DAC_PHASE 3

Date :26/10/2023

Project Title :Public Transportation Efficiency Analysis

Importing The Dependencies

```python
In [2]:  import pandas as pd
         import numpy as np
         import matplotlib.pyplot as plt
         import seaborn as sns
```

```python
In [5]:  data = pd.read_csv("C://Users//Indarjith K//Desktop//Indrajithdataset.csv")
```

```
C:\Users\Indarjith K\AppData\Local\Temp\ipykernel_10320\2758608790.py:1: Dtyp
eWarning: Columns (1) have mixed types. Specify dtype option on import or set
low_memory=False.
  data = pd.read_csv("C://Users//Indarjith K//Desktop//Indrajithdataset.csv")
```

In [6]: data

Out[6]:

	TripID	RouteID	StopID	StopName	WeekBeginning	NumberOfBoardings
0	23631	100	14156	181 Cross Rd	30-06-2013 00:00	1
1	23631	100	14144	177 Cross Rd	30-06-2013 00:00	1
2	23632	100	14132	175 Cross Rd	30-06-2013 00:00	1
3	23633	100	12266	Zone A Arndale Interchange	30-06-2013 00:00	2
4	23633	100	14147	178 Cross Rd	30-06-2013 00:00	1
...	...	...	...	...	...	...
1048570	45682	171	13929	8 Fullarton Rd	29-09-2013 00:00	2
1048571	45682	171	13758	3 Glen Osmond Rd	29-09-2013 00:00	3
1048572	45682	171	13967	9 Fullarton Rd	29-09-2013 00:00	1
1048573	45682	171	13808	5 Fullarton Rd	29-09-2013 00:00	1
1048574	45682	171	13845	6 Fullarton Rd	29-09-2013 00:00	3

1048575 rows × 6 columns

EXPLORING THE DATASET

1. Displaying The Top 5 Rows

In [7]: data.head()

Out[7]:

	TripID	RouteID	StopID	StopName	WeekBeginning	NumberOfBoardings
0	23631	100	14156	181 Cross Rd	30-06-2013 00:00	1
1	23631	100	14144	177 Cross Rd	30-06-2013 00:00	1
2	23632	100	14132	175 Cross Rd	30-06-2013 00:00	1
3	23633	100	12266	Zone A Arndale Interchange	30-06-2013 00:00	2
4	23633	100	14147	178 Cross Rd	30-06-2013 00:00	1

2. Displaying The Bottom 5 Rows

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

Out[8]:

	TripID	RouteID	StopID	StopName	WeekBeginning	NumberOfBoardings
1048570	45682	171	13929	8 Fullarton Rd	29-09-2013 00:00	2
1048571	45682	171	13758	3 Glen Osmond Rd	29-09-2013 00:00	3
1048572	45682	171	13967	9 Fullarton Rd	29-09-2013 00:00	1
1048573	45682	171	13808	5 Fullarton Rd	29-09-2013 00:00	1
1048574	45682	171	13845	6 Fullarton Rd	29-09-2013 00:00	3

3. Find The Shape Of The Dataset

In [9]: `data.shape`

Out[9]: `(1048575, 6)`

4. Displaying The Information

In [10]: `data.info`

Out[10]: `<bound method DataFrame.info of`   `TripID RouteID  StopID`
`StopName     WeekBeginning  \`
```
0         23631    100   14156         181 Cross Rd  30-06-2013 00:00
1         23631    100   14144         177 Cross Rd  30-06-2013 00:00
2         23632    100   14132         175 Cross Rd  30-06-2013 00:00
3         23633    100   12266  Zone A Arndale Interchange  30-06-2013 00:00
4         23633    100   14147         178 Cross Rd  30-06-2013 00:00
...        ...    ...    ...                      ...               ...
1048570   45682    171   13929     8 Fullarton Rd  29-09-2013 00:00
1048571   45682    171   13758   3 Glen Osmond Rd  29-09-2013 00:00
1048572   45682    171   13967     9 Fullarton Rd  29-09-2013 00:00
1048573   45682    171   13808     5 Fullarton Rd  29-09-2013 00:00
1048574   45682    171   13845     6 Fullarton Rd  29-09-2013 00:00

         NumberOfBoardings
0                        1
1                        1
2                        1
3                        2
4                        1
...                    ...
1048570                  2
1048571                  3
1048572                  1
1048573                  1
1048574                  3

[1048575 rows x 6 columns]>
```

5. Cheking For Null Values

In [11]: `data.isnull().sum()`

Out[11]:
```
TripID              0
RouteID             0
StopID              0
StopName            0
WeekBeginning       0
NumberOfBoardings   0
dtype: int64
```

6. Check For Duplicate And Drop Them

In [12]: `dup = data.duplicated().any()`

In [13]: `print(dup)`

```
False
```

## 7. Get The Entire Statistics Of The Data

In [14]: `data.describe()`

Out[14]:

	TripID	StopID	NumberOfBoardings
count	1.048575e+06	1.048575e+06	1.048575e+06
mean	2.860299e+04	1.330114e+04	4.132290e+00
std	1.674656e+04	1.119243e+03	6.291338e+00
min	3.017000e+03	1.081700e+04	1.000000e+00
25%	1.162200e+04	1.269800e+04	1.000000e+00
50%	3.423400e+04	1.333500e+04	2.000000e+00
75%	4.512600e+04	1.371600e+04	4.000000e+00
max	6.258500e+04	1.849300e+04	1.930000e+02

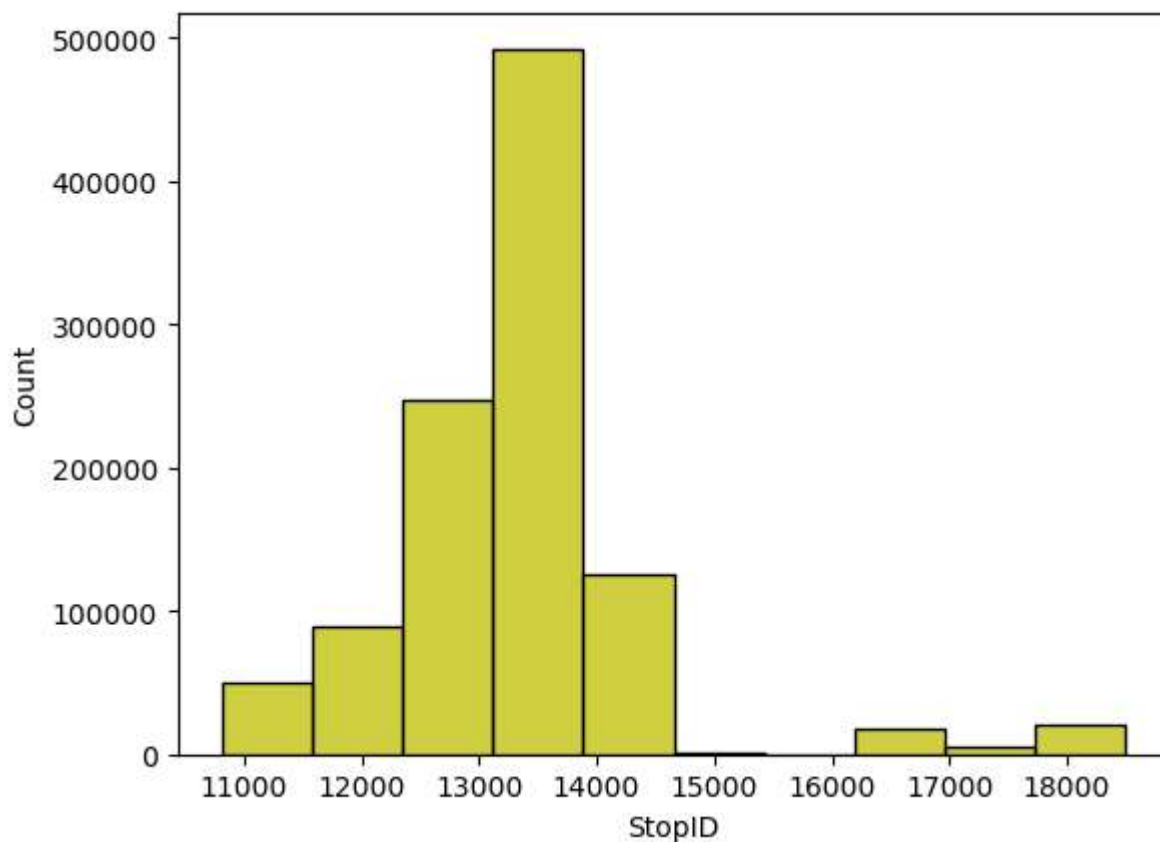# VISUALISING THE DATA

In [37]: `sns.histplot(data, x='TripID', bins=10, color='y')`

Out[37]: `<Axes: xlabel='TripID', ylabel='Count'>`
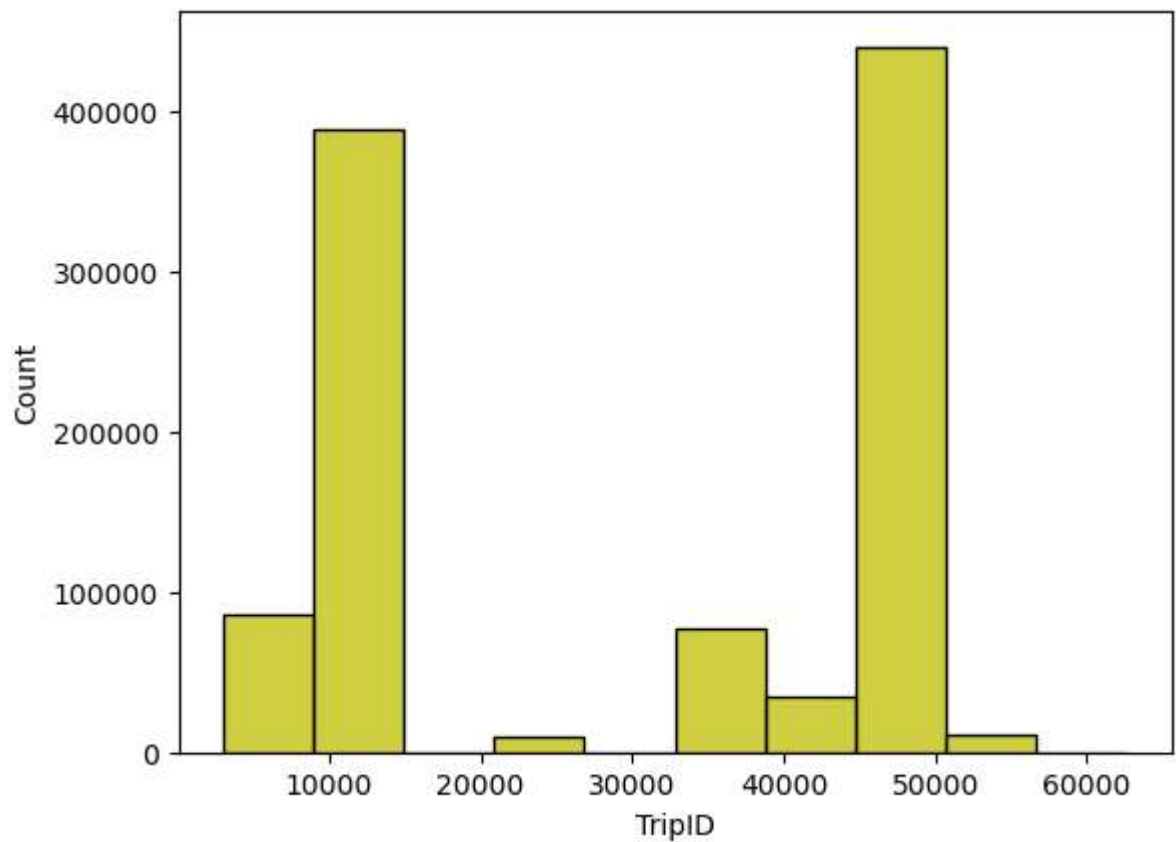
In [39]:
```python
sns.histplot(data, x='StopID', bins=10, color='y')
```
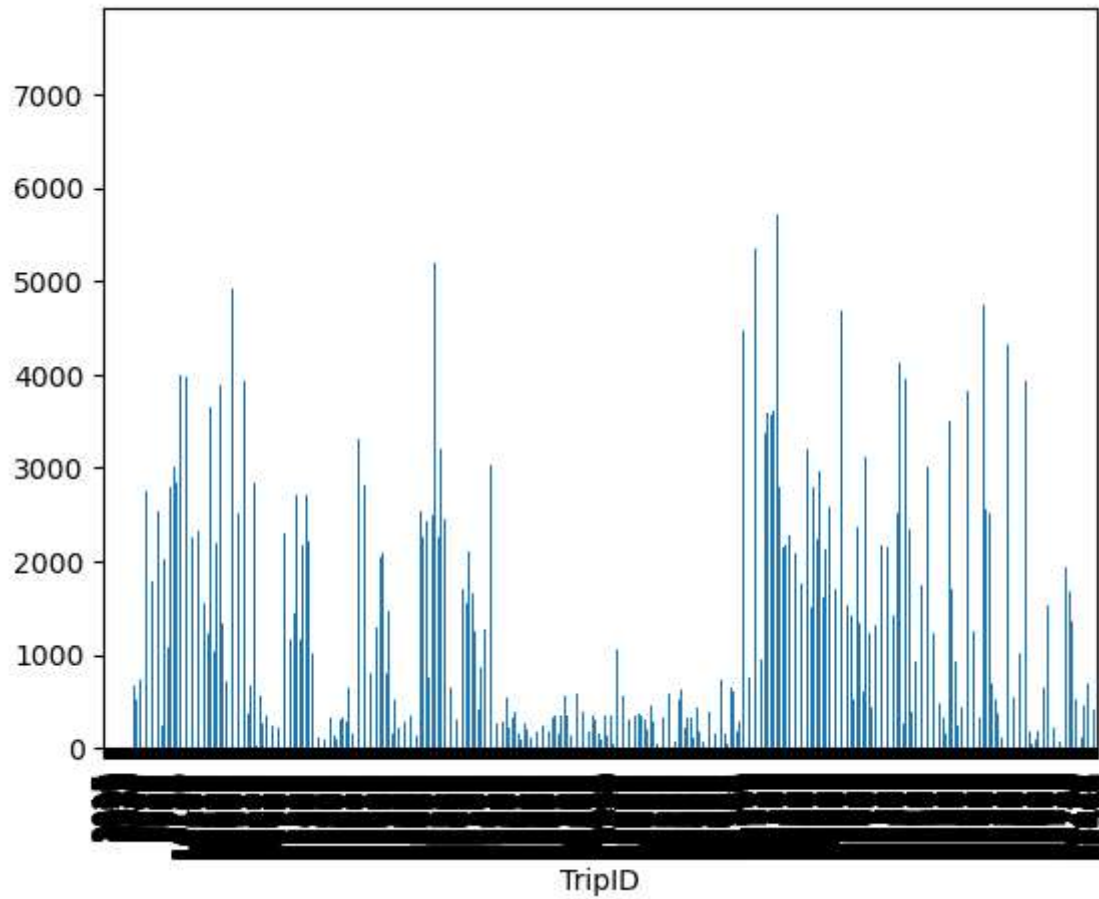
Out[39]: `<Axes: xlabel='StopID', ylabel='Count'>`

In [40]:
```python
M=(data.groupby('TripID')['NumberOfBoardings']).sum()
```

In [41]:
```python
M
```

Out[41]:
```
TripID
3017      2
3020      2
3021      1
3022      3
3023      1
         ..
62581     4
62582    11
62583     4
62584    11
62585    11
Name: NumberOfBoardings, Length: 3299, dtype: int64
```

In [42]:
```python
M.plot.bar()
plt.show()
```

In [43]:
```python
plt.figure(figsize=(12,8))
sns.pairplot(data)
```
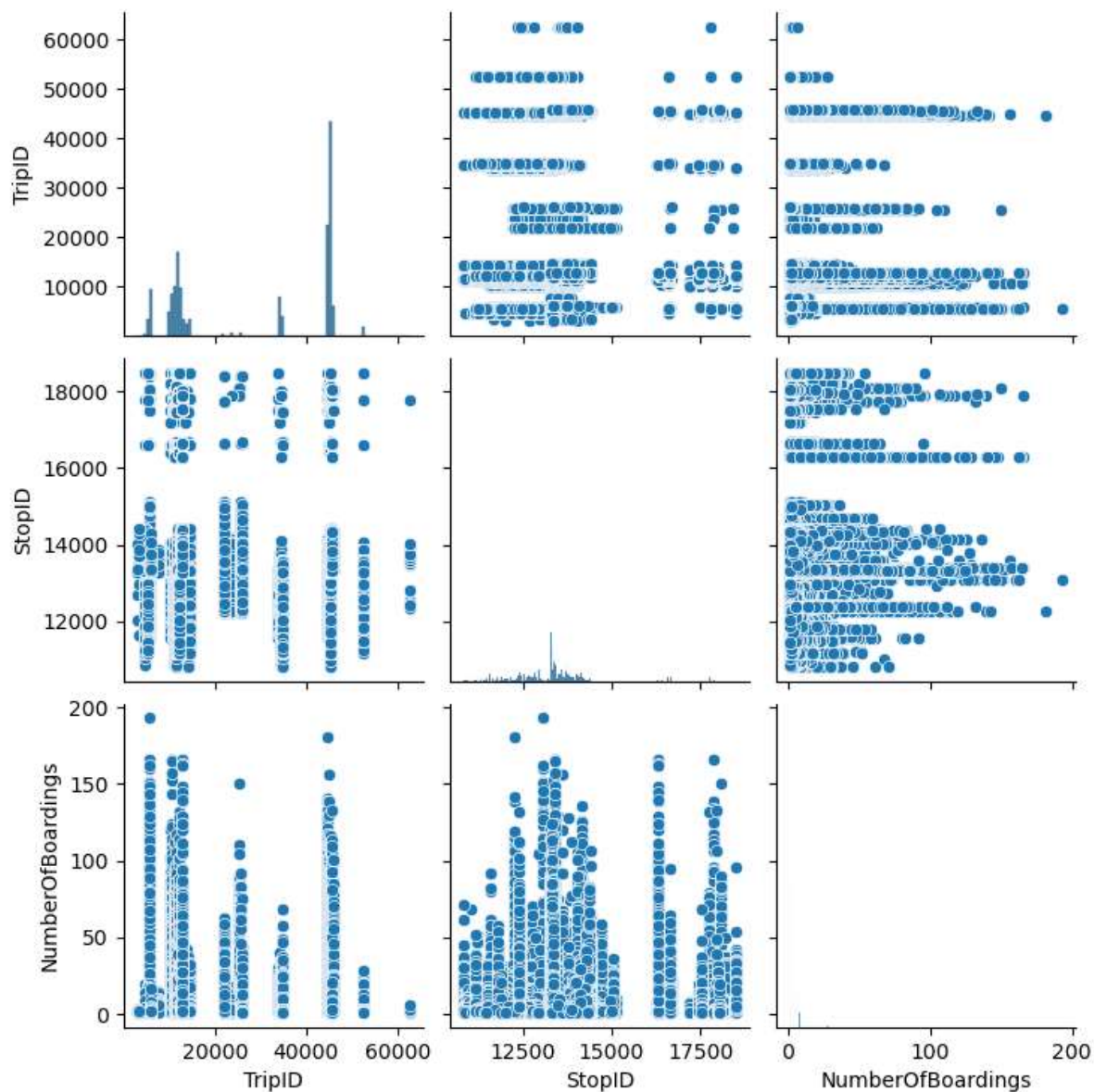
C:\Users\Indarjith K\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: Use
rWarning: The figure layout has changed to tight
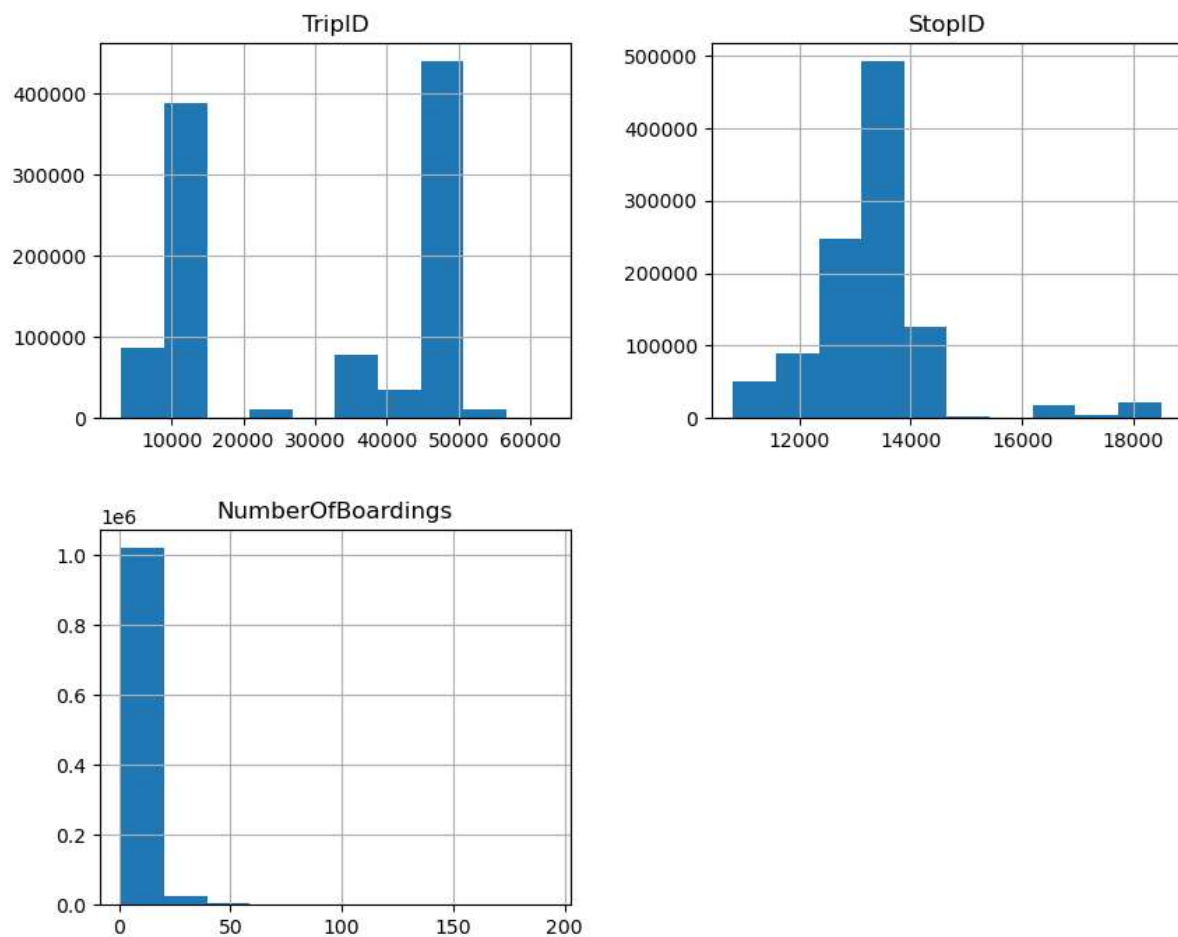   self._figure.tight_layout(*args, **kwargs)

Out[43]: <seaborn.axisgrid.PairGrid at 0x22e51e8c210>

<Figure size 1200x800 with 0 Axes>

In [44]:
```python
data.hist(figsize=(10,8))
```

Out[44]:
```
array([[<Axes: title={'center': 'TripID'}>,
        <Axes: title={'center': 'StopID'}>],
       [<Axes: title={'center': 'NumberOfBoardings'}>, <Axes: >]],
      dtype=object)
```


In [46]:
```python
C=data.groupby('StopName')['TripID'].sum().sort_values(ascending = False)
C
```

Out[46]:
```
StopName
I1 North Tce          357980471
23  Findon Rd         280075267
21 Port Rd            278666250
R1 North Tce          276122712
B1 East Tce           243863395
                         ...
X2 King William St        22448
V2 King William St        22444
I2 North Tce              12813
L1 Unley Rd               11221
11 East Av                 5613
Name: TripID, Length: 583, dtype: int64
```

```
In [54]: C.plot.bar(figsize=(20,5))
         plt.show()
```
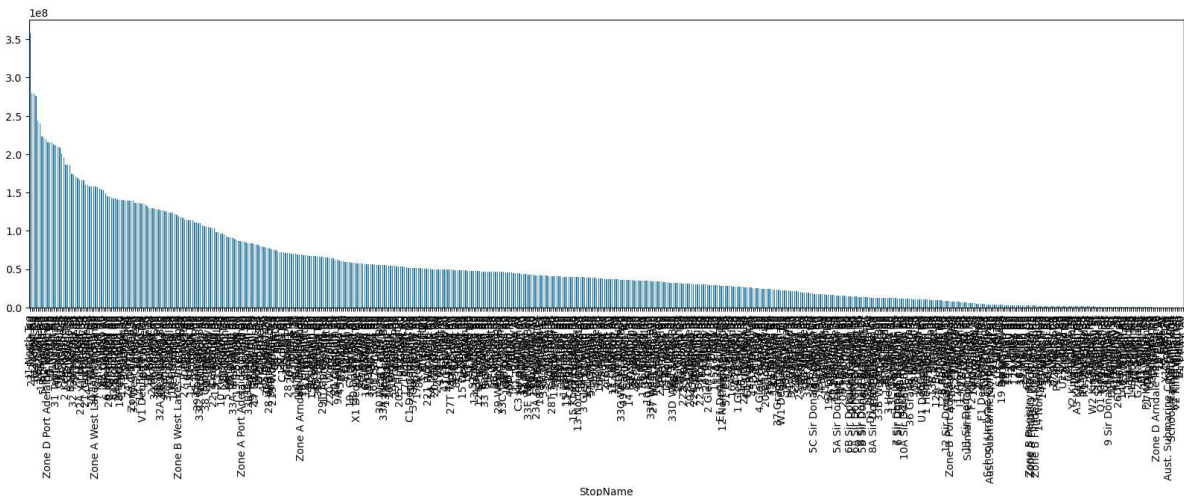

How many passengers weekBeginning

```
In [62]: WeekBeginning = data.groupby(['RouteID','WeekBeginning'])[['NumberOfBoardings']
         WeekBeginning
```

Out[62]:

		NumberOfBoardings
RouteID	**WeekBeginning**	
117	**01-06-2014 00:00**	7837
	01-09-2013 00:00	4435
	01-12-2013 00:00	7539
	02-02-2014 00:00	8272
	02-03-2014 00:00	8059
...	**...**	...
168	**07-07-2013 00:00**	5577
	14-07-2013 00:00	5411
	21-07-2013 00:00	6340
	28-07-2013 00:00	7046
	30-06-2013 00:00	6208

1519 rows × 1 columns

In [80]:
```python
WeekBeginning.unstack(level=0).plot(kind='bar',subplots=True,figsize=(20,10))
plt.show()
```


In [81]:
```python
data.corr
```

Out[81]:
```
<bound method DataFrame.corr of          TripID RouteID  StopID
StopName         WeekBeginning  \
0          23631      100    14156              181 Cross Rd  30-06-2013 00:00
1          23631      100    14144              177 Cross Rd  30-06-2013 00:00
2          23632      100    14132              175 Cross Rd  30-06-2013 00:00
3          23633      100    12266   Zone A Arndale Interchange  30-06-2013 00:00
4          23633      100    14147              178 Cross Rd  30-06-2013 00:00
...          ...      ...      ...                      ...               ...
1048570    45682      171    13929            8 Fullarton Rd  29-09-2013 00:00
1048571    45682      171    13758         3 Glen Osmond Rd  29-09-2013 00:00
1048572    45682      171    13967            9 Fullarton Rd  29-09-2013 00:00
1048573    45682      171    13808            5 Fullarton Rd  29-09-2013 00:00
1048574    45682      171    13845            6 Fullarton Rd  29-09-2013 00:00

         NumberOfBoardings
0                        1
1                        1
2                        1
3                        2
4                        1
...                    ...
1048570                  2
1048571                  3
1048572                  1
1048573                  1
1048574                  3

[1048575 rows x 6 columns]>
```

In [109]:
```python
sns.lineplot(x="TripID", y="StopID", data=data)
plt.show
```

Out[109]: <function matplotlib.pyplot.show(close=None, block=None)>

In [ ]:

In [ ]: