

```

import pandas as pd

df = pd.read_csv('dataset_Facebook.csv', delimiter=';')

df

```

	Page	total likes	Type	Category	Post Month	Post Weekday
Post	Hour	\				
0	3	139441	Photo	2	12	4
1	10	139441	Status	2	12	3
2	3	139441	Photo	3	12	3
3	10	139441	Photo	2	12	2
4	3	139441	Photo	2	12	2
...
495	2	85093	Photo	3	1	7
496	8	81370	Photo	2	1	5
497	2	81370	Photo	1	1	5
498	11	81370	Photo	3	1	4
499	4	81370	Photo	2	1	4
\	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	\		
0	0.0	2752	5091			
1	0.0	10460	19057			
2	0.0	2413	4373			
3	1.0	50128	87991			
4	0.0	7244	13594			
...
495	0.0	4684	7536			
496	0.0	3480	6229			
497	0.0	3778	7216			

498	0.0	4156	7564
499	NaN	4188	7292

	Lifetime Engaged Users	Lifetime Post Consumers	\
0	178	109	
1	1457	1361	
2	177	113	
3	2211	790	
4	671	410	
..	
495	733	708	
496	537	508	
497	625	572	
498	626	574	
499	564	524	

	Lifetime Post Consumptions	\
0	159	
1	1674	
2	154	
3	1119	
4	580	
..	...	
495	985	
496	687	
497	795	
498	832	
499	743	

	Lifetime Post Impressions by people who have liked your Page	\
0	3078	
1	11710	
2	2812	
3	61027	
4	6228	
..	...	
495	4750	
496	3961	
497	4742	
498	4534	
499	3861	

	Lifetime Post reach by people who like your Page	\
0	1640	
1	6112	
2	1503	
3	32048	
4	3200	

```

..
495                                ...
496                                2876
497                                2104
498                                2388
498                                2452
499                                2200

    Lifetime People who have liked your Page and engaged with your
post \
0                                119

1                                1108
2                                132
3                                1386
4                                396

...
495                                ...
495                                392
496                                301
497                                363
498                                370
499                                316

      comment    like   share  Total Interactions
0          4    79.0   17.0           100
1          5   130.0   29.0           164
2          0    66.0   14.0            80
3         58  1572.0  147.0          1777
4         19   325.0   49.0           393
...
495          ...    ...    ...
495          5    53.0   26.0            84
496          0    53.0   22.0            75
497          4    93.0   18.0           115
498          7    91.0   38.0           136
499          0    91.0   28.0           119

[500 rows x 19 columns]

df.describe()

      Page total likes   Category Post Month Post Weekday Post
Hour \

```

count	500.000000	500.000000	500.000000	500.000000
500.000000				
mean	123194.176000	1.880000	7.038000	4.150000
7.840000				
std	16272.813214	0.852675	3.307936	2.030701
4.368589				
min	81370.000000	1.000000	1.000000	1.000000
1.000000				
25%	112676.000000	1.000000	4.000000	2.000000
3.000000				
50%	129600.000000	2.000000	7.000000	4.000000
9.000000				
75%	136393.000000	3.000000	10.000000	6.000000
11.000000				
max	139441.000000	3.000000	12.000000	7.000000
23.000000				

	Paid	Lifetime Post Total	Reach	Lifetime Post Total	Lifetime Post Total
Impressions \					
count	499.000000		500.000000		
5.000000e+02					
mean	0.278557		13903.36000		
2.958595e+04					
std	0.448739		22740.78789		
7.680325e+04					
min	0.000000		238.00000		
5.700000e+02					
25%	0.000000		3315.00000		
5.694750e+03					
50%	0.000000		5281.00000		
9.051000e+03					
75%	1.000000		13168.00000		
2.208550e+04					
max	1.000000		180480.00000		
1.110282e+06					

Lifetime Engaged Users	Lifetime Post Consumers \	
count	500.000000	500.000000
mean	920.344000	798.772000
std	985.016636	882.505013
min	9.000000	9.000000
25%	393.750000	332.500000
50%	625.500000	551.500000
75%	1062.000000	955.500000
max	11452.000000	11328.000000

Lifetime Post Consumptions \	
count	500.000000
mean	1415.130000
std	2000.594118

min	9.000000			
25%	509.250000			
50%	851.000000			
75%	1463.000000			
max	19779.000000			
Lifetime Post Impressions by people who have liked your Page \				
count	5.000000e+02			
mean	1.676638e+04			
std	5.979102e+04			
min	5.670000e+02			
25%	3.969750e+03			
50%	6.255500e+03			
75%	1.486050e+04			
max	1.107833e+06			
Lifetime Post reach by people who like your Page \				
count	500.000000			
mean	6585.488000			
std	7682.009405			
min	236.000000			
25%	2181.500000			
50%	3417.000000			
75%	7989.000000			
max	51456.000000			
Lifetime People who have liked your Page and engaged with your post \				
count	500.000000			
mean	609.986000			
std	612.725618			
min	9.000000			
25%	291.000000			
50%	412.000000			
75%	656.250000			
max	4376.000000			
	comment	like	share	Total Interactions
count	500.00000	499.000000	496.000000	500.000000
mean	7.48200	177.945892	27.266129	212.120000
std	21.18091	323.398742	42.613292	380.233118
min	0.00000	0.000000	0.000000	0.000000

25%	1.00000	56.50000	10.00000	71.00000
50%	3.00000	101.00000	19.00000	123.50000
75%	7.00000	187.50000	32.25000	228.50000
max	372.00000	5172.00000	790.00000	6334.00000

Create Data Subsets

```
# First subset: Like and Share
df_subset_1 = df[['like', 'share']]
df_subset_1
```

	like	share
0	79.0	17.0
1	130.0	29.0
2	66.0	14.0
3	1572.0	147.0
4	325.0	49.0
..
495	53.0	26.0
496	53.0	22.0
497	93.0	18.0
498	91.0	38.0
499	91.0	28.0

[500 rows x 2 columns]

```
# second subset: Comment and Type
df_subset_2 = df[['comment', 'Type']]
df_subset_2
```

	comment	Type
0	4	Photo
1	5	Status
2	0	Photo
3	58	Photo
4	19	Photo
..
495	5	Photo
496	0	Photo
497	4	Photo
498	7	Photo
499	0	Photo

[500 rows x 2 columns]

Merge Data

```
merged_data = pd.merge(df_subset_2, df_subset_1, left_on='comment',
right_on= 'like')
merged_data

      comment  Type  like  share
0            4  Photo   4.0    2.0
1            4  Photo   4.0    1.0
2            4  Photo   4.0    0.0
3            4  Photo   4.0    1.0
4            5  Status   5.0    2.0
...
1462          0  Photo   0.0    0.0
1463          0  Photo   0.0    0.0
1464          0  Photo   0.0    0.0
1465          0  Photo   0.0    0.0
1466          0  Photo   0.0    0.0

[1467 rows x 4 columns]
```

Example for merge

```
# Define a dictionary containing employee data
data1 = {
'key': ['K0', 'K1', 'K2', 'K3'],
'Name': ['Jai', 'Princi', 'Gaurav', 'Anuj'],
'Age':[27, 24, 22, 32],}

# Define a dictionary containing employee data
data2 = {
'key': ['K0', 'K1', 'K2', 'K3'],
'Address': ['Nagpur', 'Kanpur', 'Allahabad', 'Kannuaj'],
'Qualification': ['Btech', 'B.A', 'Bcom', 'B.hons']}

# Convert the dictionary into DataFrame
data1 = pd.DataFrame(data1)
# Convert the dictionary into DataFrame
data2 = pd.DataFrame(data2)

# print(df, "\n\n", df1)
res = pd.merge(data1, data2, on='key')
res

   key    Name  Age   Address Qualification
0   K0     Jai  27     Nagpur        Btech
1   K1  Princi  24     Kanpur         B.A
2   K2   Gaurav  22  Allahabad        Bcom
3   K3    Anuj  32    Kannuaj       B.hons
```

Sort Data

```
# Sorting merged_data in descending order wrt 'Like'  
merged_data.sort_values(by=['like'], ascending=False)
```

	comment	Type	like	share
1351	146	Photo	146.0	9.0
1352	146	Photo	146.0	15.0
549	144	Photo	144.0	10.0
550	144	Photo	144.0	29.0
711	64	Photo	64.0	19.0
...
891	0	Photo	0.0	0.0
892	0	Photo	0.0	0.0
895	0	Photo	0.0	0.0
896	0	Photo	0.0	0.0
1466	0	Photo	0.0	0.0

[1467 rows x 4 columns]

Transposing Data

```
# Method 1  
merged_data.transpose()
```

	0	1	2	3	4	5	6	7	8
comment	4	4	4	4	4	5	0	0	0
0									
Type	Photo	Photo	Photo	Photo	Status	Photo	Photo	Photo	Photo
Photo									
like	4.0	4.0	4.0	4.0	5.0	0.0	0.0	0.0	0.0
0.0									
share	2.0	1.0	0.0	1.0	2.0	0.0	0.0	0.0	0.0
0.0									
	9	...	1457	1458	1459	1460	1461	1462	1463
1464	\								
comment	0	...	7	7	7	7	7	0	0
0									
Type	Photo	...	Photo	Photo	Photo	Photo	Photo	Photo	Photo
Photo									
like	0.0	...	7.0	7.0	7.0	7.0	7.0	0.0	0.0
0.0									
share	0.0	...	2.0	NaN	2.0	3.0	2.0	0.0	0.0
0.0									
	1465	1466							

```

comment      0      0
Type        Photo   Photo
like        0.0    0.0
share       0.0    0.0

```

[4 rows x 1467 columns]

```

# Method 2
merged_data.T

```

	0	1	2	3	4	5	6	7	8
\comment	4	4	4	4	5	0	0	0	0
Type	Photo	Photo	Photo	Photo	Status	Photo	Photo	Photo	Photo
Photo									
like	4.0	4.0	4.0	4.0	5.0	0.0	0.0	0.0	0.0
0.0									
share	2.0	1.0	0.0	1.0	2.0	0.0	0.0	0.0	0.0
0.0									
1464 \comment	9	...	1457	1458	1459	1460	1461	1462	1463
Type	Photo	...	Photo	Photo	Photo	Photo	Photo	Photo	Photo
Photo									
like	0.0	...	7.0	7.0	7.0	7.0	7.0	0.0	0.0
0.0									
share	0.0	...	2.0	NaN	2.0	3.0	2.0	0.0	0.0
0.0									
comment	1465	1466							
Type	Photo	Photo							
like	0.0	0.0							
share	0.0	0.0							

[4 rows x 1467 columns]

Shape And Reshape Data

```
df
```

	Page	total	likes	Type	Category	Post	Month	Post	Weekday
Post	Hour	\							
0		139441	Photo		2		12		4
3									
1		139441	Status		2		12		3

10						
2	139441	Photo	3	12	3	
3						
3	139441	Photo	2	12	2	
10						
4	139441	Photo	2	12	2	
3						
..
..						
495	85093	Photo	3	1	7	
2						
496	81370	Photo	2	1	5	
8						
497	81370	Photo	1	1	5	
2						
498	81370	Photo	3	1	4	
11						
499	81370	Photo	2	1	4	
4						
Paid Lifetime Post Total Reach Lifetime Post Total Impressions						
\						
0	0.0		2752			5091
1	0.0		10460			19057
2	0.0		2413			4373
3	1.0		50128			87991
4	0.0		7244			13594
..
495	0.0		4684			7536
496	0.0		3480			6229
497	0.0		3778			7216
498	0.0		4156			7564
499	Nan		4188			7292
Lifetime Engaged Users Lifetime Post Consumers \						
0		178		109		
1		1457		1361		
2		177		113		
3		2211		790		

4	671	410
..
495	733	708
496	537	508
497	625	572
498	626	574
499	564	524
Lifetime Post Consumptions \		
0	159	
1	1674	
2	154	
3	1119	
4	580	
..	...	
495	985	
496	687	
497	795	
498	832	
499	743	
Lifetime Post Impressions by people who have liked your Page \		
0	3078	
1	11710	
2	2812	
3	61027	
4	6228	
..	...	
495	4750	
496	3961	
497	4742	
498	4534	
499	3861	
Lifetime Post reach by people who like your Page \		
0	1640	
1	6112	
2	1503	
3	32048	
4	3200	
..	...	
495	2876	
496	2104	
497	2388	
498	2452	
499	2200	
Lifetime People who have liked your Page and engaged with your post \		
0	119	

1		1108
2		132
3		1386
4		396
..		...
495		392
496		301
497		363
498		370
499		316

	comment	like	share	Total Interactions
0	4	79.0	17.0	100
1	5	130.0	29.0	164
2	0	66.0	14.0	80
3	58	1572.0	147.0	1777
4	19	325.0	49.0	393
..
495	5	53.0	26.0	84
496	0	53.0	22.0	75
497	4	93.0	18.0	115
498	7	91.0	38.0	136
499	0	91.0	28.0	119

[500 rows x 19 columns]

```
df.Type.unique()

array(['Photo', 'Status', 'Link', 'Video'], dtype=object)

# Reshape
# Comment is id_vars and Type is value_vars
pd.melt(df, id_vars =['Type'], value_vars =[ 'comment'])
```

	Type	variable	value
0	Photo	comment	4
1	Status	comment	5
2	Photo	comment	0
3	Photo	comment	58
4	Photo	comment	19

```
...     ...     ...     ...
495  Photo  comment      5
496  Photo  comment      0
497  Photo  comment      4
498  Photo  comment      7
499  Photo  comment      0
```

```
[500 rows x 3 columns]
```

Examples

```
# Reshape
df_temp = pd.DataFrame({'foo': ['one', 'one', 'one', 'two', 'two',
'two'],
                        'bar': ['A', 'B', 'C', 'A', 'B', 'C'],
                        'baz': [1, 2, 3, 4, 5, 6],
                        'zoo': ['x', 'y', 'z', 'q', 'w', 't']})
df_temp
   foo bar  baz zoo
0  one   A    1   x
1  one   B    2   y
2  one   C    3   z
3  two   A    4   q
4  two   B    5   w
5  two   C    6   t

df_temp.pivot(index='foo', columns='bar', values='baz')

bar  A  B  C
foo
one  1  2  3
two  4  5  6
```