IMPORTING LIBRARIES

In [1]:
```
import numpy as np
import pandas as pd
import warnings
warnings.filterwarnings("ignore")
import seaborn as sns
import matplotlib.pyplot as plt
```

Exploratory Data Analysis

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

In [3]:
```
df
```

Out[3]:

	Unnamed: 0	key	fare_amount	pickup_datetime	pickup_longitude	pick
0	24238194	2015-05-07 19:52:06.0000003	7.5	2015-05-07 19:52:06 UTC	-73.999817	
1	27835199	2009-07-17 20:04:56.0000002	7.7	2009-07-17 20:04:56 UTC	-73.994355	
2	44984355	2009-08-24 21:45:00.00000061	12.9	2009-08-24 21:45:00 UTC	-74.005043	
3	25894730	2009-06-26 08:22:21.0000001	5.3	2009-06-26 08:22:21 UTC	-73.976124	
4	17610152	2014-08-28 17:47:00.000000188	16.0	2014-08-28 17:47:00 UTC	-73.925023	
...	...	...	...	...	...	...
199995	42598914	2012-10-28 10:49:00.00000053	3.0	2012-10-28 10:49:00 UTC	-73.987042	
199996	16382965	2014-03-14 01:09:00.0000008	7.5	2014-03-14 01:09:00 UTC	-73.984722	
199997	27804658	2009-06-29 00:42:00.00000078	30.9	2009-06-29 00:42:00 UTC	-73.986017	
199998	20259894	2015-05-20 14:56:25.0000004	14.5	2015-05-20 14:56:25 UTC	-73.997124	
199999	11951496	2010-05-15 04:08:00.00000076	14.1	2010-05-15 04:08:00 UTC	-73.984395	

200000 rows × 9 columns

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200000 entries, 0 to 199999
Data columns (total 9 columns):
 #   Column             Non-Null Count   Dtype  
---  ------             --------------   -----  
 0   Unnamed: 0         200000 non-null  int64  
 1   key                200000 non-null  object 
 2   fare_amount        200000 non-null  float64
 3   pickup_datetime    200000 non-null  object 
 4   pickup_longitude   200000 non-null  float64
 5   pickup_latitude    200000 non-null  float64
 6   dropoff_longitude  199999 non-null  float64
 7   dropoff_latitude   199999 non-null  float64
 8   passenger_count    200000 non-null  int64  
dtypes: float64(5), int64(2), object(2)
memory usage: 13.7+ MB
```

In [5]:
```
list(df)
```

Out[5]:
```
['Unnamed: 0',
 'key',
 'fare_amount',
 'pickup_datetime',
 'pickup_longitude',
 'pickup_latitude',
 'dropoff_longitude',
 'dropoff_latitude',
 'passenger_count']
```

In [6]:
```
df.shape
```

Out[6]:
```
(200000, 9)
```

In [7]:
```
df.head(5)
```

Out[7]:

	Unnamed: 0	key	fare_amount	pickup_datetime	pickup_longitude	pickup_lat
0	24238194	2015-05-07 19:52:06.0000003	7.5	2015-05-07 19:52:06 UTC	-73.999817	40.73
1	27835199	2009-07-17 20:04:56.0000002	7.7	2009-07-17 20:04:56 UTC	-73.994355	40.72
2	44984355	2009-08-24 21:45:00.00000061	12.9	2009-08-24 21:45:00 UTC	-74.005043	40.74
3	25894730	2009-06-26 08:22:21.0000001	5.3	2009-06-26 08:22:21 UTC	-73.976124	40.79
4	17610152	2014-08-28 17:47:00.000000188	16.0	2014-08-28 17:47:00 UTC	-73.925023	40.74

In [8]:
```
df.tail()
```

Out[8]:

	Unnamed: 0	key	fare_amount	pickup_datetime	pickup_longitude	picku
199995	42598914	2012-10-28 10:49:00.00000053	3.0	2012-10-28 10:49:00 UTC	-73.987042	
199996	16382965	2014-03-14 01:09:00.0000008	7.5	2014-03-14 01:09:00 UTC	-73.984722	
199997	27804658	2009-06-29 00:42:00.00000078	30.9	2009-06-29 00:42:00 UTC	-73.986017	
199998	20259894	2015-05-20 14:56:25.0000004	14.5	2015-05-20 14:56:25 UTC	-73.997124	
199999	11951496	2010-05-15 04:08:00.00000076	14.1	2010-05-15 04:08:00 UTC	-73.984395	

In [9]:
```
df.describe()
```

Out[9]:

	Unnamed: 0	fare_amount	pickup_longitude	pickup_latitude	dropoff_longitude	drc
count	2.000000e+05	200000.000000	200000.000000	200000.000000	199999.000000	19
mean	2.771250e+07	11.359955	-72.527638	39.935885	-72.525292	
std	1.601382e+07	9.901776	11.437787	7.720539	13.117408	
min	1.000000e+00	-52.000000	-1340.648410	-74.015515	-3356.666300	
25%	1.382535e+07	6.000000	-73.992065	40.734796	-73.991407	
50%	2.774550e+07	8.500000	-73.981823	40.752592	-73.980093	
75%	4.155530e+07	12.500000	-73.967154	40.767158	-73.963658	
max	5.542357e+07	499.000000	57.418457	1644.421482	1153.572603	

In [10]:
```
df.isnull().sum()
```

Out[10]:
```
Unnamed: 0           0
key                  0
fare_amount          0
pickup_datetime      0
pickup_longitude     0
pickup_latitude      0
dropoff_longitude    1
dropoff_latitude     1
passenger_count      0
dtype: int64
```

In [11]:
```
df.min()
```

Out[11]:
```
Unnamed: 0                                  1
key                 2009-01-01 01:15:22.0000006
fare_amount                               -52.0
pickup_datetime          2009-01-01 01:15:22 UTC
pickup_longitude                     -1340.64841
pickup_latitude                       -74.015515
dropoff_longitude                     -3356.6663
dropoff_latitude                     -881.985513
passenger_count                                0
dtype: object
```

In [12]:
```
df.max()
```

Out[12]:
```
Unnamed: 0                              55423567
key                 2015-06-30 23:40:39.0000001
fare_amount                               499.0
pickup_datetime          2015-06-30 23:40:39 UTC
pickup_longitude                      57.418457
pickup_latitude                     1644.421482
dropoff_longitude                   1153.572603
dropoff_latitude                     872.697628
passenger_count                             208
dtype: object
```

In [13]:
```
df.groupby('passenger_count').sum()
```

Out[13]:

passenger_count	Unnamed: 0	fare_amount	pickup_longitude	pickup_latitude	dropoff_lon
0	18978654087	6707.10	-5.097014e+04	2.807805e+04	-5.09687
1	3831523506157	1557856.86	-1.003961e+07	5.528875e+06	-1.00378
2	821538663509	346792.84	-2.134961e+06	1.175793e+06	-2.13677
3	243991738353	102013.66	-6.441367e+05	3.546023e+05	-6.44414
4	118778722636	49783.21	-3.096661e+05	1.705347e+05	-3.09814
5	388522141206	156896.57	-1.016264e+06	5.590608e+05	-1.01551
6	119131416750	51929.11	-3.098461e+05	1.701921e+05	-3.09546
208	35893772	11.70	-7.393779e+01	4.075850e+01	-7.39378

In [14]:
```
df.groupby('fare_amount').sum()
```

Out[14]:

fare_amount	Unnamed: 0	pickup_longitude	pickup_latitude	dropoff_longitude	dropoff_latitude
-52.00	65106419	-147.985023	81.487755	-74.005699	40.728680
-50.50	29575269	-73.784868	40.648677	-73.976975	40.763522
-49.57	26673143	-73.972772	40.785657	-73.972867	40.785500
-23.70	10267585	-73.952740	40.768233	-74.007028	40.707338
-10.90	34292338	-73.964257	40.760630	-73.994222	40.761533
...	...	...	...	...	...
230.00	54143082	-73.937765	40.758267	-74.382200	40.700890
250.00	38680012	0.000000	0.000000	0.000000	0.000000
275.00	20013003	0.000000	0.000000	0.000000	0.000000
350.00	33491441	0.000000	0.000000	0.000000	0.000000
499.00	51151143	-73.968377	40.764602	-73.968368	40.764600

1244 rows × 6 columns

DROPPING THE UNWANTED COLUMNS

In [15]:
```
df=df.drop(['Unnamed: 0','pickup_longitude','pickup_latitude','dropoff_long
```

In [16]:
```
df
```

Out[16]:

	key	fare_amount	pickup_datetime	passenger_count
0	2015-05-07 19:52:06.0000003	7.5	2015-05-07 19:52:06 UTC	1
1	2009-07-17 20:04:56.0000002	7.7	2009-07-17 20:04:56 UTC	1
2	2009-08-24 21:45:00.00000061	12.9	2009-08-24 21:45:00 UTC	1
3	2009-06-26 08:22:21.0000001	5.3	2009-06-26 08:22:21 UTC	3
4	2014-08-28 17:47:00.000000188	16.0	2014-08-28 17:47:00 UTC	5
...	...	...	...	...
199995	2012-10-28 10:49:00.00000053	3.0	2012-10-28 10:49:00 UTC	1
199996	2014-03-14 01:09:00.0000008	7.5	2014-03-14 01:09:00 UTC	1
199997	2009-06-29 00:42:00.00000078	30.9	2009-06-29 00:42:00 UTC	2
199998	2015-05-20 14:56:25.0000004	14.5	2015-05-20 14:56:25 UTC	1
199999	2010-05-15 04:08:00.00000076	14.1	2010-05-15 04:08:00 UTC	1

200000 rows × 4 columns

In [17]:
```
df=df.drop(['key'],axis=1)
```

In [18]:
```
df
```

Out[18]:

	fare_amount	pickup_datetime	passenger_count
0	7.5	2015-05-07 19:52:06 UTC	1
1	7.7	2009-07-17 20:04:56 UTC	1
2	12.9	2009-08-24 21:45:00 UTC	1
3	5.3	2009-06-26 08:22:21 UTC	3
4	16.0	2014-08-28 17:47:00 UTC	5
...	...	...	...
199995	3.0	2012-10-28 10:49:00 UTC	1
199996	7.5	2014-03-14 01:09:00 UTC	1
199997	30.9	2009-06-29 00:42:00 UTC	2
199998	14.5	2015-05-20 14:56:25 UTC	1
199999	14.1	2010-05-15 04:08:00 UTC	1

200000 rows × 3 columns

DATA CLEANING

In [43]:
```
df['pickup_datetime']=pd.to_datetime(df['pickup_datetime'])
```

In [44]:
```
df['year']=df['pickup_datetime'].dt.year
```

In [45]:
```
df['date']=df['pickup_datetime'].dt.date
```

In [46]:
```
df['time']=df['pickup_datetime'].dt.time
```

In [47]:
```
df['month']=df['pickup_datetime'].dt.month
```

In [48]:
```
df
```

Out[48]:

	fare_amount	pickup_datetime	passenger_count	year	date	time	month
0	7.5	2015-05-07 19:52:06+00:00	1	2015	2015-05-07	19:52:06	5
1	7.7	2009-07-17 20:04:56+00:00	1	2009	2009-07-17	20:04:56	7
2	12.9	2009-08-24 21:45:00+00:00	1	2009	2009-08-24	21:45:00	8
3	5.3	2009-06-26 08:22:21+00:00	3	2009	2009-06-26	08:22:21	6
4	16.0	2014-08-28 17:47:00+00:00	5	2014	2014-08-28	17:47:00	8
...	...	...	...	...	...	...	...
199995	3.0	2012-10-28 10:49:00+00:00	1	2012	2012-10-28	10:49:00	10
199996	7.5	2014-03-14 01:09:00+00:00	1	2014	2014-03-14	01:09:00	3
199997	30.9	2009-06-29 00:42:00+00:00	2	2009	2009-06-29	00:42:00	6
199998	14.5	2015-05-20 14:56:25+00:00	1	2015	2015-05-20	14:56:25	5
199999	14.1	2010-05-15 04:08:00+00:00	1	2010	2010-05-15	04:08:00	5

200000 rows × 7 columns

In [49]:
```
print(df[['pickup_datetime','date','time','year','month']].head())
```

```
           pickup_datetime        date      time  year  month
0 2015-05-07 19:52:06+00:00  2015-05-07  19:52:06  2015      5
1 2009-07-17 20:04:56+00:00  2009-07-17  20:04:56  2009      7
2 2009-08-24 21:45:00+00:00  2009-08-24  21:45:00  2009      8
3 2009-06-26 08:22:21+00:00  2009-06-26  08:22:21  2009      6
4 2014-08-28 17:47:00+00:00  2014-08-28  17:47:00  2014      8
```

GROUPING THE DATA

In [50]:
```
df.groupby('year').sum()
```

Out[50]:

year	fare_amount	passenger_count	month
2009	305637.75	51398	199401
2010	306002.55	50849	197608
2011	332326.24	53079	208766
2012	363298.45	54156	208773
2013	396489.39	53343	201286
2014	390094.57	50923	192192
2015	178142.10	23159	48333

In [51]:
```
df.groupby('month').sum()
```

Out[51]:

month	fare_amount	passenger_count	year
1	189499.77	29432	35546978
2	182453.99	28028	33591011
3	208300.37	31032	37750839
4	210972.89	31061	37434524
5	220246.02	31847	37943337
6	206421.84	29959	35785919
7	168478.59	25693	30363131
8	159351.40	24314	28605137
9	180011.21	25349	30707495
10	190058.67	27492	32610508
11	177806.02	25944	30799876
12	178390.28	26756	31209733

In [52]:
```
df.groupby('passenger_count').sum()
```

Out[52]:

passenger_count	fare_amount	year	month
0	6707.10	1426053	4184
1	1557856.86	278479123	867262
2	346792.84	59198340	184971
3	102013.66	17865809	56340
4	49783.21	8601877	27906
5	156896.57	28177895	88751
6	51929.11	8597381	26933
208	11.70	2010	12

In [53]:
```
df['year']=pd.to_datetime(df['date']).dt.year
```

In [54]:
```
result=df.groupby('year')['passenger_count'].sum().reset_index()
result
```

Out[54]:

	year	passenger_count
0	2009	51398
1	2010	50849
2	2011	53079
3	2012	54156
4	2013	53343
5	2014	50923
6	2015	23159

In [55]:
```
result=df.groupby('month')['passenger_count'].sum().reset_index()
result
```

Out[55]:

	month	passenger_count
0	1	29432
1	2	28028
2	3	31032
3	4	31061
4	5	31847
5	6	29959
6	7	25693
7	8	24314
8	9	25349
9	10	27492
10	11	25944
11	12	26756

In [56]:
```
result=df.groupby('date')['passenger_count'].sum().reset_index()
result
```

Out[56]:

	date	passenger_count
0	2009-01-01	113
1	2009-01-02	113
2	2009-01-03	147
3	2009-01-04	132
4	2009-01-05	109
...	...	...
2367	2015-06-26	145
2368	2015-06-27	133
2369	2015-06-28	123
2370	2015-06-29	99
2371	2015-06-30	103

2372 rows × 2 columns

CORRELATION MATRIX

In [57]:
```
cor_mat=df.corr()
```

In [58]:
```
cor_mat
```

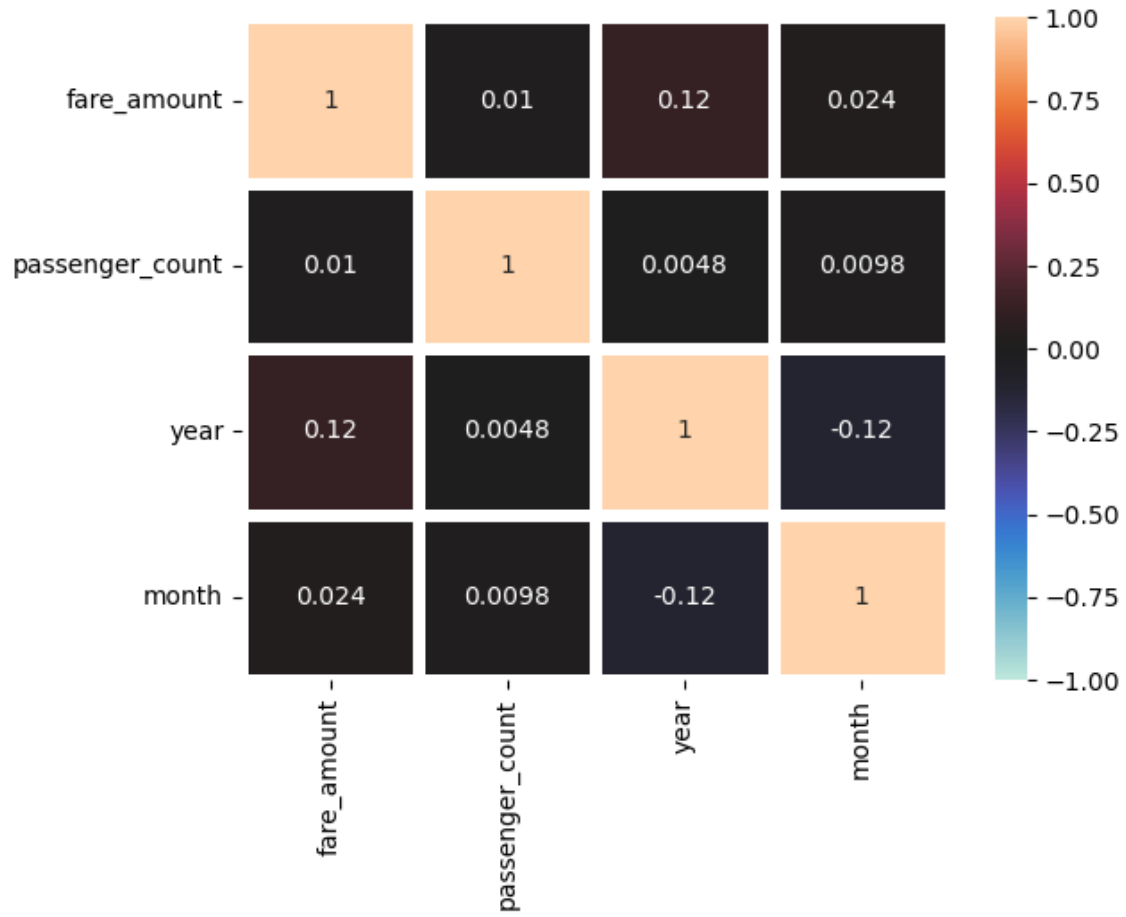
Out[58]:

	fare_amount	passenger_count	year	month
fare_amount	1.000000	0.010150	0.118335	0.023814
passenger_count	0.010150	1.000000	0.004798	0.009773
year	0.118335	0.004798	1.000000	-0.115859
month	0.023814	0.009773	-0.115859	1.000000

HEATMAP

In [59]:
```
import seaborn as sns
sns.heatmap(cor_mat,vmax=1,vmin=-1,annot=True,linewidth=5,cmap='icefire')
```

Out[59]: <Axes: >

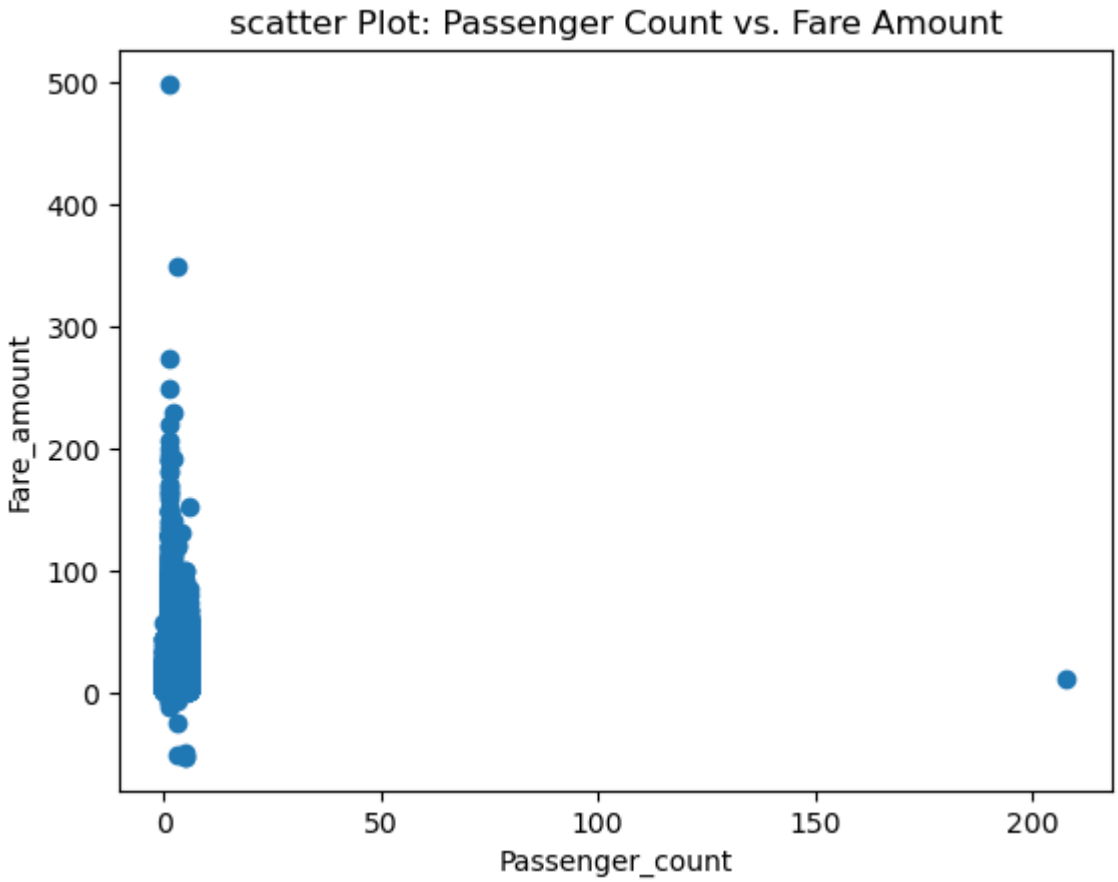


In [ ]:
```

```

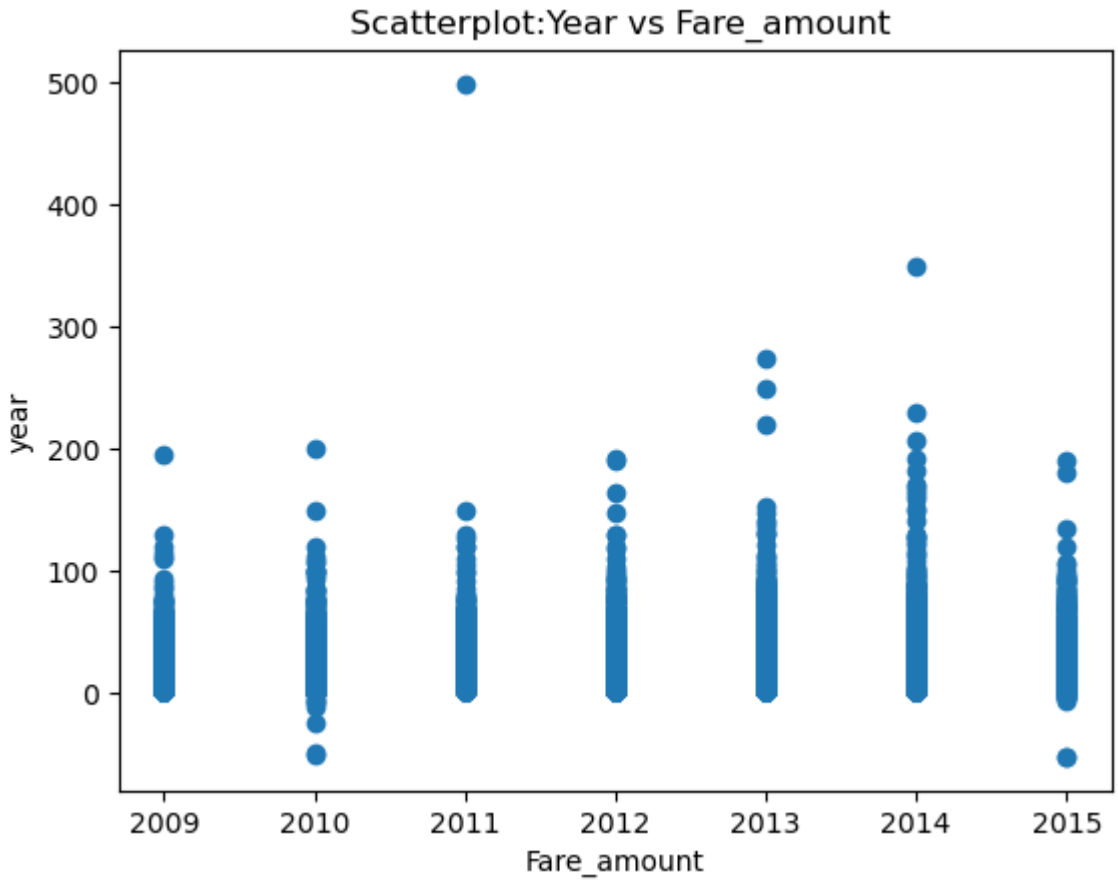
GRAPHICAL REPRESENTATION USING SCATTER PLOT

In [60]:
```
plt.scatter(df['passenger_count'],df['fare_amount'])
plt.xlabel('Passenger_count')
plt.ylabel('Fare_amount')
plt.title('scatter Plot: Passenger Count vs. Fare Amount')
plt.show()
```



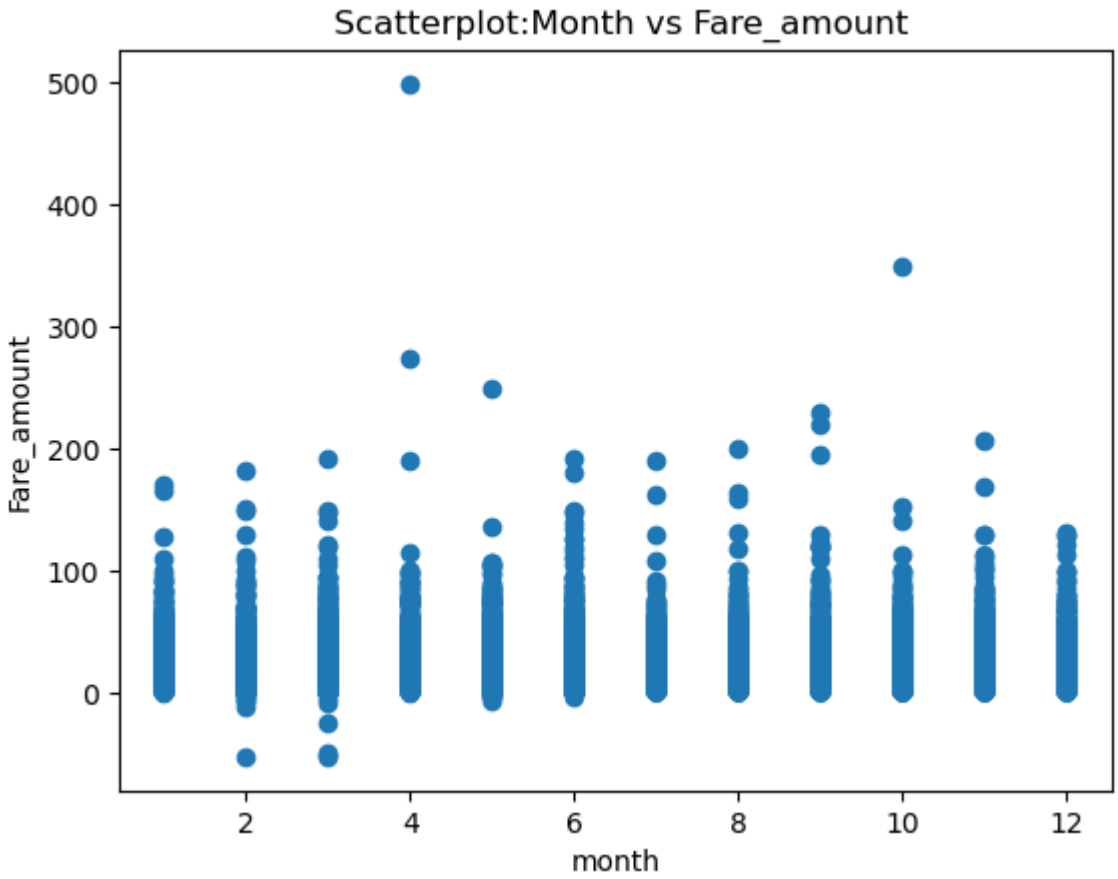
In [61]:
```
#YEAR
```

In [62]:
```
plt.scatter(df['year'],df['fare_amount'])
plt.ylabel('year')
plt.xlabel('Fare_amount')
plt.title(' Scatterplot:Year vs Fare_amount')
plt.show()
```



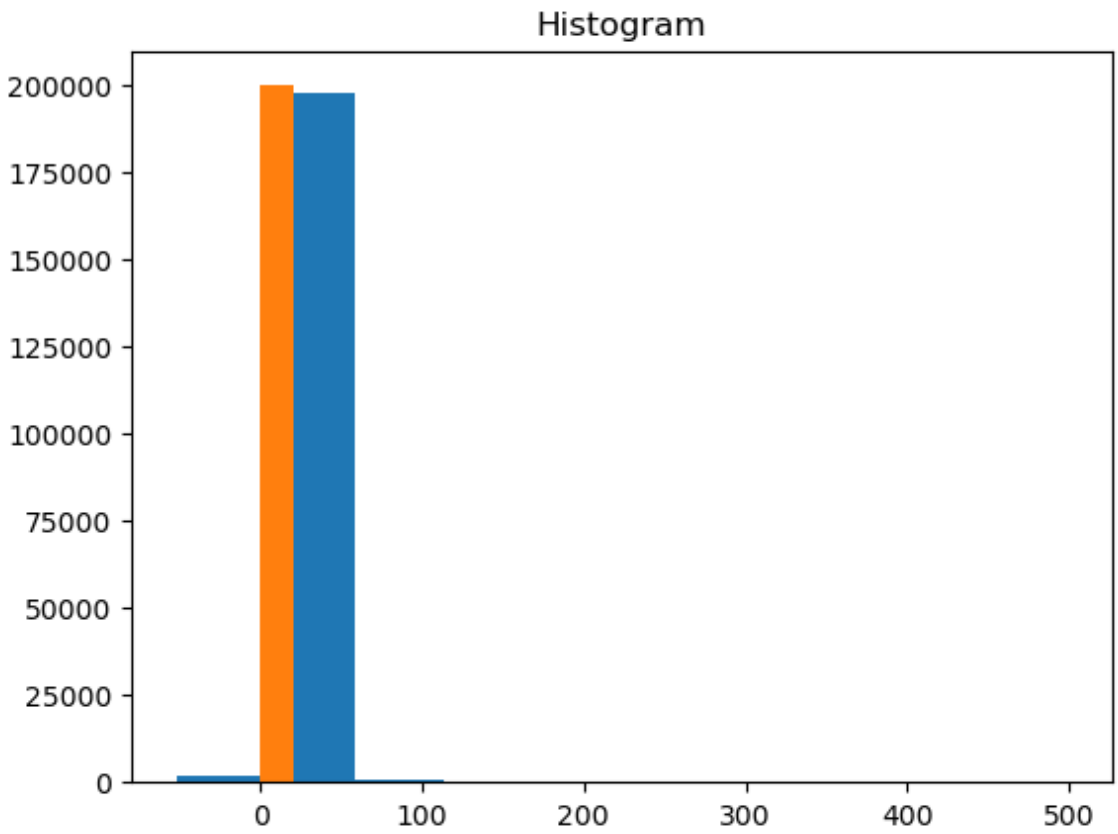
In [63]:
```
#MONTH
```

In [64]:
```
plt.scatter(df['month'],df['fare_amount'])
plt.xlabel('month')
plt.ylabel('Fare_amount')
plt.title(' Scatterplot:Month vs Fare_amount')
plt.show()
```



HISTOGRAM REPRESENTATION

In [65]:
```
plt.hist(df['fare_amount'])
plt.hist(df['passenger_count'])
plt.title('Histogram')
plt.show()
```



In [ ]:

In [ ]:

In [66]:
```
df.to_csv('newfile_uber.csv')
```

In [ ]:

In [ ]:

In [ ]:

In [ ]: