hkoh0nydh

August 4, 2023

1 20104169 - SUMESH R

2 Importing Libraries

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

[2]:
```
from google.colab import drive
drive.mount('/content/drive')
df=pd.read_csv("/content/drive/MyDrive/mydatasets/rainfall/rainfall_haryana␣
 ↪delhi & chandigarh.csv")
df
```

```
Mounted at /content/drive
```

[2]:
```
     index                  SUBDIVISION  YEAR   JAN   FEB   MAR   APR   MAY  \
0     1357  HARYANA DELHI & CHANDIGARH  1901  35.4  28.9  11.1   0.0   5.1
1     1358  HARYANA DELHI & CHANDIGARH  1902   0.0   0.7   2.9  10.2  15.8
2     1359  HARYANA DELHI & CHANDIGARH  1903  14.7   0.5   2.3   0.5   8.5
3     1360  HARYANA DELHI & CHANDIGARH  1904   7.6   0.7  48.0   0.5  29.3
4     1361  HARYANA DELHI & CHANDIGARH  1905  44.8  20.8  14.0   1.3   7.4
..     …                           …    …     …     …     …     …     …
110   1467  HARYANA DELHI & CHANDIGARH  2011   0.7  26.7   6.9   8.9  28.7
111   1468  HARYANA DELHI & CHANDIGARH  2012   8.2   0.2   0.1  11.8   3.8
112   1469  HARYANA DELHI & CHANDIGARH  2013  21.1  52.2   5.3   3.3   1.4
113   1470  HARYANA DELHI & CHANDIGARH  2014  13.0  17.3  26.8   7.5  20.3
114   1471  HARYANA DELHI & CHANDIGARH  2015  12.4   6.6  71.8  34.8   8.4

      JUN    JUL    AUG    SEP  OCT   NOV   DEC  ANNUAL  Jan-Feb  Mar-May  \
0    13.2  126.4  151.5   10.5  2.0   0.0   6.1   390.1     64.3     16.2
1    74.6  149.3   97.1   59.8  9.3   0.0   0.0   419.8      0.7     28.9
2     8.6  151.6  138.2   97.7  4.0   0.0   2.3   428.9     15.2     11.3
3    34.3  109.7  162.9  102.3  1.5  10.4  20.3   527.8      8.3     77.8
4    20.1   93.6   23.1   92.6  0.0   0.0   5.1   322.9     65.7     22.7
..    …      …      …      …    …    …     …      …        …        …
```

```
110  94.4   85.0  127.3  133.1   0.0    0.0    0.4   512.0      27.5      44.4
111   5.3   68.1  196.6   90.7   2.4    0.6    3.5   391.4       8.4      15.8
112  62.1   96.5  161.9   42.8  10.9    1.7    2.1   461.2      73.2      10.0
113  25.9   72.3   34.8   67.3  10.5    0.2    9.6   305.5      30.3      54.6
114  43.7  130.3   89.2   32.1   3.7    2.3    0.2   435.3      18.9     114.9

     Jun-Sep  Oct-Dec
0      301.6      8.1
1      380.9      9.3
2      396.1      6.2
3      409.3     32.3
4      229.4      5.1
..       …        …
110    439.8      0.4
111    360.7      6.5
112    363.3     14.7
113    200.2     20.3
114    295.2      6.2

[115 rows x 20 columns]
```

3 Data Cleaning and Data Preprocessing

[3]:
```
df=df.dropna()
```

[4]:
```
df.columns
```

[4]:
```
Index(['index', 'SUBDIVISION', 'YEAR', 'JAN', 'FEB', 'MAR', 'APR', 'MAY',
       'JUN', 'JUL', 'AUG', 'SEP', 'OCT', 'NOV', 'DEC', 'ANNUAL', 'Jan-Feb',
       'Mar-May', 'Jun-Sep', 'Oct-Dec'],
      dtype='object')
```

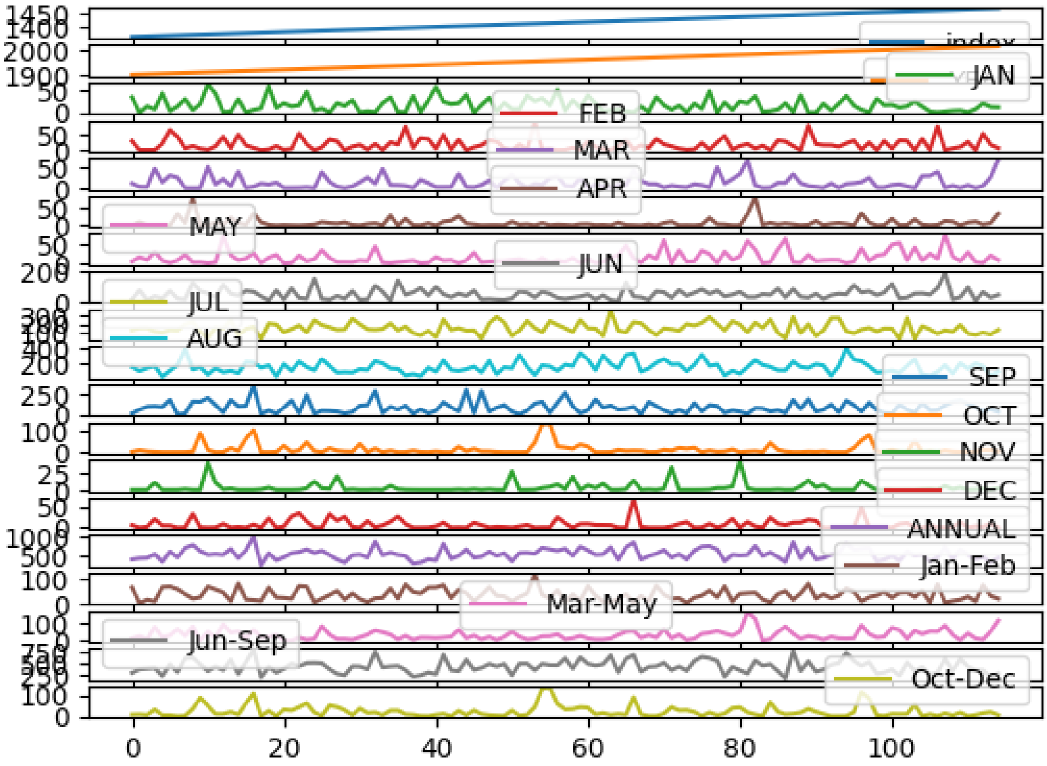
[5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 115 entries, 0 to 114
Data columns (total 20 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   index        115 non-null    int64
 1   SUBDIVISION  115 non-null    object
 2   YEAR         115 non-null    int64
 3   JAN          115 non-null    float64
 4   FEB          115 non-null    float64
 5   MAR          115 non-null    float64
 6   APR          115 non-null    float64
 7   MAY         115 non-null    float64
 8   JUN         115 non-null    float64
 9   JUL         115 non-null    float64
 10  AUG         115 non-null    float64
 11  SEP         115 non-null    float64
 12  OCT         115 non-null    float64
 13  NOV         115 non-null    float64
 14  DEC         115 non-null    float64
 15  ANNUAL      115 non-null    float64
 16  Jan-Feb     115 non-null    float64
 17  Mar-May     115 non-null    float64
 18  Jun-Sep     115 non-null    float64
 19  Oct-Dec     115 non-null    float64
dtypes: float64(17), int64(2), object(1)
memory usage: 18.1+ KB
```

4 Line chart

```
[6]: df.plot.line(subplots=True)
```

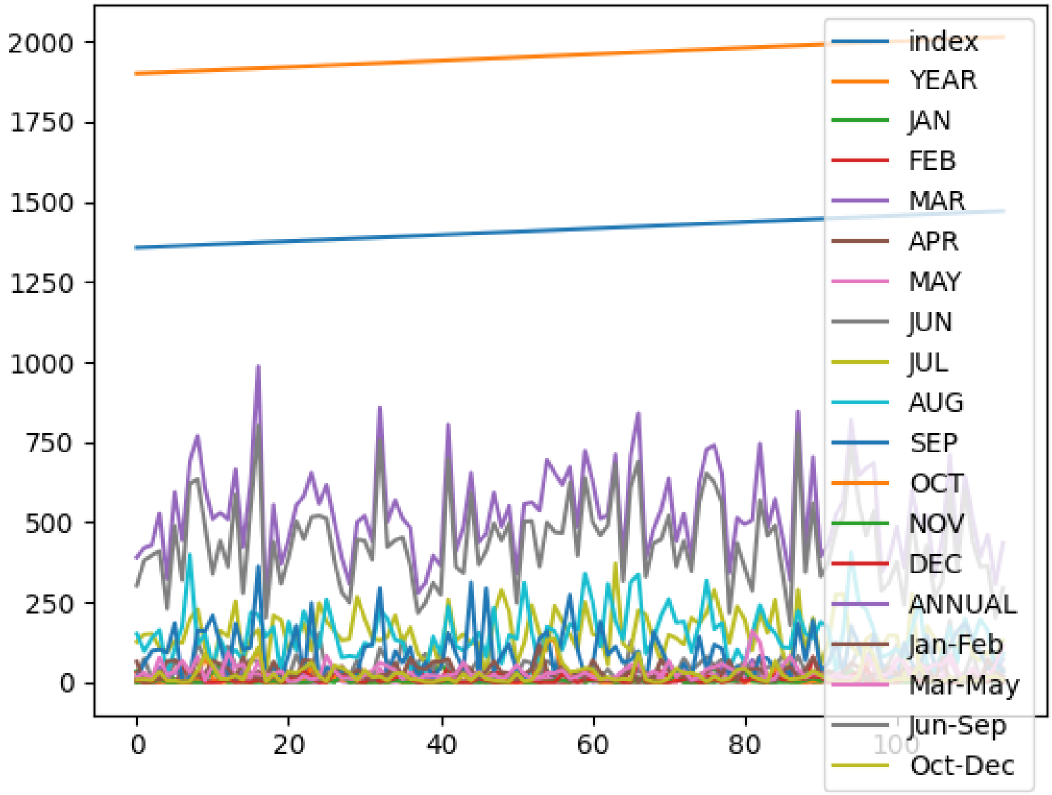
```
[6]: array([<Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
            <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
            <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
            <Axes: >], dtype=object)
```



5 Line chart

[7]:
```
df.plot.line()
```

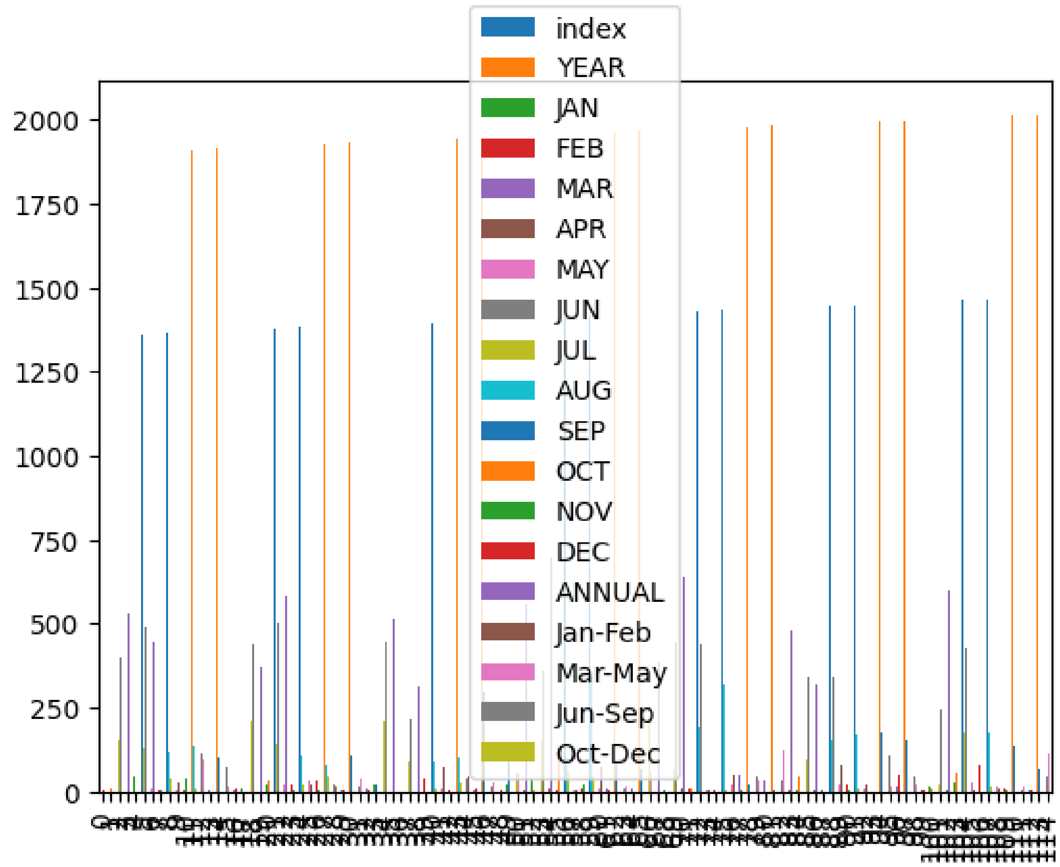
[7]:
```
<Axes: >
```



6 Bar chart

[8]:
```
df.plot.bar()
```

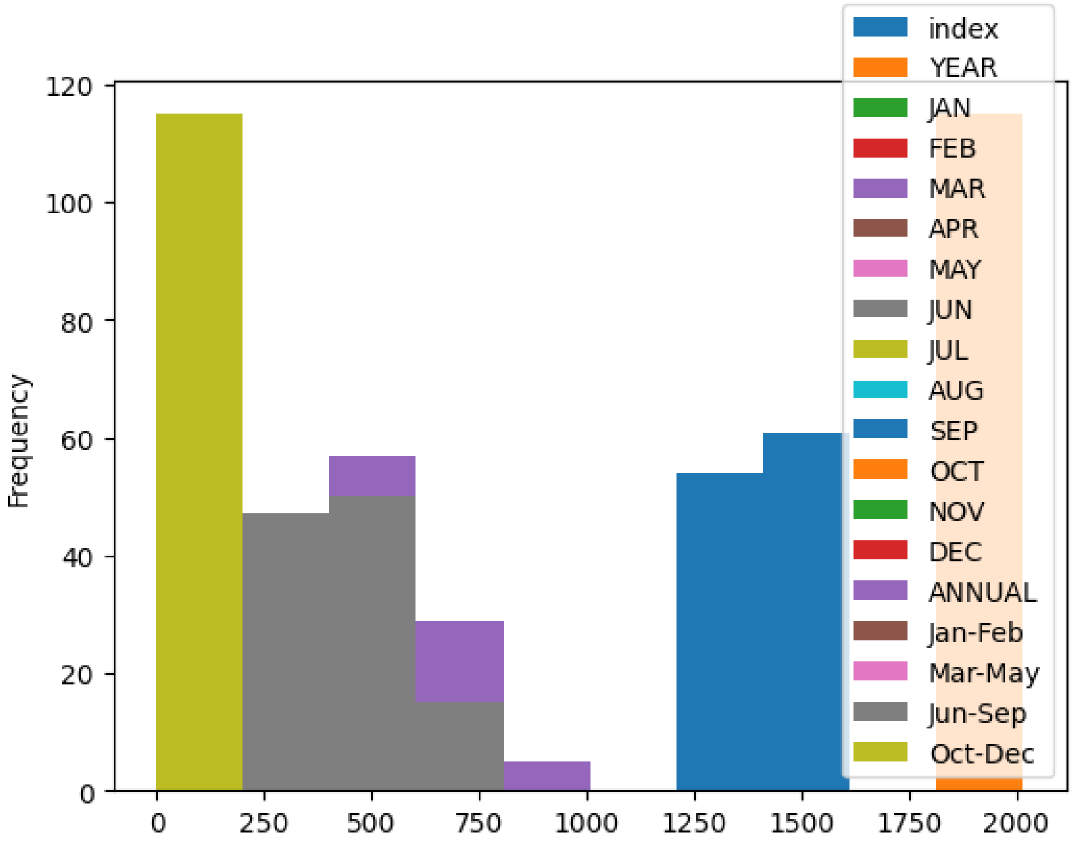
[8]:
```
<Axes: >
```



7 Histogram

```
[9]: df.plot.hist()
```

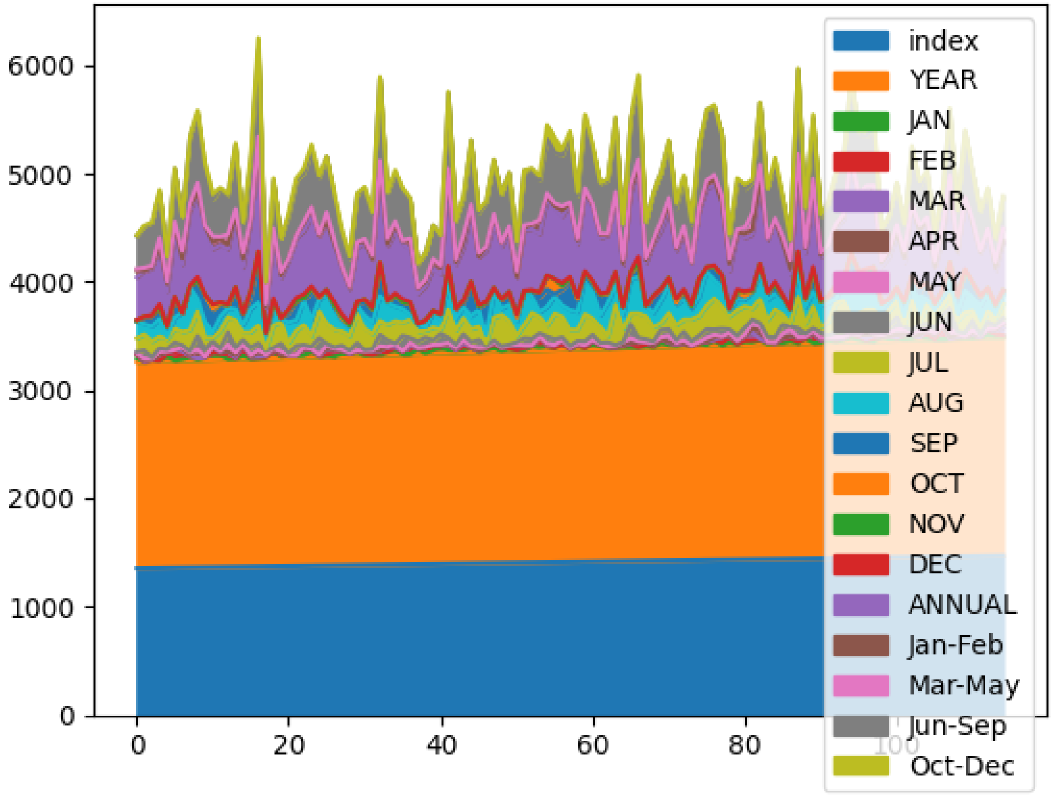
```
[9]: <Axes: ylabel='Frequency'>
```



8 Area chart

[10]:
```
df.plot.area()
```

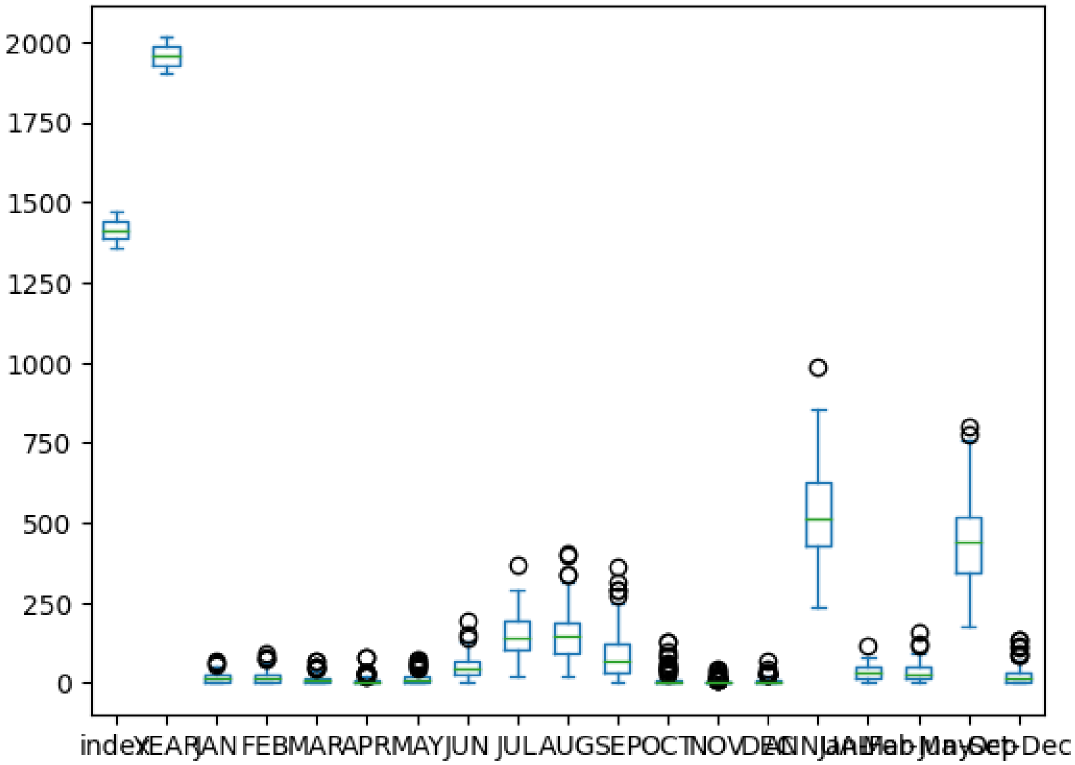
[10]:
```
<Axes: >
```



9 Box chart

[11]:
```
df.plot.box()
```

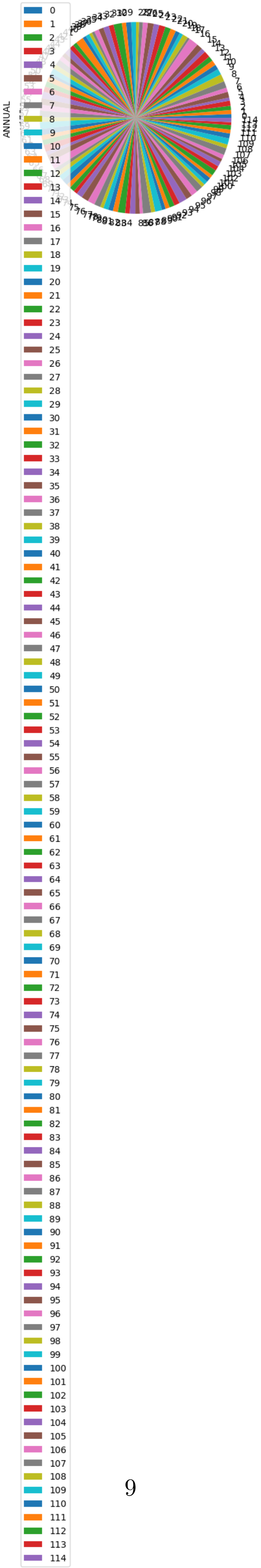
[11]:
```
<Axes: >
```



10 Pie chart

```
[12]: df.plot.pie(y='ANNUAL' )
```

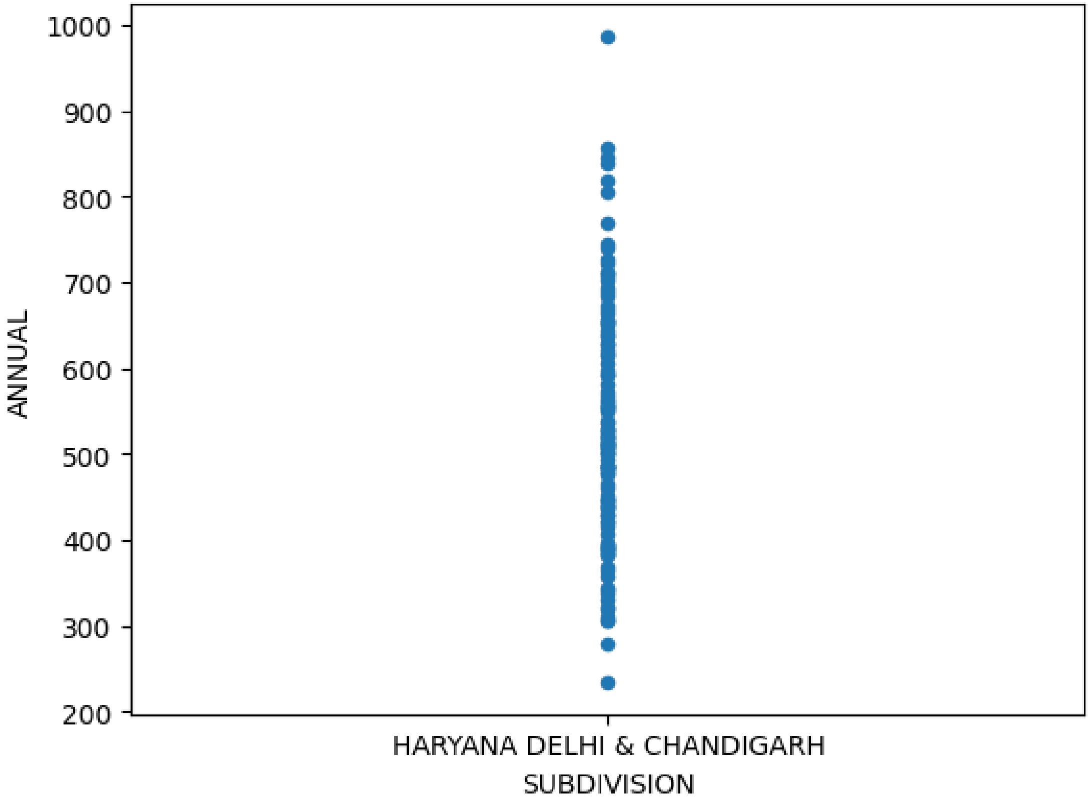
```
[12]: <Axes: ylabel='ANNUAL'>
```


ANNUAL
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114

11 Scatter chart

[13]:
```
df.plot.scatter(x='SUBDIVISION' ,y='ANNUAL')
```

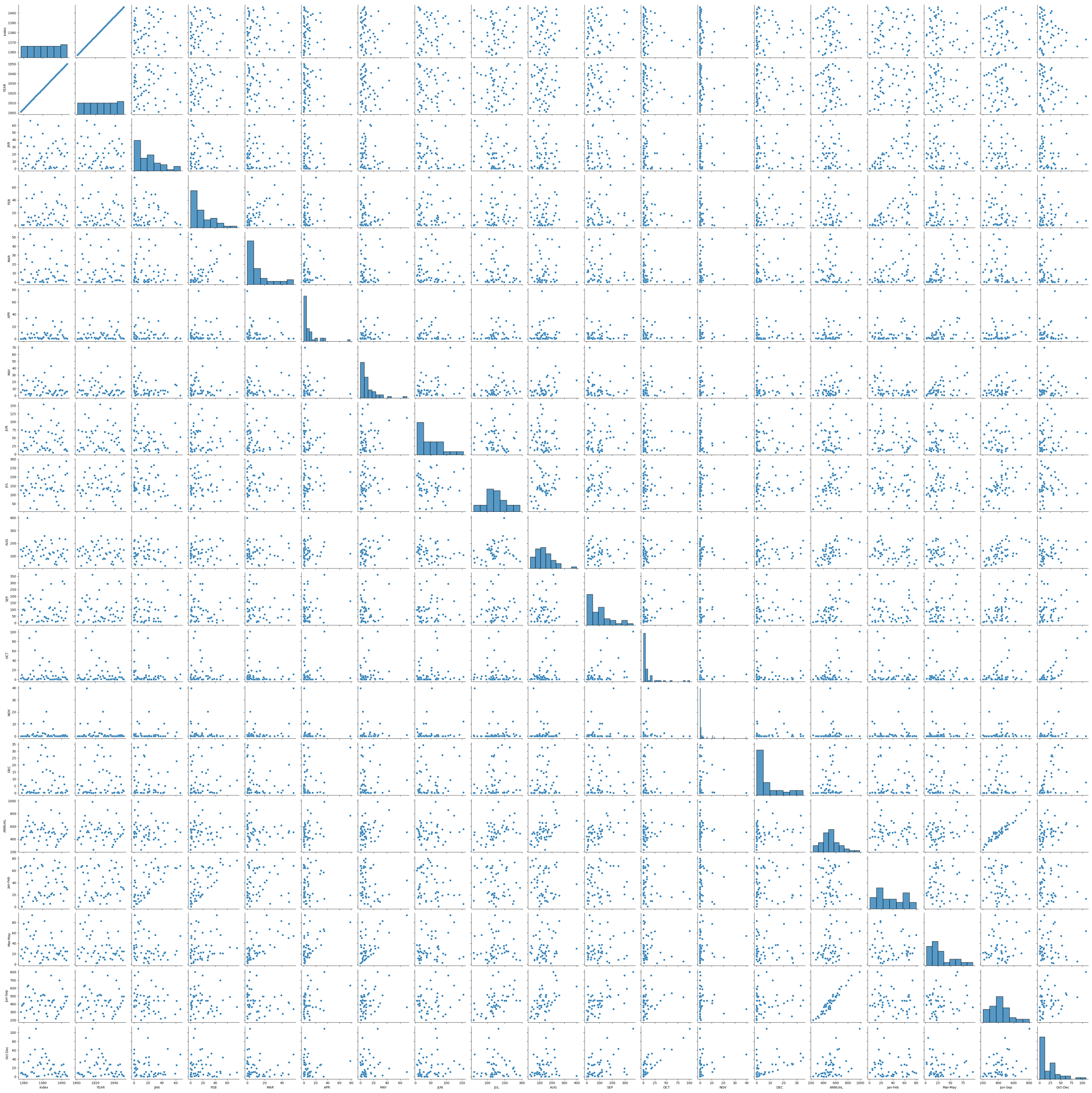
[13]:
```
<Axes: xlabel='SUBDIVISION', ylabel='ANNUAL'>
```



12 Seaborn

[14]:
```
sns.pairplot(df[0:50])
```

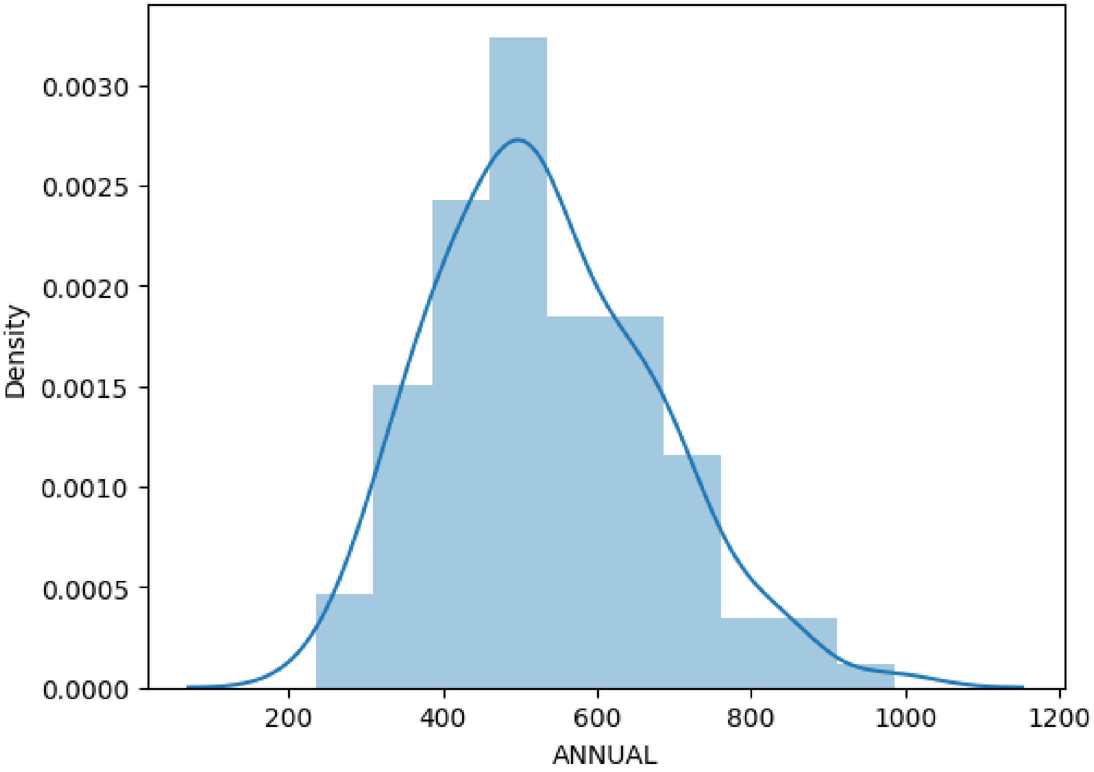
[14]:
```
<seaborn.axisgrid.PairGrid at 0x7d5460223220>
```



```
[15]: sns.distplot(df['ANNUAL'])
```

```
<ipython-input-15-5daa97052ca5>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df['ANNUAL'])
```

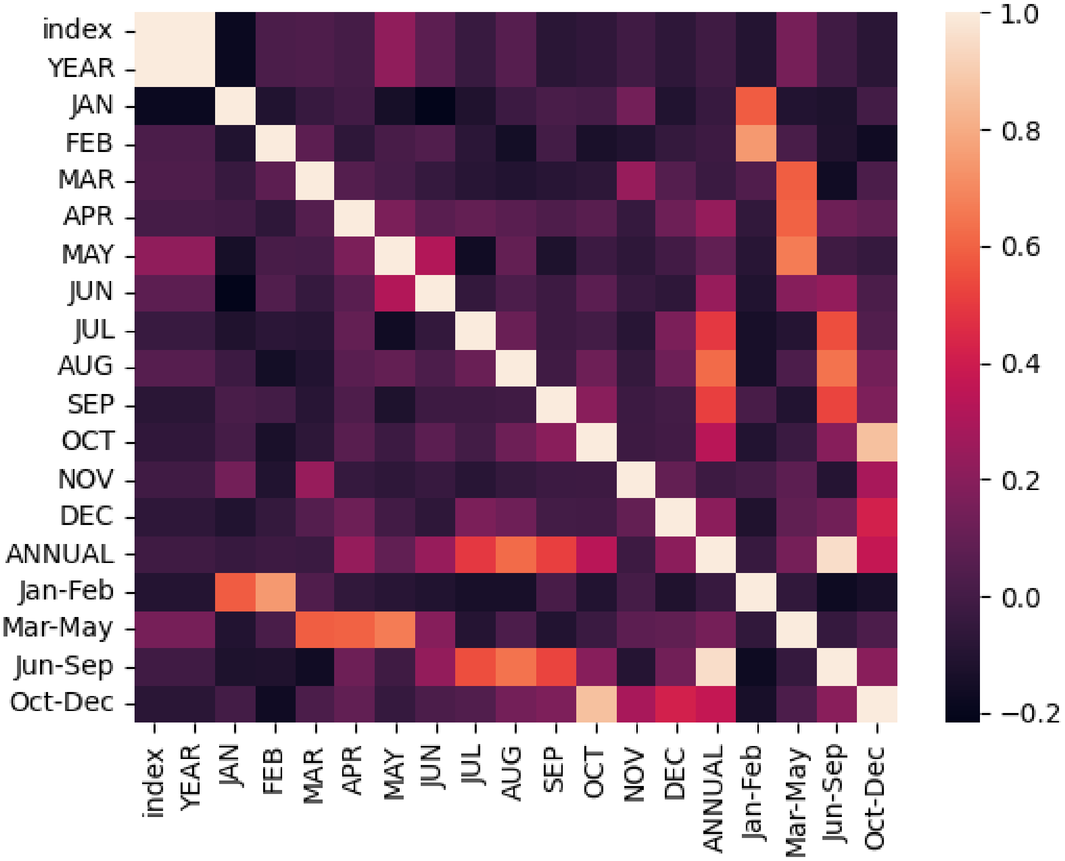
[15]: <Axes: xlabel='ANNUAL', ylabel='Density'>



[16]:
```
sns.heatmap(df.corr())
```

```
<ipython-input-16-aa4f4450a243>:1: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only
to silence this warning.
  sns.heatmap(df.corr())
```

[16]: <Axes: >


index
YEAR
JAN
FEB
MAR
APR
MAY
JUN
JUL
AUG
SEP
OCT
NOV
DEC
ANNUAL
Jan-Feb
Mar-May
Jun-Sep
Oct-Dec
1.0
0.8
0.6
0.4
0.2
0.0
−0.2