In [1]:
```
#Finding revenu generate by each year and month
#Finding each year how much revenue will be generated(split the year,date,t
```

In [27]:
```
import pandas as pd
import numpy as np
import warnings
warnings.filterwarnings("ignore")
import matplotlib.pyplot as plt
```

In [28]:
```
df=pd.read_csv('uber.csv')
df
```

Out[28]:

	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 [29]:

```
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 [30]:

```
list(df)
```

Out[30]:

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

In [31]:

```
df.describe()
```

Out[31]:

	Unnamed: 0	fare_amount	pickup_longitude	pickup_latitude	dropoff_longitude	drc
count	2.000000e+05	200000.000000	200000.000000	200000.000000	199999.000000	1
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 [32]:
```
df.isna().sum()
```

Out[32]:
```
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 [33]:
```
df.min()
```

Out[33]:
```
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 [34]:
```
df.max()
```

Out[34]:
```
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 [35]:

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

# Extract year, date, and time into separate columns
df['date'] = df['pickup_datetime'].dt.date
df['year'] = df['pickup_datetime'].dt.year
df['month'] = df['pickup_datetime'].dt.month
df['day'] = df['pickup_datetime'].dt.day
df['time'] = df['pickup_datetime'].dt.time


df
```

Out[35]:

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

200000 rows × 14 columns

In [36]:
```
df = df.drop(['Unnamed: 0','key','pickup_datetime','pickup_longitude','pick
df
```

Out[36]:

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

200000 rows × 7 columns

In [37]:
```
list(df)
```

Out[37]:
```
['fare_amount', 'passenger_count', 'date', 'year', 'month', 'day', 'time']
```

In [38]:
```
df.to_csv('resuber.csv') #tableau public file
```

In [39]:
```
df.groupby('passenger_count').count()
```

Out[39]:

	fare_amount	date	year	month	day	time
passenger_count						
0	709	709	709	709	709	709
1	138425	138425	138425	138425	138425	138425
2	29428	29428	29428	29428	29428	29428
3	8881	8881	8881	8881	8881	8881
4	4276	4276	4276	4276	4276	4276
5	14009	14009	14009	14009	14009	14009
6	4271	4271	4271	4271	4271	4271
208	1	1	1	1	1	1

In [40]:

```
df.groupby('year').count()
```

Out[40]:

	fare_amount	passenger_count	date	month	day	time
year						
2009	30536	30536	30536	30536	30536	30536
2010	30194	30194	30194	30194	30194	30194
2011	31945	31945	31945	31945	31945	31945
2012	32396	32396	32396	32396	32396	32396
2013	31195	31195	31195	31195	31195	31195
2014	29968	29968	29968	29968	29968	29968
2015	13766	13766	13766	13766	13766	13766

In [41]:

```
df.groupby('day').count()
```

Out[41]:

	fare_amount	passenger_count	date	year	month	time
day						
1	6203	6203	6203	6203	6203	6203
2	6220	6220	6220	6220	6220	6220
3	6281	6281	6281	6281	6281	6281
4	6340	6340	6340	6340	6340	6340
5	6517	6517	6517	6517	6517	6517
6	6566	6566	6566	6566	6566	6566
7	6643	6643	6643	6643	6643	6643
8	6869	6869	6869	6869	6869	6869
9	6790	6790	6790	6790	6790	6790
10	6689	6689	6689	6689	6689	6689
11	6749	6749	6749	6749	6749	6749
12	6773	6773	6773	6773	6773	6773
13	6681	6681	6681	6681	6681	6681
14	6826	6826	6826	6826	6826	6826
15	6526	6526	6526	6526	6526	6526
16	6850	6850	6850	6850	6850	6850
17	6876	6876	6876	6876	6876	6876
18	6910	6910	6910	6910	6910	6910
19	6774	6774	6774	6774	6774	6774
20	6747	6747	6747	6747	6747	6747
21	6579	6579	6579	6579	6579	6579
22	6683	6683	6683	6683	6683	6683
23	6752	6752	6752	6752	6752	6752
24	6481	6481	6481	6481	6481	6481
25	6220	6220	6220	6220	6220	6220
26	6280	6280	6280	6280	6280	6280
27	6232	6232	6232	6232	6232	6232
28	6409	6409	6409	6409	6409	6409
29	5960	5960	5960	5960	5960	5960
30	5841	5841	5841	5841	5841	5841
31	3733	3733	3733	3733	3733	3733

In [42]:
```
df.groupby('time').count()
```

Out[42]:

	fare_amount	passenger_count	date	year	month	day
time						
00:00:00	79	79	79	79	79	79
00:00:02	1	1	1	1	1	1
00:00:03	3	3	3	3	3	3
00:00:07	4	4	4	4	4	4
00:00:09	2	2	2	2	2	2
...	...	...	...	...	...	...
23:59:54	4	4	4	4	4	4
23:59:55	2	2	2	2	2	2
23:59:57	1	1	1	1	1	1
23:59:58	2	2	2	2	2	2
23:59:59	4	4	4	4	4	4

59072 rows × 6 columns

In [43]:
```
df['year']=pd.to_datetime(df['date']).dt.year
```

In [44]:
```
result=df.groupby('year')['passenger_count'].sum().reset_index()
result
```

Out[44]:

	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 [45]:

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

Out[45]:

	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 [46]:

```
numeric_df = df.select_dtypes(include='number')
cor_mat = numeric_df.corr()
```

In [47]:

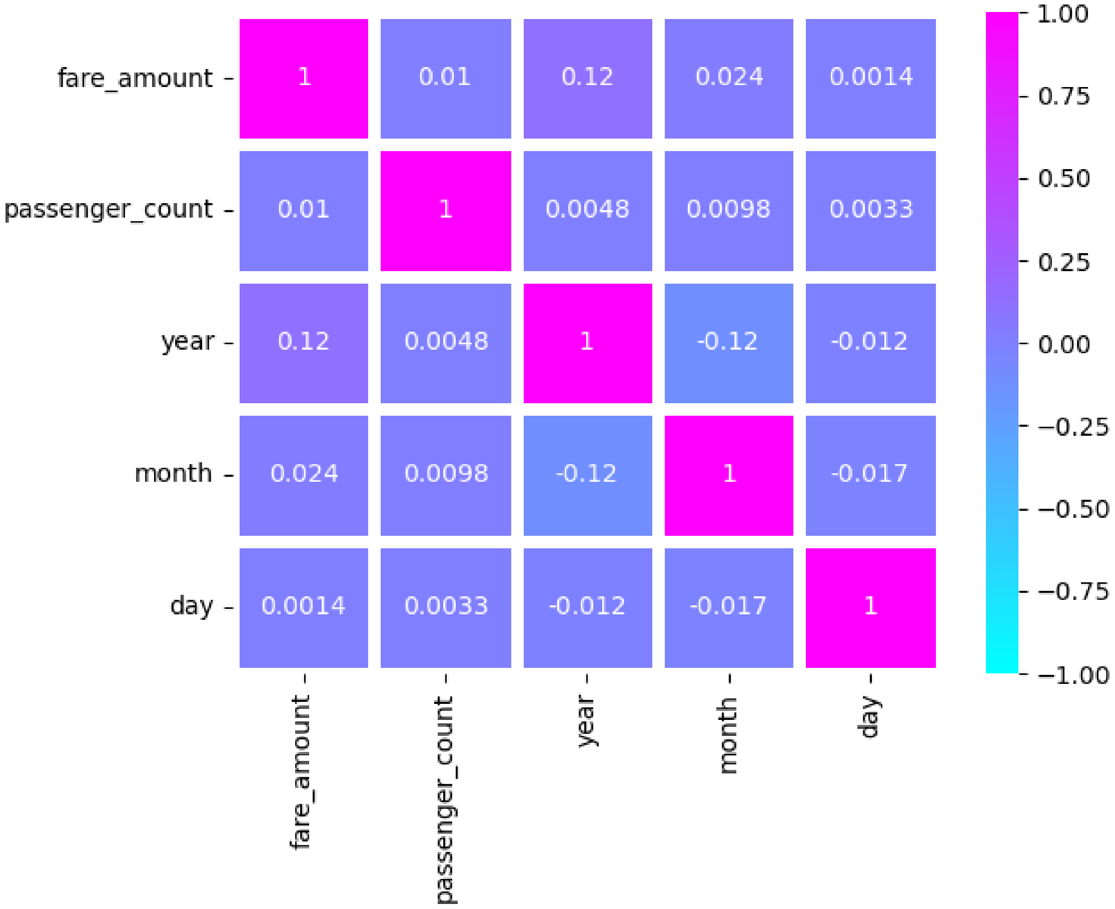
```
cor_mat
```

Out[47]:

	fare_amount	passenger_count	year	month	day
fare_amount	1.000000	0.010150	0.118335	0.023814	0.001374
passenger_count	0.010150	1.000000	0.004798	0.009773	0.003252
year	0.118335	0.004798	1.000000	-0.115859	-0.012170
month	0.023814	0.009773	-0.115859	1.000000	-0.017360
day	0.001374	0.003252	-0.012170	-0.017360	1.000000

In [48]:
```
import seaborn as sns
sns.heatmap(cor_mat,vmax=1,vmin=-1,annot=True,linewidth=5,cmap='cool')
```

Out[48]:
```
<Axes: >
```

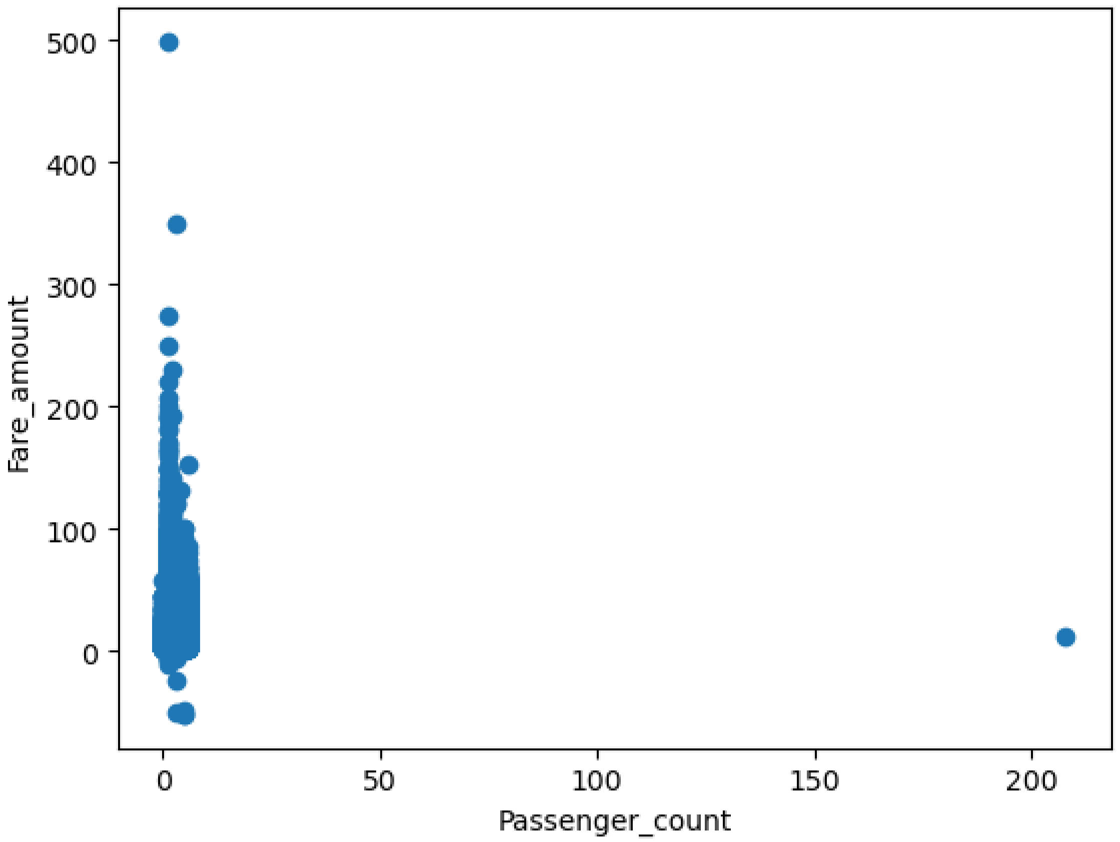


In [49]:
```
df.isnull().sum()
```

Out[49]:
```
fare_amount        0
passenger_count    0
date               0
year               0
month              0
day                0
time               0
dtype: int64
```

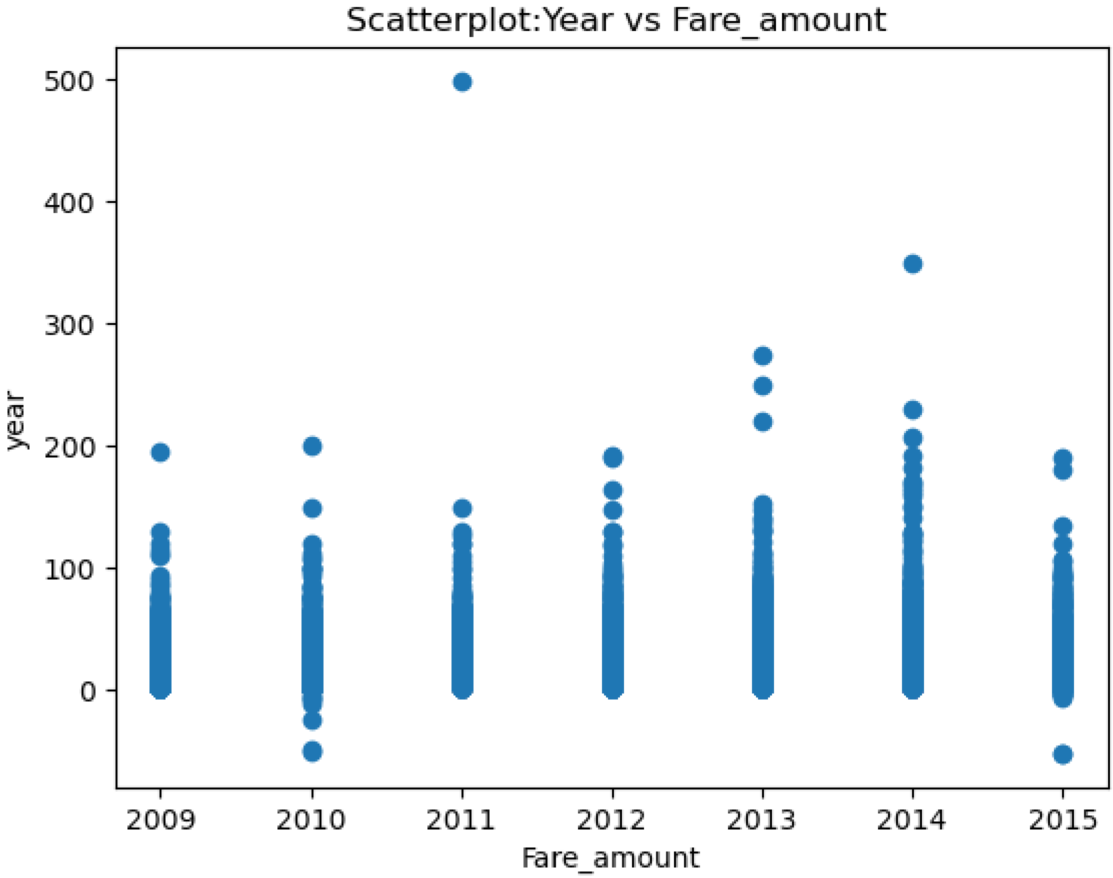
In [50]:

```
plt.scatter(df['passenger_count'],df['fare_amount'])
plt.xlabel('Passenger_count')
plt.ylabel('Fare_amount')
plt.show()
```

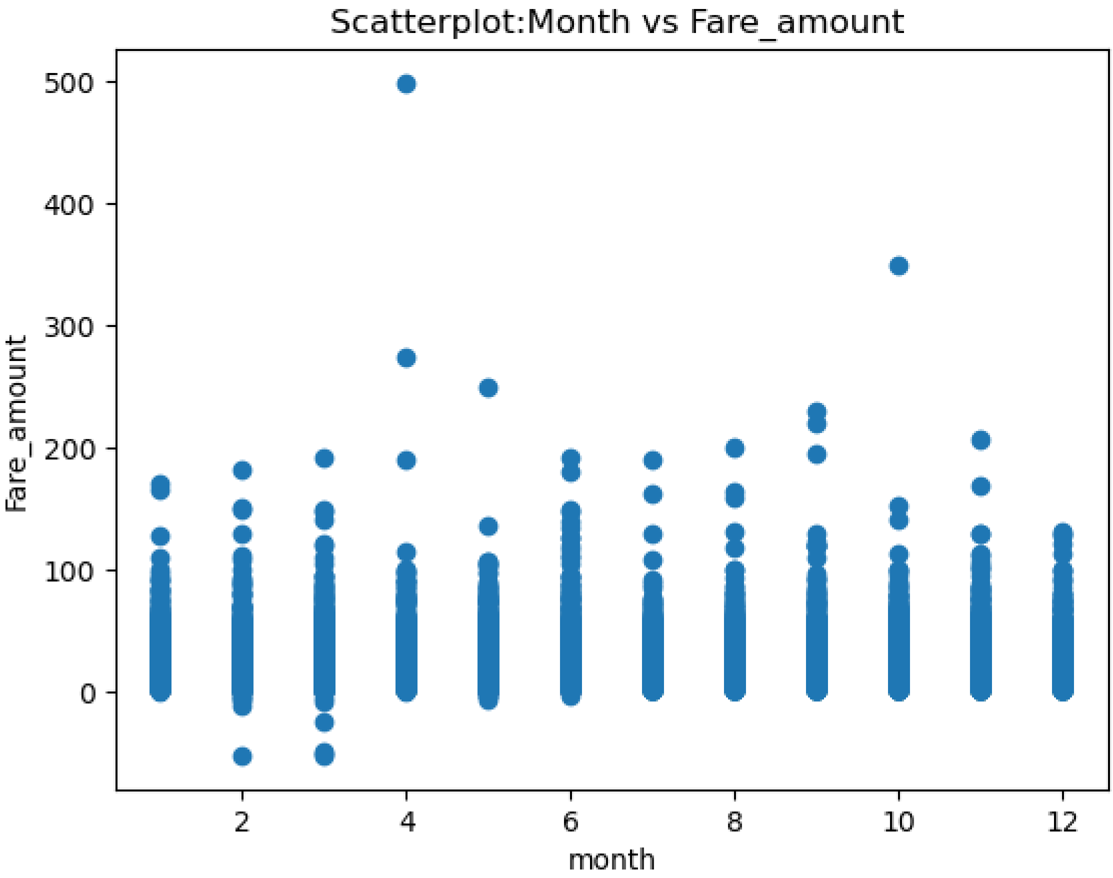


In [51]:

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

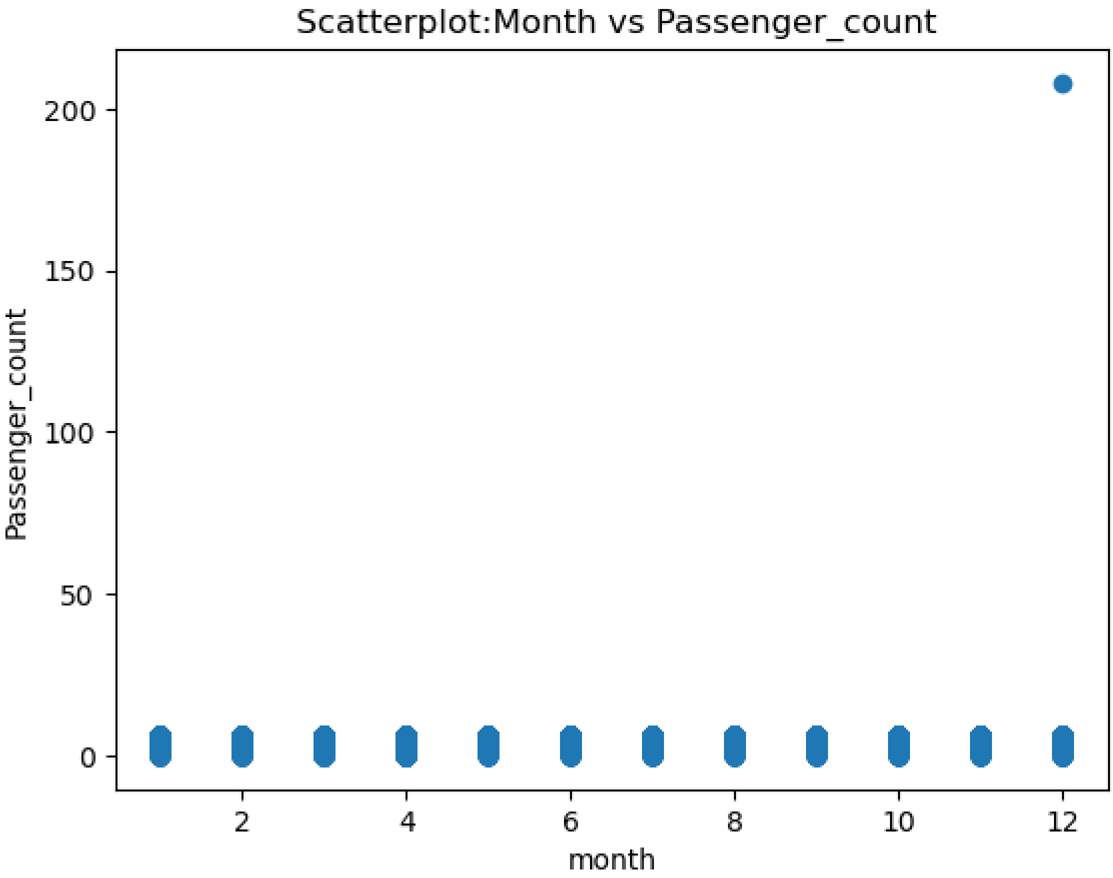


In [52]:
```
plt.scatter(df['month'],df['fare_amount'])
plt.xlabel('month')
plt.ylabel('Fare_amount')
plt.title(' Scatterplot:Month vs Fare_amount')
plt.show()
```

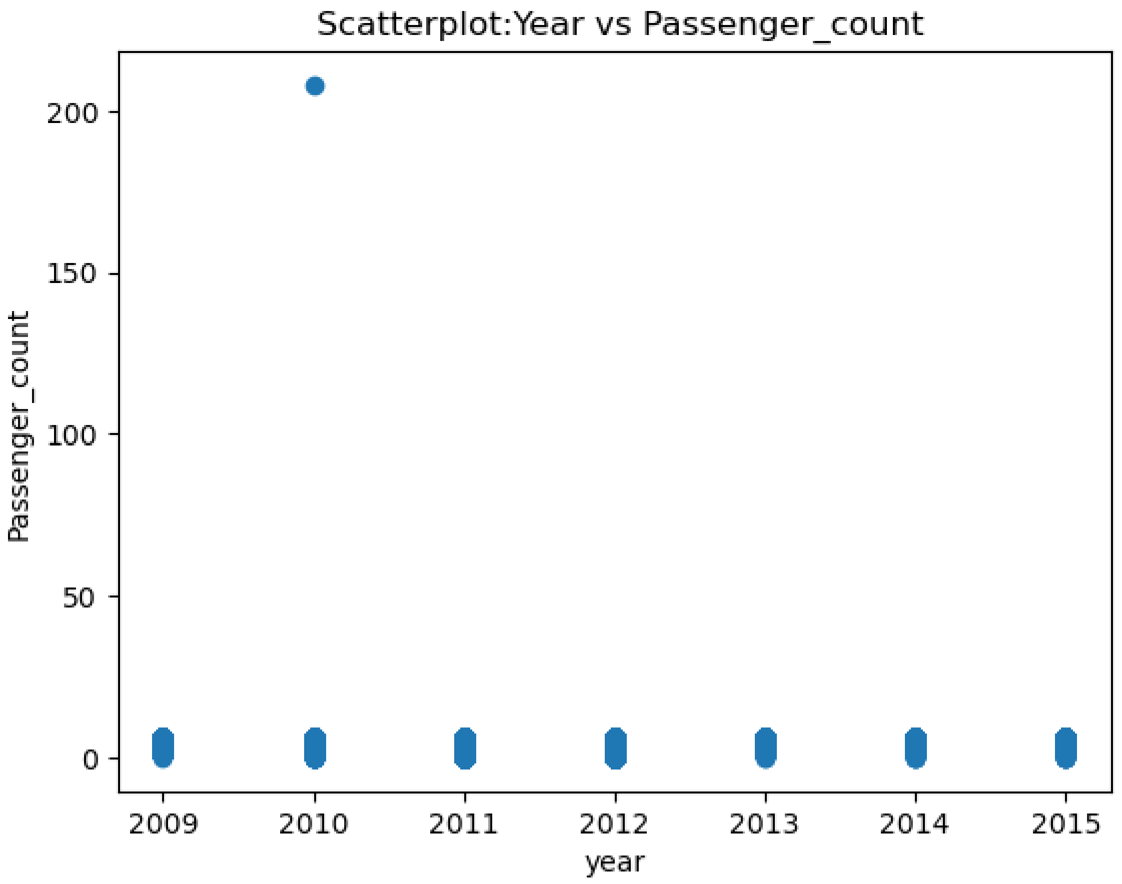


In [53]:

```
plt.scatter(df['month'],df['passenger_count'])
plt.xlabel('month')
plt.ylabel('Passenger_count')
plt.title(' Scatterplot:Month vs Passenger_count')
plt.show()
```



In [54]:
```
plt.scatter(df['year'],df['passenger_count'])
plt.xlabel('year')
plt.ylabel('Passenger_count')
plt.title(' Scatterplot:Year vs Passenger_count')
plt.show()
```


Scatterplot:Year vs Passenger_count
Passenger_count
year