In [1]:

```
import datetime
import matplotlib.pyplot as plt
from matplotlib import gridspec
import matplotlib.cm as cm
import numpy as np
import pandas as pd
import seaborn as sns
plt.rcParams["figure.figsize"] = (10,5)

import warnings
sns.set_style("whitegrid")
warnings.filterwarnings('ignore')
```

In [2]:

```
#Loading the data from csv files.
df=pd.read_csv('Rilders_List2020-2.csv')
df.head()
```

Out[2]:

	LeadID	State	City	RiderName	LL EMP Code	RiderNumber	DOJ	Client	
0	31	Karnataka	Bengaluru	SRIKANTH P	LLBBC001	6363376901	04-04-2019	Bigbasket Bangalore	
1	32	Karnataka	Bengaluru	SUPREEM .	LLBBC003	7483505921	4/16/2019	Bigbasket Bangalore	
2	33	Tamil Nadu	Chennai	Silambarasan V	LLBBC004	8695084040	03-01-2019	Bigbasket Chennai	C
3	35	Karnataka	Bengaluru	VEERESH U	LLBBC011	9110444430	6/23/2019	Bigbasket Bangalore	
4	37	Karnataka	Bengaluru	Harish M	LLBBC027	9008816586	6/17/2019	Bigbasket Bangalore	

In [3]:

```
df.shape
```

Out[3]:

```
(7349, 15)
```

In [4]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7349 entries, 0 to 7348
Data columns (total 15 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   LeadID                7349 non-null   int64  
 1   State                 7299 non-null   object 
 2   City                  7299 non-null   object 
 3   RiderName             7349 non-null   object 
 4   LL EMP Code           7344 non-null   object 
 5   RiderNumber           7349 non-null   object 
 6   DOJ                   7257 non-null   object 
 7   Client                7299 non-null   object 
 8   Hub                   7299 non-null   object 
 9   Previous_Status       7295 non-null   object 
 10  Current_Status        7294 non-null   object 
 11  Status_Changed        7294 non-null   object 
 12  Modified_by           7294 non-null   object 
 13  ReasonForLeaving      5737 non-null   float64
 14  RelevingDate_Entered  5769 non-null   object 
dtypes: float64(1), int64(1), object(13)
memory usage: 861.3+ KB
```

In [5]:

```
df.isnull().sum()
```

Out[5]:

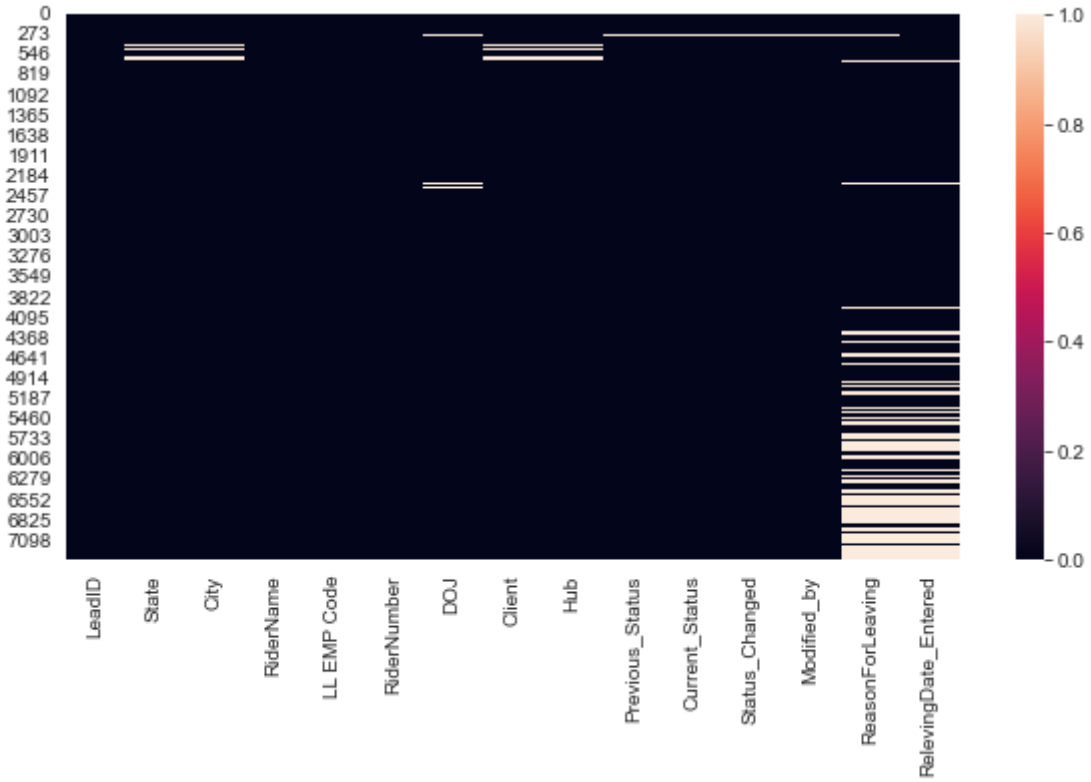
```
LeadID                   0
State                   50
City                    50
RiderName                0
LL EMP Code              5
RiderNumber              0
DOJ                     92
Client                  50
Hub                     50
Previous_Status         54
Current_Status          55
Status_Changed          55
Modified_by             55
ReasonForLeaving      1612
RelevingDate_Entered  1580
dtype: int64
```

In [6]:

```
sns.heatmap(df.isnull())
```

Out[6]:

```
<AxesSubplot:>
```



In [7]:

```
df1=df.copy()
```

In [8]:

```
df1.shape
```

Out[8]:

```
(7349, 15)
```

In [9]:

```
def format_color_groups(df1):
    colors = ['violet', 'pink']
    x = df1.copy()
    factors = list(x['RiderNumber'].unique())
    i = 0
    for factor in factors:
        style = f'background-color: {colors[i]}'
        x.loc[x['RiderNumber'] == factor, :] = style
        i = not i
    return x
```

In [10]:

```
# df1.style.apply(format_color_groups, axis=None)
```

In [11]:

```
df1=df1.drop_duplicates(subset='RiderNumber').copy()
```

In [12]:

```
df1.head(5)
```

Out[12]:

	LeadID	State	City	RiderName	LL EMP Code	RiderNumber	DOJ	Client	
0	31	Karnataka	Bengaluru	SRIKANTH P	LLBBC001	6363376901	04-04-2019	Bigbasket Bangalore	
1	32	Karnataka	Bengaluru	SUPREEM .	LLBBC003	7483505921	4/16/2019	Bigbasket Bangalore	
2	33	Tamil Nadu	Chennai	Silambarasan V	LLBBC004	8695084040	03-01-2019	Bigbasket Chennai	C
3	35	Karnataka	Bengaluru	VEERESH U	LLBBC011	9110444430	6/23/2019	Bigbasket Bangalore	
4	37	Karnataka	Bengaluru	Harish M	LLBBC027	9008816586	6/17/2019	Bigbasket Bangalore	

In [13]:

```
df1[df1['DOJ'] == '0000-00-00'].head()
```

Out[13]:

	LeadID	State	City	RiderName	LL EMP Code	RiderNumber	DOJ	Client	
42	94	Tamil Nadu	Chennai	Karthick Subbaih	LLC0072	8220322678	0000-00-00	Bigbasket Chennai	V
56	123	Telangana	Hyderabad	Nohan Kumar	LLH0001	8431291109	0000-00-00	Bigbasket Hyderabad	N
124	278	Tamil Nadu	Chennai	Vino Th .	LLC0162	9677059440	0000-00-00	Bigbasket Chennai	Chi
125	284	Karnataka	Bengaluru	Manjunath Dhage	LLB0257	7795166002	0000-00-00	Bigbasket Bangalore	
127	287	Tamil Nadu	Chennai	Senthilkumar. V	LLC0379	9790914122	0000-00-00	Bigbasket Chennai	V

In [14]:

```
df1=df1.drop([2,1349,1428,4400,5748],axis=0).copy()
df1.head(10)
```

Out[14]:

	LeadID	State	City	RiderName	LL EMP Code	RiderNumber	DOJ	Client	
0	31	Karnataka	Bengaluru	SRIKANTH P	LLBBC001	6363376901	04-04-2019	Bigbasket Bangalore	
1	32	Karnataka	Bengaluru	SUPREEM .	LLBBC003	7483505921	4/16/2019	Bigbasket Bangalore	
3	35	Karnataka	Bengaluru	VEERESH U	LLBBC011	9110444430	6/23/2019	Bigbasket Bangalore	
4	37	Karnataka	Bengaluru	Harish M	LLBBC027	9008816586	6/17/2019	Bigbasket Bangalore	
5	38	Karnataka	Bengaluru	ANANDH .	LLBBC028	9095272783	2/26/2019	TOW Bangalore	H
6	39	Karnataka	Bengaluru	Deepak Badiya	LLBBC029	9382654318	05-10-2022	BB_Now Bangalore	
7	44	Karnataka	Bengaluru	Arun Kumar .	LLBBC046	9535331836	7/17/2019	Bigbasket Bangalore	
8	46	Karnataka	Bengaluru	Naveen Kumar V	LLBBC088	7349716619	08-07-2019	Bigbasket Bangalore	
9	47	Karnataka	Bengaluru	Lokesh S	LLBBC092	8951907480	8/13/2019	Bigbasket Bangalore	
10	48	Karnataka	Bengaluru	BANAJ KUMAR SAHOO .	LLBBA011	9337077633	07-04-2019	Bigbasket Bangalore	

In [15]:

```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 6680 entries, 0 to 7348
Data columns (total 15 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   LeadID                6680 non-null   int64  
 1   State                 6631 non-null   object 
 2   City                  6631 non-null   object 
 3   RiderName             6680 non-null   object 
 4   LL EMP Code           6675 non-null   object 
 5   RiderNumber           6680 non-null   object 
 6   DOJ                   6588 non-null   object 
 7   Client                6631 non-null   object 
 8   Hub                   6631 non-null   object 
 9   Previous_Status       6626 non-null   object 
 10  Current_Status        6625 non-null   object 
 11  Status_Changed        6625 non-null   object 
 12  Modified_by           6625 non-null   object 
 13  ReasonForLeaving      5111 non-null   float64
 14  RelevingDate_Entered  5143 non-null   object 
dtypes: float64(1), int64(1), object(13)
memory usage: 835.0+ KB
```

In [16]:

```
df1['DOJ'] = pd.to_datetime(df1['DOJ'],errors='coerce')
```

In [17]:

```
df1['RelevingDate_Entered'] = pd.to_datetime(df1['RelevingDate_Entered'], errors='coerce')
```

In [18]:

```
df1['Status_Changed'] = pd.to_datetime(df1['Status_Changed'], errors='ignore')
```

In [19]:

```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 6680 entries, 0 to 7348
Data columns (total 15 columns):
 #   Column                Non-Null Count  Dtype         
---  ------                --------------  -----         
 0   LeadID                6680 non-null   int64         
 1   State                 6631 non-null   object        
 2   City                  6631 non-null   object        
 3   RiderName             6680 non-null   object        
 4   LL EMP Code           6675 non-null   object        
 5   RiderNumber           6680 non-null   object        
 6   DOJ                   5493 non-null   datetime64[ns]
 7   Client                6631 non-null   object        
 8   Hub                   6631 non-null   object        
 9   Previous_Status       6626 non-null   object        
 10  Current_Status        6625 non-null   object        
 11  Status_Changed        6625 non-null   datetime64[ns]
 12  Modified_by           6625 non-null   object        
 13  ReasonForLeaving      5111 non-null   float64       
 14  RelevingDate_Entered  3068 non-null   datetime64[ns]
dtypes: datetime64[ns](3), float64(1), int64(1), object(10)
memory usage: 835.0+ KB
```

In [20]:

```
df2= df1.drop(['LeadID','State','City','LL EMP Code','RiderNumber',
    'Client','Hub','Previous_Status','Current_Status','Modified_by','ReasonForLeaving'], ax
df2.head
```

Out[20]:

```
<bound method NDFrame.head of             RiderName        DOJ Status_Chang
ed RelevingDate_Entered
0           SRIKANTH P 2019-04-04     2020-05-19           2020-04-02
1            SUPREEM . 2019-04-16     2020-05-19           2020-02-20
3            VEERESH U 2019-06-23     2020-11-10                  NaT
4             Harish M 2019-06-17            NaT                  NaT
5            ANANDH . 2019-02-26     2020-07-20                  NaT
...                ...        ...            ...                  ...
7344    Kranthi Kumara 2022-06-14     2022-06-14                  NaT
7345         Darshan P 2022-06-14     2022-06-14                  NaT
7346       Madhu Kumar 2022-06-14     2022-06-14                  NaT
7347            Ravi S 2022-06-15     2022-06-16                  NaT
7348  Aravinda reddy K 2022-06-15     2022-06-15                  NaT

[6680 rows x 4 columns]>
```

In [21]:

```
df2.head()
```

Out[21]:

	RiderName	DOJ	Status_Changed	RelevingDate_Entered
0	SRIKANTH P	2019-04-04	2020-05-19	2020-04-02
1	SUPREEM .	2019-04-16	2020-05-19	2020-02-20
3	VEERESH U	2019-06-23	2020-11-10	NaT
4	Harish M	2019-06-17	NaT	NaT
5	ANANDH .	2019-02-26	2020-07-20	NaT

In [22]:

```
df2.sample(5)
```

Out[22]:

	RiderName	DOJ	Status_Changed	RelevingDate_Entered
5136	Vijay Choudhary	2022-02-23	2022-03-26	2022-03-13
3138	Kiran H G	2021-11-20	2021-12-27	2022-04-04
5785	Supriyo Chanda	2022-03-26	2022-04-05	2022-04-05
3848	Sharath HB	2021-12-29	2021-12-18	2022-02-11
1163	Siva Raj	NaT	2021-11-02	NaT

In [23]:

```
df2.isna().sum()
```

Out[23]:

```
RiderName                 0
DOJ                    1187
Status_Changed           55
RelevingDate_Entered    3612
dtype: int64
```

In [24]:

```
df2.isna().any(axis=1).value_counts()
df2.isna().sum()
```

Out[24]:

```
RiderName                 0
DOJ                    1187
Status_Changed           55
RelevingDate_Entered    3612
dtype: int64
```

In [25]:

```
df3=df2.dropna().copy()
df3
```

Out[25]:

	RiderName	DOJ	Status_Changed	RelevingDate_Entered
0	SRIKANTH P	2019-04-04	2020-05-19	2020-04-02
1	SUPREEM .	2019-04-16	2020-05-19	2020-02-20
6	Deepak Badiya	2022-05-10	2022-05-14	2022-05-14
8	Naveen Kumar V	2019-08-07	2022-02-10	2022-02-10
10	BANAJ KUMAR SAHOO .	2019-07-04	2020-07-20	2020-04-29
...	...	...	...	...
7287	G Reddy Vasu	2022-06-10	2022-06-13	2022-06-13
7289	Thomson Thaddiues	2022-06-10	2022-06-13	2022-06-13
7290	Bharath N	2022-06-10	2022-06-13	2022-06-13
7292	Vijay R	2022-06-10	2022-06-15	2022-06-15
7325	Ranganatha C	2022-06-13	2022-06-15	2022-06-15

2675 rows × 4 columns

In [26]:

```
df3.shape
```

Out[26]:

```
(2675, 4)
```

In [27]:

```
df3['no_of_working_days']=(df3['RelevingDate_Entered']-df3['DOJ']).dt.days
```

In [28]:

```
df3
```

Out[28]:

	RiderName	DOJ	Status_Changed	RelevingDate_Entered	no_of_working_days
0	SRIKANTH P	2019-04-04	2020-05-19	2020-04-02	364
1	SUPREEM .	2019-04-16	2020-05-19	2020-02-20	310
6	Deepak Badiya	2022-05-10	2022-05-14	2022-05-14	4
8	Naveen Kumar V	2019-08-07	2022-02-10	2022-02-10	918
10	BANAJ KUMAR SAHOO .	2019-07-04	2020-07-20	2020-04-29	300
...	...	...	...	...	...
7287	G Reddy Vasu	2022-06-10	2022-06-13	2022-06-13	3
7289	Thomson Thaddiues	2022-06-10	2022-06-13	2022-06-13	3
7290	Bharath N	2022-06-10	2022-06-13	2022-06-13	3
7292	Vijay R	2022-06-10	2022-06-15	2022-06-15	5
7325	Ranganatha C	2022-06-13	2022-06-15	2022-06-15	2

2675 rows × 5 columns

In [29]:

```
df3[df3['no_of_working_days'] <= 0]
```

Out[29]:

	RiderName	DOJ	Status_Changed	RelevingDate_Entered	no_of_working_days
16	Rudresh R	2021-10-20	2021-11-10	2020-03-10	-589
53	Maria Victas .	2021-07-22	2022-04-06	2020-02-24	-514
60	Nitin s	2022-03-18	2020-10-24	2020-06-25	-631
69	VIGNESH D	2021-01-18	2021-03-10	2020-02-28	-325
72	Dineshkumar R	2022-04-03	2022-04-07	2020-03-16	-748
...	...	...	...	...	...
7045	Devana Dharaneedhar Reddy	2022-05-31	2022-06-01	2022-05-31	0
7162	Karthikeyan K	2022-06-07	2022-06-16	2022-05-28	-10
7163	S.Gopi Krishnan	2022-06-07	2022-06-16	2022-05-31	-7
7167	Srihari R	2022-07-04	2022-06-07	2022-06-06	-28
7180	B.Aravinda Pandian	2022-06-07	2022-06-16	2022-05-31	-7

250 rows × 5 columns

In [30]:

```
df3 = df3[df3['no_of_working_days'] >= 0]
```

In [31]:

```
df3.shape
```

Out[31]:

```
(2481, 5)
```

In [32]:

```
df3.head()
```

Out[32]:

	RiderName	DOJ	Status_Changed	RelevingDate_Entered	no_of_working_days
0	SRIKANTH P	2019-04-04	2020-05-19	2020-04-02	364
1	SUPREEM .	2019-04-16	2020-05-19	2020-02-20	310
6	Deepak Badiya	2022-05-10	2022-05-14	2022-05-14	4
8	Naveen Kumar V	2019-08-07	2022-02-10	2022-02-10	918
10	BANAJ KUMAR SAHOO .	2019-07-04	2020-07-20	2020-04-29	300

In [33]:

```
df3.shape
```

Out[33]:

```
(2481, 5)
```

In [34]:

```
min(df3['no_of_working_days'])
```

Out[34]:

```
0
```

In [35]:

```
max(df3['no_of_working_days'])
```

Out[35]:

```
976
```

In [36]:

```
sum(df3['no_of_working_days'])
```

Out[36]:

```
170754
```

In [37]:

```
df3['Riders_Joined_by_month'] = pd.DatetimeIndex(df3['DOJ']).month

df3['Riders_Quitted_by_month'] = pd.DatetimeIndex(df3['RelevingDate_Entered']).month
df3['Riders_Joined_by_year'] = pd.DatetimeIndex(df3['DOJ']).year
df3['Riders_Quitted_by_year'] = pd.DatetimeIndex(df3['RelevingDate_Entered']).year
```

In [38]:

```
df3['Riders_Joined_by_weeks']=pd.to_datetime(df3["DOJ"]).dt.week
df3['Riders_Quitted_by_weeks']=pd.to_datetime(df3["RelevingDate_Entered"]).dt.week
```

In [39]:

```
df3.shape
```

Out[39]:

```
(2481, 11)
```

No of joiner and quitter by Month wise

In [40]:

```
df_joinbymonth=pd.DataFrame(df3.groupby(pd.Grouper(key='DOJ', freq='M')).size().sort_index(
df_joinbymonth.reset_index(inplace=True)
```

In [41]:

```
df_joinbymonth
```

Out[41]:

	DOJ	0
0	2019-04-30	2
1	2019-05-31	0
2	2019-06-30	2
3	2019-07-31	1
4	2019-08-31	4
5	2019-09-30	2
6	2019-10-31	5
7	2019-11-30	4
8	2019-12-31	5
9	2020-01-31	6
10	2020-02-29	18
11	2020-03-31	3
12	2020-04-30	9
13	2020-05-31	5
14	2020-06-30	5
15	2020-07-31	5
16	2020-08-31	2
17	2020-09-30	11
18	2020-10-31	4
19	2020-11-30	8
20	2020-12-31	5
21	2021-01-31	12
22	2021-02-28	4
23	2021-03-31	15
24	2021-04-30	7
25	2021-05-31	13
26	2021-06-30	13
27	2021-07-31	34
28	2021-08-31	46
29	2021-09-30	108
30	2021-10-31	102
31	2021-11-30	139
32	2021-12-31	208
33	2022-01-31	426
34	2022-02-28	383

	DOJ	0
35	2022-03-31	313
36	2022-04-30	310
37	2022-05-31	212
38	2022-06-30	30

In [42]:

```
df_quitbymonth=pd.DataFrame(df3.groupby(pd.Grouper(key='RelevingDate_Entered', freq='M')).s
```

In [43]:

```
df_quitbymonth.reset_index(inplace=True)
```

In [44]:

```
df_cumulative_month=df_joinbymonth.merge(df_quitbymonth, left_on='DOJ', right_on='RelevingD
df_cumulative_month
```

Out[44]:

	DOJ	0_x	RelevingDate_Entered	0_y
0	2019-04-30	2	NaT	NaN
1	2019-05-31	0	NaT	NaN
2	2019-06-30	2	NaT	NaN
3	2019-07-31	1	NaT	NaN
4	2019-08-31	4	NaT	NaN
5	2019-09-30	2	NaT	NaN
6	2019-10-31	5	NaT	NaN
7	2019-11-30	4	NaT	NaN
8	2019-12-31	5	NaT	NaN
9	2020-01-31	6	NaT	NaN
10	2020-02-29	18	2020-02-29	9.0
11	2020-03-31	3	2020-03-31	20.0
12	2020-04-30	9	2020-04-30	5.0
13	2020-05-31	5	2020-05-31	9.0
14	2020-06-30	5	2020-06-30	5.0
15	2020-07-31	5	2020-07-31	2.0
16	2020-08-31	2	2020-08-31	0.0
17	2020-09-30	11	2020-09-30	1.0
18	2020-10-31	4	2020-10-31	0.0
19	2020-11-30	8	2020-11-30	0.0
20	2020-12-31	5	2020-12-31	0.0
21	2021-01-31	12	2021-01-31	0.0
22	2021-02-28	4	2021-02-28	0.0
23	2021-03-31	15	2021-03-31	0.0
24	2021-04-30	7	2021-04-30	1.0
25	2021-05-31	13	2021-05-31	0.0
26	2021-06-30	13	2021-06-30	3.0
27	2021-07-31	34	2021-07-31	1.0
28	2021-08-31	46	2021-08-31	0.0
29	2021-09-30	108	2021-09-30	3.0
30	2021-10-31	102	2021-10-31	5.0
31	2021-11-30	139	2021-11-30	16.0
32	2021-12-31	208	2021-12-31	73.0
33	2022-01-31	426	2022-01-31	261.0

	DOJ	0_x	RelevingDate_Entered	0_y
34	2022-02-28	383	2022-02-28	594.0
35	2022-03-31	313	2022-03-31	426.0
36	2022-04-30	310	2022-04-30	380.0
37	2022-05-31	212	2022-05-31	531.0
38	2022-06-30	30	2022-06-30	136.0

In [45]:

```
df_cumulative_month=df_cumulative_month.drop('RelevingDate_Entered',axis=1)
df_cumulative_month= df_cumulative_month.rename(columns={'0_x':'joiners', '0_y':'quitters'}
```

In [46]:

```
df_cumulative_month=df_cumulative_month.rename(columns={'DOJ':'Month'})
```

In [47]:

```
df_cumulative_month.fillna(0, inplace=True)
```

In [48]:

```
df_cumulative_month
```

Out[48]:

	Month	joiners	quitters
0	2019-04-30	2	0.0
1	2019-05-31	0	0.0
2	2019-06-30	2	0.0
3	2019-07-31	1	0.0
4	2019-08-31	4	0.0
5	2019-09-30	2	0.0
6	2019-10-31	5	0.0
7	2019-11-30	4	0.0
8	2019-12-31	5	0.0
9	2020-01-31	6	0.0
10	2020-02-29	18	9.0
11	2020-03-31	3	20.0
12	2020-04-30	9	5.0
13	2020-05-31	5	9.0
14	2020-06-30	5	5.0
15	2020-07-31	5	2.0
16	2020-08-31	2	0.0
17	2020-09-30	11	1.0
18	2020-10-31	4	0.0
19	2020-11-30	8	0.0
20	2020-12-31	5	0.0
21	2021-01-31	12	0.0
22	2021-02-28	4	0.0
23	2021-03-31	15	0.0
24	2021-04-30	7	1.0
25	2021-05-31	13	0.0
26	2021-06-30	13	3.0
27	2021-07-31	34	1.0
28	2021-08-31	46	0.0
29	2021-09-30	108	3.0
30	2021-10-31	102	5.0
31	2021-11-30	139	16.0
32	2021-12-31	208	73.0
33	2022-01-31	426	261.0
34	2022-02-28	383	594.0

	Month	joiners	quitters
35	2022-03-31	313	426.0
36	2022-04-30	310	380.0
37	2022-05-31	212	531.0
38	2022-06-30	30	136.0

In [49]:

```
df_cumulative_month['cumulative_joiners'] = df_cumulative_month['joiners'].cumsum()
```

In [50]:

```
df_cumulative_month['cumulative_quitters'] = df_cumulative_month['quitters'].cumsum()
```

In [51]:

```
df_cumulative_month
```

Out[51]:

	Month	joiners	quitters	cumulative_joiners	cumulative_quitters
0	2019-04-30	2	0.0	2	0.0
1	2019-05-31	0	0.0	2	0.0
2	2019-06-30	2	0.0	4	0.0
3	2019-07-31	1	0.0	5	0.0
4	2019-08-31	4	0.0	9	0.0
5	2019-09-30	2	0.0	11	0.0
6	2019-10-31	5	0.0	16	0.0
7	2019-11-30	4	0.0	20	0.0
8	2019-12-31	5	0.0	25	0.0
9	2020-01-31	6	0.0	31	0.0
10	2020-02-29	18	9.0	49	9.0

In [52]:

```
df3['no_of_working_days'].std()
```

Out[52]:

```
105.72875888700554
```

In [53]:

```
df3['no_of_working_days'].quantile([.1, .2,.3,.4,.5,.6,.7,.8,.9])
```

Out[53]:

```
0.1      4.0
0.2      9.0
0.3     16.0
0.4     23.0
0.5     32.0
0.6     46.0
0.7     65.0
0.8    102.0
0.9    159.0
Name: no_of_working_days, dtype: float64
```

In [54]:

```
df3['no_of_working_days'].agg(['size','sum','mean','max','min','median'])
```

Out[54]:

```
size        2481.000000
sum       170754.000000
mean          68.824667
max          976.000000
min            0.000000
median        32.000000
Name: no_of_working_days, dtype: float64
```

No of joiner and quitter by Week wise

In [55]:

```
df_joinbyweek=pd.DataFrame(df3.groupby(pd.Grouper(key='DOJ', freq='W')).size().sort_index(a
df_joinbyweek.reset_index(inplace=True)
```

In [56]:

```
df_quitbyweek=pd.DataFrame(df3.groupby(pd.Grouper(key='RelevingDate_Entered', freq='W')).si
```

In [57]:

```
# df3[df3['no_of_working_days'] == 976]
```

In [58]:

```
df3.isnull().sum()
```

Out[58]:

```
RiderName                 0
DOJ                       0
Status_Changed            0
RelevingDate_Entered      0
no_of_working_days        0
Riders_Joined_by_month    0
Riders_Quitted_by_month   0
Riders_Joined_by_year     0
Riders_Quitted_by_year    0
Riders_Joined_by_weeks    0
Riders_Quitted_by_weeks   0
dtype: int64
```

In [59]:

```
df_quitbyweek.reset_index(inplace=True)
```

In [60]:

```
df_cumulative_week=df_joinbyweek.merge(df_quitbyweek, left_on='DOJ', right_on='RelevingDate
df_cumulative_week
```

Out[60]:

	DOJ	0_x	RelevingDate_Entered	0_y
0	2019-04-07	1	NaT	NaN
1	2019-04-14	0	NaT	NaN
2	2019-04-21	1	NaT	NaN
3	2019-04-28	0	NaT	NaN
4	2019-05-05	0	NaT	NaN
...	...	...	...	...
163	2022-05-22	42	2022-05-22	87.0
164	2022-05-29	47	2022-05-29	103.0
165	2022-06-05	20	2022-06-05	92.0
166	2022-06-12	17	2022-06-12	54.0
167	2022-06-19	2	2022-06-19	37.0

168 rows × 4 columns

In [61]:

```
df_cumulative_week=df_cumulative_week.drop('RelevingDate_Entered',axis=1)
df_cumulative_week= df_cumulative_week.rename(columns={'0_x':'joiners', '0_y':'quitters'})
```

In [62]:

```
df_cumulative_week=df_cumulative_week.rename(columns={'DOJ':'Month'})
```

In [64]:

```
df_cumulative_week.fillna(0, inplace=True)
```

In [63]:

```
df_cumulative_week['quitters']=(df_cumulative_week['quitters']*-1)
```

In [64]:

```
df_cumulative_week
```

Out[64]:

	Month	joiners	quitters
0	2019-04-07	1	NaN
1	2019-04-14	0	NaN
2	2019-04-21	1	NaN
3	2019-04-28	0	NaN
4	2019-05-05	0	NaN
...	...	...	...
163	2022-05-22	42	-87.0
164	2022-05-29	47	-103.0
165	2022-06-05	20	-92.0
166	2022-06-12	17	-54.0
167	2022-06-19	2	-37.0

168 rows × 3 columns

In [66]:

```
df_cumulative_week['cumulative_joiners'] = df_cumulative_week['joiners'].cumsum()
```

In [67]:

```
df_cumulative_week['cumulative_quitters'] = df_cumulative_week['quitters'].cumsum()
```

In [68]:

```
df_cumulative_week
```

Out[68]:

	Month	joiners	quitters	cumulative_joiners	cumulative_quitters
0	2019-04-07	1	0.0	1	0.0
1	2019-04-14	0	0.0	1	0.0
2	2019-04-21	1	0.0	2	0.0
3	2019-04-28	0	0.0	2	0.0
4	2019-05-05	0	0.0	2	0.0
...	...	...	...	...	...
163	2022-05-22	42	87.0	2395	2195.0
164	2022-05-29	47	103.0	2442	2298.0
165	2022-06-05	20	92.0	2462	2390.0
166	2022-06-12	17	54.0	2479	2444.0
167	2022-06-19	2	37.0	2481	2481.0

168 rows × 5 columns

In [69]:

```
df_cumulative_week['joiners'].std()
```

Out[69]:

```
27.031800570573434
```

In [70]:

```
df_cumulative_week['joiners'].quantile([.1, .2,.3,.4,.5,.6,.7,.8,.9])
```

Out[70]:

```
0.1     0.0
0.2     0.0
0.3     1.0
0.4     1.0
0.5     2.0
0.6     3.0
0.7     6.0
0.8    24.2
0.9    59.9
Name: joiners, dtype: float64
```

In [71]:

```
df_cumulative_week['joiners'].agg(['size','sum','mean','max','min'])
```

Out[71]:

```
size     168.000000
sum     2481.000000
mean      14.767857
max      126.000000
min        0.000000
Name: joiners, dtype: float64
```

Examine how many riders join in the following weeks, months, and years

In [72]:

```
sum(df3['Riders_Quitted_by_weeks'])
```

Out[72]:

```
34302
```

In [73]:

```
sum(df3['Riders_Quitted_by_weeks'])/365
```

Out[73]:

```
93.97808219178083
```

In [74]:

```
sum(df3['Riders_Quitted_by_weeks'])/52
```

Out[74]:

```
659.6538461538462
```

In [75]:

```
df3['Riders_Joined_by_weeks'].agg(['sum'])
```

Out[75]:

```
sum    48415
Name: Riders_Joined_by_weeks, dtype: int64
```

In [76]:

```
sum(df3['Riders_Joined_by_weeks'])/365
```

Out[76]:

```
132.64383561643837
```

In [77]:

```
sum(df3['Riders_Joined_by_weeks'])/(52)
```

Out[77]:

```
931.0576923076923
```

In [78]:

```
df_cumulative_agg=df_cumulative_week.drop('Month',axis=1)
df_cumulative_agg=df_cumulative_agg.groupby("joiners").agg(['count','median','mean'])
```

In [79]:

```
df_cumulative_agg
```

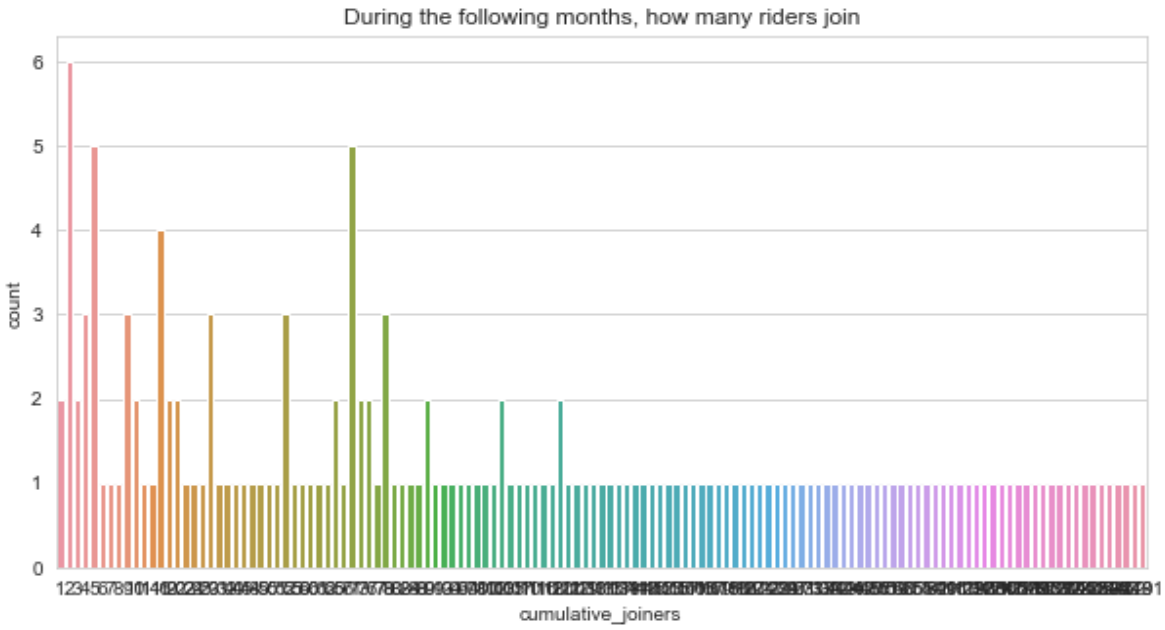
Out[79]:

	quitters			cumulative_joiners			cumulative_quitters		
	count	median	mean	count	median	mean	count	median	mean
joiners									
0	37	0.0	0.297297	37	16.0	34.729730	37	0.0	17.405405
1	37	0.0	0.702703	37	60.0	62.243243	37	33.0	26.648649
2	23	0.0	1.695652	23	100.0	199.391304	23	51.0	147.956522
3	9	0.0	1.000000	9	65.0	74.888889	9	38.0	31.222222
4	5	0.0	0.200000	5	88.0	94.400000	5	51.0	37.400000
5	5	1.0	1.400000	5	116.0	115.600000	5	51.0	41.800000
6	3	0.0	0.000000	3	175.0	171.000000	3	55.0	54.333333
7	2	0.0	0.000000	2	166.5	166.500000	2	53.5	53.500000
8	3	0.0	0.000000	3	212.0	166.333333	3	56.0	37.333333
10	1	0.0	0.000000	1	222.0	222.000000	1	56.0	56.000000
11	1	0.0	0.000000	1	197.0	197.000000	1	56.0	56.000000
12	1	0.0	0.000000	1	273.0	273.000000	1	56.0	56.000000
14	1	0.0	0.000000	1	261.0	261.000000	1	56.0	56.000000
16	1	0.0	0.000000	1	442.0	442.000000	1	64.0	64.000000
17	2	27.0	27.000000	2	1359.0	1359.000000	2	1250.0	1250.000000
18	1	0.0	0.000000	1	460.0	460.000000	1	64.0	64.000000
20	1	92.0	92.000000	1	2462.0	2462.000000	1	2390.0	2390.000000
23	1	4.0	4.000000	1	426.0	426.000000	1	64.0	64.000000
25	1	9.0	9.000000	1	511.0	511.000000	1	73.0	73.000000
26	2	0.5	0.500000	2	412.5	412.500000	2	61.0	61.000000
30	1	7.0	7.000000	1	655.0	655.000000	1	91.0	91.000000
31	2	1.0	1.000000	2	496.5	496.500000	2	70.0	70.000000
33	1	1.0	1.000000	1	372.0	372.000000	1	59.0	59.000000
35	1	4.0	4.000000	1	625.0	625.000000	1	84.0	84.000000
40	1	1.0	1.000000	1	313.0	313.000000	1	57.0	57.000000
42	1	87.0	87.000000	1	2395.0	2395.000000	1	2195.0	2195.000000
46	1	206.0	206.000000	1	2241.0	2241.000000	1	1895.0	1895.000000
47	2	96.5	96.500000	2	2397.5	2397.500000	2	2203.0	2203.000000
48	2	7.0	7.000000	2	660.5	660.500000	2	91.5	91.500000
57	1	58.0	58.000000	1	819.0	819.000000	1	162.0	162.000000
59	1	5.0	5.000000	1	714.0	714.000000	1	96.0	96.000000
62	1	177.0	177.000000	1	1667.0	1667.000000	1	1163.0	1163.000000
65	3	114.0	97.333333	3	2104.0	2122.666667	3	1614.0	1705.666667

	quitters			cumulative_joiners			cumulative_quitters	
	count	median	mean	count	median	mean	count	median	mean
joiners									
68	1	54.0	54.000000	1	1816.0	1816.000000	1	1309.0	1309.000000
77	1	62.0	62.000000	1	1893.0	1893.000000	1	1371.0	1371.000000
81	2	83.0	83.000000	2	1893.5	1893.500000	2	1407.0	1407.000000
84	1	93.0	93.000000	1	1223.0	1223.000000	1	388.0	388.000000
88	1	88.0	88.000000	1	1512.0	1512.000000	1	894.0	894.000000
91	1	75.0	75.000000	1	2195.0	2195.000000	1	1689.0	1689.000000
93	1	92.0	92.000000	1	1605.0	1605.000000	1	986.0	986.000000
96	1	196.0	196.000000	1	1424.0	1424.000000	1	806.0	806.000000
97	2	44.5	44.500000	2	964.5	964.500000	2	223.5	223.500000
105	1	222.0	222.000000	1	1328.0	1328.000000	1	610.0	610.000000
126	1	44.0	44.000000	1	1139.0	1139.000000	1	295.0	295.000000

In [80]:

```
sns.countplot(df_cumulative_week["cumulative_joiners"],data=df3)
plt.title('During the following months, how many riders join')
plt.show()
```



In [81]:

```
df3["Riders_Joined_by_weeks"].mode()
```

Out[81]:

```
0    3
Name: Riders_Joined_by_weeks, dtype: int64
```

In [82]:

```
df3.groupby("Riders_Joined_by_month").agg(['count','mean','median'])
```

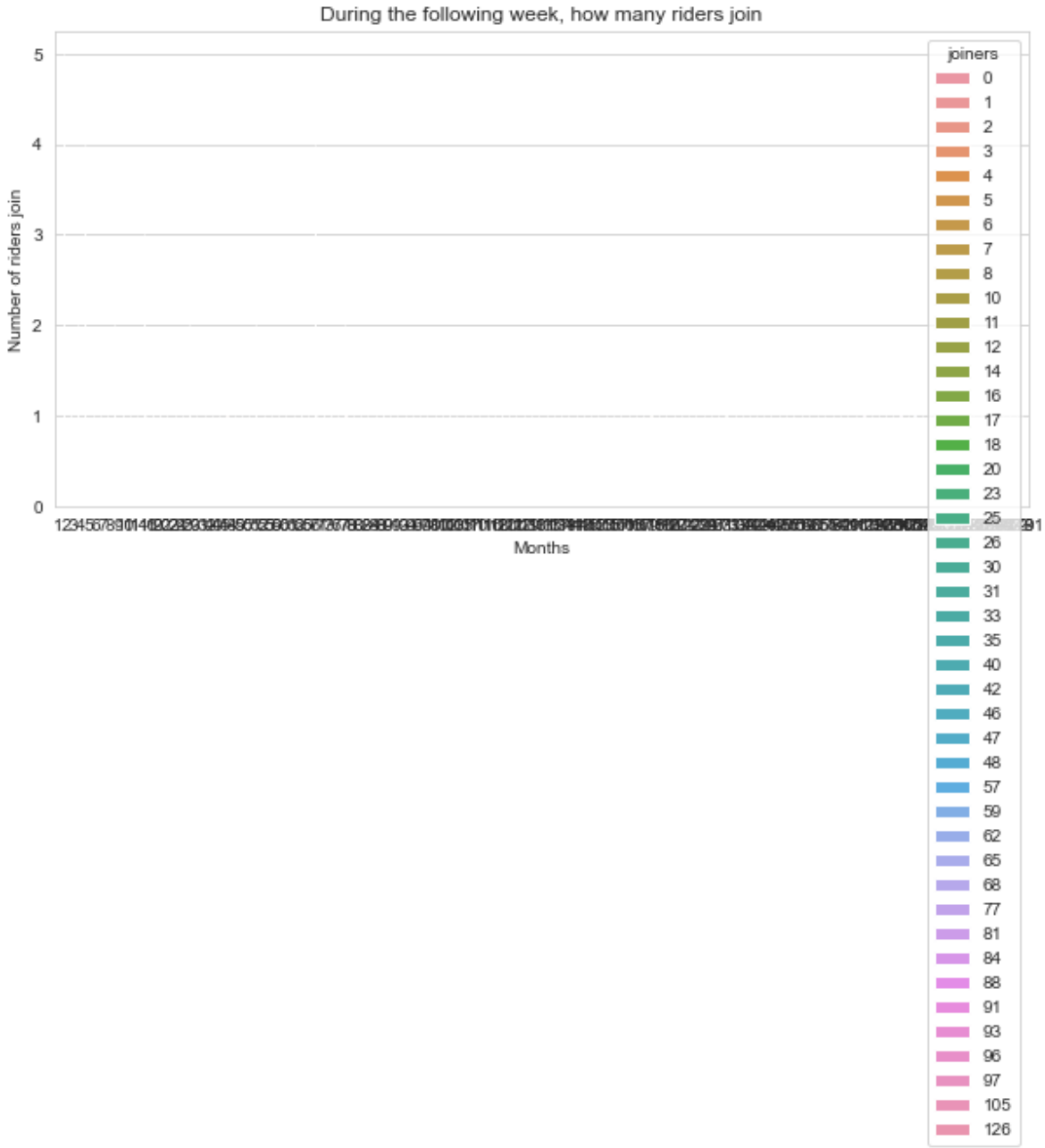
Out[82]:

	DOJ			Status_Change		
	count	mean	median	count	mean	media
Riders_Joined_by_month						
1	444	2021-12-26 23:43:47.027026944	2022-01-16	444	2022-02-21 03:50:16.216216064	2022 02-1 00:00:0
2	405	2022-01-08 01:11:06.666666752	2022-02-12	405	2022-03-01 09:18:13.333333504	2022 03-1 00:00:0
3	331	2022-02-20 11:49:07.432024064	2022-03-15	331	2022-04-08 03:46:13.413897216	2022 04-1 00:00:0
4	328	2022-03-11 09:08:46.829268224	2022-04-14	328	2022-04-15 08:42:26.341463296	2022 05-0 12:00:0
5	230	2022-04-07 18:59:28.695652096	2022-05-12	230	2022-05-10 13:40:10.434782720	2022 05-2 00:00:0
6	50	2021-11-07 17:16:48.000000000	2022-06-01	50	2022-02-20 05:16:48.000000000	2022 06-0 00:00:0
7	40	2021-05-13 15:36:00.000000000	2021-07-17	40	2022-01-18 03:00:00.000000000	2022 02-1 12:00:0
8	52	2021-06-06 18:55:23.076923136	2021-08-16	52	2021-12-27 17:32:18.461538560	2022 02-1 00:00:0
9	121	2021-08-03 11:18:20.826446336	2021-09-17	121	2022-02-01 00:35:42.148760320	2022 02-1 00:00:0
10	111	2021-08-28 10:09:43.783783680	2021-10-11	111	2022-01-21 18:35:40.540540416	2022 02-1 00:00:0
11	151	2021-10-08 03:20:15.894039808	2021-11-16	151	2022-02-05 19:13:54.437086208	2022 02-1 00:00:0
12	218	2021-11-22 15:24:46.238532096	2021-12-17	218	2022-02-07 09:08:15.412844032	2022 02-1 00:00:0

12 rows × 27 columns

In [83]:

```
sns.countplot(df_cumulative_week["cumulative_joiners"],hue= df_cumulative_week['joiners'],d
plt.ylabel('Number of riders join')
plt.xlabel('Months')
plt.title('During the following week, how many riders join')
plt.show()
```



In [84]:

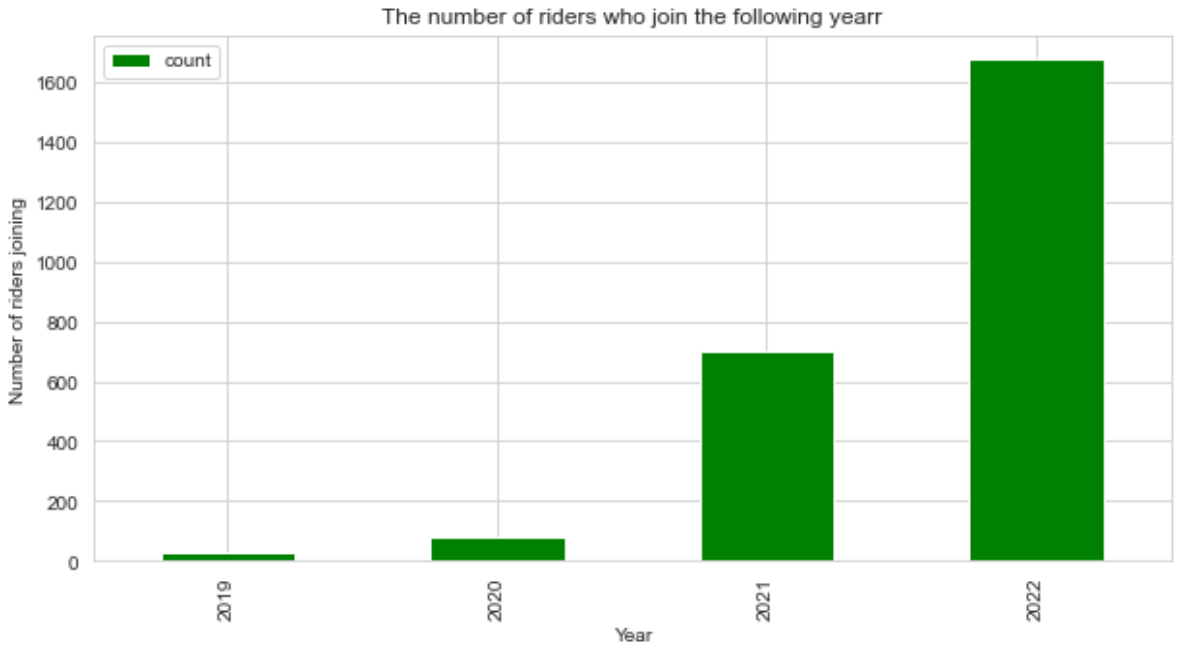
```
df_joining_year=df3.groupby(df3['DOJ'].dt.strftime('%Y'))['Riders_Joined_by_year'].agg(['co
```

In [ ]:

```
```

In [85]:

```
df_joining_year.plot(kind='bar', color= 'Green')
plt.ylabel('Number of riders joining')
plt.xlabel('Year')
plt.title('The number of riders who join the following yearr')
plt.xticks(rotation=90)
plt.show()
```



Examine how many riders quit in the following working days, weeks, months, and years

In [86]:

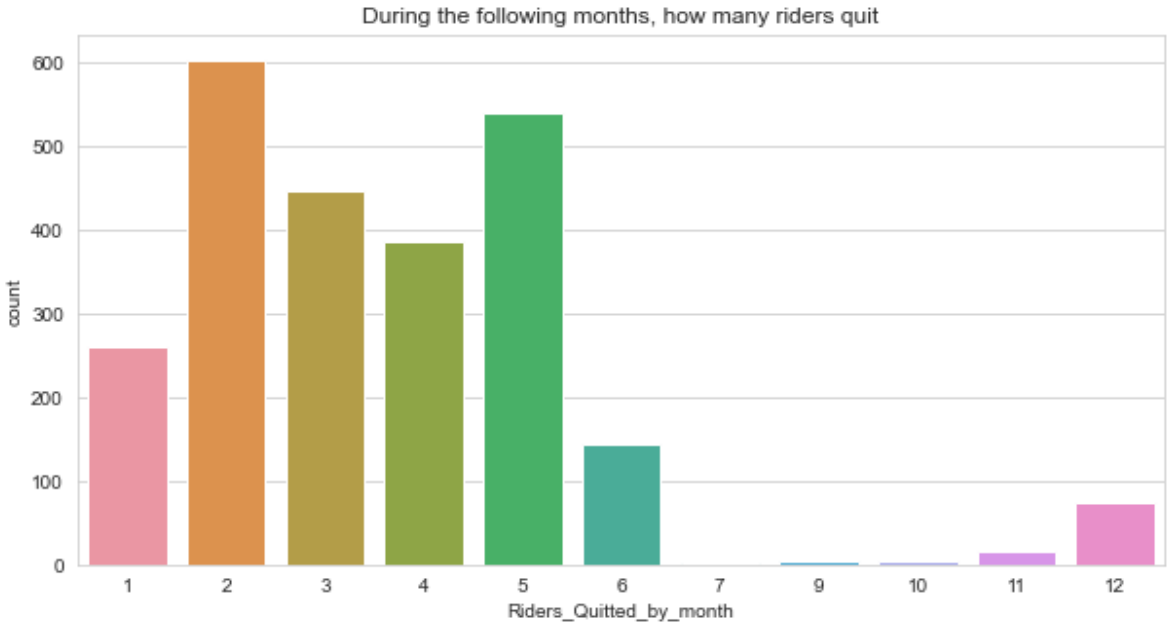
```
df3.groupby("Riders_Quitted_by_month").no_of_working_days.agg(['count'])
```

Out[86]:

Riders_Quitted_by_month	count
1	261
2	603
3	446
4	386
5	540
6	144
7	3
9	4
10	5
11	16
12	73

In [87]:

```
sns.countplot(df3["Riders_Quitted_by_month"])
plt.title('During the following months, how many riders quit')
plt.show()
```



In [88]:

```
df3.drop(['RiderName','DOJ','Status_Changed','RelevingDate_Entered'],axis=1,inplace=True)
```

In [89]:

```
df3.groupby("Riders_Quitted_by_month").agg(['count','mean','median'])
```

Out[89]:

	no_of_working_days			Riders_Joined_by_month			Riders_Jo	
	count	mean	median	count	mean	median	count	n
Riders_Quitted_by_month								
1	261	62.291188	32.0	261	7.019157	9.0	261	2021.34
2	603	74.734660	34.0	603	5.227197	2.0	603	2021.48
3	446	71.852018	34.0	446	3.923767	2.0	446	2021.62
4	386	52.049223	28.0	386	3.567358	3.0	386	2021.87
5	540	58.518519	30.0	540	4.461111	4.0	540	2021.84
6	144	70.881944	22.0	144	5.277778	5.0	144	2021.75
7	3	52.666667	76.0	3	5.000000	4.0	3	2020.33
9	4	145.000000	125.5	4	7.250000	8.0	4	2020.50
10	5	56.200000	52.0	5	8.000000	8.0	5	2021.00
11	16	169.375000	110.0	16	7.750000	7.5	16	2020.81
12	73	161.068493	71.0	73	9.493151	10.0	73	2020.79

In [90]:

```
sns.countplot(df3["Riders_Quitted_by_month"],hue=df3['Riders_Quitted_by_year'] ,data=df3,pa
plt.ylabel('Number of riders quitting')
plt.xlabel('Months')
plt.title('During the following week, how many riders quit')
# plt.show()
```

Out[90]:

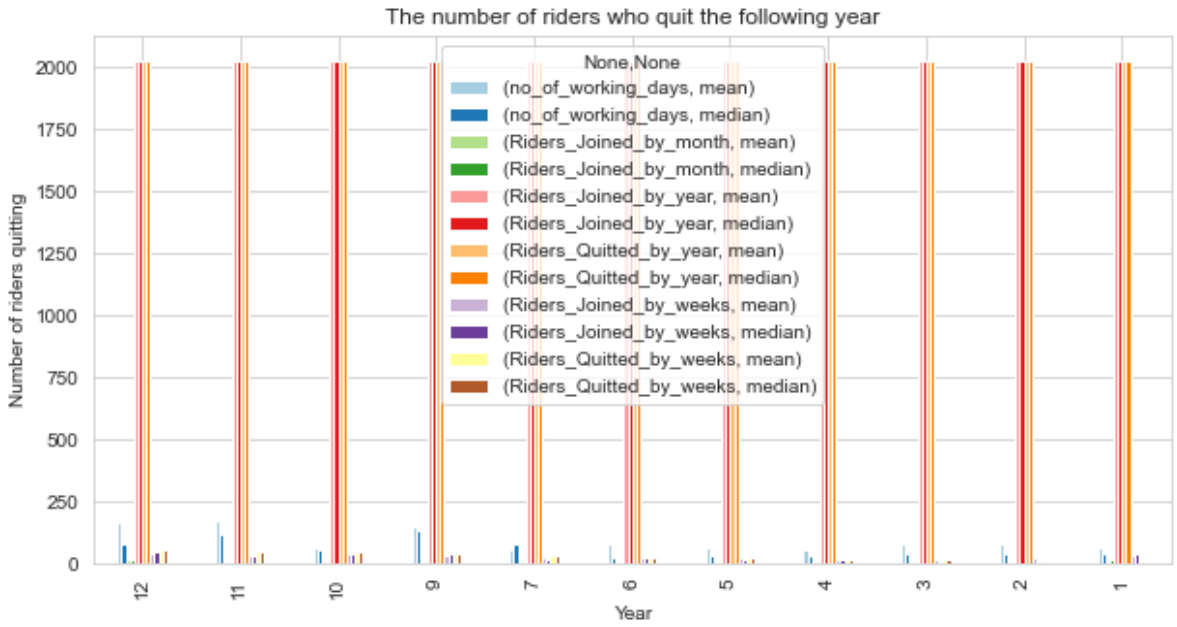
```
Text(0.5, 1.0, 'During the following week, how many riders quit')
```

In [91]:

```
df_quitting_year=df3.groupby(df3["Riders_Quitted_by_month"]).agg(['mean','median']).sort_in
                                                                                    inpla
```

In [92]:

```
df_quitting_year.plot(kind='bar', colormap= 'Paired')
plt.ylabel('Number of riders quitting')
plt.xlabel('Year')
plt.title('The number of riders who quit the following year')
plt.xticks(rotation=90)
plt.show()
```



In [ ]:

```

```

Analyze how many different types of clients there are

In [93]:

```
df1
```

Out[93]:

	LeadID	State	City	RiderName	LL EMP Code	RiderNumber	DOJ	Client	
0	31	Karnataka	Bengaluru	SRIKANTH P	LLBBC001	6363376901	2019-04-04	Bigbasket Bangalore	
1	32	Karnataka	Bengaluru	SUPREEM .	LLBBC003	7483505921	2019-04-16	Bigbasket Bangalore	
3	35	Karnataka	Bengaluru	VEERESH U	LLBBC011	9110444430	2019-06-23	Bigbasket Bangalore	
4	37	Karnataka	Bengaluru	Harish M	LLBBC027	9008816586	2019-06-17	Bigbasket Bangalore	
5	38	Karnataka	Bengaluru	ANANDH .	LLBBC028	9095272783	2019-02-26	TOW Bangalore	H
...	...	...	...	...	...	...	...	...	
7344	27758	Karnataka	Bengaluru	Kranthi Kumara	13944	8861590554	2022-06-14	1kart Bangalore	C
7345	27761	Karnataka	Bengaluru	Darshan P	13948	7337715815	2022-06-14	1kart Bangalore	N
7346	27763	Telangana	Hyderabad	Madhu Kumar	13950	9180746571	2022-06-14	Bigbasket Hyderabad	B
7347	27764	Karnataka	Bengaluru	Ravi S	13951	7975174381	2022-06-15	1kart Bangalore	R
7348	27782	Karnataka	Bengaluru	Aravinda reddy K	13960	7013137501	2022-06-15	1kart Bangalore	T

6680 rows × 15 columns

In [94]:

```
client_df=df1.drop(['LeadID','State','City','LL EMP Code','RiderNumber',
                    'Previous_Status','Current_Status','Modified_by','ReasonForLeaving'], a
client_df.shape
```

Out[94]:

```
(6680, 6)
```

In [95]:

```
client_df.isna().sum()
```

Out[95]:

```
RiderName                0
DOJ                   1187
Client                  49
Hub                     49
Status_Changed          55
RelevingDate_Entered  3612
dtype: int64
```

In [96]:

```
client_df.isna().any(axis=1).value_counts()
```

Out[96]:

```
True     4006
False    2674
dtype: int64
```

In [97]:

```
client_df.sample(5)
```

Out[97]:

	RiderName	DOJ	Client	Hub	Status_Changed	RelevingDate_Entered
2210	MAHESH J C	2021-09-13	1kart Bangalore	1Kart Vijayanagar	2021-12-15	NaT
108	SHIVU S	2020-02-11	1kart Bangalore	1Kart Mysore Road	2021-12-16	NaT
3002	Shoaib M.Z	NaT	1kart Bangalore	1Kart E_City	2021-12-16	NaT
4076	Joel Kumar	2022-01-15	Grofers Delhi	Grofers_Super Store - Delhi Chattarpur ES2	2022-02-02	2022-02-02
1714	Rakshith Gowda	NaT	1kart Bangalore	1Kart Srinagar	2021-08-13	NaT

In [98]:

```
client_df=client_df.dropna().copy()
```

In [99]:

```
client_df.shape
```

Out[99]:

```
(2674, 6)
```

In [100]:

```
client_df.sample(20)
```

Out[100]:

	RiderName	DOJ	Client	Hub	Status_Changed	Releving
4933	Naveen Vijay	2022-02-14	PORTER BLR-SMARTSHIFT LOGISTICS SOLUTIONS P LTD	Porter BLR - LLOffice	2022-05-13	
5597	Ajmal MI	2022-03-22	ShadowFax E-Comm Kerala	SFX E-com Aluva	2022-05-25	
5070	Avi Patel	2022-02-17	Flipkart Ahmedabad	Flipkart_ManiNagarhub_AHM	2022-03-26	
2157	Naveen Kumar B N	2022-03-10	Shadowfax E-Commerce Bangalore	Shadowfax E-Commerce Bangalore	2022-04-12	
2160	Aman Kumar Jha	2021-09-19	Flipkart Delhi	Flipkart_Vikashpuri hub_DEL	2022-02-11	
3006	Raju Raju	2021-11-02	1kart Bangalore	1kart Nelamangla	2021-12-16	
5612	Jansi Daniel	2022-03-31	Bigbasket Bangalore	Bigbasket HRBR	2022-05-30	
5635	Saarathy A	2022-03-21	Flipkart Chennai	Flipkart Guindy	2022-05-14	
4059	Fayazuddin	2022-01-03	Bigbasket Chennai	Bigbasket Mylapore	2022-02-15	
2903	Dinesh Kumar R	2021-10-28	Bigbasket Chennai	Bigbasket Chitlapakkam	2022-03-11	
5805	Siddaraju N	2022-04-06	1kart Bangalore	1Kart RajajiNagar	2022-05-04	
2394	Muhammed Niyas	2021-09-24	Flipkart Kerala	Thamarassery_Hub	2022-02-10	
4649	Vijaykumar R	2022-01-31	Shadowfax Food Bangalore	ShadowFax Food LL Villa	2022-03-12	
3386	Sudarshana Bhosle	2021-11-28	Bigbasket Bangalore	BigBasket Mahalakshmi Layout	2022-03-21	
5301	Vinay R	2022-03-05	Bigbasket Bangalore	Bigbasket Yeshwanthpur	2022-04-30	
5777	S. Manikanta S. Manikanta	2022-03-18	Bigbasket Bangalore	Bigbasket Jayanagar	2022-04-30	
5475	Rakesh Tm	2022-04-01	1kart Bangalore	1kart_YelahankaHub_BLR	2022-04-28	
3071	Chethan C	2021-11-09	PORTER BLR-SMARTSHIFT LOGISTICS SOLUTIONS P LTD	Porter BLR - LLOffice	2022-02-24	

	RiderName	DOJ	Client	Hub	Status_Changed	Releving
2738	Devanand M	2022-03-22	Flipkart Kerala	AluvaHub_ALU	2022-05-25	
4133	Rajappa G	2022-01-08	Shadowfax Food Bangalore	ShadowFax Food LL Villa	2022-02-23	

In [101]:

```
client_df[client_df['Client']=='1kart Bangalore'].shape
```

Out[101]:

```
(554, 6)
```

In [102]:

```
client_df[client_df['Client']=='Shadowfax E-Commerce ( Chennai )']
```

Out[102]:

	RiderName	DOJ	Client	Hub	Status_Changed	RelevingDate_
134	Subramani G	2021-10-25	Shadowfax E-Commerce (Chennai)	SFX E-Com CHN_Velachery	2021-08-13	20
159	Amudhan Manavalan	2022-04-06	Shadowfax E-Commerce (Chennai)	SFX E-Com CHN_Pallikaranai	2022-04-19	20
199	Govarthanan B	2022-04-08	Shadowfax E-Commerce (Chennai)	SFX E-Com CHN_Pallikaranai	2022-05-10	20
340	BHARATH Kumar	2022-04-07	Shadowfax E-Commerce (Chennai)	SFX E-Com CHN_Pallikaranai	2022-05-10	20
940	Surya M	2022-03-01	Shadowfax E-Commerce (Chennai)	SFX E-Com CHN_SHOLINGANALLUR/ Thoraipakkam	2022-04-19	20
...	...	...	...	...	...	
6227	Goverdhan. R	2022-04-22	Shadowfax E-Commerce (Chennai)	SFX E-Com CHN_Ayyanavaram	2022-05-14	20
6232	Jeyapandi R	2022-04-22	Shadowfax E-Commerce (Chennai)	SFX E-Com CHN_Pallavaram	2022-05-07	20
6252	Srinivasan K	2022-04-23	Shadowfax E-Commerce (Chennai)	SFX E-Com CHN_Pallavaram	2022-05-07	20
6303	MANIKANDAN MOORTHY	2022-04-26	Shadowfax E-Commerce (Chennai)	SFX E-Com CHN_Madhavaram_E	2022-05-14	20
6552	Sai Kumar	2022-05-07	Shadowfax E-Commerce (Chennai)	SFX E-Com_CHN-Periyar Nagar	2022-05-14	20

83 rows × 6 columns

In [103]:

```
client_df[client_df['Client']=='Flipkart Mumbai']
```

Out[103]:

	RiderName	DOJ	Client	Hub	Status_Changed	RelevingDate
3123	VijayaKumar Pandian	2021-10-08	Flipkart Mumbai	Flipkart_Dadar WestHub_MUM	2022-05-16	20
4116	Santosh wadekar	2022-01-13	Flipkart Mumbai	Flipkart_ChemburSplitHub_MUM	2022-02-11	20
4470	Yash Singh	2022-01-21	Flipkart Mumbai	Flipkart_KalyanHub_Mum	2022-02-11	20
4542	Purushottam Birkad	2022-01-24	Flipkart Mumbai	Flipkart_ChemburSplitHub_MUM	2022-03-26	20
4640	Namitesh Gurav	2022-01-28	Flipkart Mumbai	Flipkart_ThaneHub_MUM	2022-03-28	20
4683	Parasram Navnath Dhope	2022-01-30	Flipkart Mumbai	Flipkart_KalyanHub_Mum	2022-03-28	20
4808	Vishal Hande	2022-02-05	Flipkart Mumbai	Flipkart_Boriwali_ MUM	2022-03-26	20
4997	Sagar Koli	2022-02-17	Flipkart Mumbai	Flipkart_Dadar WestHub_MUM	2022-03-15	20
5188	Farhan Shaikh	2022-02-25	Flipkart Mumbai	Flipkart_WadalaHub_MUM_PL	2022-03-28	20
5267	Deepak Mishra	2022-03-05	Flipkart Mumbai	Flipkart_KalyanHub_Mum	2022-05-16	20
5324	Ashish Shinde	2022-03-01	Flipkart Mumbai	Flipkart_Dadar WestHub_MUM	2022-03-28	20
5350	ashwin choudhari	2022-03-05	Flipkart Mumbai	Flipkart_KalyanHub_Mum	2022-06-04	20
5564	Rahul Ghegadmal	2022-03-21	Flipkart Mumbai	Flipkart_KalyanHub_Mum	2022-04-23	20

In [104]:

```
client_df[client_df['Client']=='Flipkart Kerala']
```

Out[104]:

	RiderName	DOJ	Client	Hub	Status_Changed	RelevingDate_Entei
1933	Visakhan Pv	2021-08-12	Flipkart Kerala	IrinjalakudaHub_IJK	2022-02-10	2022-02
2084	NISAR N	2021-08-27	Flipkart Kerala	KayamkulamHub_KKM	2022-02-10	2022-02
2092	Prajith C Chandrakumar	2021-09-24	Flipkart Kerala	TrivandrumHub_TRV	2022-02-10	2022-02
2105	Shaji Mp	2021-09-26	Flipkart Kerala	Thamarassery_Hub	2022-02-10	2022-02
2139	Bibin Raj	2021-09-02	Flipkart Kerala	TrivandrumHub_TRV	2022-02-21	2022-02
...	...	...	...	...	...	
6174	AJMAL SHAJAHAN V	2022-04-23	Flipkart Kerala	VettichiraHub_VTT	2022-05-07	2022-05
6210	Sanad Mp	2022-04-21	Flipkart Kerala	ThenjipalamHub_TJI	2022-05-07	2022-05
6254	Sumesh K K	2022-04-29	Flipkart Kerala	MananthavadyHub_MDY	2022-05-25	2022-05
6359	Vishnu Vc	2022-05-05	Flipkart Kerala	TripunithuraHub_TRT	2022-05-25	2022-05
6589	Hashif S	2022-05-19	Flipkart Kerala	MalaPuramHub_MLP	2022-05-28	2022-05

87 rows × 6 columns

In [134]:

```
client_df1=client_df.groupby(['Client','Hub']).count().copy()
client_df1
```

Out[134]:

Client	Hub	RiderName	DOJ	Status_Changed	RelevingDate_Entered
1kart Bangalore	1Kart Bangalore Anekal	1	1	1	1
	1Kart Bangalore Jakkasandra	1	1	1	1
	1Kart Bangalore Sarjapur	3	3	3	3
	1Kart Banshankari	9	9	9	9
	1Kart E_City	17	17	17	17
...	...	...	...	...	...
TOW Bangalore	TOW Whitefield	3	3	3	3
Village Milk	Ramapuram	1	1	1	1
shadowfax Bangalore FLKRT QUICK	Flipkart Quick HSR	2	2	2	2
shadowfax Hyderabad FLKRT QUICK	FlipkartHyperLocal _Golconda	2	2	2	2
	FlipkartHyperLocal_NTR Nagar	1	1	1	1

361 rows × 4 columns

In [106]:

```
gk = client_df.groupby('Client')
gk.get_group('1kart Bangalore').shape
```

Out[106]:

```
(554, 6)
```

In [107]:

```
min(client_df['Client'])
```

Out[107]:

```
'1kart Bangalore'
```

In [108]:

```
client_df[client_df['Client']=='shadowfax Hyderabad FLKRT QUICK']
```

Out[108]:

	RiderName	DOJ	Client	Hub	Status_Changed	RelevingDate_Entei
1361	Mohd Rashed	2021-06-03	shadowfax Hyderabad FLKRT QUICK	FlipkartHyperLocal_NTR Nagar	2022-02-10	2021-12
1479	Suleman Khan	2021-07-03	shadowfax Hyderabad FLKRT QUICK	FlipkartHyperLocal _Golconda	2022-02-10	2021-12
1480	Gulam Rasool	2021-07-03	shadowfax Hyderabad FLKRT QUICK	FlipkartHyperLocal _Golconda	2022-02-10	2021-12

In [109]:

```
client_df[client_df['Client']=='1kart Bangalore'].shape
```

Out[109]:

```
(554, 6)
```

In [110]:

```
client_df.groupby(client_df['Client']=='1kart Bangalore').count()
```

Out[110]:

Client	RiderName	DOJ	Client	Hub	Status_Changed	RelevingDate_Entered
False	2120	2120	2120	2120	2120	2120
True	554	554	554	554	554	554

In [111]:

```
client_df.groupby(client_df['Client']=='shadowfax Hyderabad FLKRT QUICK').count()
```

Out[111]:

Client	RiderName	DOJ	Client	Hub	Status_Changed	RelevingDate_Entered
False	2671	2671	2671	2671	2671	2671
True	3	3	3	3	3	3

In [112]:

```
client_df.groupby(client_df['Client']=='Shadowfax E-Commerce (Uttar Pradesh)').count()
```

Out[112]:

Client	RiderName	DOJ	Client	Hub	Status_Changed	RelevingDate_Entered
False	2672	2672	2672	2672	2672	2672
True	2	2	2	2	2	2

Pairs of data that are correlated

In [113]:

```
# correlated=df3.corr()
```

In [114]:

```
# sns.heatmap(correlated, annot = True, cmap = 'Blues')
```

In [ ]:

```

```

Analyze how many riders are satisfied each year

No of the riders who joined following 2019

In [115]:

```
df3_2019=df3[df3['Riders_Joined_by_year']==2019].copy()
df3_2019.sample(5)
```

Out[115]:

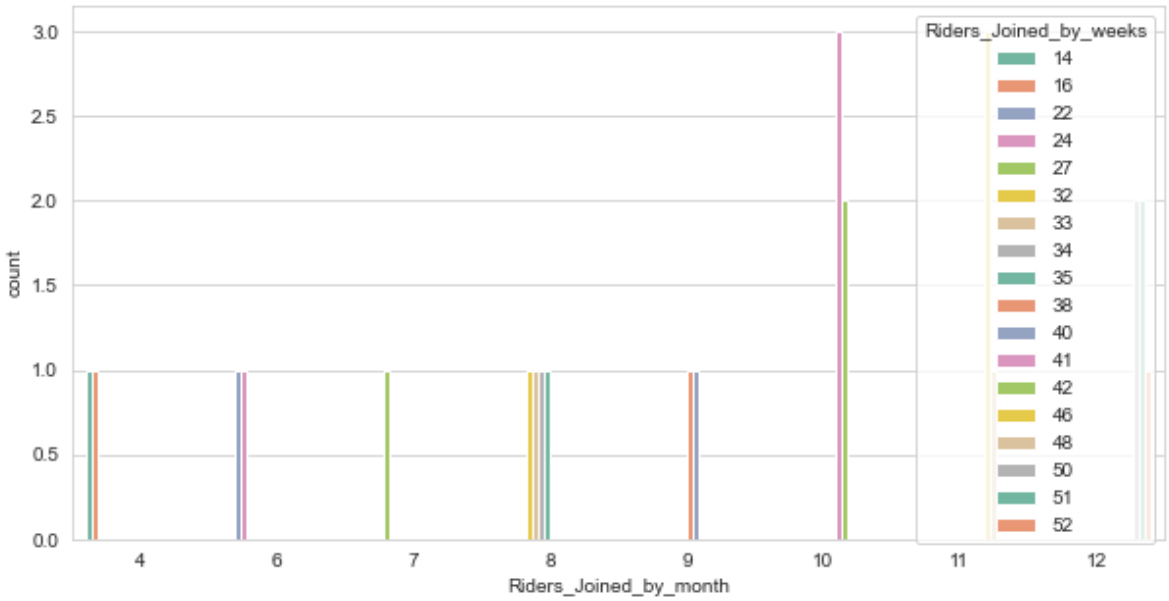
	no_of_working_days	Riders_Joined_by_month	Riders_Joined_by_year	Riders_Quitted_by_mo
62	81	12	2019	
40	329	10	2019	
59	742	12	2019	
54	103	11	2019	
26	231	9	2019	

In [116]:

```
sns.countplot(df3_2019["Riders_Joined_by_month"],hue=df3_2019['Riders_Joined_by_weeks'] ,da
```

Out[116]:

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



No of the riders who joined following 2020

In [117]:

```
df3_2020=df3[df3['Riders_Joined_by_year']==2020].copy()
df3_2020.sample(5)
```

Out[117]:

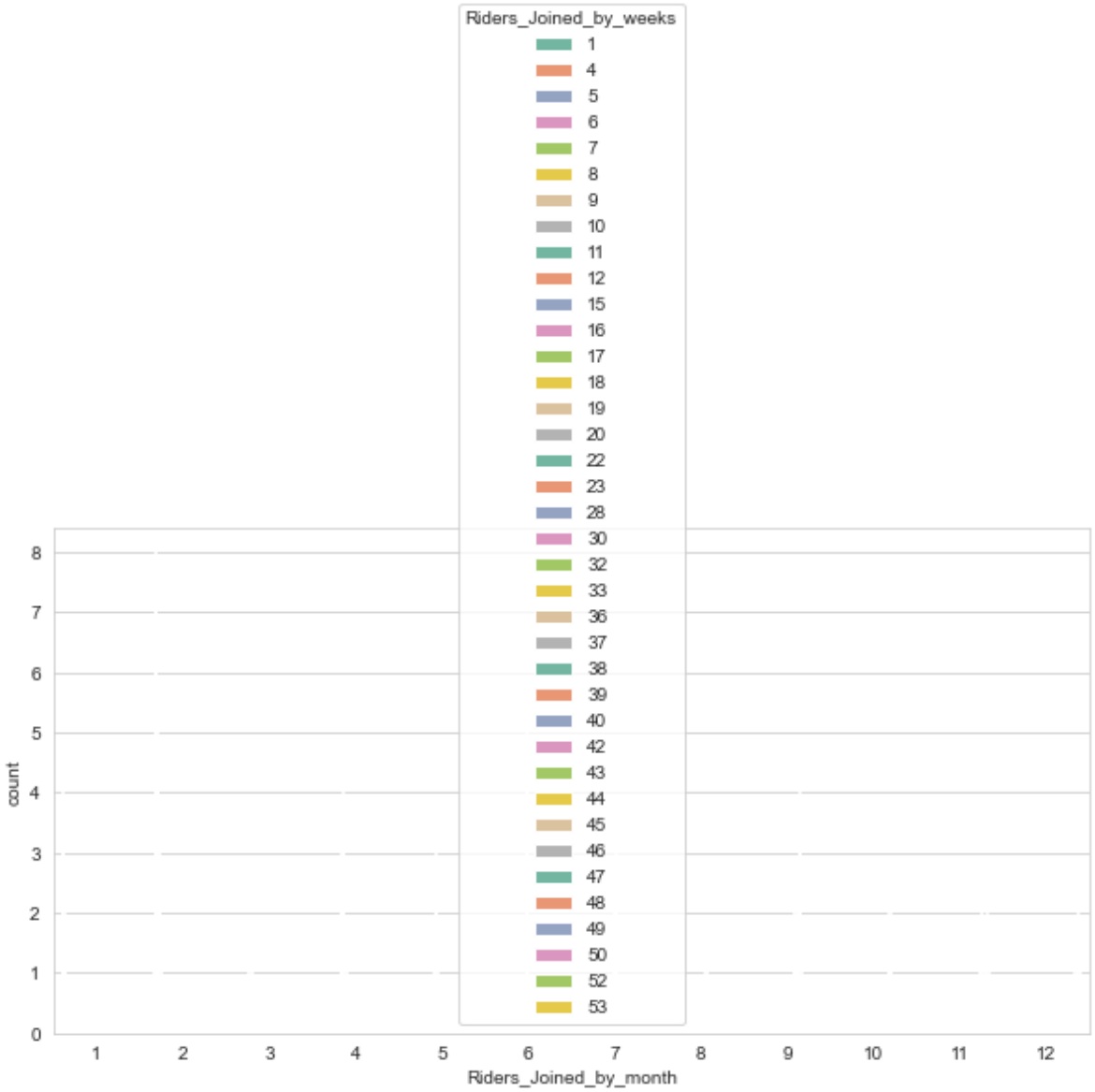
	no_of_working_days	Riders_Joined_by_month	Riders_Joined_by_year	Riders_Quitted_by_m
243	29	2	2020	
499	482	9	2020	
856	438	12	2020	
225	785	1	2020	
96	92	2	2020	

In [118]:

```
sns.countplot(df3_2020["Riders_Joined_by_month"],hue=df3_2020['Riders_Joined_by_weeks'] ,da
```

Out[118]:

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



No of riders who joined following the 2021

In [119]:

```
df3_2021=df3[df3['Riders_Joined_by_year']==2021].copy()
df3_2021.sample(5)
```

Out[119]:

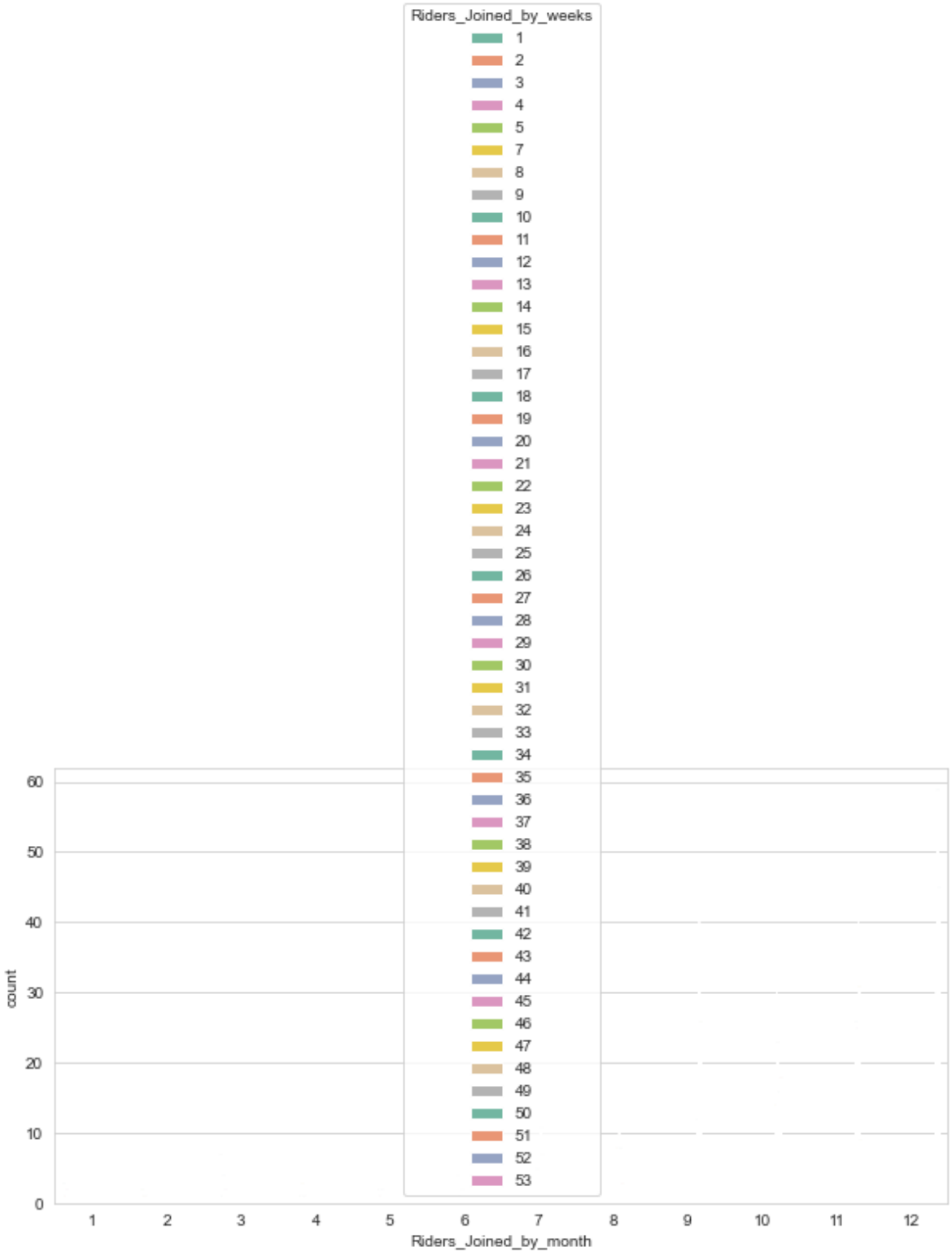
	no_of_working_days	Riders_Joined_by_month	Riders_Joined_by_year	Riders_Quitted_by_
3750	47	12	2021	
2140	126	9	2021	
2976	88	11	2021	
3272	137	11	2021	
1930	170	8	2021	

In [120]:

```
sns.countplot(df3_2021["Riders_Joined_by_month"],hue=df3_2021['Riders_Joined_by_weeks'] ,da
```

Out[120]:

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



No of riders who joined following the 2022

In [121]:

```
df3_2022=df3[df3['Riders_Joined_by_year']==2022].copy()
df3_2022.sample(5)
```

Out[121]:

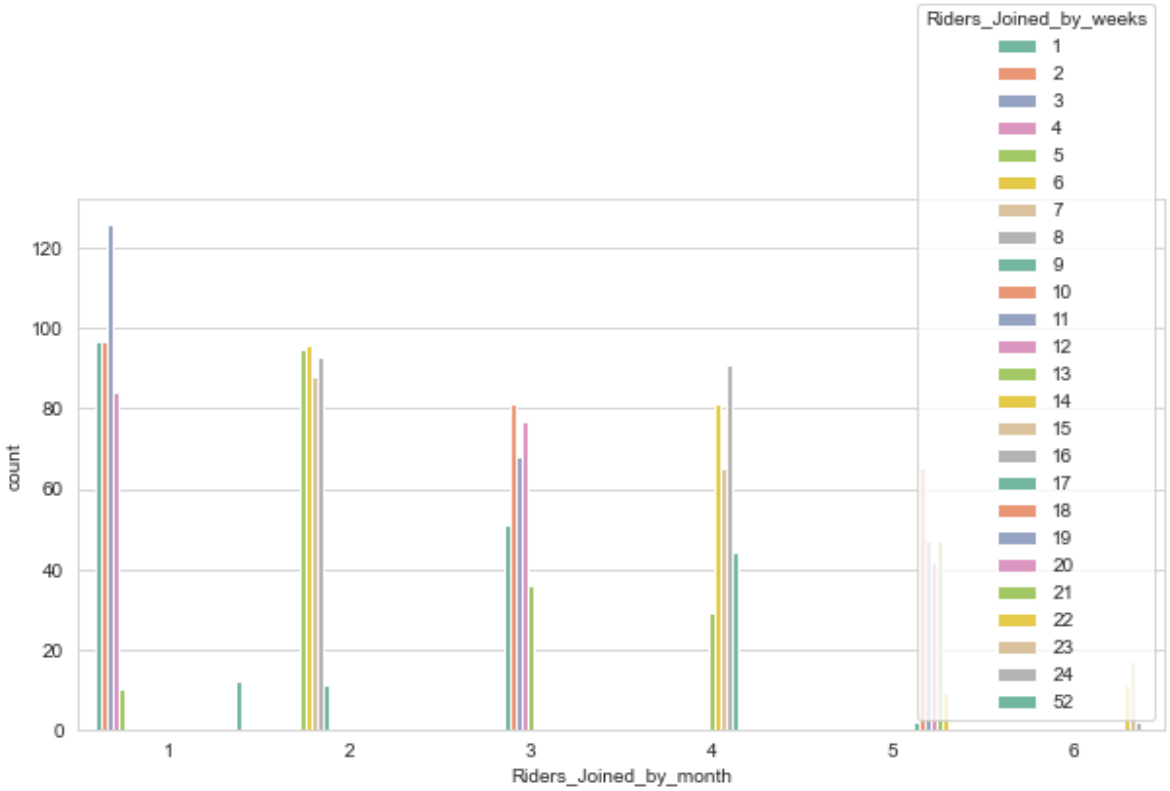
	no_of_working_days	Riders_Joined_by_month	Riders_Joined_by_year	Riders_Quitted_by_
4368	76	2	2022	
6085	17	4	2022	
7169	2	6	2022	
5591	37	4	2022	
4875	0	2	2022	

In [122]:

```
sns.countplot(df3_2022["Riders_Joined_by_month"],hue=df3_2022['Riders_Joined_by_weeks'] ,da
```

Out[122]:

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



No of riders who quitted following the 2020

In [123]:

```
df3_2020=df3[df3['Riders_Quitted_by_year']==2020].copy()
df3_2020
```

Out[123]:

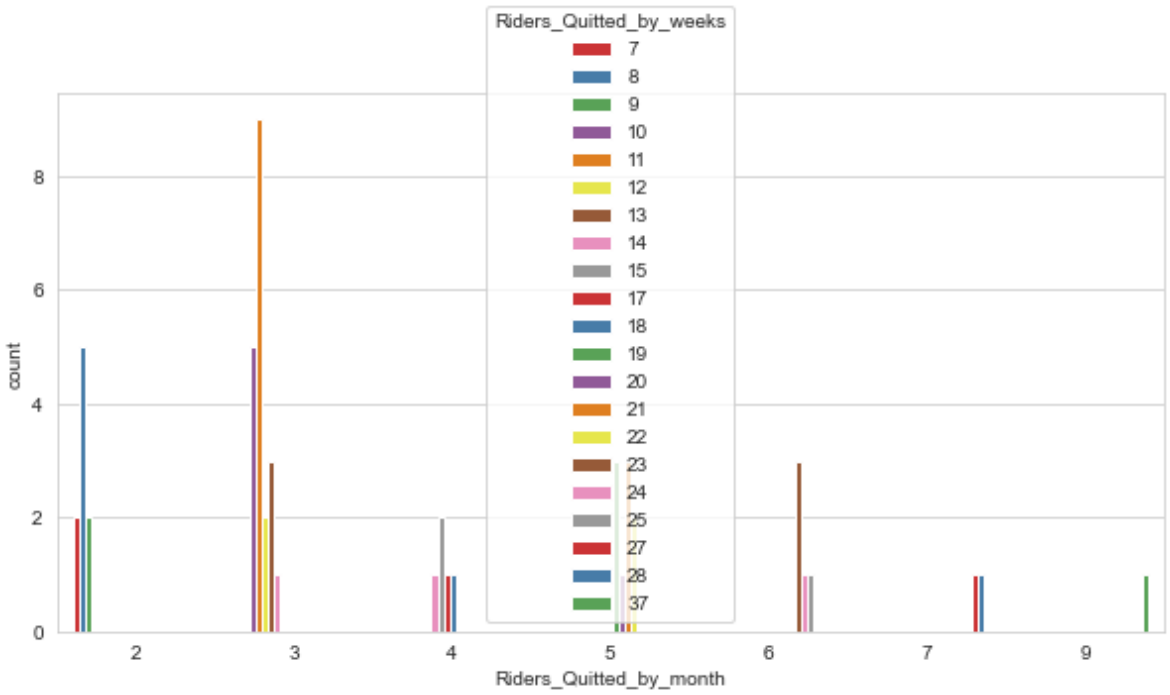
	no_of_working_days	Riders_Joined_by_month	Riders_Joined_by_year	Riders_Quitted_by_month	Riders_
0	364	4	2019	4	
1	310	4	2019	2	
10	300	7	2019	4	
15	243	8	2019	4	
18	200	8	2019	3	
19	187	8	2019	3	
20	169	9	2019	3	
21	141	10	2019	2	
26	231	9	2019	5	
27	229	10	2019	5	

In [124]:

```
sns.countplot(df3_2020["Riders_Quitted_by_month"],hue=df3_2020['Riders_Quitted_by_weeks'] ,
```

Out[124]:

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



No of riders who quitted following the 2021

In [125]:

```
df3_2021=df3[df3['Riders_Quitted_by_year']==2021].copy()
df3_2021.sample(5)
```

Out[125]:

	no_of_working_days	Riders_Joined_by_month	Riders_Joined_by_year	Riders_Quitted_by_
2893	0	11	2021	
3688	7	12	2021	
3583	6	12	2021	
1364	155	6	2021	
1096	276	3	2021	

In [126]:

```
df3[df3['Riders_Quitted_by_year']==2021].shape
```

Out[126]:

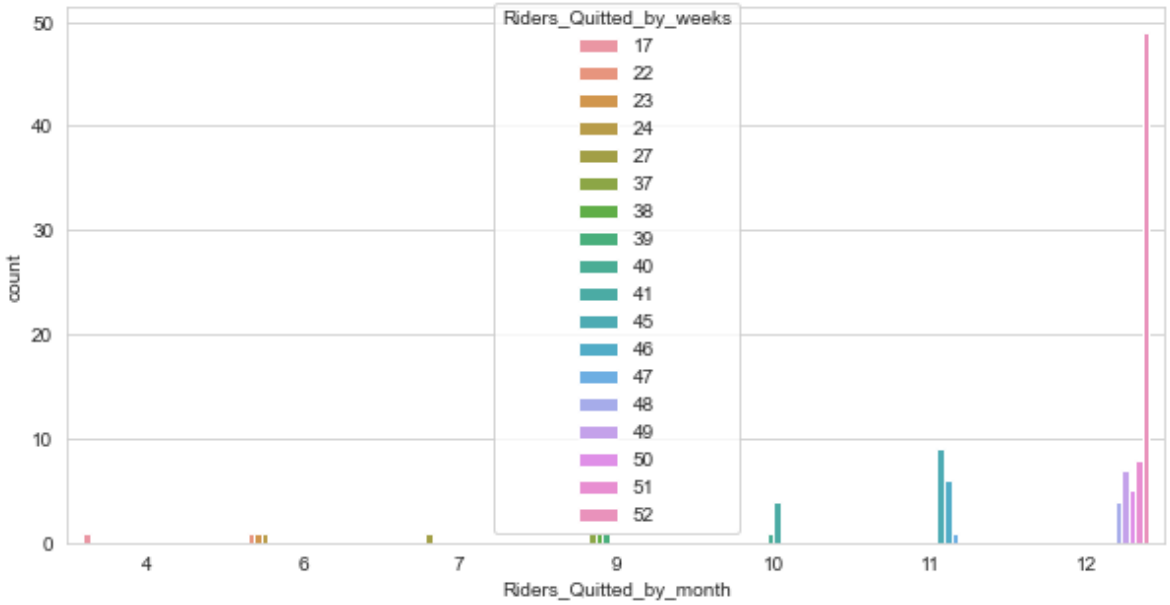
```
(102, 7)
```

In [127]:

```
sns.countplot(df3_2021["Riders_Quitted_by_month"],hue=df3_2021['Riders_Quitted_by_weeks'] ,
```

Out[127]:

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



No of riders who quitted following the 2022

In [128]:

```
df3_2022=df3[df3['Riders_Quitted_by_year']==2022].copy()
df3_2022.sample(5)
```

Out[128]:

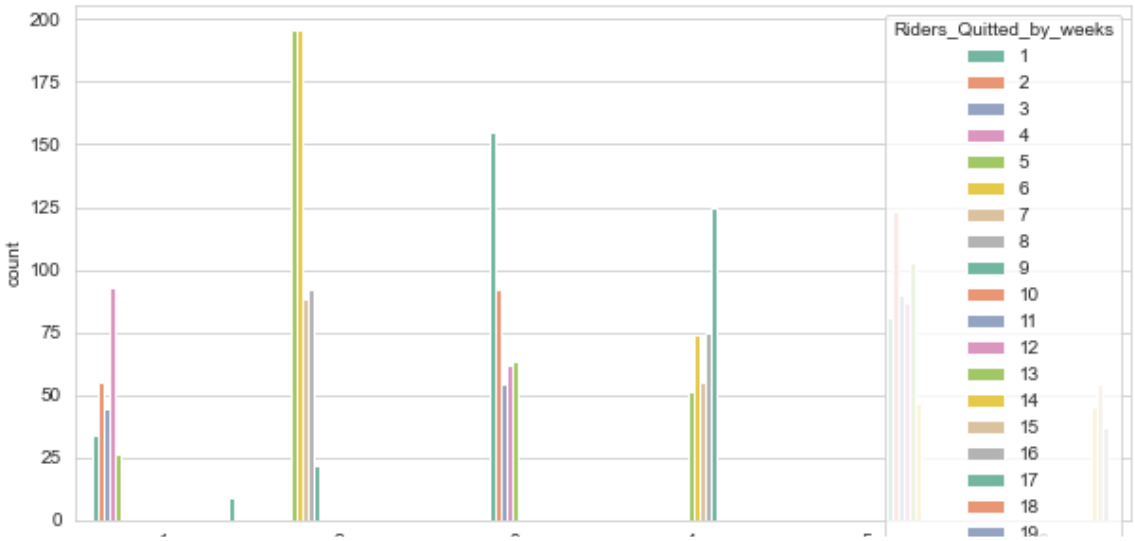
	no_of_working_days	Riders_Joined_by_month	Riders_Joined_by_year	Riders_Quitted_by_
6757	13	5	2022	
5408	0	3	2022	
5002	65	2	2022	
4923	51	2	2022	
4746	731	2	2020	

In [129]:

```
sns.countplot(df3_2022["Riders_Quitted_by_month"],hue=df3_2022['Riders_Quitted_by_weeks'] ,
```

Out[129]:

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



In [131]:

```
df3[df3['Riders_Quitted_by_year']==2022]
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
File ~\Anaconda3\lib\site-packages\pandas\core\indexes\base.py:3621, in Inde
x.get_loc(self, key, method, tolerance)
   3620 try:
-> 3621     return self._engine.get_loc(casted_key)
   3622 except KeyError as err:

File ~\Anaconda3\lib\site-packages\pandas\_libs\index.pyx:136, in pandas._li
bs.index.IndexEngine.get_loc()

File ~\Anaconda3\lib\site-packages\pandas\_libs\index.pyx:163, in pandas._li
bs.index.IndexEngine.get_loc()

File pandas\_libs\hashtable_class_helper.pxi:5198, in pandas._libs.hashtabl
e.PyObjectHashTable.get_item()

File pandas\_libs\hashtable_class_helper.pxi:5206, in pandas._libs.hashtabl
e.PyObjectHashTable.get_item()

KeyError: 'Riders_Quitted_by_year'

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
Input In [131], in <cell line: 1>()
----> 1 df2[df2['Riders_Quitted_by_year']==2022]

File ~\Anaconda3\lib\site-packages\pandas\core\frame.py:3505, in DataFrame._
_getitem__(self, key)
   3503 if self.columns.nlevels > 1:
   3504     return self._getitem_multilevel(key)
-> 3505 indexer = self.columns.get_loc(key)
   3506 if is_integer(indexer):
   3507     indexer = [indexer]

File ~\Anaconda3\lib\site-packages\pandas\core\indexes\base.py:3623, in Inde
x.get_loc(self, key, method, tolerance)
   3621     return self._engine.get_loc(casted_key)
   3622 except KeyError as err:
-> 3623     raise KeyError(key) from err
   3624 except TypeError:
   3625     # If we have a listlike key, _check_indexing_error will raise
   3626     #  InvalidIndexError. Otherwise we fall through and re-raise
   3627     #  the TypeError.
   3628     self._check_indexing_error(key)

KeyError: 'Riders_Quitted_by_year'
```

In [ ]:

```

```