Perform the following operations using Python on the Facebook metrics data sets

a. Create data subsets

b. Merge Data

c. Sort Data

d. Transposing Data

e. Shape and reshape Data

Uploading dataset and Importing required libraries

```
import numpy as np # linear algebra
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
df = pd.read_csv('dataset_Facebook.csv', sep=';')
df.head()
```

	Page total likes	Type	Category	Post Month	Post Weekday	Post Hour	Paid	Lifetime Post Total Reach	Lifetime Post Total Impressions	Lifeti Engag Use
0	139441	Photo	2	12	4	3	0.0	2752	5091	1
1	139441	Status	2	12	3	10	0.0	10460	19057	14
2	139441	Photo	3	12	3	3	0.0	2413	4373	1
3	139441	Photo	2	12	2	10	1.0	50128	87991	22
4	139441	Photo	2	12	2	3	0.0	7244	13594	6

```
#describing the dataframe
df.describe()
```

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

```
df.shape #displays total rows and columns in dataset
```

```
(500, 19)
```

Creating Subsets

```
#subset 1
df1 =df[['Page total likes', 'Category', 'Post Month', 'Post Weekday']].loc[0:15]
```

```
df1
```

	Page total likes	Category	Post Month	Post Weekday
0	139441	2	12	4
1	139441	2	12	3
2	139441	3	12	3
3	139441	2	12	2
4	139441	2	12	2
5	139441	2	12	1
6	139441	3	12	1
7	139441	3	12	7
8	139441	2	12	7
9	139441	3	12	6
10	139441	2	12	5
11	139441	2	12	5
12	139441	2	12	5
13	139441	2	12	5
14	138414	2	12	4
15	138414	2	12	3

```
#subset 2
df2 =df[['Page total likes', 'Category', 'Post Month', 'Post Weekday']].loc[16:30]
df2
```

	Page total likes	Category	Post Month	Post Weekday
16	138414	3	12	3
17	138414	1	12	2
18	138414	3	12	2
19	138414	3	12	1
20	138414	2	12	1
21	138414	1	12	7
22	138414	1	12	7
23	138414	3	12	7
24	138414	2	12	6
25	138458	2	12	6
26	138458	2	12	5
27	138458	3	12	5
28	138895	2	12	5
29	138895	1	12	4
30	138895	2	12	4

```
#subset 3
df3 =df[['Page total likes', 'Category', 'Post Month', 'Post Weekday']].loc[31:50]
df3
```

	Page total likes	Category	Post Month	Post Weekday
31	138895	2	12	3
32	138895	3	12	3
33	138895	3	12	2
34	138895	1	12	2
35	138895	2	12	1
36	138895	3	12	1
37	138895	1	12	7
38	138895	2	12	7
39	138895	1	12	7
40	138895	2	12	6
41	138895	1	12	6
42	138853	1	12	5

Merge Data

```
merging=pd.concat([df1,df2,df3])
merging
```

	Page total likes	Category	Post Month	Post Weekday
0	139441	2	12	4
1	139441	2	12	3
2	139441	3	12	3
3	139441	2	12	2
4	139441	2	12	2
5	139441	2	12	1
6	139441	3	12	1
7	139441	3	12	7
8	139441	2	12	7
9	139441	3	12	6
10	139441	2	12	5
11	139441	2	12	5
12	139441	2	12	5
13	139441	2	12	5
14	138414	2	12	4
15	138414	2	12	3
16	138414	3	12	3
17	138414	1	12	2

Sort Data

```
sort_values=df.sort_values('Page total likes', ascending=False)
sort_values
```

	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	C
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

Transposing data

```
# transpose
df.transpose()
```

	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	...	(
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

Shape and reshape of data

```
#shaping
shaping=df.shape
shaping
```

```
(500, 19)
```

```
#reshape
pivot_table = pd.pivot_table(df,index= ['Type', 'Category'], values='like')
print(pivot_table)
```

```
                      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
```

```
#extra command to reshape data using array
reshaping_arr=np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
reshaping_arr.reshape(5,2)
```

```
array([[ 1,  2],
       [ 3,  4],
```

```
[ 5,  6],
[ 7,  8],
[ 9, 10]])
```