```
In [1]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
```

```
In [93]: data=pd.read_csv(r"C:\Users\user\Desktop\Vicky\3 Fitness-1.csv")
```

```
In [94]: data.head()
```

Out[94]:

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
0	A	5.62%	7.73%	6.16%	75
1	B	4.21%	17.27%	19.21%	160
2	C	9.83%	11.60%	5.17%	101
3	D	2.81%	21.91%	7.88%	127
4	E	25.28%	10.57%	11.82%	179

```
In [95]: data.tail()
```

Out[95]:

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
4	E	25.28%	10.57%	11.82%	179
5	F	8.15%	16.24%	18.47%	167
6	G	18.54%	8.76%	17.49%	171
7	H	25.56%	5.93%	13.79%	170
8	Grand Total	100.00%	100.00%	100.00%	1150

```
In [96]: data.isna()
```

Out[96]:

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
5	False	False	False	False	False
6	False	False	False	False	False
7	False	False	False	False	False
8	False	False	False	False	False

In [109]: `data.fillna(value=5)`

Out[109]:

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
0	A	5.62%	7.73%	6.16%	75
1	B	4.21%	17.27%	19.21%	160
2	C	9.83%	11.60%	5.17%	101
3	D	2.81%	21.91%	7.88%	127
4	E	25.28%	10.57%	11.82%	179
5	F	8.15%	16.24%	18.47%	167
6	G	18.54%	8.76%	17.49%	171
7	H	25.56%	5.93%	13.79%	170
8	Grand Total	100.00%	100.00%	100.00%	1150

In [108]: `data.shape`

Out[108]: `(9, 5)`

In [107]: `data.size`

Out[107]: 45

In [106]: `pd.isna(data)`

Out[106]:

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
5	False	False	False	False	False
6	False	False	False	False	False
7	False	False	False	False	False
8	False	False	False	False	False

In [105]: `data.dropna()`

Out[105]:

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
0	A	5.62%	7.73%	6.16%	75
1	B	4.21%	17.27%	19.21%	160
2	C	9.83%	11.60%	5.17%	101
3	D	2.81%	21.91%	7.88%	127
4	E	25.28%	10.57%	11.82%	179
5	F	8.15%	16.24%	18.47%	167
6	G	18.54%	8.76%	17.49%	171
7	H	25.56%	5.93%	13.79%	170
8	Grand Total	100.00%	100.00%	100.00%	1150

In [104]: `data.describe()`

Out[104]:

	Sum of Total Sales
count	9.000000
mean	255.555556
std	337.332963
min	75.000000
25%	127.000000
50%	167.000000
75%	171.000000
max	1150.000000

In [99]:
```python
x="Sum of Jan"
y="Sum of Feb"
x1="Sum of Mar"
y1="Sum of Total Sales"
plt.scatter(x,y,label="JanandFeb",color="r")
plt.scatter(x1,y1,label="MAr",color="b")

plt.title("Fitness")
plt.xlabel("JAN")
plt.ylabel("Feb")
plt.legend()
plt.show
```
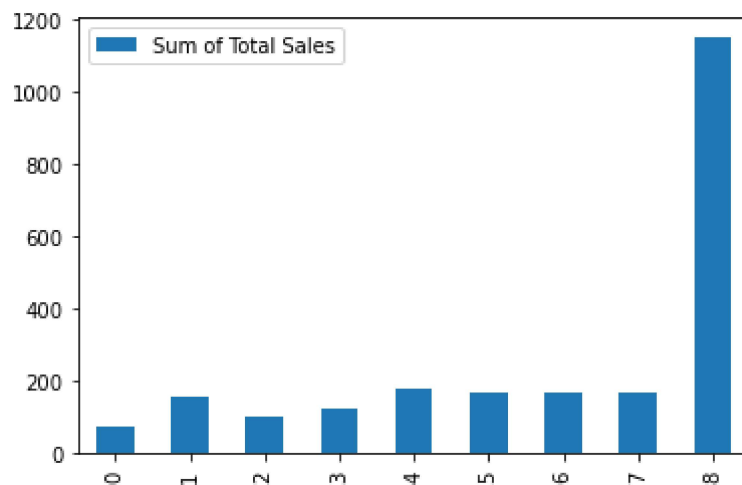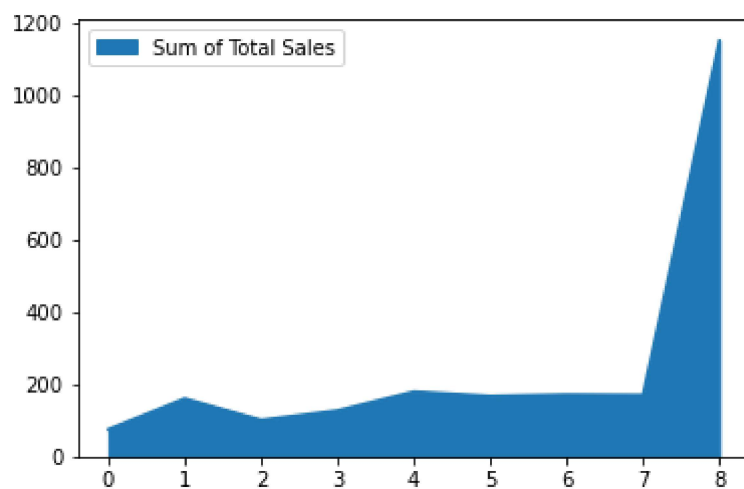
Out[99]: <function matplotlib.pyplot.show(close=None, block=None)>
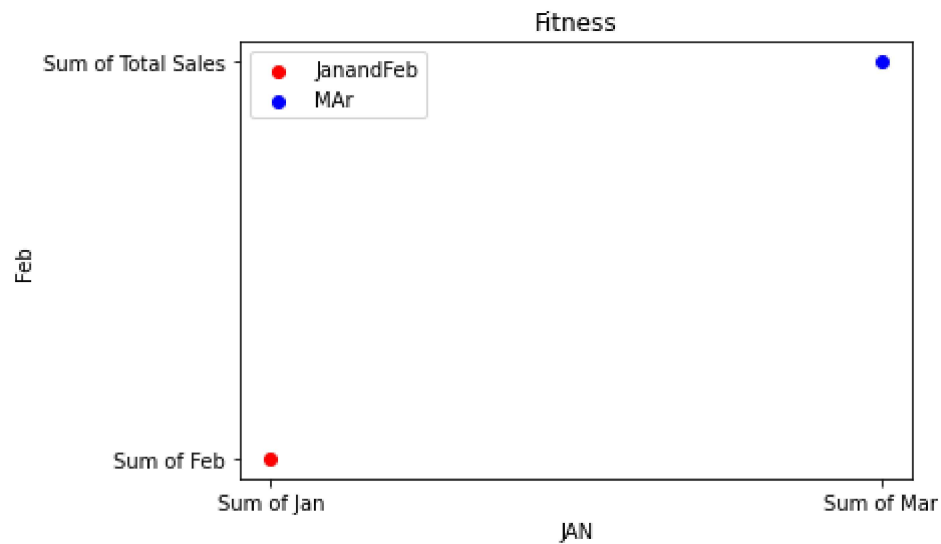
In [101]:
```python
data.plot.bar()
```

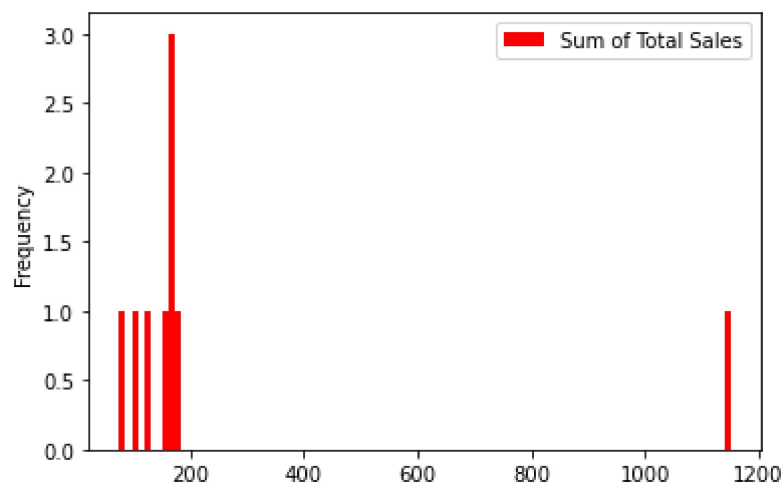Out[101]: <AxesSubplot:>

In [102]: `data.plot.area()`

Out[102]: `<AxesSubplot:>`

In [103]: `data.plot.hist(bins=100,color="r")`

Out[103]: `<AxesSubplot:ylabel='Frequency'>`

In [40]:
```python
data.plot.box()
```

Out[40]: <AxesSubplot:>
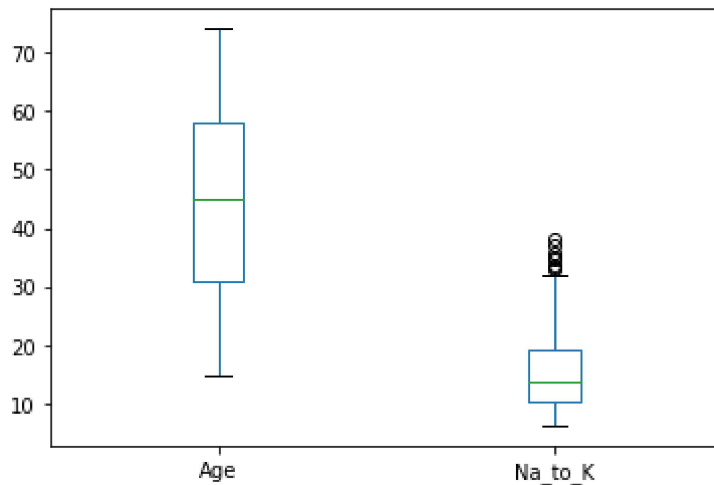
In [ ]:
```python
#cols=["r","b"]
#data.plot.pie(y="Row_Labels",shadow=True,startangle=90,colors=cols)
```

In [115]:
```python
data.plot.scatter(x="Sum of Feb",y="Sum of Jan")


plt.title("Drugs")
plt.legend()
```

No handles with labels found to put in legend.

Out[115]: <matplotlib.legend.Legend at 0x2447d42f7c0>
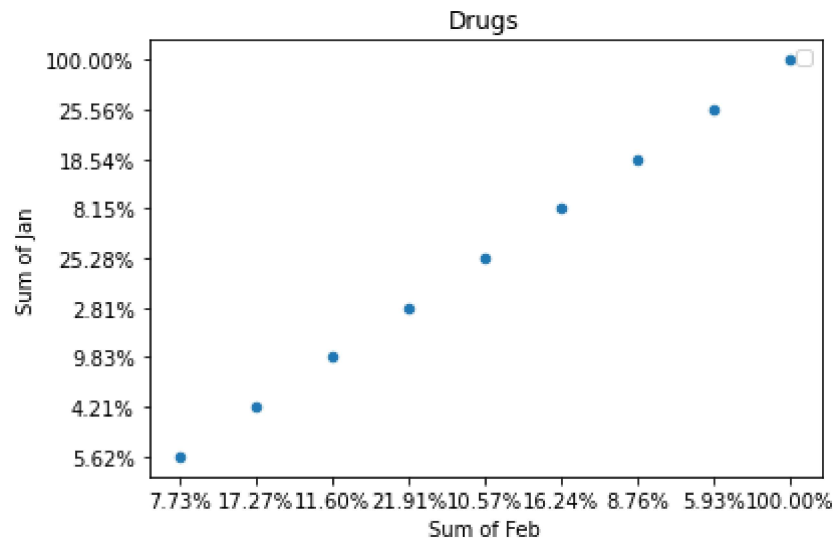
In [116]:
```python
from numpy import linalg as la
```

In [117]:
```python
data.mean()
```

Out[117]:
```
Sum of Total Sales      255.555556
dtype: float64
```

In [118]: `data.median()`

Out[118]: Sum of Total Sales   167.0
          dtype: float64

In [119]: `data.mode()`

Out[119]:

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
0	A	100.00%	10.57%	100.00%	75
1	B	18.54%	100.00%	11.82%	101
2	C	2.81%	11.60%	13.79%	127
3	D	25.28%	16.24%	17.49%	160
4	E	25.56%	17.27%	18.47%	167
5	F	4.21%	21.91%	19.21%	170
6	G	5.62%	5.93%	5.17%	171
7	Grand Total	8.15%	7.73%	6.16%	179
8	H	9.83%	8.76%	7.88%	1150

In [120]: `data.std()`

Out[120]: Sum of Total Sales   337.332963
          dtype: float64

In [121]: `data.var()`

Out[121]: Sum of Total Sales   113793.527778
          dtype: float64

In [122]: `data.max()`

Out[122]: Row Labels   H
          Sum of Jan   9.83%
          Sum of Feb   8.76%
          Sum of Mar   7.88%
          Sum of Total Sales   1150
          dtype: object

In [123]: `data.min()`

Out[123]: Row Labels   A
          Sum of Jan   100.00%
          Sum of Feb   10.57%
          Sum of Mar   100.00%
          Sum of Total Sales   75
          dtype: object

In [125]:
```python
data1=data[["Sum of Jan","Sum of Feb"]]
data1
```

Out[125]:

	Sum of Jan	Sum of Feb
0	5.62%	7.73%
1	4.21%	17.27%
2	9.83%	11.60%
3	2.81%	21.91%
4	25.28%	10.57%
5	8.15%	16.24%
6	18.54%	8.76%
7	25.56%	5.93%
8	100.00%	100.00%

In [126]:
```python
data1.sum()
```

Out[126]:
```
Sum of Jan    5.62%4.21%9.83%2.81%25.28%8.15%18.54%25.56%100...
Sum of Feb    7.73%17.27%11.60%21.91%10.57%16.24%8.76%5.93%1...
dtype: object
```

In [127]:
```python
data.cumsum()
```

Out[127]:

	Row Labels	Sum of Jan	
0	A	5.62%	
1	AB	5.62%4.21%	
2	ABC	5.62%4.21%9.83%	
3	ABCD	5.62%4.21%9.83%2.81%	
4	ABCDE	5.62%4.21%9.83%2.81%25.28%	7
5	ABCDEF	5.62%4.21%9.83%2.81%25.28%8.15%	7.73%17
6	ABCDEFG	5.62%4.21%9.83%2.81%25.28%8.15%18.54%	7.73%17.27%1
7	ABCDEFGH	5.62%4.21%9.83%2.81%25.28%8.15%18.54%25.56%	7.73%17.27%11.60%
8	ABCDEFGHGrand Total	5.62%4.21%9.83%2.81%25.28%8.15%18.54%25.56%100...	7.73%17.27%11.60%21.

In [128]: `data1.cumsum()`

Out[128]:

	Sum of Jan	
0	5.62%	
1	5.62%4.21%	
2	5.62%4.21%9.83%	7.73%
3	5.62%4.21%9.83%2.81%	7.73%17.27%
4	5.62%4.21%9.83%2.81%25.28%	7.73%17.27%11.60%
5	5.62%4.21%9.83%2.81%25.28%8.15%	7.73%17.27%11.60%21.91%
6	5.62%4.21%9.83%2.81%25.28%8.15%18.54%	7.73%17.27%11.60%21.91%10.57%
7	5.62%4.21%9.83%2.81%25.28%8.15%18.54%25.56%	7.73%17.27%11.60%21.91%10.57%16.24
8	5.62%4.21%9.83%2.81%25.28%8.15%18.54%25.56%100...	7.73%17.27%11.60%21.91%10.57%16.24%8

In [129]:
```python
from scipy.stats import spearmanr
from scipy.stats import pearsonr
```

In [72]:
```python
print(spearmanr(data,data1))
```

```
SpearmanrResult(correlation=array([[ 1.       ,  0.10446701,  0.06283512, -
0.0688114 , -0.04727388,
        -0.02040842,  1.       , -0.04727388],
       [ 0.10446701,  1.       , -0.00552946, -0.00881101, -0.14587075,
        -0.09109896,  0.10446701, -0.14587075],
       [ 0.06283512, -0.00552946,  1.       , -0.1354242 , -0.15091871,
         0.1910008 ,  0.06283512, -0.15091871],
       [-0.0688114 , -0.00881101, -0.1354242 ,  1.       , -0.01282312,
         0.04802219, -0.0688114 , -0.01282312],
       [-0.04727388, -0.14587075, -0.15091871, -0.01282312,  1.       ,
         0.775643  , -0.04727388,  1.       ],
       [-0.02040842, -0.09109896,  0.1910008 ,  0.04802219,  0.775643  ,
         1.       , -0.02040842,  0.775643  ],
       [ 1.       ,  0.10446701,  0.06283512, -0.0688114 , -0.04727388,
        -0.02040842,  1.       , -0.04727388],
       [-0.04727388, -0.14587075, -0.15091871, -0.01282312,  1.       ,
         0.775643  , -0.04727388,  1.       ]]), pvalue=array([[0.00000000e+
00, 1.40978395e-01, 3.76735771e-01, 3.32950156e-01,
        5.06220058e-01, 7.74235282e-01, 0.00000000e+00, 5.06220058e-01],
       [1.40978395e-01, 0.00000000e+00, 9.38059728e-01, 9.01451523e-01,
        3.93001141e-02, 1.99517379e-01, 1.40978395e-01, 3.93001141e-02],
       [3.76735771e-01, 9.38059728e-01, 0.00000000e+00, 5.58761447e-02,
        3.29109301e-02, 6.74424545e-03, 3.76735771e-01, 3.29109301e-02],
       [3.32950156e-01, 9.01451523e-01, 5.58761447e-02, 0.00000000e+00,
        8.56982221e-01, 4.99504731e-01, 3.32950156e-01, 8.56982221e-01],
       [5.06220058e-01, 3.93001141e-02, 3.29109301e-02, 8.56982221e-01,
        0.00000000e+00, 1.95525864e-41, 5.06220058e-01, 0.00000000e+00],
       [7.74235282e-01, 1.99517379e-01, 6.74424545e-03, 4.99504731e-01,
        1.95525864e-41, 0.00000000e+00, 7.74235282e-01, 1.95525864e-41],
       [0.00000000e+00, 1.40978395e-01, 3.76735771e-01, 3.32950156e-01,
        5.06220058e-01, 7.74235282e-01, 0.00000000e+00, 5.06220058e-01],
       [5.06220058e-01, 3.93001141e-02, 3.29109301e-02, 8.56982221e-01,
        0.00000000e+00, 1.95525864e-41, 5.06220058e-01, 0.00000000e+00]]))
```

In [75]:
```python
from numpy import mean,std,cov
from numpy.random import randn,seed
from matplotlib import pyplot
```

```
In [76]: print(cov(data,data1))
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-76-81961316810f> in <module>
----> 1 print(cov(data,data1))

<__array_function__ internals> in cov(*args, **kwargs)

C:\ProgramData\Anaconda3\lib\site-packages\numpy\lib\function_base.py in cov
(m, y, rowvar, bias, ddof, fweights, aweights, dtype)
   2426            if not rowvar and y.shape[0] != 1:
   2427                y = y.T
-> 2428            X = np.concatenate((X, y), axis=0)
   2429
   2430        if ddof is None:

<__array_function__ internals> in concatenate(*args, **kwargs)

ValueError: all the input array dimensions for the concatenation axis must ma
tch exactly, but along dimension 1, the array at index 0 has size 6 and the a
rray at index 1 has size 2
```

```
In [130]: data.fillna(value=5)
```

Out[130]:

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
0	A	5.62%	7.73%	6.16%	75
1	B	4.21%	17.27%	19.21%	160
2	C	9.83%	11.60%	5.17%	101
3	D	2.81%	21.91%	7.88%	127
4	E	25.28%	10.57%	11.82%	179
5	F	8.15%	16.24%	18.47%	167
6	G	18.54%	8.76%	17.49%	171
7	H	25.56%	5.93%	13.79%	170
8	Grand Total	100.00%	100.00%	100.00%	1150

```
In [80]: data.count()
```

```
Out[80]: Age           200
         Sex           200
         BP            200
         Cholesterol   200
         Na_to_K       200
         Drug          200
         dtype: int64
```

In [81]: `data.count`

Out[81]:
```
<bound method DataFrame.count of       Age Sex      BP Cholesterol   Na_to_K
Drug
0      23   F    HIGH       HIGH    25.355   drugY
1      47   M     LOW       HIGH    13.093   drugC
2      47   M     LOW       HIGH    10.114   drugC
3      28   F  NORMAL       HIGH     7.798   drugX
4      61   F     LOW       HIGH    18.043   drugY
..    ...  ..     ...        ...       ...     ...
195    56   F     LOW       HIGH    11.567   drugC
196    16   M     LOW       HIGH    12.006   drugC
197    52   M  NORMAL       HIGH     9.894   drugX
198    23   M  NORMAL     NORMAL    14.020   drugX
199    40   F     LOW     NORMAL    11.349   drugX

[200 rows x 6 columns]>
```

In [131]: `data.columns`

Out[131]:
```
Index(['Row Labels', 'Sum of Jan', 'Sum of Feb', 'Sum of Mar',
       'Sum of Total Sales'],
      dtype='object')
```

In [132]: `data.iloc[3]`

Out[132]:
```
Row Labels              D
Sum of Jan          2.81%
Sum of Feb         21.91%
Sum of Mar          7.88%
Sum of Total Sales    127
Name: 3, dtype: object
```

In [133]: `data.dropna(axis=1,how="any")`

Out[133]:

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
0	A	5.62%	7.73%	6.16%	75
1	B	4.21%	17.27%	19.21%	160
2	C	9.83%	11.60%	5.17%	101
3	D	2.81%	21.91%	7.88%	127
4	E	25.28%	10.57%	11.82%	179
5	F	8.15%	16.24%	18.47%	167
6	G	18.54%	8.76%	17.49%	171
7	H	25.56%	5.93%	13.79%	170
8	Grand Total	100.00%	100.00%	100.00%	1150

In [134]: `data.index`

Out[134]: `RangeIndex(start=0, stop=9, step=1)`

In [ ]: