

```
In [ ]: # 5 and 13 task
```

```
In [1]: import pandas as pd
```

```
In [4]: df=pd.read_csv('C:/Users/Admin/Desktop/mayank/facebook.csv',delimiter=';')
```

```
In [5]: df.head()
```

Out[5]:

	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	Lifetime Post Consumptions	Lifetin Po Impressio by peop who ha liked yo Pa
0	139441	Photo	2	12	4	3	0.0	2752	5091	178	109	159	30
1	139441	Status	2	12	3	10	0.0	10460	19057	1457	1361	1674	117
2	139441	Photo	3	12	3	3	0.0	2413	4373	177	113	154	28
3	139441	Photo	2	12	2	10	1.0	50128	87991	2211	790	1119	610
4	139441	Photo	2	12	2	3	0.0	7244	13594	671	410	580	62

```
In [6]: #creating data subset
```

```
In [7]: sub1=df[10:61]  
sub1
```

Out[7]:

	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	Lifetime Post Consumptions	Lifeti P Impressi by peo who h liked y P
10	139441	Status	2	12	5	10	0.0	21744	42334	4258	4100	4540	37
11	139441	Photo	2	12	5	10	0.0	3112	5590	208	127	145	3
12	139441	Photo	2	12	5	10	0.0	2847	5133	193	115	133	3
13	139441	Photo	2	12	5	3	0.0	2549	4896	249	134	168	3
14	138414	Photo	2	12	4	5	1.0	22784	39941	887	337	417	34
15	138414	Status	2	12	3	10	0.0	10060	19680	1264	1209	1425	17
16	138414	Photo	3	12	3	3	0.0	1722	2981	163	123	148	1
17	138414	Photo	1	12	2	12	1.0	53264	111785	1706	1103	1655	92
18	138414	Status	3	12	2	3	0.0	3930	7509	130	86	112	5
19	138414	Photo	3	12	1	11	0.0	1591	2825	121	88	111	2
20	138414	Photo	2	12	1	3	0.0	2848	5066	200	142	184	3
21	138414	Photo	1	12	7	10	0.0	1384	2467	15	15	20	2
22	138414	Link	1	12	7	10	0.0	3454	6853	118	104	130	6

	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	Lifetime Post Consumptions	Lifetime P Impressi by peo who h liked y P:
23	138414	Photo	3	12	7	3	0.0	2723	4888	176	118	143	2
24	138414	Status	2	12	6	10	0.0	8488	15294	1341	1270	1489	9
25	138458	Status	2	12	6	3	0.0	8284	15104	1521	1462	1711	10
26	138458	Status	2	12	5	11	0.0	19552	34143	2806	2531	3420	17
27	138458	Photo	3	12	5	3	0.0	2478	4306	212	124	149	2
28	138895	Photo	2	12	5	3	0.0	9560	18264	973	559	885	9
29	138895	Video	1	12	4	11	1.0	36208	61262	1141	1068	1728	30
30	138895	Photo	2	12	4	2	0.0	4940	9390	385	306	501	5
31	138895	Photo	2	12	3	10	0.0	1683	2929	192	171	221	1
32	138895	Photo	3	12	3	3	0.0	5280	9578	368	237	345	4
33	138895	Photo	3	12	2	9	0.0	3002	5318	268	185	247	3
34	138895	Photo	1	12	2	3	0.0	3766	7149	298	260	431	5
35	138895	Photo	2	12	1	11	0.0	4512	7808	423	284	431	5
36	138895	Photo	3	12	1	3	0.0	2690	4628	252	168	226	3
37	138895	Photo	1	12	7	10	1.0	19800	28663	479	424	805	5
38	138895	Status	2	12	7	9	0.0	17576	33058	5352	5202	6547	23
39	138895	Photo	1	12	7	3	0.0	3290	6085	306	284	402	4
40	138895	Status	2	12	6	11	0.0	13280	24198	2055	1912	2720	17
41	138895	Link	1	12	6	3	1.0	18480	28438	517	366	460	12
42	138353	Photo	1	12	5	10	0.0	7268	13989	2087	2079	12074	13
43	138353	Link	1	12	5	3	1.0	2645	4270	134	109	170	2
44	138353	Photo	1	12	4	11	0.0	4284	8387	355	316	513	7
45	138353	Link	1	12	4	3	1.0	7968	13023	206	158	223	6
46	138353	Status	1	12	3	11	0.0	16576	30612	3572	3464	4802	24
47	138353	Link	1	12	3	2	0.0	1925	3481	97	83	126	2
48	138353	Photo	1	12	2	11	0.0	3786	7329	338	283	450	6
49	138353	Link	1	12	2	2	0.0	1536	3094	84	76	99	2
50	138353	Photo	2	11	1	9	0.0	1728	3155	108	65	95	2
51	138329	Photo	1	11	1	3	1.0	25248	40125	726	467	863	16
52	138329	Photo	1	11	7	9	0.0	4894	8899	355	181	264	4
53	138329	Photo	1	11	7	3	0.0	2935	5439	237	182	401	3
54	138329	Photo	1	11	6	10	0.0	2425	4462	260	213	433	3
55	138329	Video	1	11	6	2	1.0	16416	31950	459	411	539	21
56	138329	Photo	1	11	5	11	0.0	5812	10465	343	204	301	4
57	138329	Photo	1	11	5	3	0.0	2545	4846	165	131	167	3
58	138329	Photo	1	11	4	10	0.0	2257	4372	230	173	327	3
59	138329	Photo	1	11	4	3	1.0	27072	84885	421	304	487	35
60	138185	Photo	1	11	3	11	1.0	10940	27951	417	335	591	15

```
In [9]: sub2=df.iloc[10:21,1:4]
sub2
```

```
Out[9]:
```

	Type	Category	Post Month
10	Status	2	12
11	Photo	2	12
12	Photo	2	12
13	Photo	2	12
14	Photo	2	12
15	Status	2	12
16	Photo	3	12
17	Photo	1	12
18	Status	3	12
19	Photo	3	12
20	Photo	2	12

```
In [11]: sub3=df.loc[10:21,['Type','Category','Post Month']]
sub3
```

```
Out[11]:
```

	Type	Category	Post Month
10	Status	2	12
11	Photo	2	12
12	Photo	2	12
13	Photo	2	12
14	Photo	2	12
15	Status	2	12
16	Photo	3	12
17	Photo	1	12
18	Status	3	12
19	Photo	3	12
20	Photo	2	12
21	Photo	1	12

```
In [12]: #concat datasets
```

```
In [13]: up=df.iloc[1:11,0:4]
down=df.iloc[11:21,0:4]
```

```
In [14]: up
```

```
Out[14]:
```

	Page total likes	Type	Category	Post Month
1	139441	Status	2	12
2	139441	Photo	3	12
3	139441	Photo	2	12
4	139441	Photo	2	12
5	139441	Status	2	12
6	139441	Photo	3	12
7	139441	Photo	3	12
8	139441	Status	2	12
9	139441	Photo	3	12

	Page total likes	Type	Category	Post Month
10	139441	Status	2	12

In [15]:

```
down
```

Out[15]:

	Page total likes	Type	Category	Post Month
11	139441	Photo	2	12
12	139441	Photo	2	12
13	139441	Photo	2	12
14	138414	Photo	2	12
15	138414	Status	2	12
16	138414	Photo	3	12
17	138414	Photo	1	12
18	138414	Status	3	12
19	138414	Photo	3	12
20	138414	Photo	2	12

In [17]:

```
pd.concat([up,down])
```

Out[17]:

	Page total likes	Type	Category	Post Month
1	139441	Status	2	12
2	139441	Photo	3	12
3	139441	Photo	2	12
4	139441	Photo	2	12
5	139441	Status	2	12
6	139441	Photo	3	12
7	139441	Photo	3	12
8	139441	Status	2	12
9	139441	Photo	3	12
10	139441	Status	2	12
11	139441	Photo	2	12
12	139441	Photo	2	12
13	139441	Photo	2	12
14	138414	Photo	2	12
15	138414	Status	2	12
16	138414	Photo	3	12
17	138414	Photo	1	12
18	138414	Status	3	12
19	138414	Photo	3	12
20	138414	Photo	2	12

In [18]:

```
#how to create dataframe

data=pd.DataFrame({
    'A':[1,2,3],
    'B':[4,5,6]
})
data
```

Out[18]:

	A	B
0	1	4

A B

1 2 5

2 3 6

In [19]:

```
#merge
```

In [20]:

```
df.shape
```

Out[20]:

```
(500, 19)
```

In [30]:

```
left=df.iloc[:,0:10]
right=df.iloc[:,9:20]
```

In [22]:

```
left
```

Out[22]:

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

500 rows × 10 columns

In [23]:

```
right
```

Out[23]:

	Lifetime Engaged Users	Lifetime Post Consumers	Lifetime Post Consumptions	Lifetime Post Impressions by people who have liked your Page	Lifetime Post reach by people who like your Page	Lifetime People who have liked your Page and engaged with your post	comment	like	share	Total Interactions
0	178	109	159	3078	1640	119	4	79.0	17.0	100
1	1457	1361	1674	11710	6112	1108	5	130.0	29.0	164
2	177	113	154	2812	1503	132	0	66.0	14.0	80
3	2211	790	1119	61027	32048	1386	58	1572.0	147.0	1777
4	671	410	580	6228	3200	396	19	325.0	49.0	393
...
495	733	708	985	4750	2876	392	5	53.0	26.0	84
496	537	508	687	3961	2104	301	0	53.0	22.0	75
497	625	572	795	4742	2388	363	4	93.0	18.0	115
498	626	574	832	4534	2452	370	7	91.0	38.0	136
499	564	524	743	3861	2200	316	0	91.0	28.0	119

500 rows × 10 columns

In [24]:

```
merged=pd.merge(left,right,how='inner',on='Lifetime Engaged Users')
```

merged

Out[24]:

														Life
	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	Lifetime Post Consumptions	Impress by pe who liked	
0	139441	Photo		2	12	4	3	0.0	2752	5091	178	109	159	
1	139441	Status		2	12	3	10	0.0	10460	19057	1457	1361	1674	1
2	139441	Photo		3	12	3	3	0.0	2413	4373	177	113	154	
3	139441	Photo		2	12	2	10	1.0	50128	87991	2211	790	1119	6
4	139441	Photo		2	12	2	3	0.0	7244	13594	671	410	580	
...	
701	85979	Photo		3	1	1	2	0.0	4908	7491	957	937	1153	
702	85979	Photo		3	1	7	2	0.0	4800	7754	975	938	1278	
703	85979	Photo		3	1	6	3	1.0	6184	10228	956	901	1140	
704	85093	Photo		3	1	7	10	0.0	5400	9218	810	756	1003	
705	85093	Photo		3	1	7	2	0.0	4684	7536	733	708	985	

706 rows × 19 columns



In [26]:

```
left_merged=pd.merge(left,right,how='right',on='Lifetime Engaged Users')
left_merged
```

Out[26]:

														Life
	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	Lifetime Post Consumptions	Impress by pe who liked	
0	139441	Photo		2	12	4	3	0.0	2752	5091	178	109	159	
1	139441	Status		2	12	3	10	0.0	10460	19057	1457	1361	1674	1
2	139441	Photo		3	12	3	3	0.0	2413	4373	177	113	154	
3	139441	Photo		2	12	2	10	1.0	50128	87991	2211	790	1119	6
4	139441	Photo		2	12	2	3	0.0	7244	13594	671	410	580	
...	
701	135195	Photo		2	9	5	2	0.0	4518	8533	626	574	832	
702	81370	Photo		3	1	4	11	0.0	4156	7564	626	574	832	
703	125612	Photo		3	6	1	3	0.0	2763	4388	564	524	743	
704	111620	Photo		3	4	3	7	0.0	6596	9835	564	524	743	
705	81370	Photo		2	1	4	4	NaN	4188	7292	564	524	743	

706 rows × 19 columns



In []:

```
In [32]: left_hard=pd.DataFrame({
        'key':['A','B','C'],
        'A':[1,2,3],
        'B':[4,5,6]
    })
left_hard
```

```
Out[32]:
```

	key	A	B
0	A	1	4
1	B	2	5
2	C	3	6

```
In [34]: right_hard=pd.DataFrame({
        'key':['A','B','D'],
        'C':[1,2,3],
        'D':[4,5,6]
    })
right_hard
```

```
Out[34]:
```

	key	C	D
0	A	1	4
1	B	2	5
2	D	3	6

```
In [35]: inner_join=pd.merge(left_hard,right_hard,how='inner',on='key')
inner_join
```

```
Out[35]:
```

	key	A	B	C	D
0	A	1	4	1	4
1	B	2	5	2	5

```
In [36]: left_join=pd.merge(left_hard,right_hard,how='left',on='key')
left_join
```

```
Out[36]:
```

	key	A	B	C	D
0	A	1	4	1.0	4.0
1	B	2	5	2.0	5.0
2	C	3	6	NaN	NaN

```
In [37]: right_join=pd.merge(left_hard,right_hard,how='right',on='key')
right_join
```

```
Out[37]:
```

	key	A	B	C	D
0	A	1.0	4.0	1	4
1	B	2.0	5.0	2	5
2	D	NaN	NaN	3	6

```
In [42]: soorted=df.sort_values('Lifetime Post Total Reach')
soorted
```

```
Out[42]:
```

	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	Lifetime Post Consumptions	Life: Impress by pe who l liked ; I
422	102112	Photo	1	3	1	19	0.0	238	570	143	142	834	
426	100732	Photo	1	3	7	18	0.0	391	746	131	130	766	
432	100732	Photo	1	3	6	17	0.0	452	726	186	184	889	
120	136393	Photo	1	10	7	9	0.0	584	1029	273	271	308	
123	136393	Photo	1	10	7	7	0.0	617	1071	229	223	265	
...	
380	111620	Photo	1	4	7	14	0.0	128064	251269	1539	1408	2506	8.
277	126424	Video	1	6	2	13	0.0	139008	277100	1779	1643	2356	10
463	92186	Photo	3	2	7	2	1.0	153536	497910	1713	1633	2493	;
464	92079	Photo	1	2	6	13	0.0	158208	453213	2482	2319	3412	8
244	130791	Photo	2	7	3	5	1.0	180480	319133	8072	4010	6242	10;

500 rows × 19 columns

◀		▶
---	--	---

In [43]:

```
as_soorted=df.sort_values('Lifetime Post Total Reach',ascending=False)
as_soorted
```

Out[43]:

	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	Lifetime Post Consumptions	Life: Impress by pe who l liked ; I
244	130791	Photo	2	7	3	5	1.0	180480	319133	8072	4010	6242	10;
464	92079	Photo	1	2	6	13	0.0	158208	453213	2482	2319	3412	8
463	92186	Photo	3	2	7	2	1.0	153536	497910	1713	1633	2493	;
277	126424	Video	1	6	2	13	0.0	139008	277100	1779	1643	2356	10
380	111620	Photo	1	4	7	14	0.0	128064	251269	1539	1408	2506	8.
...	
123	136393	Photo	1	10	7	7	0.0	617	1071	229	223	265	
120	136393	Photo	1	10	7	9	0.0	584	1029	273	271	308	
432	100732	Photo	1	3	6	17	0.0	452	726	186	184	889	
426	100732	Photo	1	3	7	18	0.0	391	746	131	130	766	
422	102112	Photo	1	3	1	19	0.0	238	570	143	142	834	

500 rows × 19 columns

◀		▶
---	--	---

In [44]:

```
trans=df.T
trans
```


Out[44]:

	0	1	2	3	4	5	6	7	8	9	...	490	491	492	493
Page total likes	139441	139441	139441	139441	139441	139441	139441	139441	139441	139441	...	85979	85979	85979	85093
Type	Photo	Status	Photo	Photo	Photo	Status	Photo	Photo	Status	Photo	...	Photo	Photo	Link	Photo
Category	2	2	3	2	2	2	3	3	2	3	...	3	3	1	3
Post Month	12	12	12	12	12	12	12	12	12	12	...	1	1	1	1
Post Weekday	4	3	3	2	2	1	1	7	7	6	...	6	6	5	1
Post Hour	3	10	3	10	3	9	3	9	3	10	...	11	3	11	2
Paid	0.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	0.0	0.0	...	0.0	1.0	0.0	0.0
Lifetime Post Total Reach	2752	10460	2413	50128	7244	10472	11692	13720	11844	4694	...	5280	6184	45920	8412
Lifetime Post Total Impressions	5091	19057	4373	87991	13594	20849	19479	24137	22538	8668	...	8703	10228	5808	13960
Lifetime Engaged Users	178	1457	177	2211	671	1191	481	537	1530	280	...	951	956	753	1179
Lifetime Post Consumers	109	1361	113	790	410	1073	265	232	1407	183	...	911	901	655	1111
Lifetime Post Consumptions	159	1674	154	1119	580	1389	364	305	1692	250	...	1237	1140	763	1632
Lifetime Post Impressions by people who have liked your Page	3078	11710	2812	61027	6228	16034	15432	19728	15220	4309	...	5757	6085	15766	8632
Lifetime Post reach by people who like your Page	1640	6112	1503	32048	3200	7852	9328	11056	7912	2324	...	3300	3502	10720	5348
Lifetime People who have liked your Page and engaged with your post	119	1108	132	1386	396	1016	379	422	1250	199	...	431	437	220	699
comment	4	5	0	58	19	1	3	0	0	3	...	1	1	0	17
like	79.0	130.0	66.0	1572.0	325.0	152.0	249.0	325.0	161.0	113.0	...	79.0	105.0	128.0	185.0
share	17.0	29.0	14.0	147.0	49.0	33.0	27.0	14.0	31.0	26.0	...	30.0	46.0	9.0	55.0
Total Interactions	100	164	80	1777	393	186	279	339	192	142	...	110	152	137	257

19 rows × 500 columns



In [45]:

```
import numpy as np
```

In [48]:

```
a=np.array([[1,2,3],[4,5,6],[7,8,9],[7,8,9]])  
a
```

Out[48]:

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

In [49]:

```
a.shape
```

Out[49]:

```
(4, 3)
```

In [50]:

```
new_a=a.reshape((3,4))
```

```
In [51]: new_a

Out[51]: array([[1, 2, 3, 4],
               [5, 6, 7, 8],
               [9, 7, 8, 9]])

In [53]: new_a.shape

Out[53]: (3, 4)
```

cleaning and integration (6 and 7)

```
In [55]: dfh=pd.read_csv('C:/Users/Admin/Desktop/mayank/Heart.csv')
```

```
In [56]: dfh.head()
```

```
Out[56]:
```

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpeak	Slope	Ca	Thal	AHD
0	1	63	1	typical	145	233	1	2	150	0	2.3	3	0.0	fixed	No
1	2	67	1	asymptomatic	160	286	0	2	108	1	1.5	2	3.0	normal	Yes
2	3	67	1	asymptomatic	120	229	0	2	129	1	2.6	2	2.0	reversable	Yes
3	4	37	1	nonanginal	130	250	0	0	187	0	3.5	3	0.0	normal	No
4	5	41	0	nontypical	130	204	0	2	172	0	1.4	1	0.0	normal	No

```
In [68]: dfh.isnull().sum()
```

```
Out[68]: Unnamed: 0    0
Age          0
Sex          0
ChestPain    0
RestBP       0
Chol         0
Fbs          0
RestECG      0
MaxHR        0
ExAng        0
Oldpeak      0
Slope        0
Ca           4
Thal         2
AHD          0
dtype: int64
```

```
In [69]: mean=dfh['Ca'].mean()
mean
```

```
Out[69]: 0.6722408026755853
```

```
In [71]: dfh['Ca']=dfh['Ca'].fillna(mean,inplace=False)
```

```
In [72]: dfh.isnull().sum()
```

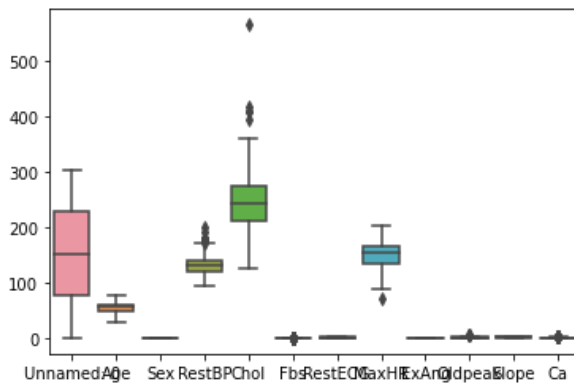
```
Out[72]: Unnamed: 0    0
Age          0
Sex          0
ChestPain    0
RestBP       0
Chol         0
Fbs          0
RestECG      0
MaxHR        0
ExAng        0
Oldpeak      0
Slope        0
Ca           0
Thal         2
```

```
AHD
dtype: int64
```

```
In [66]: import seaborn as sns
```

```
In [73]: sns.boxplot(data=dfh)
```

```
Out[73]: <AxesSubplot:>
```



```
In [ ]: # Q3 is 75
#Q1 is 25

#range=Q3-Q1

#upper=Q3+1.5*range
#Lower=Q1-1.5*range
```

```
In [75]: Q1=np.percentile(dfh['RestBP'],25)
Q1
```

```
Out[75]: 120.0
```

```
In [76]: Q3=np.percentile(dfh['RestBP'],75)
Q3
```

```
Out[76]: 140.0
```

```
In [78]: range=Q3-Q1
range
```

```
Out[78]: 20.0
```

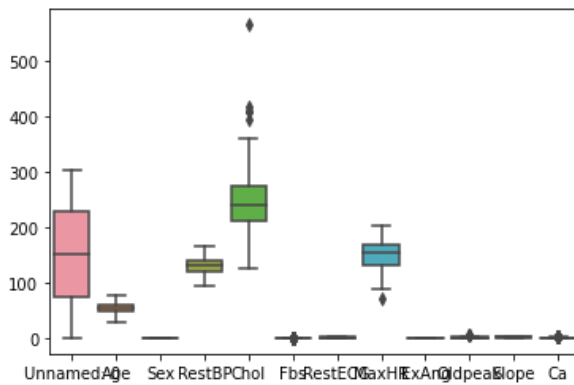
```
In [82]: upper=Q3+1.5*range
lower=Q1-1.5*range
print(upper,lower)
```

```
170.0 90.0
```

```
In [84]: rmout=dfh[(dfh['RestBP']>lower)&(dfh['RestBP']<upper)]
```

```
In [86]: sns.boxplot(data=rmout)
```

```
Out[86]: <AxesSubplot:>
```



```
In [90]: from sklearn.linear_model import LogisticRegression
```

```
In [92]: model=LogisticRegression()
```

```
In [93]: dfh.head()
```

```
Out[93]:
```

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpeak	Slope	Ca	Thal	AHD
0	1	63	1	typical	145	233	1	2	150	0	2.3	3	0.0	fixed	No
1	2	67	1	asymptomatic	160	286	0	2	108	1	1.5	2	3.0	normal	Yes
2	3	67	1	asymptomatic	120	229	0	2	129	1	2.6	2	2.0	reversable	Yes
3	4	37	1	nonanginal	130	250	0	0	187	0	3.5	3	0.0	normal	No
4	5	41	0	nontypical	130	204	0	2	172	0	1.4	1	0.0	normal	No

```
In [96]: x=dfh[['Age']]
         y=dfh[['Fbs']]
```

```
In [97]: model.fit(x,y)
```

C:\Users\Admin\anaconda3\lib\site-packages\sklearn\utils\validation.py:63: DataConversionWarning: A column-vector or y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().

```
return f(*args, **kwargs)
```

```
Out[97]: LogisticRegression()
```

```
In [100... pre=model.predict([[12]])
pre
```

```
Out[100... array([0], dtype=int64)
```

```
In [103... from sklearn.model_selection import train_test_split
```

```
In [104... x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=42)
```

```
In [105... newmodel=LogisticRegression()
```

```
In [106... newmodel.fit(x_train,y_train)
```

C:\Users\Admin\anaconda3\lib\site-packages\sklearn\utils\validation.py:63: DataConversionWarning: A column-vector or y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using ravel().

```
return f(*args, **kwargs)
```

```
Out[106... LogisticRegression()
```

```
In [112... x_test
```

```
Out[112...]
      Age
179    53
228    54
111    56
246    58
 60    51
...    ...
249    62
104    49
300    57
193    62
184    60
```

61 rows × 1 columns

```
In [110...]
y_pred=newmodel.predict(x_test)
```

```
In [111...]
y_pred
```

```
Out[111...]
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], dtype=int64)
```

```
In [113...]
from sklearn.metrics import confusion_matrix
```

```
In [114...]
confusion_matrix(y_test,y_pred)
```

```
Out[114...]
array([[49,  0],
       [12,  0]], dtype=int64)
```

visualization

```
In [115...]
dfh.head()
```

```
Out[115...]
   Unnamed: 0  Age  Sex  ChestPain  RestBP  Chol  Fbs  RestECG  MaxHR  ExAng  Oldpeak  Slope  Ca  Thal  AHD
0           1    63    1    typical    145   233    1         2    150     0      2.3     3  0.0   fixed    No
1           2    67    1  asymptomatic    160   286    0         2    108     1      1.5     2  3.0   normal    Yes
2           3    67    1  asymptomatic    120   229    0         2    129     1      2.6     2  2.0  reversable  Yes
3           4    37    1   nonanginal    130   250    0         0    187     0      3.5     3  0.0   normal    No
4           5    41    0   nontypical    130   204    0         2    172     0      1.4     1  0.0   normal    No
```

```
In [116...]
df=pd.read_csv('C:/Users/Admin/Desktop/mayank/iris.csv')
```

```
In [124...]
df['Species']
```

```
Out[124...]
   Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm  Species
0    1             5.1             3.5             1.4             0.2  Iris-setosa
1    2             4.9             3.0             1.4             0.2  Iris-setosa
2    3             4.7             3.2             1.3             0.2  Iris-setosa
3    4             4.6             3.1             1.5             0.2  Iris-setosa
4    5             5.0             3.6             1.4             0.2  Iris-setosa
...  ...           ...           ...           ...           ...  ...
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

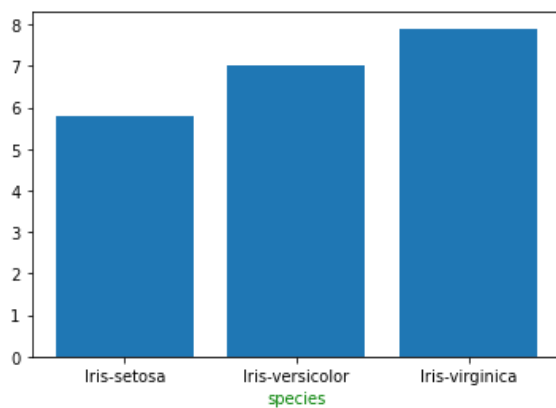
150 rows × 6 columns

```
In [119... import matplotlib.pyplot as plt
```

```
In [128... x=df['Species']
y=df['SepalLengthCm']

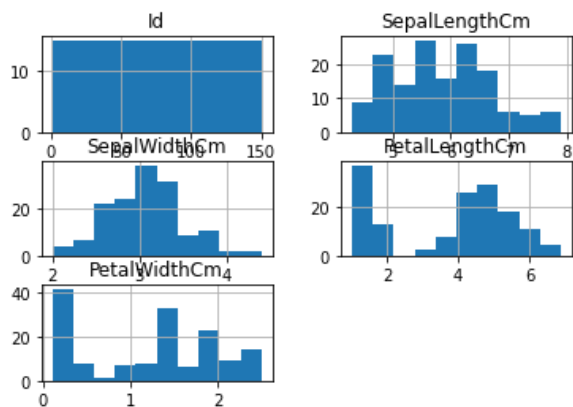
plt.bar(x,y)
```

Out[128... <BarContainer object of 150 artists>



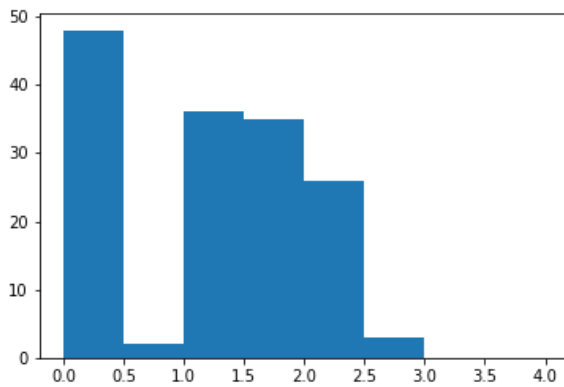
```
In [136... df.hist()
```

Out[136... array([[<AxesSubplot:title={'center':'Id'}>,
<AxesSubplot:title={'center':'SepalLengthCm'}>],
[<AxesSubplot:title={'center':'SepalWidthCm'}>,
<AxesSubplot:title={'center':'PetalLengthCm'}>],
[<AxesSubplot:title={'center':'PetalWidthCm'}>],
<AxesSubplot:~>]],
dtype=object)
<Figure size 1440x1440 with 0 Axes>



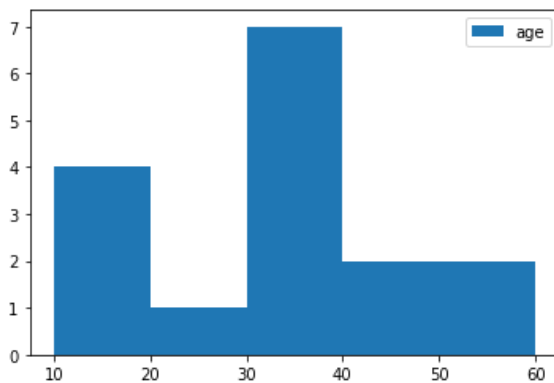
```
In [140... plt.hist(df['PetalWidthCm'],bins=[0,0.5,1,1.5,2,2.5,3,4])
```

Out[140... (array([48., 2., 36., 35., 26., 3., 0.]),
array([0. , 0.5, 1. , 1.5, 2. , 2.5, 3. , 4.]),
<BarContainer object of 7 artists>)



```
In [144... age=[10,10,10,10,20,30,30,30,30,30,30,30,40,40,50,60]
plt.hist(age,bins=[10,20,30,40,50,60],label='age')
plt.legend()
```

```
Out[144... <matplotlib.legend.Legend at 0x288331427c0>
```



```
In [145... #pie-chart
```

```
In [148... df['Species'].unique()
```

```
Out[148... array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'], dtype=object)
```

```
In [158... one=df[df['Species']=='Iris-setosa']
two=df[df['Species']=='Iris-versicolor']
three=df[df['Species']=='Iris-virginica']
```

```
In [159... x=[one,two,three]
x
```

```
Out[159... [  Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm  Species
0    1         5.1           3.5         1.4         0.2  Iris-setosa
1    2         4.9           3.0         1.4         0.2  Iris-setosa
2    3         4.7           3.2         1.3         0.2  Iris-setosa
3    4         4.6           3.1         1.5         0.2  Iris-setosa
4    5         5.0           3.6         1.4         0.2  Iris-setosa
5    6         5.4           3.9         1.7         0.4  Iris-setosa
6    7         4.6           3.4         1.4         0.3  Iris-setosa
7    8         5.0           3.4         1.5         0.2  Iris-setosa
8    9         4.4           2.9         1.4         0.2  Iris-setosa
9   10         4.9           3.1         1.5         0.1  Iris-setosa
10  11         5.4           3.7         1.5         0.2  Iris-setosa
11  12         4.8           3.4         1.6         0.2  Iris-setosa
12  13         4.8           3.0         1.4         0.1  Iris-setosa
13  14         4.3           3.0         1.1         0.1  Iris-setosa
14  15         5.8           4.0         1.2         0.2  Iris-setosa
15  16         5.7           4.4         1.5         0.4  Iris-setosa
16  17         5.4           3.9         1.3         0.4  Iris-setosa
17  18         5.1           3.5         1.4         0.3  Iris-setosa
18  19         5.7           3.8         1.7         0.3  Iris-setosa
19  20         5.1           3.8         1.5         0.3  Iris-setosa
20  21         5.4           3.4         1.7         0.2  Iris-setosa
21  22         5.1           3.7         1.5         0.4  Iris-setosa
22  23         4.6           3.6         1.0         0.2  Iris-setosa
```

23	24	5.1	3.3	1.7	0.5	Iris-setosa
24	25	4.8	3.4	1.9	0.2	Iris-setosa
25	26	5.0	3.0	1.6	0.2	Iris-setosa
26	27	5.0	3.4	1.6	0.4	Iris-setosa
27	28	5.2	3.5	1.5	0.2	Iris-setosa
28	29	5.2	3.4	1.4	0.2	Iris-setosa
29	30	4.7	3.2	1.6	0.2	Iris-setosa
30	31	4.8	3.1	1.6	0.2	Iris-setosa
31	32	5.4	3.4	1.5	0.4	Iris-setosa
32	33	5.2	4.1	1.5	0.1	Iris-setosa
33	34	5.5	4.2	1.4	0.2	Iris-setosa
34	35	4.9	3.1	1.5	0.1	Iris-setosa
35	36	5.0	3.2	1.2	0.2	Iris-setosa
36	37	5.5	3.5	1.3	0.2	Iris-setosa
37	38	4.9	3.1	1.5	0.1	Iris-setosa
38	39	4.4	3.0	1.3	0.2	Iris-setosa
39	40	5.1	3.4	1.5	0.2	Iris-setosa
40	41	5.0	3.5	1.3	0.3	Iris-setosa
41	42	4.5	2.3	1.3	0.3	Iris-setosa
42	43	4.4	3.2	1.3	0.2	Iris-setosa
43	44	5.0	3.5	1.6	0.6	Iris-setosa
44	45	5.1	3.8	1.9	0.4	Iris-setosa
45	46	4.8	3.0	1.4	0.3	Iris-setosa
46	47	5.1	3.8	1.6	0.2	Iris-setosa
47	48	4.6	3.2	1.4	0.2	Iris-setosa
48	49	5.3	3.7	1.5	0.2	Iris-setosa
49	50	5.0	3.3	1.4	0.2	Iris-setosa,
	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	\
50	51	7.0	3.2	4.7	1.4	
51	52	6.4	3.2	4.5	1.5	
52	53	6.9	3.1	4.9	1.5	
53	54	5.5	2.3	4.0	1.3	
54	55	6.5	2.8	4.6	1.5	
55	56	5.7	2.8	4.5	1.3	
56	57	6.3	3.3	4.7	1.6	
57	58	4.9	2.4	3.3	1.0	
58	59	6.6	2.9	4.6	1.3	
59	60	5.2	2.7	3.9	1.4	
60	61	5.0	2.0	3.5	1.0	
61	62	5.9	3.0	4.2	1.5	
62	63	6.0	2.2	4.0	1.0	
63	64	6.1	2.9	4.7	1.4	
64	65	5.6	2.9	3.6	1.3	
65	66	6.7	3.1	4.4	1.4	
66	67	5.6	3.0	4.5	1.5	
67	68	5.8	2.7	4.1	1.0	
68	69	6.2	2.2	4.5	1.5	
69	70	5.6	2.5	3.9	1.1	
70	71	5.9	3.2	4.8	1.8	
71	72	6.1	2.8	4.0	1.3	
72	73	6.3	2.5	4.9	1.5	
73	74	6.1	2.8	4.7	1.2	
74	75	6.4	2.9	4.3	1.3	
75	76	6.6	3.0	4.4	1.4	
76	77	6.8	2.8	4.8	1.4	
77	78	6.7	3.0	5.0	1.7	
78	79	6.0	2.9	4.5	1.5	
79	80	5.7	2.6	3.5	1.0	
80	81	5.5	2.4	3.8	1.1	
81	82	5.5	2.4	3.7	1.0	
82	83	5.8	2.7	3.9	1.2	
83	84	6.0	2.7	5.1	1.6	
84	85	5.4	3.0	4.5	1.5	
85	86	6.0	3.4	4.5	1.6	
86	87	6.7	3.1	4.7	1.5	
87	88	6.3	2.3	4.4	1.3	
88	89	5.6	3.0	4.1	1.3	
89	90	5.5	2.5	4.0	1.3	
90	91	5.5	2.6	4.4	1.2	
91	92	6.1	3.0	4.6	1.4	
92	93	5.8	2.6	4.0	1.2	
93	94	5.0	2.3	3.3	1.0	
94	95	5.6	2.7	4.2	1.3	
95	96	5.7	3.0	4.2	1.2	
96	97	5.7	2.9	4.2	1.3	
97	98	6.2	2.9	4.3	1.3	
98	99	5.1	2.5	3.0	1.1	
99	100	5.7	2.8	4.1	1.3	

Species
50 Iris-versicolor

51	Iris-versicolor				
52	Iris-versicolor				
53	Iris-versicolor				
54	Iris-versicolor				
55	Iris-versicolor				
56	Iris-versicolor				
57	Iris-versicolor				
58	Iris-versicolor				
59	Iris-versicolor				
60	Iris-versicolor				
61	Iris-versicolor				
62	Iris-versicolor				
63	Iris-versicolor				
64	Iris-versicolor				
65	Iris-versicolor				
66	Iris-versicolor				
67	Iris-versicolor				
68	Iris-versicolor				
69	Iris-versicolor				
70	Iris-versicolor				
71	Iris-versicolor				
72	Iris-versicolor				
73	Iris-versicolor				
74	Iris-versicolor				
75	Iris-versicolor				
76	Iris-versicolor				
77	Iris-versicolor				
78	Iris-versicolor				
79	Iris-versicolor				
80	Iris-versicolor				
81	Iris-versicolor				
82	Iris-versicolor				
83	Iris-versicolor				
84	Iris-versicolor				
85	Iris-versicolor				
86	Iris-versicolor				
87	Iris-versicolor				
88	Iris-versicolor				
89	Iris-versicolor				
90	Iris-versicolor				
91	Iris-versicolor				
92	Iris-versicolor				
93	Iris-versicolor				
94	Iris-versicolor				
95	Iris-versicolor				
96	Iris-versicolor				
97	Iris-versicolor				
98	Iris-versicolor				
99	Iris-versicolor	,			
	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm \
100	101	6.3	3.3	6.0	2.5
101	102	5.8	2.7	5.1	1.9
102	103	7.1	3.0	5.9	2.1
103	104	6.3	2.9	5.6	1.8
104	105	6.5	3.0	5.8	2.2
105	106	7.6	3.0	6.6	2.1
106	107	4.9	2.5	4.5	1.7
107	108	7.3	2.9	6.3	1.8
108	109	6.7	2.5	5.8	1.8
109	110	7.2	3.6	6.1	2.5
110	111	6.5	3.2	5.1	2.0
111	112	6.4	2.7	5.3	1.9
112	113	6.8	3.0	5.5	2.1
113	114	5.7	2.5	5.0	2.0
114	115	5.8	2.8	5.1	2.4
115	116	6.4	3.2	5.3	2.3
116	117	6.5	3.0	5.5	1.8
117	118	7.7	3.8	6.7	2.2
118	119	7.7	2.6	6.9	2.3
119	120	6.0	2.2	5.0	1.5
120	121	6.9	3.2	5.7	2.3
121	122	5.6	2.8	4.9	2.0
122	123	7.7	2.8	6.7	2.0
123	124	6.3	2.7	4.9	1.8
124	125	6.7	3.3	5.7	2.1
125	126	7.2	3.2	6.0	1.8
126	127	6.2	2.8	4.8	1.8
127	128	6.1	3.0	4.9	1.8
128	129	6.4	2.8	5.6	2.1
129	130	7.2	3.0	5.8	1.6
130	131	7.4	2.8	6.1	1.9

131	132	7.9	3.8	6.4	2.0
132	133	6.4	2.8	5.6	2.2
133	134	6.3	2.8	5.1	1.5
134	135	6.1	2.6	5.6	1.4
135	136	7.7	3.0	6.1	2.3
136	137	6.3	3.4	5.6	2.4
137	138	6.4	3.1	5.5	1.8
138	139	6.0	3.0	4.8	1.8
139	140	6.9	3.1	5.4	2.1
140	141	6.7	3.1	5.6	2.4
141	142	6.9	3.1	5.1	2.3
142	143	5.8	2.7	5.1	1.9
143	144	6.8	3.2	5.9	2.3
144	145	6.7	3.3	5.7	2.5
145	146	6.7	3.0	5.2	2.3
146	147	6.3	2.5	5.0	1.9
147	148	6.5	3.0	5.2	2.0
148	149	6.2	3.4	5.4	2.3
149	150	5.9	3.0	5.1	1.8

```

Species
100 Iris-virginica
101 Iris-virginica
102 Iris-virginica
103 Iris-virginica
104 Iris-virginica
105 Iris-virginica
106 Iris-virginica
107 Iris-virginica
108 Iris-virginica
109 Iris-virginica
110 Iris-virginica
111 Iris-virginica
112 Iris-virginica
113 Iris-virginica
114 Iris-virginica
115 Iris-virginica
116 Iris-virginica
117 Iris-virginica
118 Iris-virginica
119 Iris-virginica
120 Iris-virginica
121 Iris-virginica
122 Iris-virginica
123 Iris-virginica
124 Iris-virginica
125 Iris-virginica
126 Iris-virginica
127 Iris-virginica
128 Iris-virginica
129 Iris-virginica
130 Iris-virginica
131 Iris-virginica
132 Iris-virginica
133 Iris-virginica
134 Iris-virginica
135 Iris-virginica
136 Iris-virginica
137 Iris-virginica
138 Iris-virginica
139 Iris-virginica
140 Iris-virginica
141 Iris-virginica
142 Iris-virginica
143 Iris-virginica
144 Iris-virginica
145 Iris-virginica
146 Iris-virginica
147 Iris-virginica
148 Iris-virginica
149 Iris-virginica ]

```

In [160...

```
plt.pie(x)
```

```
-----
ValueError                                Traceback (most recent call last)
```

```
~\AppData\Local\Temp\ipykernel_4900\3416303025.py in <module>
```

```
----> 1 plt.pie(x)
```

```
~\anaconda3\lib\site-packages\matplotlib\pyplot.py in pie(x, explode, labels, colors, autopct, pctdistance, sha
```

```

dow, labeldistance, startangle, radius, counterclock, wedgeprops, textprops, center, frame, rotatelabels, normalize, data)
3004     textprops=None, center=(0, 0), frame=False,
3005     rotatelabels=False, *, normalize=None, data=None):
-> 3006     return gca().pie(
3007         x, explode=explode, labels=labels, colors=colors,
3008         autopct=autopct, pctdistance=pctdistance, shadow=shadow,

~\anaconda3\lib\site-packages\matplotlib\__init__.py in inner(ax, data, *args, **kwargs)
1359     def inner(ax, *args, data=None, **kwargs):
1360         if data is None:
-> 1361             return func(ax, *map(sanitize_sequence, args), **kwargs)
1362
1363         bound = new_sig.bind(ax, *args, **kwargs)

~\anaconda3\lib\site-packages\matplotlib\axes\_axes.py in pie(self, x, explode, labels, colors, autopct, pctdistance, shadow, labeldistance, startangle, radius, counterclock, wedgeprops, textprops, center, frame, rotatelabels, normalize)
3028     # The use of float32 is "historical", but can't be changed without
3029     # regenerating the test baselines.
-> 3030     x = np.asarray(x, np.float32)
3031     if x.ndim > 1:
3032         raise ValueError("x must be 1D")

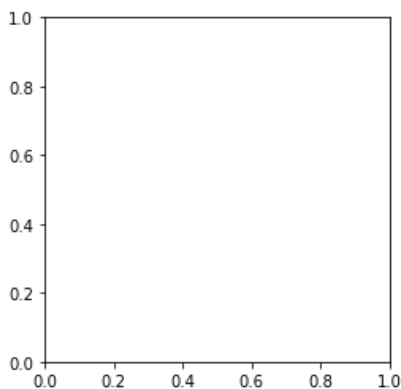
~\anaconda3\lib\site-packages\numpy\core\_asarray.py in asarray(a, dtype, order, like)
100     return _asarray_with_like(a, dtype=dtype, order=order, like=like)
101
--> 102     return array(a, dtype, copy=False, order=order)
103
104

~\anaconda3\lib\site-packages\pandas\core\generic.py in __array__(self, dtype)
1991
1992     def __array__(self, dtype: NpDtype | None = None) -> np.ndarray:
-> 1993     return np.asarray(self._values, dtype=dtype)
1994
1995     def __array_wrap__(

~\anaconda3\lib\site-packages\numpy\core\_asarray.py in asarray(a, dtype, order, like)
100     return _asarray_with_like(a, dtype=dtype, order=order, like=like)
101
--> 102     return array(a, dtype, copy=False, order=order)
103
104

```

ValueError: could not convert string to float: 'Iris-setosa'



In [154...

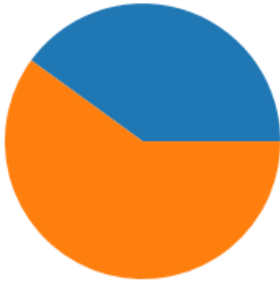
```
plt.pie([20,30])
```

Out[154...

```

(<matplotlib.patches.Wedge at 0x28833233100>,
 <matplotlib.patches.Wedge at 0x2883096f940>],
 [Text(0.33991867422268784, 1.0461621742897658, ''),
 Text(-0.3399185762739153, -1.046162206115244, '')])

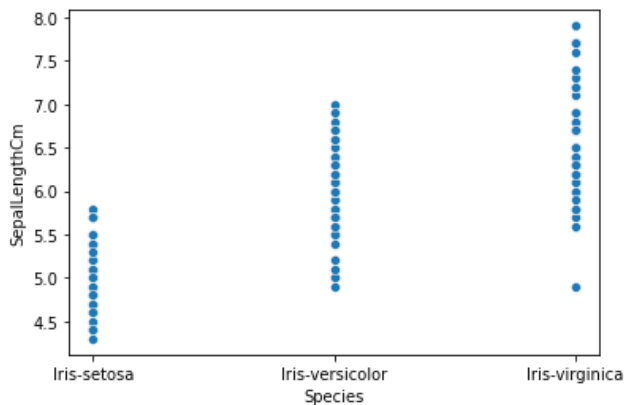
```



```
In [161...] import seaborn as sns
```

```
In [162...] sns.scatterplot(x='Species',y='SepallLengthCm',data=df)
```

```
Out[162...] <AxesSubplot:xlabel='Species', ylabel='SepallLengthCm'>
```



```
In [164...] sns.get_dataset_names()
```

```
Out[164...] ['anagrams',
'anscombe',
'attention',
'brain_networks',
'car_crashes',
'diamonds',
'dots',
'exercise',
'flights',
'fmri',
'gammas',
'geyser',
'iris',
'mpg',
'penguins',
'planets',
'taxis',
'tips',
'titanic']
```

```
In [165...] df=sns.load_dataset('titanic')
```

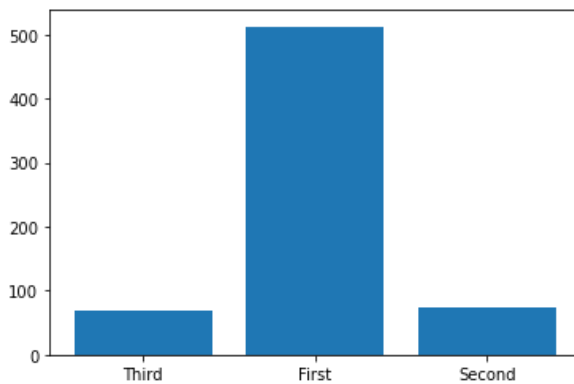
```
In [166...] df.head()
```

```
Out[166...]   survived  pclass   sex  age  sibsp  parch   fare  embarked  class  who  adult_male  deck  embark_town  alive  alone
0         0        3  male  22.0     1     0   7.2500         S  Third   man         True   NaN  Southampton    no    False
1         1        1 female  38.0     1     0  71.2833         C   First  woman        False    C    Cherbourg   yes    False
2         1        3 female  26.0     0     0   7.9250         S  Third  woman        False   NaN  Southampton   yes     True
3         1        1 female  35.0     1     0  53.1000         S   First  woman        False    C    Southampton   yes    False
4         0        3  male  35.0     0     0   8.0500         S  Third   man         True   NaN  Southampton    no     True
```

```
In [167... x=df['class']
y=df['fare']

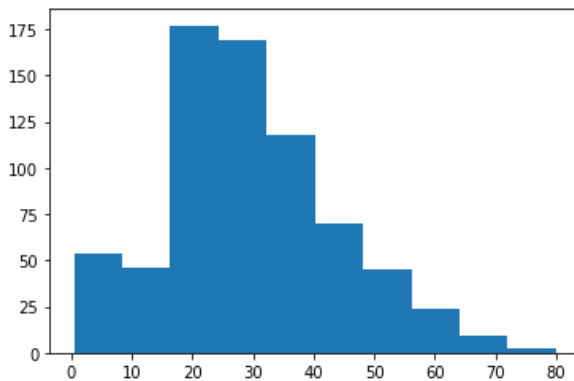
plt.bar(x,y)
```

Out[167... <BarContainer object of 891 artists>



```
In [168... plt.hist(df['age'])
```

Out[168... (array([54., 46., 177., 169., 118., 70., 45., 24., 9., 2.]),
array([0.42 , 8.378, 16.336, 24.294, 32.252, 40.21 , 48.168, 56.126,
64.084, 72.042, 80.]),
<BarContainer object of 10 artists>)



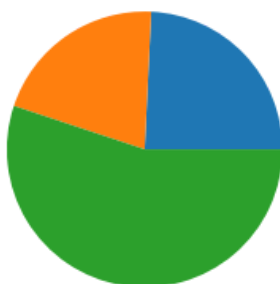
```
In [174... first=df[df['class']=='First'].count()[0].sum()
second=df[df['class']=='Second'].count()[0].sum()
Third=df[df['class']=='Third'].count()[0].sum()

x=[first,second,Third]
x
```

Out[174... [216, 184, 491]

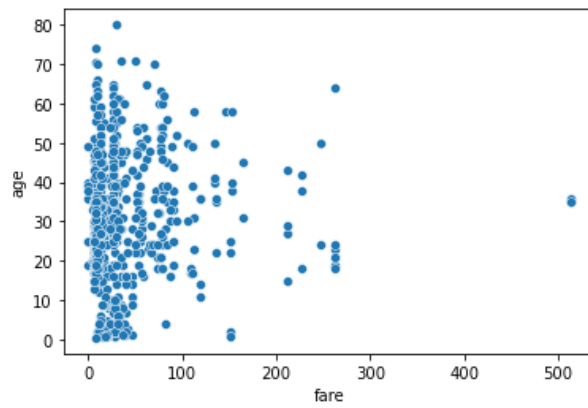
```
In [175... plt.pie(x)
```

Out[175... (<matplotlib.patches.Wedge at 0x2883638f550>,
<matplotlib.patches.Wedge at 0x2883638fc40>,
<matplotlib.patches.Wedge at 0x288363901c0>],
[Text(0.796107424686281, 0.7590869306998885, ''),
Text(-0.6221676201813416, 0.9071424653260843, ''),
Text(-0.175716065426671, -1.0858747001155198, '')])



```
In [177... sns.scatterplot(x='fare',y='age',data=df)
```

```
Out[177... <AxesSubplot:xlabel='fare', ylabel='age'>
```



```
In [ ]:
```