Import Libraries

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
import warnings
warnings.filterwarnings('ignore')
```

Load & Read the Dataset

In [2]:

```
survey = pd.read_csv(r'D:\DatSets\Survey_Data_Analysis\survey.csv')
survey.head(2)
```

Out[2]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfactior
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Dissatisfiec
1	2	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	RLSP	JD(U)	Somewha Satisfiec

How many Samples Collected Each Day

In [3]:

```
survey.head()
```

Out[3]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfactior
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Dissatisfiec
1	2	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	RLSP	JD(U)	Somewha Satisfiec
2	3	5-24-2017	Rural	Graduate	₹ 5,000 - ₹ 10,000	JAP(L)	BJP	Fully Dissatisfiec
3	4	5-24-2017	Urban	Graduate	₹ 10,000 - ₹ 20,000	RJD	RJD	Fully Dissatisfiec
4	5	5-24-2017	Rural	Graduate	₹ 50,000 - ₹ 80,000	JD(U)	Did not vote	Fully Satisfiec

In [4]:

```
survey['collection_date'].unique()
```

Out[4]:

```
array(['5-24-2017', '5-25-2017', '5-26-2017', '5-27-2017', '5-28-2017',
       '5-29-2017', '5-30-2017', '5-31-2017', '6-1-2017'], dtype=object)
```

In [5]:

```
survey['collection_date'].nunique()
```

Out[5]:

```
9
```

In [6]:

```
survey['collection_date'].value_counts(ascending=True)
```

Out[6]:

```
5-31-2017     557
5-30-2017     582
5-26-2017     598
6-1-2017      607
5-29-2017     620
5-27-2017     665
5-28-2017     761
5-25-2017     998
5-24-2017    1479
Name: collection_date, dtype: int64
```

What Propotion of the total respodents were age less than 45 ?

In [7]:

```
survey.head(2)
```

Out[7]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfactior
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Dissatisfiec
1	2	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	RLSP	JD(U)	Somewha Satisfiec

In [8]:
```
# survey[survey['age']<45]  ERROR '<' not supported between instances of 'str'
```

In [9]:
```
survey.dtypes
```

Out[9]:
```
response_id           int64
collection_date      object
urban_rural          object
education            object
income               object
Vote_Now             object
Past_Vote            object
CM_satisfaction      object
MLA_satisfaction     object
age                  object
gender               object
assembly_no           int64
category             object
weight              float64
dtype: object
```

In [10]:
```
survey.age.unique()
```

Out[10]:
```
array(['26', '34', '25', '36', '22', '38', '23', '42', '51', '29', '24',
       '30', '39', '37', '19', '44', '53', '32', '21', '18', '20', '27',
       '48', '28', '45', '74', '35', '31', '70', '40', '49', '46', '43',
       '41', '33', '58', '24ko', '56', '50', '55', '54', '62', '60', '59',
       '47', '61', '52', '66', '57', '67', '65', '71', '63', '64', '68',
       '69', '99', '72', '75', '76'], dtype=object)
```

In [11]:
```
survey.age.value_counts()
```

Out[11]:

```
25      598
24      449
30      440
28      396
22      392
26      389
27      360
20      328
23      317
21      311
32      263
35      263
29      239
40      177
34      173
19      171
31      159
33      143
38      135
36      127
18      125
42      104
37       90
45       82
43       72
39       69
50       60
41       54
44       54
48       35
47       34
46       32
49       27
52       24
51       23
55       20
60       15
53       12
61       12
54       11
57       10
56        9
58        9
64        8
67        7
62        5
59        5
65        5
68        4
63        4
66        4
70        3
71        2
74        1
24ko      1
69        1
99        1
72        1
75        1
```

```
76        1
Name: age, dtype: int64
```

In [12]:
```
survey['age'].replace({'24ko' : 24}, inplace=True)
```

In [13]:
```
survey.age.unique()
```

Out[13]:
```
array(['26', '34', '25', '36', '22', '38', '23', '42', '51', '29', '24',
       '30', '39', '37', '19', '44', '53', '32', '21', '18', '20', '27',
       '48', '28', '45', '74', '35', '31', '70', '40', '49', '46', '43',
       '41', '33', '58', 24, '56', '50', '55', '54', '62', '60', '59',
       '47', '61', '52', '66', '57', '67', '65', '71', '63', '64', '68',
       '69', '99', '72', '75', '76'], dtype=object)
```

In [14]:
```
survey['age'] = survey.age.astype(int)
```

In [15]:
```
survey.age.dtypes
```

Out[15]:
```
dtype('int32')
```

In [16]:
```
survey[survey['age']<45]
```

Out[16]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_sati
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Di
1	2	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	RLSP	JD(U)	Sc
2	3	5-24-2017	Rural	Graduate	₹ 5,000 - ₹ 10,000	JAP(L)	BJP	Fully Di
3	4	5-24-2017	Urban	Graduate	₹ 10,000 - ₹ 20,000	RJD	RJD	Fully Di
4	5	5-24-2017	Rural	Graduate	₹ 50,000 - ₹ 80,000	JD(U)	Did not vote	Fully
...	...	...	...	...	...	...	...	
6862	6863	6-1-2017	Rural	10th Pass	BPL/Below ₹ 5,000	JD(U)	BJP	Fully
6863	6864	6-1-2017	Rural	Graduate	BPL/Below ₹ 5,000	RJD	RJD	Fully Di
6864	6865	6-1-2017	Urban	Graduate	₹ 5,000 - ₹ 10,000	JAP(L)	RJD	
6865	6866	6-1-2017	Rural	Professional Education	₹ 10,000 - ₹ 20,000	BJP	Did not vote	Fully
6866	6867	6-1-2017	Rural	10th Pass	₹ 5,000 - ₹ 10,000	JAP(L)	JAP(L)	Fully

6399 rows × 14 columns

In [17]:
```
survey.shape
```

Out[17]:
```
(6867, 14)
```

In [18]:
```
6399/6867 *100
```

Out[18]:
```
93.18479685452162
```

create new column in dataframe age_group. the age groups are 18-25,25-40,40-55,55+

In [19]:
```
survey.head(2)
```

Out[19]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfactior
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Dissatisfiec
1	2	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	RLSP	JD(U)	Somewha Satisfiec

In [20]:
```
sdata = survey.copy()
```

In [21]:
```
sdata.head(2)
```

Out[21]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfactior
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Dissatisfiec
1	2	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	RLSP	JD(U)	Somewha Satisfiec

In [22]:
```
sdata['age_group'] = survey['age']
```

In [23]:
```
sdata.head(2)
```

Out[23]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfactior
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Dissatisfiec
1	2	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	RLSP	JD(U)	Somewha Satisfiec

In [24]:
```python
s1 = sdata[(sdata['age_group']>18) & (sdata['age_group']<25)]
```

In [25]:
```python
s1['age_group'] ='18-25'
```

In [26]:
```python
s1.head(6)
```

Out[26]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfa
4	5	5-24-2017	Rural	Graduate	₹ 50,000 - ₹ 80,000	JD(U)	Did not vote	Fully Sat
8	9	5-24-2017	Urban	12th Pass	BPL/Below ₹ 5,000	BJP	BJP	Fully Sat
14	15	5-24-2017	Rural	Graduate	₹ 1,50,000 & above	BJP	BJP	Fully Sat
16	17	5-24-2017	Rural	Post - Graduation	BPL/Below ₹ 5,000	RJD	RJD	Fully Dissat
18	19	5-24-2017	Rural	Graduate	₹ 30,000 - ₹ 50,000	JAP(L)	JD(U)	Some Dissat
24	25	5-24-2017	Rural	12th Pass	₹ 50,000 - ₹ 80,000	Others	BJP	Fully Dissat

In [27]:
```python
s2= sdata[(sdata['age_group']>25) &( sdata['age_group']<40)]
```

In [28]:
```python
s2['age_group']='25-40'
```

In [29]:
```python
s2.head(6)
```

Out[29]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfac
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Dissati
1	2	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	RLSP	JD(U)	Some Sati
3	4	5-24-2017	Urban	Graduate	₹ 10,000 - ₹ 20,000	RJD	RJD	Fully Dissati
5	6	5-24-2017	Urban	Graduate	₹ 20,000 - ₹ 30,000	Others	BJP	Some Sati
6	7	5-24-2017	Urban	Graduate	BPL/Below ₹ 5,000	RJD	BJP	Fully Dissati
7	8	5-24-2017	Rural	Professional Education	₹ 1,50,000 & above	BJP	JD(U)	Some Dissati

```
In [30]: s3 = sdata[(sdata['age_group']>40 ) & (sdata['age_group']<55)]
```

```
In [31]: s3['age_group']='40-55'
```

```
In [32]: s3.head(6)
```

Out[32]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfacti
11	12	5-24-2017	Urban	Graduate	₹ 10,000 - ₹ 20,000	LJP	JD(U)	Somewh Satisfi
12	13	5-24-2017	Urban	Post - Graduation	₹ 5,000 - ₹ 10,000	Undecided	JD(U)	Somewh Dissatisfi
28	29	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	JD(U)	BJP	Fully Satisfi
30	31	5-24-2017	Urban	Graduate	₹ 30,000 - ₹ 50,000	NOTA	NOTA	Fully Dissatisfi
33	34	5-24-2017	Urban	Graduate	₹ 50,000 - ₹ 80,000	BJP	BJP	Fully Satisfi
59	60	5-24-2017	Rural	12th Pass	₹ 5,000 - ₹ 10,000	JD(U)	JD(U)	Fully Satisfi

```
In [33]: s4 = sdata[sdata['age_group']>55]
```

In [34]:
```python
s4['age_group'] ='55+'
```

In [35]:
```python
s4.head(6)
```

Out[35]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfac
69	70	5-24-2017	Rural	Graduate	₹ 30,000 - ₹ 50,000	RJD	INC	Fully Dissatis
74	75	5-24-2017	Urban	Graduate	₹ 30,000 - ₹ 50,000	BJP	BJP	Somev Dissatis
121	122	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	NOTA	Did not vote	Somev Satis
126	127	5-24-2017	Urban	Graduate	₹ 80,000 - ₹ 1,50,000	NOTA	BJP	Fully Dissatis
277	278	5-24-2017	Urban	Post - Graduation	₹ 30,000 - ₹ 50,000	BJP	BJP	Somev Satis
282	283	5-24-2017	Urban	Graduate	₹ 10,000 - ₹ 20,000	BJP	BJP	Somev Satis

In [36]:
```python
sdata= pd.concat([s1,s2,s3,s4])
```

In [37]:
```python
sdata.head()
```

Out[37]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfa
4	5	5-24-2017	Rural	Graduate	₹ 50,000 - ₹ 80,000	JD(U)	Did not vote	Fully Sat
8	9	5-24-2017	Urban	12th Pass	BPL/Below ₹ 5,000	BJP	BJP	Fully Sat
14	15	5-24-2017	Rural	Graduate	₹ 1,50,000 & above	BJP	BJP	Fully Sat
16	17	5-24-2017	Rural	Post - Graduation	BPL/Below ₹ 5,000	RJD	RJD	Fully Dissat
18	19	5-24-2017	Rural	Graduate	₹ 30,000 - ₹ 50,000	JAP(L)	JD(U)	Some Dissat

In [38]:
```
sdata.age_group.unique()
```

Out[38]:
```
array(['18-25', '25-40', '40-55', '55+'], dtype=object)
```

How many samples collected from each age group ? witch age group had most samples?

In [39]:
```
sdata['age_group'].value_counts()
```

Out[39]:
```
25-40    3246
18-25    1969
40-55     624
55+       108
Name: age_group, dtype: int64
```

What proportion of residents voted BJP vote now & past vote_now ?

In [40]:
```
survey.head()
```

Out[40]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfactior
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Dissatisfiec
1	2	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	RLSP	JD(U)	Somewha Satisfiec
2	3	5-24-2017	Rural	Graduate	₹ 5,000 - ₹ 10,000	JAP(L)	BJP	Fully Dissatisfiec
3	4	5-24-2017	Urban	Graduate	₹ 10,000 - ₹ 20,000	RJD	RJD	Fully Dissatisfiec
4	5	5-24-2017	Rural	Graduate	₹ 50,000 - ₹ 80,000	JD(U)	Did not vote	Fully Satisfiec

In [41]:
```
survey[(survey['Vote_Now']=='BJP') & (survey['Past_Vote']=='BJP')]
```

Out[41]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satis
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Diss
8	9	5-24-2017	Urban	12th Pass	BPL/Below ₹ 5,000	BJP	BJP	Fully S
10	11	5-24-2017	Urban	12th Pass	₹ 5,000 - ₹ 10,000	BJP	BJP	Sor S
14	15	5-24-2017	Rural	Graduate	₹ 1,50,000 & above	BJP	BJP	Fully S
15	16	5-24-2017	Urban	12th Pass	₹ 5,000 - ₹ 10,000	BJP	BJP	Fully Diss
...	...	...	...	...	...	...	...	
6824	6825	6-1-2017	Rural	Graduate	BPL/Below ₹ 5,000	BJP	BJP	Fully Diss
6832	6833	6-1-2017	Rural	Graduate	BPL/Below ₹ 5,000	BJP	BJP	Sor S
6845	6846	6-1-2017	Rural	12th Pass	BPL/Below ₹ 5,000	BJP	BJP	Sor Diss
6851	6852	6-1-2017	Urban	Graduate	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully S
6861	6862	6-1-2017	Rural	Graduate	BPL/Below ₹ 5,000	BJP	BJP	Fully Diss

1630 rows × 14 columns

In [42]:
```
survey.shape
```

Out[42]:
```
(6867, 14)
```

In [43]:
```
1630/6867  * 100
```

Out[43]:
```
23.736711810106307
```

Determine the residents who are fully satisfied with the performance of CM.So if there are collected 1000 samples on day 1, out of 1000 samples 300 are fully satisfied.so our answer is 0.3 for that day

In [44]:
```
survey.head(2)
```

Out[44]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfactior
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Dissatisfiec
1	2	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	RLSP	JD(U)	Somewha Satisfiec

In [45]:
```
survey.CM_satisfaction.unique()
```

Out[45]:
```
array(['Fully Dissatisfied', 'Somewhat Satisfied', 'Fully Satisfied',
       'Somewhat Dissatisfied', 'Can’t say'], dtype=object)
```

In [46]:
```
cm = survey[survey['CM_satisfaction']=='Fully Satisfied']
```

In [47]:
```
cm.head(2)
```

Out[47]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfact
4	5	5-24-2017	Rural	Graduate	₹ 50,000 - ₹ 80,000	JD(U)	Did not vote	Fully Satisf
8	9	5-24-2017	Urban	12th Pass	BPL/Below ₹ 5,000	BJP	BJP	Fully Satisf

In [48]:
```
a = cm.collection_date.value_counts()
print(a)
```

```
5-24-2017    146
5-25-2017     95
5-28-2017     92
6-1-2017      75
5-29-2017     73
5-30-2017     65
5-31-2017     62
5-26-2017     59
5-27-2017     54
Name: collection_date, dtype: int64
```

In [49]:
```
b = survey.collection_date.value_counts()
b
```

Out[49]:
```
5-24-2017    1479
5-25-2017     998
5-28-2017     761
5-27-2017     665
5-29-2017     620
6-1-2017      607
5-26-2017     598
5-30-2017     582
5-31-2017     557
Name: collection_date, dtype: int64
```

In [50]:

```
c= a/b *100
c
```

Out[50]:

```
5-24-2017     9.871535
5-25-2017     9.519038
5-26-2017     9.866221
5-27-2017     8.120301
5-28-2017    12.089356
5-29-2017    11.774194
5-30-2017    11.168385
5-31-2017    11.131059
6-1-2017     12.355848
Name: collection_date, dtype: float64
```

Create a day wise proporation of respondents that opted fully dissatisfied with their MLA. Create a line plotof the result with date on x-axis, and proporations on y-axis

In [51]:

```
survey.head(2)
```

Out[51]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfactior
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Dissatisfiec
1	2	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	RLSP	JD(U)	Somewha Satisfiec

In [52]:

```
survey.MLA_satisfaction.unique()
```

Out[52]:

```
array(['Fully Dissatisfied', 'Somewhat Satisfied', 'Fully Satisfied',
       'Somewhat Dissatisfied', 'Can’t say'], dtype=object)
```

In [53]:

```
mla = survey[survey['MLA_satisfaction']=='Fully Dissatisfied']
```

In [54]:

```
d=mla.collection_date.value_counts()
```

In [55]:

```
e =survey.collection_date.value_counts()
```

In [56]:

```
f=d/e *100
```

In [57]:

```
f
```

Out[57]:

```
5-24-2017    59.972955
5-25-2017    63.126253
5-26-2017    59.030100
5-27-2017    60.601504
5-28-2017    60.709593
5-29-2017    56.451613
5-30-2017    62.027491
5-31-2017    57.630162
6-1-2017     63.426689
Name: collection_date, dtype: float64
```

In [58]:

```
g= pd.DataFrame(f)
g.columns
```

Out[58]:

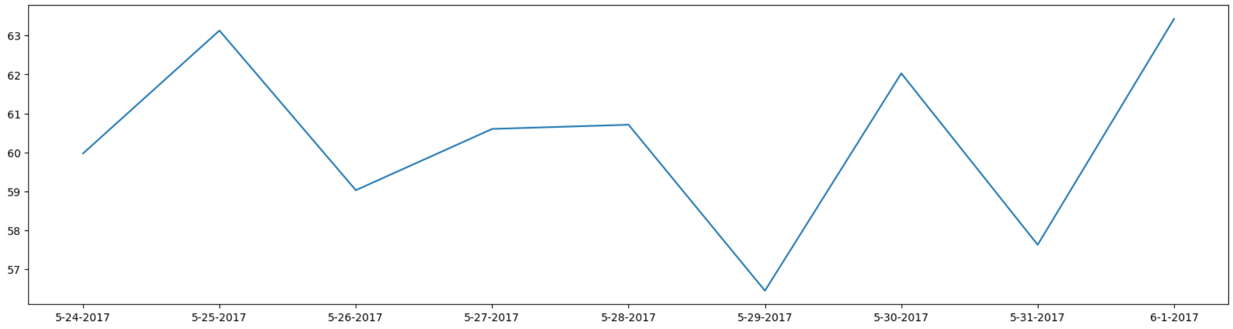
```
Index(['collection_date'], dtype='object')
```

In [59]:

```
g.collection_date.plot(kind='line', figsize=(20,5))
```

Out[59]:

```
<Axes: >
```



Create a pivot table with index as Past_vote, Columns as vote_now and cell values as thecount of samples

In [60]:

```
survey.head(2)
```

Out[60]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfactior
0	1	5-24-2017	Rural	12th Pass	₹ 30,000 - ₹ 50,000	BJP	BJP	Fully Dissatisfiec
1	2	5-24-2017	Rural	Graduate	₹ 20,000 - ₹ 30,000	RLSP	JD(U)	Somewha Satisfiec

In [61]:

```
survey.pivot_table(index='Past_Vote',columns='Vote_Now', aggfunc='count')
```

Out[61]:

	CM_satisfaction									...			
Vote_Now	**AAP**	**AIMIM**	**BJP**	**BSP**	**HAM**	**INC**	**JAP(L)**	**JD(U)**	**LJP**	**Left Front**	**...**	**LJP**	**Left Front**	**NOT**
Past_Vote														
BJP	7.0	1.0	1630.0	3.0	5.0	42.0	101.0	110.0	44.0	13.0	...	44.0	13.0	194.
BSP	NaN	1.0	2.0	5.0	NaN	1.0	NaN	NaN	NaN	NaN	...	NaN	NaN	Nal
Did not vote	8.0	5.0	231.0	2.0	1.0	22.0	41.0	69.0	10.0	16.0	...	10.0	16.0	83.
HAM	2.0	NaN	25.0	NaN	1.0	NaN	3.0	2.0	2.0	NaN	...	2.0	NaN	4.
INC	NaN	6.0	6.0	NaN	NaN	57.0	10.0	9.0	NaN	3.0	...	NaN	3.0	6.
JAP(L)	NaN	NaN	4.0	NaN	NaN	NaN	35.0	NaN	NaN	NaN	...	NaN	NaN	Nal
JD(U)	6.0	10.0	254.0	2.0	2.0	45.0	72.0	279.0	29.0	9.0	...	29.0	9.0	90.
LJP	NaN	NaN	76.0	1.0	NaN	4.0	3.0	6.0	9.0	2.0	...	9.0	2.0	3.
Left Front	NaN	1.0	2.0	NaN	NaN	1.0	NaN	NaN	NaN	18.0	...	NaN	18.0	1.
NOTA	NaN	1.0	29.0	1.0	NaN	9.0	7.0	7.0	3.0	5.0	...	3.0	5.0	46.
Others	1.0	5.0	45.0	1.0	1.0	4.0	7.0	4.0	1.0	4.0	...	1.0	4.0	10.
RJD	3.0	7.0	46.0	NaN	1.0	31.0	31.0	39.0	4.0	9.0	...	4.0	9.0	18.
RLSP	NaN	NaN	17.0	NaN	NaN	NaN	3.0	NaN	NaN	NaN	...	NaN	NaN	3.
VIP	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	NaN	NaN	...	NaN	NaN	Nal

14 rows × 216 columns

Repeted the above Question, with the cell values as the sum of 'Weights

In [62]:

```
survey.pivot_table(index='Past_Vote',columns='Vote_Now',values='weight', aggfunc='sum'
```

Out[62]:

Vote_Now	AAP	AIMIM	BJP	BSP	HAM	INC	JAP(L)	JD(U
Past_Vote								
BJP	4.807335	0.349996	1049.344112	0.452551	10.171283	50.055502	63.832368	68.02958
BSP	NaN	1.654731	5.174487	32.010277	NaN	2.298223	NaN	Na
Did not vote	8.865896	5.379527	114.203906	3.329611	2.497170	27.791290	26.359190	58.98345
HAM	0.312607	NaN	25.348884	NaN	1.648856	NaN	0.712161	1.00671
INC	NaN	11.105958	1.642875	NaN	NaN	161.742302	18.942597	11.07732
JAP(L)	NaN	NaN	6.512846	NaN	NaN	NaN	57.876224	Na
JD(U)	12.457216	29.198610	156.243100	11.005903	2.639574	91.806262	106.217216	498.79686
LJP	NaN	NaN	85.010450	2.481377	NaN	21.613385	7.375946	11.38281
Left Front	NaN	2.298223	0.209586	NaN	NaN	2.291368	NaN	Na
NOTA	NaN	1.061204	12.685435	0.051485	NaN	3.844332	8.645440	12.13236
Others	13.661335	21.212705	34.866218	0.684568	1.505864	1.389539	5.333891	5.10197
RJD	5.515847	24.149369	32.285093	NaN	0.960400	67.072640	28.661774	55.38567
RLSP	NaN	NaN	23.376575	NaN	NaN	NaN	3.269255	Na
VIP	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.33350

Create a dataframe by performinga group by over age_group and calculate the count of total samplesunder each age_group

In [63]:

```
sdata.head(2)
```

Out[63]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfact
4	5	5-24-2017	Rural	Graduate	₹ 50,000 - ₹ 80,000	JD(U)	Did not vote	Fully Satisf
8	9	5-24-2017	Urban	12th Pass	BPL/Below ₹ 5,000	BJP	BJP	Fully Satisf

In [64]:

```
df1 = sdata.groupby('age_group').count()
```

In [65]:

```
df1.head()
```

Out[65]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_sa
age_group								
18-25	1969	1969	1969	1969	1969	1969	1969	
25-40	3246	3246	3246	3246	3246	3246	3246	
40-55	624	624	624	624	624	624	624	
55+	108	108	108	108	108	108	108	

Create a dataframe by performinga group by over age_group and calculate the count of total samplesunder each age_group that opted for the BJP party Vote_NOw

In [66]:

```
sdata.head(2)
```

Out[66]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_satisfact
4	5	5-24-2017	Rural	Graduate	₹ 50,000 - ₹ 80,000	JD(U)	Did not vote	Fully Satis
8	9	5-24-2017	Urban	12th Pass	BPL/Below ₹ 5,000	BJP	BJP	Fully Satis

In [67]:

```
df_bjp = sdata[sdata['Vote_Now']=='BJP']
```

In [68]:

```
type(df_bjp)
```

Out[68]:

```
pandas.core.frame.DataFrame
```

In [69]:

```
df2 = df_bjp.groupby('age_group').count()
```

In [70]:

```
df2
```

Out[70]:

	response_id	collection_date	urban_rural	education	income	Vote_Now	Past_Vote	CM_sa
age_group								
18-25	569	569	569	569	569	569	569	
25-40	1149	1149	1149	1149	1149	1149	1149	
40-55	293	293	293	293	293	293	293	
55+	58	58	58	58	58	58	58	

Join/Merge the two dataframe above two questions with common column as age_group

In [71]:

```
pd.merge(df1,df2, on='age_group')
```

Out[71]:

	response_id_x	collection_date_x	urban_rural_x	education_x	income_x	Vote_Now_x	Past_
age_group							
18-25	1969	1969	1969	1969	1969	1969	
25-40	3246	3246	3246	3246	3246	3246	
40-55	624	624	624	624	624	624	
55+	108	108	108	108	108	108	

4 rows × 28 columns