In [1]:

```
import pandas as pd
import numpy as np
```

DataSet1

In [2]:

```
df=pd.read_csv(r'C:\Users\user\Downloads\2015.csv')
```

In [3]:

```
print(df.mean())
```

```
Happiness Rank                   79.493671
Happiness Score                   5.375734
Standard Error                    0.047885
Economy (GDP per Capita)          0.846137
Family                            0.991046
Health (Life Expectancy)          0.630259
Freedom                           0.428615
Trust (Government Corruption)     0.143422
Generosity                        0.236917
Dystopia Residual                 2.098977
dtype: float64
```

In [4]:

```
print(df.median())
```

```
Happiness Rank                   79.500000
Happiness Score                   5.232500
Standard Error                    0.043940
Economy (GDP per Capita)          0.910245
Family                            1.029510
Health (Life Expectancy)          0.696705
Freedom                           0.435515
Trust (Government Corruption)     0.107220
Generosity                        0.215420
Dystopia Residual                 2.095415
dtype: float64
```

In [5]:

```
print(df.mode())
```

```
         Country              Region  Happiness Rank  Happiness Score  \
0    Afghanistan  Sub-Saharan Africa            82.0            5.192
1        Albania                 NaN             NaN              NaN
2        Algeria                 NaN             NaN              NaN
3         Angola                 NaN             NaN              NaN
4      Argentina                 NaN             NaN              NaN
..           ...                 ...             ...              ...
153    Venezuela                 NaN             NaN              NaN
154      Vietnam                 NaN             NaN              NaN
155        Yemen                 NaN             NaN              NaN
156       Zambia                 NaN             NaN              NaN
157     Zimbabwe                 NaN             NaN              NaN

     Standard Error  Economy (GDP per Capita)   Family  \
0           0.03751                   0.00000  0.00000
1           0.03780                   0.01530  0.13995
2           0.04394                   0.01604  0.30285
3           0.04934                   0.06940  0.35386
4           0.05051                   0.07120  0.38174
..              ...                       ...      ...
153             NaN                   1.45900  1.34043
154             NaN                   1.52186  1.34951
155             NaN                   1.55422  1.36058
156             NaN                   1.56391  1.36948
157             NaN                   1.69042  1.40223

     Health (Life Expectancy)  Freedom  Trust (Government Corruption)  \
0                     0.92356  0.00000                        0.32524
1                         NaN  0.07699                            NaN
2                         NaN  0.09245                            NaN
3                         NaN  0.10081                            NaN
4                         NaN  0.10384                            NaN
..                        ...      ...                            ...
153                       NaN  0.65821                            NaN
154                       NaN  0.65980                            NaN
155                       NaN  0.66246                            NaN
156                       NaN  0.66557                            NaN
157                       NaN  0.66973                            NaN

     Generosity  Dystopia Residual
0       0.00000            0.32858
1       0.00199            0.65429
2       0.02641            0.67042
3       0.05444            0.67108
4       0.05547            0.89991
..          ...                ...
153     0.51752            3.10712
154     0.51912            3.17728
155     0.57630            3.19131
156     0.79588            3.26001
157         NaN            3.60214

[158 rows x 12 columns]
```

In [6]:

```
df.describe()
```

Out[6]:

	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom
count	158.000000	158.000000	158.000000	158.000000	158.000000	158.000000	158.000000
mean	79.493671	5.375734	0.047885	0.846137	0.991046	0.630259	0.428615
std	45.754363	1.145010	0.017146	0.403121	0.272369	0.247078	0.150693
min	1.000000	2.839000	0.018480	0.000000	0.000000	0.000000	0.000000
25%	40.250000	4.526000	0.037268	0.545808	0.856823	0.439185	0.328330
50%	79.500000	5.232500	0.043940	0.910245	1.029510	0.696705	0.435515
75%	118.750000	6.243750	0.052300	1.158448	1.214405	0.811013	0.549092
max	158.000000	7.587000	0.136930	1.690420	1.402230	1.025250	0.669730

In [7]:

```
print(df.sum())
```

```
Country                          SwitzerlandIcelandDenmarkNorwayCanadaFinl
andNe...
Region                           Western EuropeWestern EuropeWestern Europ
eWest...
Happiness Rank
12560
Happiness Score
849.366
Standard Error
7.56579
Economy (GDP per Capita)                                                 1
33.68968
Family                                                                   1
56.58526
Health (Life Expectancy)
99.58098
Freedom
67.72116
Trust (Government Corruption)
22.66065
Generosity
37.19591
Dystopia Residual                                                        3
31.63833
dtype: object
```

In [8]:

```
print(df.cumsum())
```

```
                                          Country  \
0                                     Switzerland   
1                              SwitzerlandIceland   
2                       SwitzerlandIcelandDenmark   
3                 SwitzerlandIcelandDenmarkNorway   
4           SwitzerlandIcelandDenmarkNorwayCanada   
..                                            ...   
153  SwitzerlandIcelandDenmarkNorwayCanadaFinlandNe...   
154  SwitzerlandIcelandDenmarkNorwayCanadaFinlandNe...   
155  SwitzerlandIcelandDenmarkNorwayCanadaFinlandNe...   
156  SwitzerlandIcelandDenmarkNorwayCanadaFinlandNe...   
157  SwitzerlandIcelandDenmarkNorwayCanadaFinlandNe...   

                                                Region  Happiness Rank  \
0                                       Western Europe               1   
1                         Western EuropeWestern Europe               3   
2           Western EuropeWestern EuropeWestern Europe               6   
3    Western EuropeWestern EuropeWestern EuropeWest...              10   
4    Western EuropeWestern EuropeWestern EuropeWest...              15   
..                                                 ...             ...   
153  Western EuropeWestern EuropeWestern EuropeWest...           11934   
154  Western EuropeWestern EuropeWestern EuropeWest...           12089   
155  Western EuropeWestern EuropeWestern EuropeWest...           12245   
156  Western EuropeWestern EuropeWestern EuropeWest...           12402   
157  Western EuropeWestern EuropeWestern EuropeWest...           12560   

     Happiness Score  Standard Error  Economy (GDP per Capita)     Family  
\
0              7.587         0.03411                   1.39651    1.34951   
1             15.148         0.08295                   2.69883    2.75174   
2             22.675         0.11623                   4.02431    4.11232   
3             30.197         0.15503                   5.48331    5.44327   
4             37.624         0.19056                   6.80960    6.76588   
..               ...             ...                       ...        ...   
153          837.276         7.32523                 132.51585  155.20069   
154          840.616         7.36179                 132.80250  155.55455   
155          843.622         7.41194                 133.46570  156.02944   
156          846.527         7.49852                 133.48100  156.44531   
157          849.366         7.56579                 133.68968  156.58526   

     Health (Life Expectancy)   Freedom  Trust (Government Corruption)  \
0                     0.94143   0.66557                        0.41978   
1                     1.88927   1.29434                        0.56123   
2                     2.76391   1.94372                        1.04480   
3                     3.64912   2.61345                        1.40983   
4                     4.55475   3.24642                        1.73940   
..                        ...       ...                            ...   
153                  98.03156  66.59679                       22.18356   
154                  98.35066  67.08129                       22.26366   
155                  99.07259  67.23813                       22.45272   
156                  99.29655  67.35663                       22.55334   
157                  99.58098  67.72116                       22.66065   

     Generosity  Dystopia Residual  
0           NaN            2.51738  
1       0.43630            5.21939  
2       0.77769            7.71143  
3       1.12468           10.17674  
4       1.58279           12.62850  
..          ...                ...  
153    36.17744          326.27619  
```

```
154     36.36004           327.90947
155     36.83183           328.23805
156     37.02910           330.07107
157     37.19591           331.63833

[158 rows x 12 columns]
```

In [9]:

```
print(df.count())
```

```
Country                          158
Region                           158
Happiness Rank                   158
Happiness Score                  158
Standard Error                   158
Economy (GDP per Capita)         158
Family                           158
Health (Life Expectancy)         158
Freedom                          158
Trust (Government Corruption)    158
Generosity                       157
Dystopia Residual                158
dtype: int64
```

In [10]:

```
df.min()
```

Out[10]:

```
Country                                      Afghanistan
Region                           Australia and New Zealand
Happiness Rank                                           1
Happiness Score                                      2.839
Standard Error                                     0.01848
Economy (GDP per Capita)                               0.0
Family                                                 0.0
Health (Life Expectancy)                               0.0
Freedom                                                0.0
Trust (Government Corruption)                          0.0
Generosity                                             0.0
Dystopia Residual                                  0.32858
dtype: object
```

In [11]:

```
df.max()
```

Out[11]:

```
Country                                  Zimbabwe
Region                             Western Europe
Happiness Rank                                158
Happiness Score                             7.587
Standard Error                            0.13693
Economy (GDP per Capita)                  1.69042
Family                                    1.40223
Health (Life Expectancy)                  1.02525
Freedom                                   0.66973
Trust (Government Corruption)             0.55191
Generosity                                0.79588
Dystopia Residual                         3.60214
dtype: object
```

In [12]:

```
from numpy import cov
from scipy.stats import spearmanr
from scipy.stats import pearsonr
```

In [13]:

```
cov(df['Family'],df['Freedom'])
```

Out[13]:

```
array([[0.07418492, 0.0181217 ],
       [0.0181217 , 0.02270832]])
```

In [14]:

```
spearmanr(df['Family'],df['Freedom'])
```

Out[14]:

```
SpearmanrResult(correlation=0.5281391142435108, pvalue=9.937786974199143e-
13)
```

In [15]:

```
pearsonr(df['Family'],df['Freedom'])
```

Out[15]:

```
(0.44151821062286056, 6.363670360267173e-09)
```

DataSet2

In [16]:

```
import pandas as pd
import numpy as np
```

In [17]:

```
df2=pd.read_csv(r'C:\Users\user\Downloads\4_drug200.csv')
```

In [18]:

```
df2.mean()
```

Out[18]:

```
Age      44.315000
Value    16.084485
dtype: float64
```

In [19]:

```
df2.median()
```

Out[19]:

```
Age      45.0000
Value    13.9365
dtype: float64
```

In [20]:

```
df2.mode()
```

Out[20]:

	Age	Sex	BP	Cholesterol	Value	Drug
0	47.0	M	HIGH	HIGH	12.006	drugY
1	NaN	NaN	NaN	NaN	18.295	NaN

In [21]:

```
df2.describe()
```

Out[21]:

	Age	Value
count	200.000000	200.000000
mean	44.315000	16.084485
std	16.544315	7.223956
min	15.000000	6.269000
25%	31.000000	10.445500
50%	45.000000	13.936500
75%	58.000000	19.380000
max	74.000000	38.247000

In [22]:

```
df2.sum()
```

Out[22]:

```
Age                                                          8863
Sex            FMMFFFFMMMFFMFFFMMMFMMMFFFMFFMFMMFMMMMFMFFMMFF...
BP             HIGHLOWLOWNORMALLOWNORMALNORMALLOWNORMALLOWLOW...
Cholesterol    HIGHHIGHHIGHHIGHHIGHHIGHHIGHHIGHHIGHNORMALHIGH...
Value                                                    3216.897
Drug           drugYdrugCdrugCdrugXdrugYdrugXdrugYdrugCdrugYd...
dtype: object
```

In [23]:

```
df2.cumsum()
```

Out[23]:

	Age	Sex	
0	23	F	
1	70	FM	
2	117	FMM	
3	145	FMMF	
4	206	FMMFF	
...	...	...	
195	8732	FMMFFFFMMMFFMFFFMMMFMMMFFFMFFMFMMFMMMMFMFFMMFF...	HIGHLOWLOW
196	8748	FMMFFFFMMMFFMFFFMMMFMMMFFFMFFMFMMFMMMMFMFFMMFF...	HIGHLOWLOW
197	8800	FMMFFFFMMMFFMFFFMMMFMMMFFFMFFMFMMFMMMMFMFFMMFF...	HIGHLOWLOW
198	8823	FMMFFFFMMMFFMFFFMMMFMMMFFFMFFMFMMFMMMMFMFFMMFF...	HIGHLOWLOW
199	8863	FMMFFFFMMMFFMFFFMMMFMMMFFFMFFMFMMFMMMMFMFFMMFF...	HIGHLOWLOW

200 rows × 6 columns

In [24]:

```
df2.count()
```

Out[24]:

```
Age           200
Sex           200
BP            200
Cholesterol   200
Value         200
Drug          200
dtype: int64
```

In [25]:

```
df2.min()
```

Out[25]:

```
Age               15
Sex                F
BP              HIGH
Cholesterol     HIGH
Value          6.269
Drug           drugA
dtype: object
```

In [26]:

```
df2.max()
```

Out[26]:

```
Age                 74
Sex                  M
BP              NORMAL
Cholesterol     NORMAL
Value           38.247
Drug             drugY
dtype: object
```

DataSet3

In [27]:

```
import pandas as pd
import numpy as np
```

In [28]:

```
df3=pd.read_csv(r'C:\Users\user\Downloads\5_Instagram data.csv')
```

In [29]:

```
df3.mean()
```

Out[29]:

```
Impressions       5703.991597
From Home         2475.789916
From Hashtags     1887.512605
From Explore      1078.100840
From Other         171.092437
Saves              153.310924
Comments             6.663866
Shares               9.361345
Likes              173.781513
Profile Visits      50.621849
Follows             20.756303
dtype: float64
```

In [30]:

```
df3.median()
```

Out[30]:

```
Impressions       4289.0
From Home         2207.0
From Hashtags     1278.0
From Explore       326.0
From Other          74.0
Saves              109.0
Comments             6.0
Shares               6.0
Likes              151.0
Profile Visits      23.0
Follows              8.0
dtype: float64
```

In [31]:

```
df3.mode()
```

Out[31]:

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Profile Visits
0	5394.0	1975.0	116	45.0	34.0	40.0	6.0	3.0	114.0	19.0
1	NaN	NaN	201	84.0	NaN	135.0	NaN	NaN	151.0	21.0
2	NaN	NaN	278	NaN	NaN	144.0	NaN	NaN	NaN	NaN
3	NaN	NaN	362	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	NaN	NaN	411	NaN	NaN	NaN	NaN	NaN	NaN	NaN
5	NaN	NaN	583	NaN	NaN	NaN	NaN	NaN	NaN	NaN
6	NaN	NaN	655	NaN	NaN	NaN	NaN	NaN	NaN	NaN
7	NaN	NaN	707	NaN	NaN	NaN	NaN	NaN	NaN	NaN
8	NaN	NaN	771	NaN	NaN	NaN	NaN	NaN	NaN	NaN
9	NaN	NaN	794	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10	NaN	NaN	1248	NaN	NaN	NaN	NaN	NaN	NaN	NaN
11	NaN	NaN	1260	NaN	NaN	NaN	NaN	NaN	NaN	NaN
12	NaN	NaN	1278	NaN	NaN	NaN	NaN	NaN	NaN	NaN
13	NaN	NaN	1693	NaN	NaN	NaN	NaN	NaN	NaN	NaN
14	NaN	NaN	1938	NaN	NaN	NaN	NaN	NaN	NaN	NaN
15	NaN	NaN	2351	NaN	NaN	NaN	NaN	NaN	NaN	NaN
16	NaN	NaN	2975	NaN	NaN	NaN	NaN	NaN	NaN	NaN
17	NaN	NaN	3450	NaN	NaN	NaN	NaN	NaN	NaN	NaN
18	NaN	NaN	3551	NaN	NaN	NaN	NaN	NaN	NaN	NaN

In [32]:

```
df3.describe()
```

Out[32]:

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Co
count	119.000000	119.000000	119.000000	119.000000	119.000000	119.000000	11
mean	5703.991597	2475.789916	1887.512605	1078.100840	171.092437	153.310924	
std	4843.780105	1489.386348	1884.361443	2613.026132	289.431031	156.317731	
min	1941.000000	1133.000000	116.000000	0.000000	9.000000	22.000000	
25%	3467.000000	1945.000000	726.000000	157.500000	38.000000	65.000000	
50%	4289.000000	2207.000000	1278.000000	326.000000	74.000000	109.000000	
75%	6138.000000	2602.500000	2363.500000	689.500000	196.000000	169.000000	
max	36919.000000	13473.000000	11817.000000	17414.000000	2547.000000	1095.000000	1

In [33]:

```
df3.sum()
```

Out[33]:

```
Impressions                                                     678775
From Home                                                       294619
From Hashtags                                                   224614
From Explore                                                    128294
From Other                                                       20360
Saves                                                            18244
Comments                                                           793
Shares                                                            1114
Likes                                                            20680
Profile Visits                                                    6024
Follows                                                           2470
Caption           Here are some of the most important data visua...
Hashtags          #finance�#money�#business�#investing�#investme...
dtype: object
```

In [34]:

```
df3.cumsum()
```

Out[34]:

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Prof Vis
0	3920	2586	1028	619	56	98	9	5	162	
1	9314	5313	2866	1793	134	292	16	19	386	
2	13335	7398	4054	1793	667	333	27	20	517	1
3	17863	10098	4675	2725	740	505	37	27	730	1
4	20381	11802	4930	3004	777	601	42	31	853	1
...	...	...	...	...	...	...	...	...	...	
114	599291	266275	214385	90803	17545	16325	782	1011	19448	52
115	605022	268198	215753	93069	17610	16460	786	1012	19596	52
116	609161	269331	217291	94436	17643	16496	786	1013	19688	52
117	641856	281146	220438	111850	17813	17591	788	1088	20237	54

	Impressions	From Home	From Hashtags	From Explore	From Other	Saves	Comments	Shares	Likes	Prof Vis
118	678775	294619	224614	128294	20360	18244	793	1114	20680	60

119 rows × 13 columns

In [35]:

```
df3.count()
```

Out[35]:

```
Impressions       119
From Home         119
From Hashtags     119
From Explore      119
From Other        119
Saves             119
Comments          119
Shares            119
Likes             119
Profile Visits    119
Follows           119
Caption           119
Hashtags          119
dtype: int64
```

In [36]:

```
df3.min()
```

Out[36]:

```
Impressions                                                     1941
From Home                                                       1133
From Hashtags                                                    116
From Explore                                                       0
From Other                                                         9
Saves                                                             22
Comments                                                           0
Shares                                                             0
Likes                                                             72
Profile Visits                                                     4
Follows                                                            0
Caption           170 Python Projects with Source Code solved an...
Hashtags          #career�#job�#jobs�#jobsearch�#education�#busi...
dtype: object
```

In [37]:

```
df3.max()
```

Out[37]:

```
Impressions                                                     36919
From Home                                                       13473
From Hashtags                                                   11817
From Explore                                                    17414
From Other                                                       2547
Saves                                                            1095
Comments                                                           19
Shares                                                             75
Likes                                                             549
Profile Visits                                                    611
Follows                                                           260
Caption           You must have seen the news divided into categ...
Hashtags          #timeseries�#time�#statistics�#datascience�#bi...
dtype: object
```

In [38]:

```
from numpy import cov
from scipy.stats import spearmanr
from scipy.stats import pearsonr
```

In [39]:

```
cov(df3['Shares'],df3['Likes'])
```

Out[39]:

```
array([[ 101.79205241,  588.27453354],
       [ 588.27453354, 6786.29084176]])
```

In [40]:

```
spearmanr(df3['Shares'],df3['Likes'])
```

Out[40]:

```
SpearmanrResult(correlation=0.5692666973936509, pvalue=1.42478204825654e-1
1)
```

In [41]:

```
pearsonr(df3['Shares'],df3['Likes'])
```

Out[41]:

```
(0.7077940026881047, 2.258074786066927e-19)
```

DataSet4

In [42]:

```
import pandas as pd
import numpy as np
```

In [43]:

```
df4=pd.read_csv(r'C:\Users\user\Downloads\6_Salesworkload1.csv')
```

In [44]:

```
df4.mean()
```

Out[44]:

```
Time index      5.000000e+00
StoreID         6.199522e+04
Dept_ID         9.470588e+00
HoursLease      2.203608e+01
Sales units     1.076471e+06
Turnover        3.721393e+06
Customer                 NaN
dtype: float64
```

In [45]:

```
df4.median()
```

Out[45]:

```
Time index         5.0
StoreID        75400.5
Dept_ID            9.0
HoursLease         0.0
Sales units   293230.0
Turnover      931957.5
Customer           NaN
dtype: float64
```

In [46]:

```
df4.mode()
```

Out[46]:

	MonthYear	Time index	Country	StoreID	City	Dept_ID	Dept. Name	HoursOwn	Hours
0	01.2017	1.0	France	12227.0	Aalborg (I)	1.0	Admin	47.205	
1	02.2017	2.0	Germany	15552.0	Aalborg (II)	2.0	Checkout	NaN	
2	03.2017	3.0	United Kingdom	16927.0	Amsterdam	3.0	Clothing	NaN	
3	04.2017	4.0	NaN	17647.0	Antwerp	4.0	Customer Services	NaN	
4	05.2017	5.0	NaN	18808.0	Barcelona (I)	5.0	Delivery	NaN	
5	06.2017	6.0	NaN	19000.0	Barcelona (II)	6.0	Dry	NaN	
6	10.2016	7.0	NaN	19340.0	Berlin (I)	7.0	Fish	NaN	
7	11.2016	8.0	NaN	19769.0	Berlin (II)	8.0	Food	NaN	
8	12.2016	9.0	NaN	20166.0	Bilbao	9.0	Frozen	NaN	
9	NaN	NaN	NaN	20891.0	Birmingham	11.0	Fruits & Vegetables	NaN	
10	NaN	NaN	NaN	22117.0	Bologna	12.0	Hardware	NaN	
11	NaN	NaN	NaN	23623.0	Bordeaux	13.0	Household	NaN	
12	NaN	NaN	NaN	29650.0	Brno	14.0	Meat	NaN	
13	NaN	NaN	NaN	32949.0	Brussels (I)	15.0	Non Food	NaN	
14	NaN	NaN	NaN	34378.0	Brussels (II)	16.0	all	NaN	
15	NaN	NaN	NaN	38560.0	Cologne	17.0	other	NaN	
16	NaN	NaN	NaN	38976.0	Copenhagen (I)	18.0	others	NaN	
17	NaN	NaN	NaN	42367.0	Copenhagen (II)	NaN	NaN	NaN	
18	NaN	NaN	NaN	45583.0	Den Haag	NaN	NaN	NaN	
19	NaN	NaN	NaN	63354.0	Frankfurt	NaN	NaN	NaN	
20	NaN	NaN	NaN	64983.0	Gothenburg	NaN	NaN	NaN	
21	NaN	NaN	NaN	71991.0	Groningen	NaN	NaN	NaN	
22	NaN	NaN	NaN	73422.0	Hamburg	NaN	NaN	NaN	
23	NaN	NaN	NaN	73762.0	Krakow	NaN	NaN	NaN	
24	NaN	NaN	NaN	73949.0	Leicester	NaN	NaN	NaN	
25	NaN	NaN	NaN	76852.0	Liverpool	NaN	NaN	NaN	
26	NaN	NaN	NaN	77348.0	London (I)	NaN	NaN	NaN	
27	NaN	NaN	NaN	78325.0	London (II)	NaN	NaN	NaN	
28	NaN	NaN	NaN	78450.0	Lyon	NaN	NaN	NaN	
29	NaN	NaN	NaN	79785.0	Madrid (I)	NaN	NaN	NaN	
30	NaN	NaN	NaN	81473.0	Madrid (II)	NaN	NaN	NaN	
31	NaN	NaN	NaN	83160.0	Malmö	NaN	NaN	NaN	
32	NaN	NaN	NaN	85124.0	Manchester	NaN	NaN	NaN	

	MonthYear	Time index	Country	StoreID	City	Dept_ID	Dept. Name	HoursOwn	Hours
33	NaN	NaN	NaN	85321.0	Marseille	NaN	NaN	NaN	
34	NaN	NaN	NaN	85696.0	Milano	NaN	NaN	NaN	
35	NaN	NaN	NaN	86089.0	Munich	NaN	NaN	NaN	
36	NaN	NaN	NaN	86208.0	Nantes	NaN	NaN	NaN	
37	NaN	NaN	NaN	87703.0	Napoli	NaN	NaN	NaN	
38	NaN	NaN	NaN	88253.0	Ostrava	NaN	NaN	NaN	
39	NaN	NaN	NaN	88750.0	Paris (I)	NaN	NaN	NaN	
40	NaN	NaN	NaN	88994.0	Paris (II)	NaN	NaN	NaN	
41	NaN	NaN	NaN	90992.0	Poznan	NaN	NaN	NaN	
42	NaN	NaN	NaN	91973.0	Prague (I)	NaN	NaN	NaN	
43	NaN	NaN	NaN	93033.0	Prague (II)	NaN	NaN	NaN	
44	NaN	NaN	NaN	94153.0	Rome (I)	NaN	NaN	NaN	
45	NaN	NaN	NaN	94882.0	Rome (II)	NaN	NaN	NaN	
46	NaN	NaN	NaN	95434.0	Rotterdam	NaN	NaN	NaN	
47	NaN	NaN	NaN	96493.0	Stockholm	NaN	NaN	NaN	
48	NaN	NaN	NaN	96857.0	Warsaw (I)	NaN	NaN	NaN	
49	NaN	NaN	NaN	98422.0	Warsaw (II)	NaN	NaN	NaN	

In [47]:

```
df4.describe()
```

Out[47]:

	Time index	StoreID	Dept_ID	HoursLease	Sales units	Turnover	Cu
count	7650.000000	7650.000000	7650.000000	7650.000000	7.650000e+03	7.650000e+03	
mean	5.000000	61995.220000	9.470588	22.036078	1.076471e+06	3.721393e+06	
std	2.582158	29924.581631	5.337429	133.299513	1.728113e+06	6.003380e+06	
min	1.000000	12227.000000	1.000000	0.000000	0.000000e+00	0.000000e+00	
25%	3.000000	29650.000000	5.000000	0.000000	5.457125e+04	2.726798e+05	
50%	5.000000	75400.500000	9.000000	0.000000	2.932300e+05	9.319575e+05	
75%	7.000000	87703.000000	14.000000	0.000000	9.175075e+05	3.264432e+06	
max	9.000000	98422.000000	18.000000	3984.000000	1.124296e+07	4.271739e+07	

In [48]:

```
df4.sum()
```

Out[48]:

```
MonthYear      10.201610.201610.201610.201610.201610.201610.2...
Time index                                              38250.0
StoreID                                             474263433.0
Dept_ID                                                 72450.0
HoursLease                                             168576.0
Sales units                                        8235000965.0
Turnover                                          28468656015.0
Customer                                                    0.0
dtype: object
```

In [49]:

```
cumsum(df4['Country'],df4['City'])
```

```
--------------------------------------------------------------------------
-
NameError                                 Traceback (most recent call las
t)
<ipython-input-49-02b57bd1b938> in <module>
----> 1 cumsum(df4['Country'],df4['City'])

NameError: name 'cumsum' is not defined
```

In [50]:

```
df4.count()
```

Out[50]:

```
MonthYear        7658
Time index       7650
Country          7650
StoreID          7650
City             7650
Dept_ID          7650
Dept. Name       7650
HoursOwn         7650
HoursLease       7650
Sales units      7650
Turnover         7650
Customer            0
Area (m2)        7650
Opening hours    7650
dtype: int64
```

In [51]:

```
df4.min()
```

Out[51]:

```
MonthYear        - - - -
Time index           1.0
StoreID          12227.0
Dept_ID              1.0
HoursLease           0.0
Sales units          0.0
Turnover             0.0
Customer             NaN
dtype: object
```

In [52]:

```
df4.max()
```

Out[52]:

```
MonthYear          12.2016
Time index             9.0
StoreID            98422.0
Dept_ID               18.0
HoursLease          3984.0
Sales units     11242955.0
Turnover        42717390.0
Customer               NaN
dtype: object
```

In [53]:

```
from numpy import cov
from scipy.stats import spearmanr
from scipy.stats import pearsonr
```

In [54]:

```
cov(df4['Dept_ID'],df4['Turnover'])
```

Out[54]:

```
array([[nan, nan],
       [nan, nan]])
```

In [55]:

```
spearmanr(df4['Dept_ID'],df4['Turnover'])
```

Out[55]:

```
SpearmanrResult(correlation=nan, pvalue=nan)
```

DataSet5

In [56]:

```
import pandas as pd
import numpy as np
```

In [57]:

```
df5=pd.read_csv(r'C:\Users\user\Downloads\8_BreastCancerPrediction.csv')
```

In [58]:

```
df5.mean()
```

Out[58]:

```
id                         3.037183e+07
radius_mean                1.412729e+01
texture_mean               1.928965e+01
perimeter_mean             9.196903e+01
area_mean                  6.548891e+02
smoothness_mean            9.636028e-02
compactness_mean           1.043410e-01
concavity_mean             8.879932e-02
concave points_mean        4.891915e-02
symmetry_mean              1.811619e-01
fractal_dimension_mean     6.279761e-02
radius_se                  4.051721e-01
texture_se                 1.216853e+00
perimeter_se               2.866059e+00
area_se                    4.033708e+01
smoothness_se              7.040979e-03
compactness_se             2.547814e-02
concavity_se               3.189372e-02
concave points_se          1.179614e-02
symmetry_se                2.054230e-02
fractal_dimension_se       3.794904e-03
radius_worst               1.626919e+01
texture_worst              2.567722e+01
perimeter_worst            1.072612e+02
area_worst                 8.805831e+02
smoothness_worst           1.323686e-01
compactness_worst          2.542650e-01
concavity_worst            2.721885e-01
concave points_worst       1.146062e-01
symmetry_worst             2.900756e-01
fractal_dimension_worst    8.394582e-02
Unnamed: 32                         NaN
dtype: float64
```

In [59]:

```
df5.median()
```

Out[59]:

```
id                         906024.000000
radius_mean                    13.370000
texture_mean                   18.840000
perimeter_mean                 86.240000
area_mean                     551.100000
smoothness_mean                 0.095870
compactness_mean                0.092630
concavity_mean                  0.061540
concave points_mean             0.033500
symmetry_mean                   0.179200
fractal_dimension_mean          0.061540
radius_se                       0.324200
texture_se                      1.108000
perimeter_se                    2.287000
area_se                        24.530000
smoothness_se                   0.006380
compactness_se                  0.020450
concavity_se                    0.025890
concave points_se               0.010930
symmetry_se                     0.018730
fractal_dimension_se            0.003187
radius_worst                   14.970000
texture_worst                  25.410000
perimeter_worst                97.660000
area_worst                    686.500000
smoothness_worst                0.131300
compactness_worst               0.211900
concavity_worst                 0.226700
concave points_worst            0.099930
symmetry_worst                  0.282200
fractal_dimension_worst         0.080040
Unnamed: 32                          NaN
dtype: float64
```

In [60]:

```
df5.mode()
```

Out[60]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothr
0	8670	B	12.34	14.93	82.61	512.2	
1	8913	NaN	NaN	15.70	87.76	NaN	
2	8915	NaN	NaN	16.84	134.70	NaN	
3	9047	NaN	NaN	16.85	NaN	NaN	
4	85715	NaN	NaN	17.46	NaN	NaN	
...	...	...	...	...	...	...	
564	911157302	NaN	NaN	NaN	NaN	NaN	
565	911296201	NaN	NaN	NaN	NaN	NaN	
566	911296202	NaN	NaN	NaN	NaN	NaN	
567	911320501	NaN	NaN	NaN	NaN	NaN	
568	911320502	NaN	NaN	NaN	NaN	NaN	

569 rows × 33 columns

In [61]:

```
df5.describe()
```

Out[61]:

	id	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_
count	5.690000e+02	569.000000	569.000000	569.000000	569.000000	569.0(
mean	3.037183e+07	14.127292	19.289649	91.969033	654.889104	0.0(
std	1.250206e+08	3.524049	4.301036	24.298981	351.914129	0.0
min	8.670000e+03	6.981000	9.710000	43.790000	143.500000	0.0!
25%	8.692180e+05	11.700000	16.170000	75.170000	420.300000	0.0{
50%	9.060240e+05	13.370000	18.840000	86.240000	551.100000	0.0(
75%	8.813129e+06	15.780000	21.800000	104.100000	782.700000	0.1(
max	9.113205e+08	28.110000	39.280000	188.500000	2501.000000	0.1(

8 rows × 32 columns

In [62]:

```
df5.sum()
```

Out[62]:

```
id                                                      172815720
85
diagnosis                MMMMMMMMMMMMMMMMMMMMMBBBMMMMMMMMMMMMMMMMMMBMMMMMMM
M...
radius_mean                                                8038.4
29
texture_mean                                               10975.
81
perimeter_mean                                             52330.
38
area_mean                                                   37263
1.9
smoothness_mean                                              54.8
29
compactness_mean                                           59.370
02
concavity_mean                                            50.5268
11
concave points_mean                                       27.8349
94
symmetry_mean                                              103.08
11
fractal_dimension_mean                                     35.731
84
radius_se                                                  230.54
29
texture_se                                                 692.38
96
perimeter_se                                              1630.78
77
area_se                                                   22951.7
98
smoothness_se                                              4.0063
17
compactness_se                                            14.4970
61
```

In [63]:

```
df5.cumsum()
```

Out[63]:

	id	diagnosis
0	842302	M
1	1684819	MM
2	85985722	MMM
3	170334023	MMMM
4	254692425	MMMMM
...	...	...
564	17278698457	MMMMMMMMMMMMMMMMMMMMMBBBMMMMMMMMMMMMMMMMMMBMMMMMMMM...
565	17279625139	MMMMMMMMMMMMMMMMMMMMMBBBMMMMMMMMMMMMMMMMMMBMMMMMMMM...
566	17280552093	MMMMMMMMMMMMMMMMMMMMMBBBMMMMMMMMMMMMMMMMMMBMMMMMMMM...
567	17281479334	MMMMMMMMMMMMMMMMMMMMMBBBMMMMMMMMMMMMMMMMMMBMMMMMMMM...
568	17281572085	MMMMMMMMMMMMMMMMMMMMMBBBMMMMMMMMMMMMMMMMMMBMMMMMMMM...

569 rows × 33 columns

```
25
concave points_se                                          6.7120
02
symmetry_se                                               11.6885
68
fractal_dimension_se                                         2.15
93
radius_worst                                               9257.1
69
texture_worst                                              14610.
34
perimeter_worst                                            61031.
63
area_worst                                                  50105
1.8
smoothness_worst                                           75.317
73
compactness_worst                                         144.676
81
concavity_worst                                          154.8752
47
concave points_worst                                      65.2109
41
symmetry_worst                                              165.0
```

```
53
fractal_dimension_worst                                              47.765
17
Unnamed: 32
0.0
dtype: object
```

In [64]:

```
df5.count()
```

Out[64]:

```
id                         569
diagnosis                  569
radius_mean                569
texture_mean               569
perimeter_mean             569
area_mean                  569
smoothness_mean            569
compactness_mean           569
concavity_mean             569
concave points_mean        569
symmetry_mean              569
fractal_dimension_mean     569
radius_se                  569
texture_se                 569
perimeter_se               569
area_se                    569
smoothness_se              569
compactness_se             569
concavity_se               569
concave points_se          569
symmetry_se                569
fractal_dimension_se       569
radius_worst               569
texture_worst              569
perimeter_worst            569
area_worst                 569
smoothness_worst           569
compactness_worst          569
concavity_worst            569
concave points_worst       569
symmetry_worst             569
fractal_dimension_worst    569
Unnamed: 32                  0
dtype: int64
```

In [65]:

```
df5.min()
```

Out[65]:

```
id                              8670
diagnosis                          B
radius_mean                    6.981
texture_mean                    9.71
perimeter_mean                 43.79
area_mean                      143.5
smoothness_mean              0.05263
compactness_mean             0.01938
concavity_mean                   0.0
concave points_mean              0.0
symmetry_mean                  0.106
fractal_dimension_mean       0.04996
radius_se                     0.1115
texture_se                    0.3602
perimeter_se                   0.757
area_se                        6.802
smoothness_se               0.001713
compactness_se              0.002252
concavity_se                     0.0
concave points_se                0.0
symmetry_se                 0.007882
fractal_dimension_se        0.000895
radius_worst                    7.93
texture_worst                  12.02
perimeter_worst                50.41
area_worst                     185.2
smoothness_worst             0.07117
compactness_worst            0.02729
concavity_worst                  0.0
concave points_worst             0.0
symmetry_worst                0.1565
fractal_dimension_worst      0.05504
Unnamed: 32                      NaN
dtype: object
```

In [66]:

```
df5.max()
```

Out[66]:

```
id                          911320502
diagnosis                           M
radius_mean                     28.11
texture_mean                    39.28
perimeter_mean                  188.5
area_mean                      2501.0
smoothness_mean                0.1634
compactness_mean               0.3454
concavity_mean                 0.4268
concave points_mean            0.2012
symmetry_mean                   0.304
fractal_dimension_mean        0.09744
radius_se                       2.873
texture_se                      4.885
perimeter_se                    21.98
area_se                         542.2
smoothness_se                 0.03113
compactness_se                 0.1354
concavity_se                    0.396
concave points_se             0.05279
symmetry_se                   0.07895
fractal_dimension_se          0.02984
radius_worst                    36.04
texture_worst                   49.54
perimeter_worst                 251.2
area_worst                     4254.0
smoothness_worst               0.2226
compactness_worst               1.058
concavity_worst                 1.252
concave points_worst            0.291
symmetry_worst                 0.6638
fractal_dimension_worst        0.2075
Unnamed: 32                       NaN
dtype: object
```

In [67]:

```
from numpy import cov
from scipy.stats import spearmanr
from scipy.stats import pearsonr
```

In [68]:

```
cov(df5['perimeter_worst'],df5['area_worst'])
```

Out[68]:

```
array([[  1129.13084694,  18702.86999057],
       [ 18702.86999057, 324167.38510217]])
```

In [69]:

```
spearmanr(df5['perimeter_worst'],df5['area_worst'])
```

Out[69]:

```
SpearmanrResult(correlation=0.992432709857714, pvalue=0.0)
```

In [70]:

```
pearsonr(df5['perimeter_worst'],df5['area_worst'])
```

Out[70]:

```
(0.9775780914063879, 0.0)
```