47]:

```
<seaborn.axisgrid.FacetGrid at 0x2a31eab0520>
```



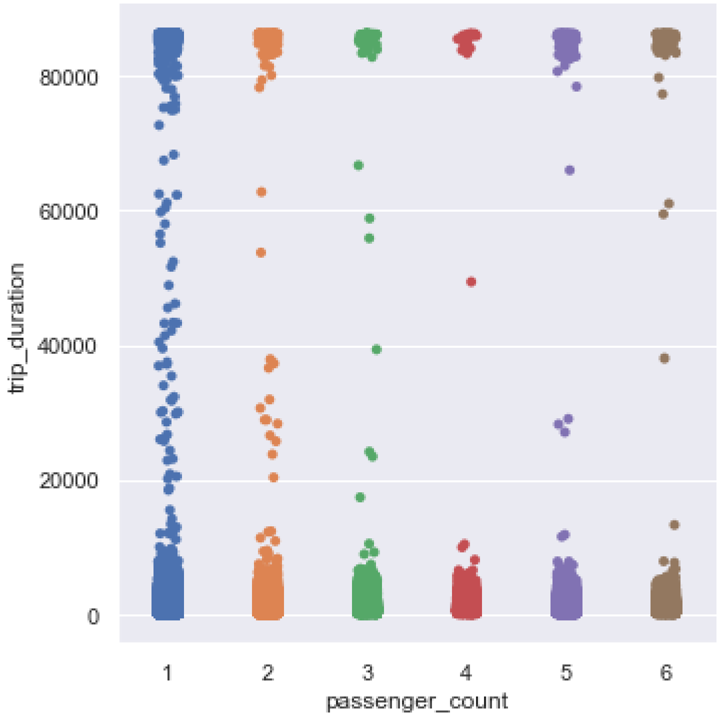
Trip Duration per passenger count

In [48]:

```
sns.catplot(y='trip_duration',x='passenger_count',data=data,kind="strip")
```

Out[48]:

```
<seaborn.axisgrid.FacetGrid at 0x2a31f949fd0>
```



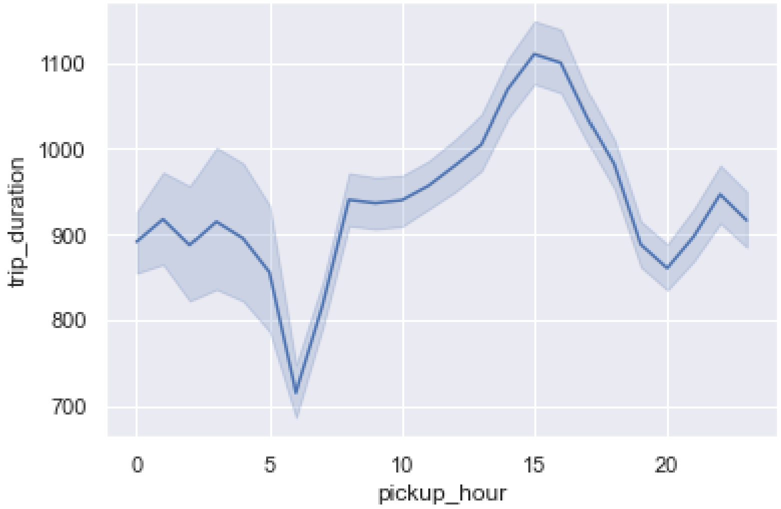
Trip Duration per hour

In [49]:

```
sns.lineplot(x='pickup_hour',y='trip_duration',data=data)
```

Out[49]:

```
<AxesSubplot:xlabel='pickup_hour', ylabel='trip_duration'>
```



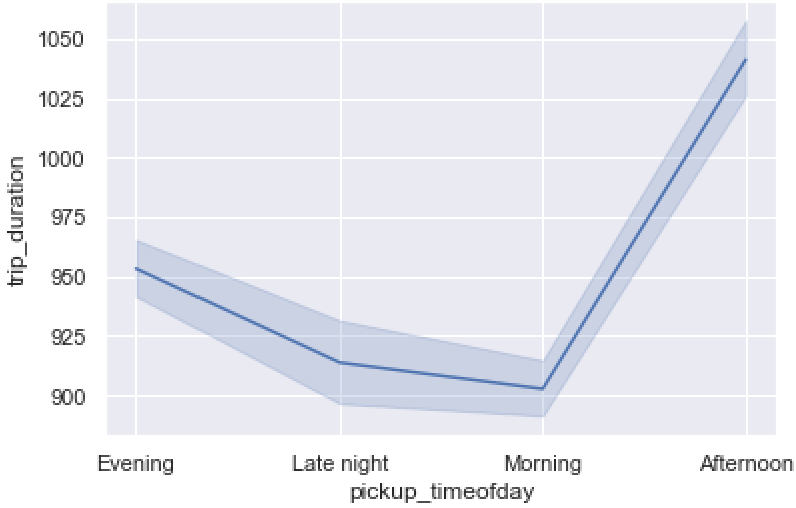
Trip Duration per time of day

In [50]:

```
sns.lineplot(x='pickup_timeofday',y='trip_duration',data=data)
```

Out[50]:

```
<AxesSubplot:xlabel='pickup_timeofday', ylabel='trip_duration'>
```



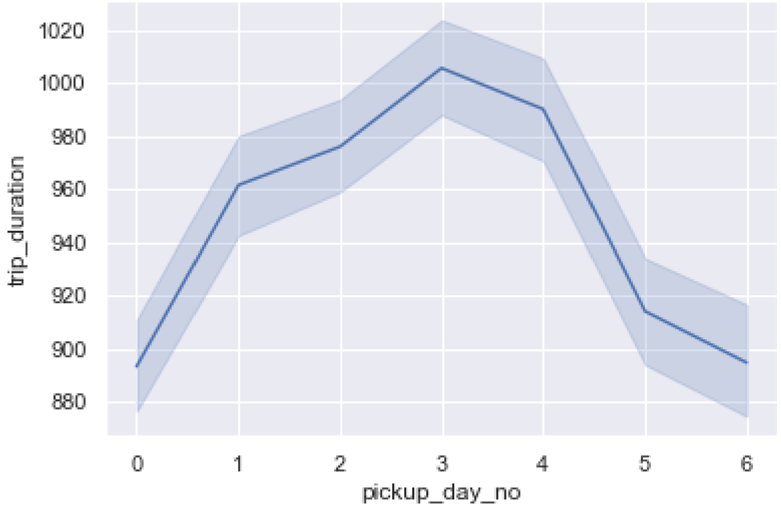
Trip Duration per Day of Week

In [51]:

```
sns.lineplot(x='pickup_day_no',y='trip_duration',data=data)
```

Out[51]:

```
<AxesSubplot:xlabel='pickup_day_no', ylabel='trip_duration'>
```



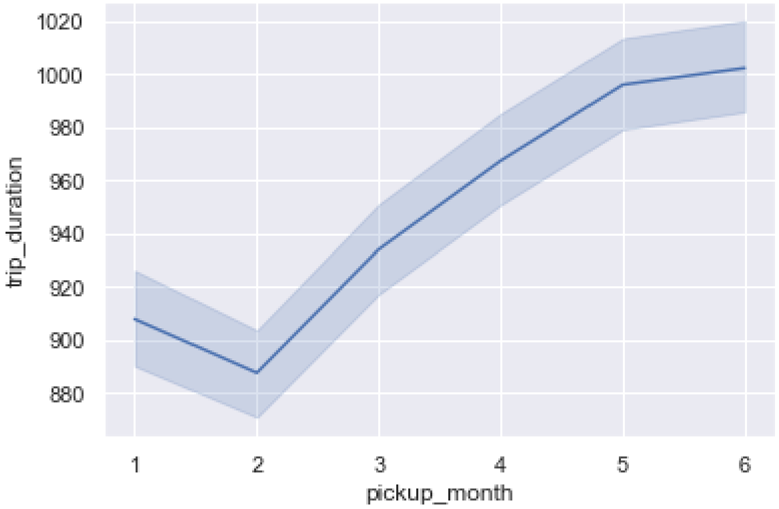
Trip Duration per month

In [52]:

```
sns.lineplot(x='pickup_month',y='trip_duration',data=data)
```

Out[52]:

```
<AxesSubplot:xlabel='pickup_month', ylabel='trip_duration'>
```



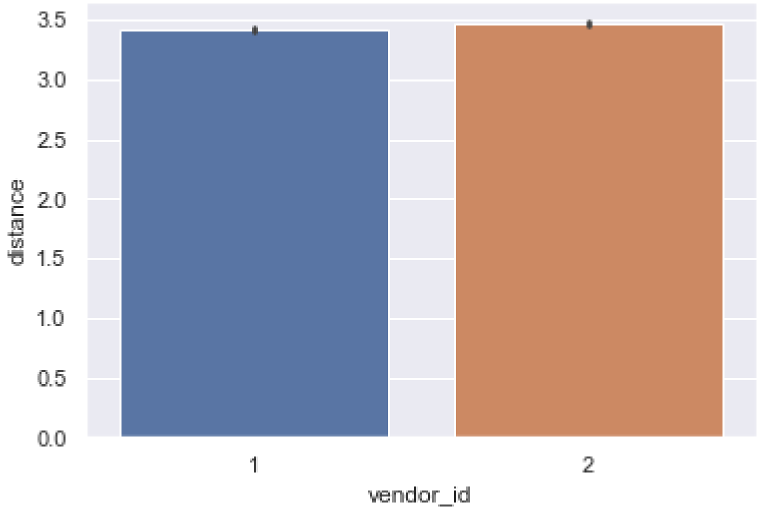
Distance and Vendor

In [53]:

```
sns.barplot(y='distance',x='vendor_id',data=data,estimator=np.mean)
```

Out[53]:

```
<AxesSubplot:xlabel='vendor_id', ylabel='distance'>
```



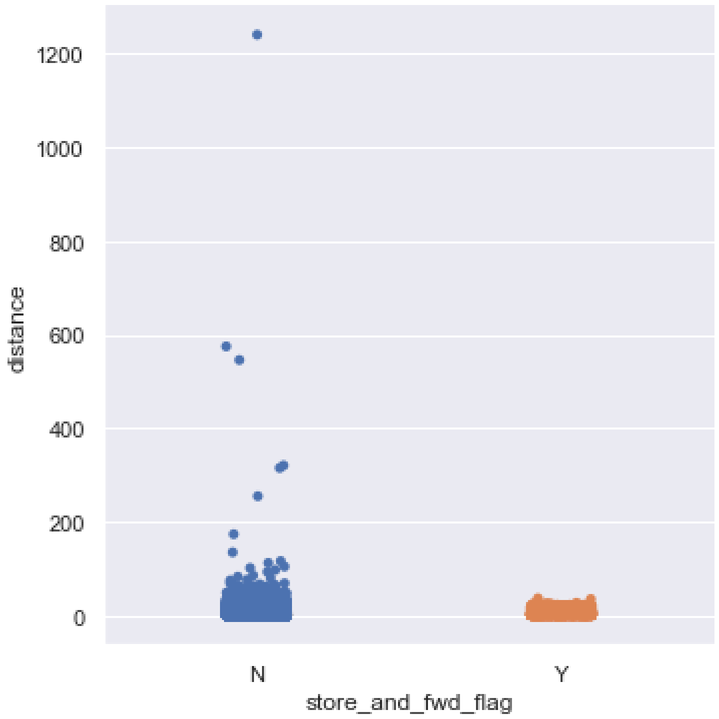
Distance and Store and Forward Flag

In [54]:

```
sns.catplot(y='distance',x='store_and_fwd_flag',data=data,kind="strip")
```

Out[54]:

```
<seaborn.axisgrid.FacetGrid at 0x2a31e7fffd0>
```



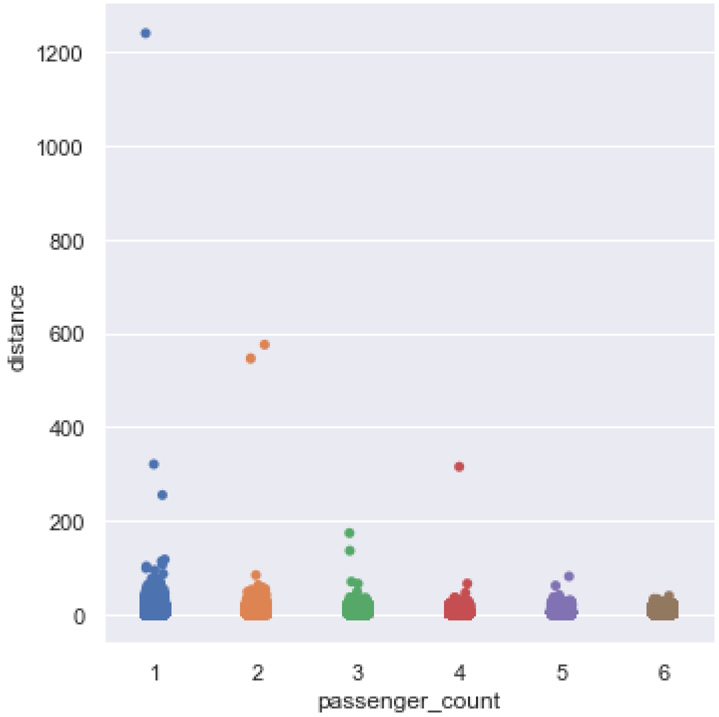
Distance per passenger count

In [55]:

```
sns.catplot(y='distance',x='passenger_count',data=data,kind="strip")
```

Out[55]:

```
<seaborn.axisgrid.FacetGrid at 0x2a31f94e160>
```



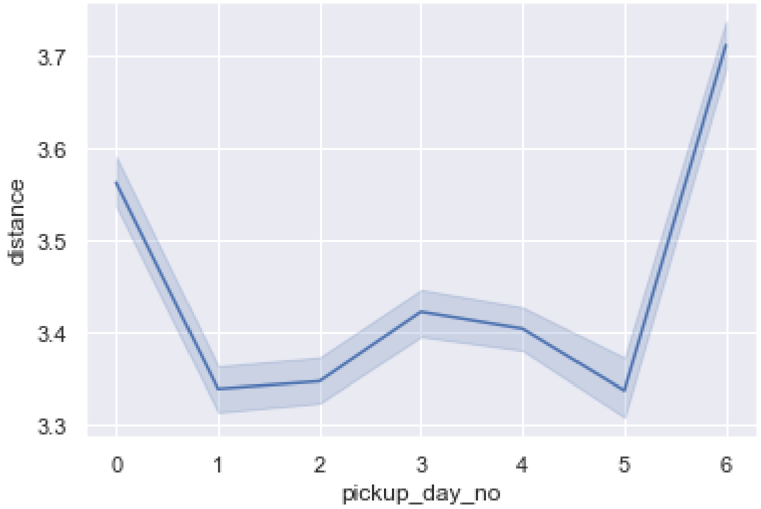
Distance per day of week

In [56]:

```
sns.lineplot(x='pickup_day_no',y='distance',data=data)
```

Out[56]:

```
<AxesSubplot:xlabel='pickup_day_no', ylabel='distance'>
```



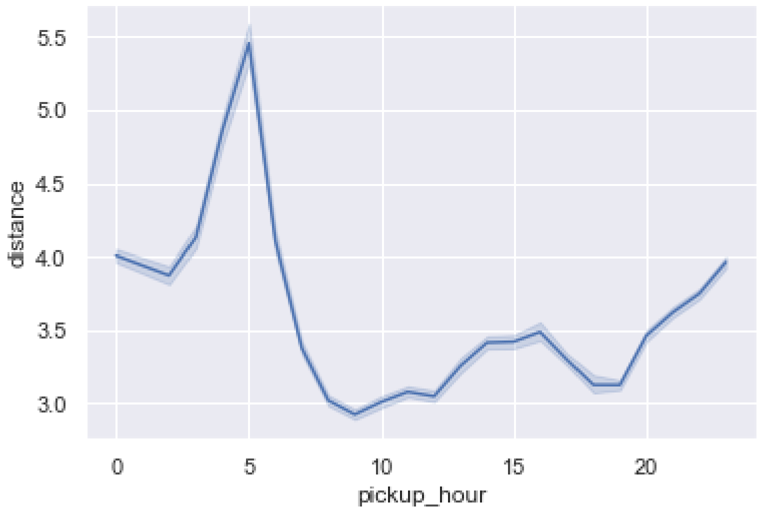
Distance per hour of day

In [57]:

```
sns.lineplot(x='pickup_hour',y='distance',data=data)
```

Out[57]:

```
<AxesSubplot:xlabel='pickup_hour', ylabel='distance'>
```



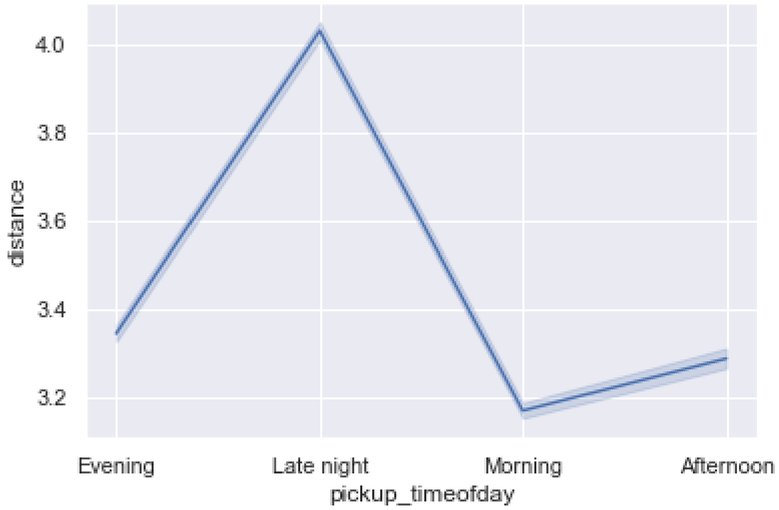
Distance per time of day

In [58]:

```
sns.lineplot(x='pickup_timeofday',y='distance',data=data)
```

Out[58]:

```
<AxesSubplot:xlabel='pickup_timeofday', ylabel='distance'>
```



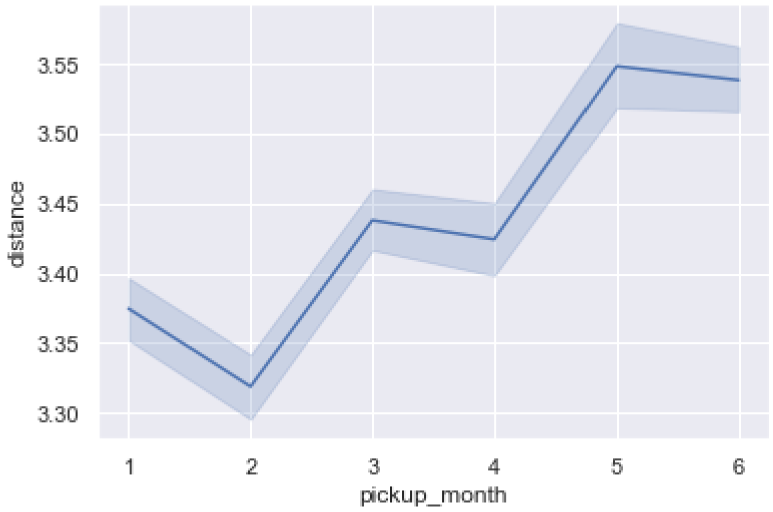
Distance per month

In [59]:

```
sns.lineplot(x='pickup_month',y='distance',data=data)
```

Out[59]:

```
<AxesSubplot:xlabel='pickup_month', ylabel='distance'>
```



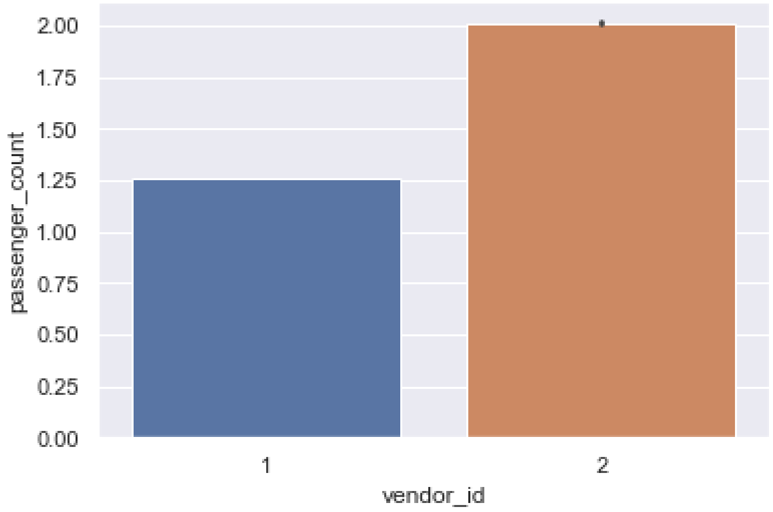
Passenger Count and Vendor id

In [60]:

```
sns.barplot(y='passenger_count',x='vendor_id',data=data)
```

Out[60]:

```
<AxesSubplot:xlabel='vendor_id', ylabel='passenger_count'>
```



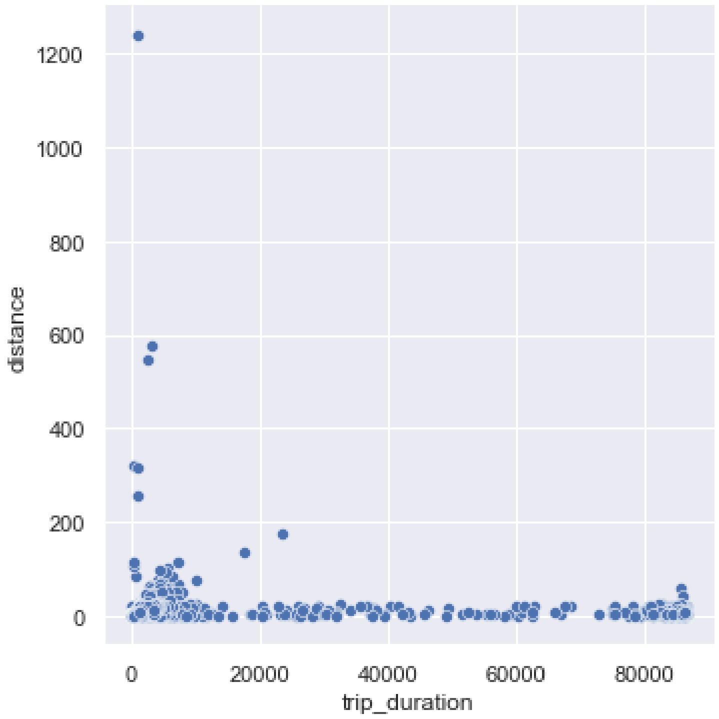
Trip Duration and Distance

In [61]:

```
sns.relplot(y=data.distance,x='trip_duration',data=data)
```

Out[61]:

```
<seaborn.axisgrid.FacetGrid at 0x2a330ad9f10>
```



In [ ]:

```
```

In [63]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 729286 entries, 0 to 729321
Data columns (total 22 columns):
 #   Column              Non-Null Count   Dtype         
---  ------              --------------   -----         
 0   id                  729286 non-null  object        
 1   vendor_id           729286 non-null  int64         
 2   pickup_datetime     729286 non-null  datetime64[ns]
 3   dropoff_datetime    729286 non-null  datetime64[ns]
 4   passenger_count     729286 non-null  int64         
 5   pickup_longitude    729286 non-null  float64       
 6   pickup_latitude     729286 non-null  float64       
 7   dropoff_longitude   729286 non-null  float64       
 8   dropoff_latitude    729286 non-null  float64       
 9   store_and_fwd_flag  729286 non-null  object        
 10  trip_duration       729286 non-null  int64         
 11  pickup_day          729286 non-null  object        
 12  dropoff_day         729286 non-null  object        
 13  pickup_day_no       729286 non-null  int64         
 14  dropoff_day_no      729286 non-null  int64         
 15  pickup_hour         729286 non-null  int64         
 16  dropoff_hour        729286 non-null  int64         
 17  pickup_month        729286 non-null  int64         
 18  dropoff_month       729286 non-null  int64         
 19  pickup_timeofday    729286 non-null  object        
 20  dropoff_timeofday   729286 non-null  object        
 21  distance            729286 non-null  float64       
dtypes: datetime64[ns](2), float64(5), int64(9), object(6)
memory usage: 144.1+ MB
```

In [64]:

```
data.isnull().sum()
```

Out[64]:

```
id                    0
vendor_id             0
pickup_datetime       0
dropoff_datetime      0
passenger_count       0
pickup_longitude      0
pickup_latitude       0
dropoff_longitude     0
dropoff_latitude      0
store_and_fwd_flag    0
trip_duration         0
pickup_day            0
dropoff_day           0
pickup_day_no         0
dropoff_day_no        0
pickup_hour           0
dropoff_hour          0
pickup_month          0
dropoff_month         0
pickup_timeofday      0
dropoff_timeofday     0
distance              0
dtype: int64
```

In [ ]:

```
data['tpep_pickup_datetime'] =  pd.to_datetime(yellow_taxi_data['tpep_pickup_datetime'], fo

data['dropoff_timeofday'] =  pd.to_datetime(yellow_taxi_data['tpep_dropoff_datetime'], form

data['trip_duration'] = (yellow_taxi_data['tpep_dropoff_datetime'] -

data['tpep_pickup_datetime']).dt.secondsyellow_taxi_data['PULocationID'].fillna(-1, inplace

data['DOLocationID'].fillna(-1, inplace = True)
```