In [37]:
```
import pandas as pd
import numpy as np
```

In [38]:
```
df = pd.read_csv('dataset_Facebook.csv',sep=';')
df
```

Out[38]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers
0	139441	Photo	2	12	4	3	0.0	2752	5091	178	109
1	139441	Status	2	12	3	10	0.0	10460	19057	1457	1361
2	139441	Photo	3	12	3	3	0.0	2413	4373	177	113
3	139441	Photo	2	12	2	10	1.0	50128	87991	2211	790
4	139441	Photo	2	12	2	3	0.0	7244	13594	671	410
...	...	...	...	...	...	...	...	...	...	...	...
495	85093	Photo	3	1	7	2	0.0	4684	7536	733	708
496	81370	Photo	2	1	5	8	0.0	3480	6229	537	508
497	81370	Photo	1	1	5	2	0.0	3778	7216	625	572
498	81370	Photo	3	1	4	11	0.0	4156	7564	626	574
499	81370	Photo	2	1	4	4	NaN	4188	7292	564	524

500 rows × 19 columns

In [39]:
```
df.describe()
```

Out[39]:

	Page total likes	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Impress
count	500.000000	500.000000	500.000000	500.000000	500.000000	499.000000	500.00000	5.000000
mean	123194.176000	1.880000	7.038000	4.150000	7.840000	0.278557	13903.36000	2.958595
std	16272.813214	0.852675	3.307936	2.030701	4.368589	0.448739	22740.78789	7.680325
min	81370.000000	1.000000	1.000000	1.000000	1.000000	0.000000	238.00000	5.700000
25%	112676.000000	1.000000	4.000000	2.000000	3.000000	0.000000	3315.00000	5.694750
50%	129600.000000	2.000000	7.000000	4.000000	9.000000	0.000000	5281.00000	9.051000
75%	136393.000000	3.000000	10.000000	6.000000	11.000000	1.000000	13168.00000	2.208550
max	139441.000000	3.000000	12.000000	7.000000	23.000000	1.000000	180480.00000	1.110282

In [40]:
```
df.shape
```

Out[40]:
```
(500, 19)
```

In [41]:
```
df[df['Type']=='Photo'].mean()
```

```
/tmp/ipykernel_25033/1850658754.py:1: FutureWarning: The default value of nume
ric_only in DataFrame.mean is deprecated. In a future version, it will default
to False. In addition, specifying 'numeric_only=None' is deprecated. Select on
ly valid columns or specify the value of numeric_only to silence this warning.
  df[df['Type']=='Photo'].mean()
```

Out[41]:
```
Page total likes                                                    122354.
171362
Category                                                                 1.
917840
Post Month                                                               6.
805164
Post Weekday                                                             4.
107981
Post Hour                                                                7.
997653
Paid                                                                     0.
280000
Lifetime Post Total Reach                                            13137.
814554
Lifetime Post Total Impressions                                      28994.
497653
Lifetime Engaged Users                                                 818.
946009
Lifetime Post Consumers                                                690.
431925
Lifetime Post Consumptions                                            1299.
025822
Lifetime Post Impressions by people who have liked your Page         16422.
483568
Lifetime Post reach by people who like your Page                      6059.
103286
Lifetime People who have liked your Page and engaged with your post    507.
305164
comment                                                                  7.
492958
like                                                                   182.
611765
share                                                                   27.
158768
Total Interactions                                                     216.
579812
dtype: float64
```

In [42]:
```
df1 = df[['Type','Category']].loc[0:15]
df1
```

Out[42]:

	Type	Category
0	Photo	2
1	Status	2
2	Photo	3
3	Photo	2
4	Photo	2
5	Status	2
6	Photo	3
7	Photo	3
8	Status	2
9	Photo	3
10	Status	2
11	Photo	2
12	Photo	2
13	Photo	2
14	Photo	2
15	Status	2

In [43]:
```
df2 = df[['Type','Category']].loc[15:30]
df2
```

Out[43]:

	Type	Category
15	Status	2
16	Photo	3
17	Photo	1
18	Status	3
19	Photo	3
20	Photo	2
21	Photo	1
22	Link	1
23	Photo	3
24	Status	2
25	Status	2
26	Status	2
27	Photo	3
28	Photo	2
29	Video	1
30	Photo	2

In [44]:
```
df3 = df[df['Type'] == 'Photo']
df3
```

Out[44]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers
0	139441	Photo	2	12	4	3	0.0	2752	5091	178	109
2	139441	Photo	3	12	3	3	0.0	2413	4373	177	113
3	139441	Photo	2	12	2	10	1.0	50128	87991	2211	790
4	139441	Photo	2	12	2	3	0.0	7244	13594	671	410
6	139441	Photo	3	12	1	3	1.0	11692	19479	481	265
...	...	...	...	...	...	...	...	...	...	...	...
495	85093	Photo	3	1	7	2	0.0	4684	7536	733	708
496	81370	Photo	2	1	5	8	0.0	3480	6229	537	508
497	81370	Photo	1	1	5	2	0.0	3778	7216	625	572
498	81370	Photo	3	1	4	11	0.0	4156	7564	626	574
499	81370	Photo	2	1	4	4	NaN	4188	7292	564	524

426 rows × 19 columns

merging

In [45]:
```
merged = pd.concat([df1,df2])
merged
```

Out[45]:

	Type	Category
0	Photo	2
1	Status	2
2	Photo	3
3	Photo	2
4	Photo	2
5	Status	2
6	Photo	3
7	Photo	3
8	Status	2
9	Photo	3
10	Status	2
11	Photo	2
12	Photo	2
13	Photo	2
14	Photo	2
15	Status	2
15	Status	2
16	Photo	3
17	Photo	1
18	Status	3
19	Photo	3
20	Photo	2
21	Photo	1
22	Link	1
23	Photo	3
24	Status	2
25	Status	2
26	Status	2
27	Photo	3
28	Photo	2
29	Video	1
30	Photo	2

In [51]:
```
sorted = df.sort_values(by='Page total likes',ascending=False,kind='quicksort',
sorted
```

Out[51]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers
0	139441	Photo	2	12	4	3	0.0	2752	5091	178	109
8	139441	Status	2	12	7	3	0.0	11844	22538	1530	1407
1	139441	Status	2	12	3	10	0.0	10460	19057	1457	1361
12	139441	Photo	2	12	5	10	0.0	2847	5133	193	115
11	139441	Photo	2	12	5	10	0.0	3112	5590	208	127
...	...	...	...	...	...	...	...	...	...	...	...
495	85093	Photo	3	1	7	2	0.0	4684	7536	733	708
496	81370	Photo	2	1	5	8	0.0	3480	6229	537	508
497	81370	Photo	1	1	5	2	0.0	3778	7216	625	572
498	81370	Photo	3	1	4	11	0.0	4156	7564	626	574
499	81370	Photo	2	1	4	4	NaN	4188	7292	564	524

500 rows × 19 columns

Transpose

In [47]:
```
transpose = df.transpose()
transpose
```

Out[47]:

	0	1	2	3	4	5	6	7	8	9	...	4
Page total likes	139441	139441	139441	139441	139441	139441	139441	139441	139441	139441	...	859
Type	Photo	Status	Photo	Photo	Photo	Status	Photo	Photo	Status	Photo	...	Pho
Category	2	2	3	2	2	2	3	3	2	3	...	
Post Month	12	12	12	12	12	12	12	12	12	12	...	
Post Weekday	4	3	3	2	2	1	1	7	7	6	...	
Post Hour	3	10	3	10	3	9	3	9	3	10	...	
Paid	0.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	0.0	0.0	...	0
Lifetime Post Total Reach	2752	10460	2413	50128	7244	10472	11692	13720	11844	4694	...	52
Lifetime Post Total Impressions	5091	19057	4373	87991	13594	20849	19479	24137	22538	8668	...	87
Lifetime Engaged Users	178	1457	177	2211	671	1191	481	537	1530	280	...	9
Lifetime Post Consumers	109	1361	113	790	410	1073	265	232	1407	183	...	9
Lifetime Post Consumptions	159	1674	154	1119	580	1389	364	305	1692	250	...	12
Lifetime Post Impressions by people who have liked your Page	3078	11710	2812	61027	6228	16034	15432	19728	15220	4309	...	57
Lifetime Post reach by people who like your Page	1640	6112	1503	32048	3200	7852	9328	11056	7912	2324	...	33
Lifetime People who have liked your Page and engaged with your post	119	1108	132	1386	396	1016	379	422	1250	199	...	4
comment	4	5	0	58	19	1	3	0	0	3	...	
like	79.0	130.0	66.0	1572.0	325.0	152.0	249.0	325.0	161.0	113.0	...	79
share	17.0	29.0	14.0	147.0	49.0	33.0	27.0	14.0	31.0	26.0	...	30
Total Interactions	100	164	80	1777	393	186	279	339	192	142	...	1

19 rows × 500 columns

In [48]:
```
df.shape
```

Out[48]:
```
(500, 19)
```

In [49]:
```
pivot_table = pd.pivot_table(df,index = ['Type','Category'],values = 'like')
pivot_table
```

Out[49]:

		like
Type	**Category**	
Link	**1**	75.650000
	2	32.000000
	3	68.000000
Photo	**1**	126.000000
	2	235.857143
	3	219.753333
Status	**1**	136.333333
	2	182.552632
	3	151.500000
Video	**1**	231.428571

In [50]:
```
arr = np.array([1,2,3,4,5,6,7,8,9])
arr.reshape(3,3)
```

Out[50]:
```
array([[1, 2, 3],
       [4, 5, 6],
       [7, 8, 9]])
```

In [56]:
```
df['index'] = range(1,len(df)+1)
df
```

Out[56]:

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifetime Engaged Users	Lifetime Post Consumers
0	139441	Photo	2	12	4	3	0.0	2752	5091	178	109
1	139441	Status	2	12	3	10	0.0	10460	19057	1457	1361
2	139441	Photo	3	12	3	3	0.0	2413	4373	177	113
3	139441	Photo	2	12	2	10	1.0	50128	87991	2211	790
4	139441	Photo	2	12	2	3	0.0	7244	13594	671	410
...	...	...	...	...	...	...	...	...	...	...	...
495	85093	Photo	3	1	7	2	0.0	4684	7536	733	708
496	81370	Photo	2	1	5	8	0.0	3480	6229	537	508
497	81370	Photo	1	1	5	2	0.0	3778	7216	625	572
498	81370	Photo	3	1	4	11	0.0	4156	7564	626	574
499	81370	Photo	2	1	4	4	NaN	4188	7292	564	524

500 rows × 20 columns

In [61]:
```
df_melted = pd.melt(df,id_vars='index',value_vars=['Type','Category'] )
df_melted
```

Out[61]:

	index	variable	value
0	1	Type	Photo
1	2	Type	Status
2	3	Type	Photo
3	4	Type	Photo
4	5	Type	Photo
...	...	...	...
995	496	Category	3
996	497	Category	2
997	498	Category	1
998	499	Category	3
999	500	Category	2

1000 rows × 3 columns

In [70]:
```
pt = pd.pivot_table(df_melted,index=['variable','value'],values='index')
pt
```

Out[70]:

		index
variable	value	
Category	1	245.227907
	2	207.153846
	3	294.167742
Type	Link	272.954545
	Photo	260.751174
	Status	160.577778
	Video	134.142857

In [75]:
```
pt2 = pd.pivot_table(df,index=['Type','Category'],values='comment')
pt2
```

Out[75]:

		comment
Type	Category	
Link	1	2.900000
	2	2.000000
	3	2.000000
Photo	1	5.897297
	2	11.692308
	3	6.913333
Status	1	4.333333
	2	9.921053
	3	2.750000
Video	1	12.285714

In [ ]:
```

```