2tarsibbk

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_east madhya␣
 ↪pradesh.csv")
df
```

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

[2]:

```
     index          SUBDIVISION  YEAR   JAN   FEB   MAR   APR   MAY    JUN  \
0      2162  EAST MADHYA PRADESH  1901  48.5  38.1  15.7  10.7   6.2   61.0
1      2163  EAST MADHYA PRADESH  1902  14.9   8.9   0.0   3.6   2.7   28.0
2      2164  EAST MADHYA PRADESH  1903   5.6   2.9   0.3   0.9  37.5   67.5
3      2165  EAST MADHYA PRADESH  1904   2.0  15.3  48.2   0.0   8.6  109.9
4      2166  EAST MADHYA PRADESH  1905  15.9   8.0  14.3  12.3  10.2   34.4
..     …                  …   …   …   …   …   …   …    …
110    2272  EAST MADHYA PRADESH  2011   0.6   1.9   0.3   7.1   4.7  332.5
111    2273  EAST MADHYA PRADESH  2012  39.4   0.7   0.6   1.1   1.2   67.8
112    2274  EAST MADHYA PRADESH  2013   2.0  43.4  14.1   9.5   0.3  311.9
113    2275  EAST MADHYA PRADESH  2014  32.1  49.7  17.8   5.1   2.5   91.8
114    2276  EAST MADHYA PRADESH  2015  37.3  11.0  73.4  25.8   6.3  139.2

       JUL    AUG    SEP    OCT   NOV   DEC  ANNUAL  Jan-Feb  Mar-May  \
0    367.5  589.2  189.9    5.9   0.0   0.0  1332.7     86.5     32.7
1    411.9  227.0  236.6   17.0  27.6   6.1   984.2     23.8      6.2
2    261.4  366.7  257.4  177.9   0.0   0.0  1178.1      8.5     38.7
3    443.2  316.6  135.6   44.8   3.2  16.9  1144.2     17.3     56.7
4    292.4  243.3  250.9    2.9   0.0   1.6   886.0     23.9     36.7
..     …    …    …    …   …   …     …      …      …
```

```
110  323.6  326.9  276.5    1.1   0.0   0.0  1275.2      2.5     12.1
111  398.9  351.7  172.6   12.7   3.8   2.7  1053.1     40.0      2.9
112  456.2  480.8   78.0  124.2   0.5   1.0  1521.9     45.4     23.9
113  283.4  231.8  139.6   56.4   1.9  12.9   924.9     81.8     25.3
114  262.2  272.1   71.6   38.2   1.2   0.9   939.2     48.3    105.5

     Jun-Sep  Oct-Dec
0     1207.7      5.9
1      903.5     50.7
2      953.0    177.9
3     1005.2     65.0
4      821.0      4.4
..       …       …
110   1259.4      1.1
111    990.9     19.2
112   1326.9    125.7
113    746.6     71.2
114    745.1     40.3

[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
```

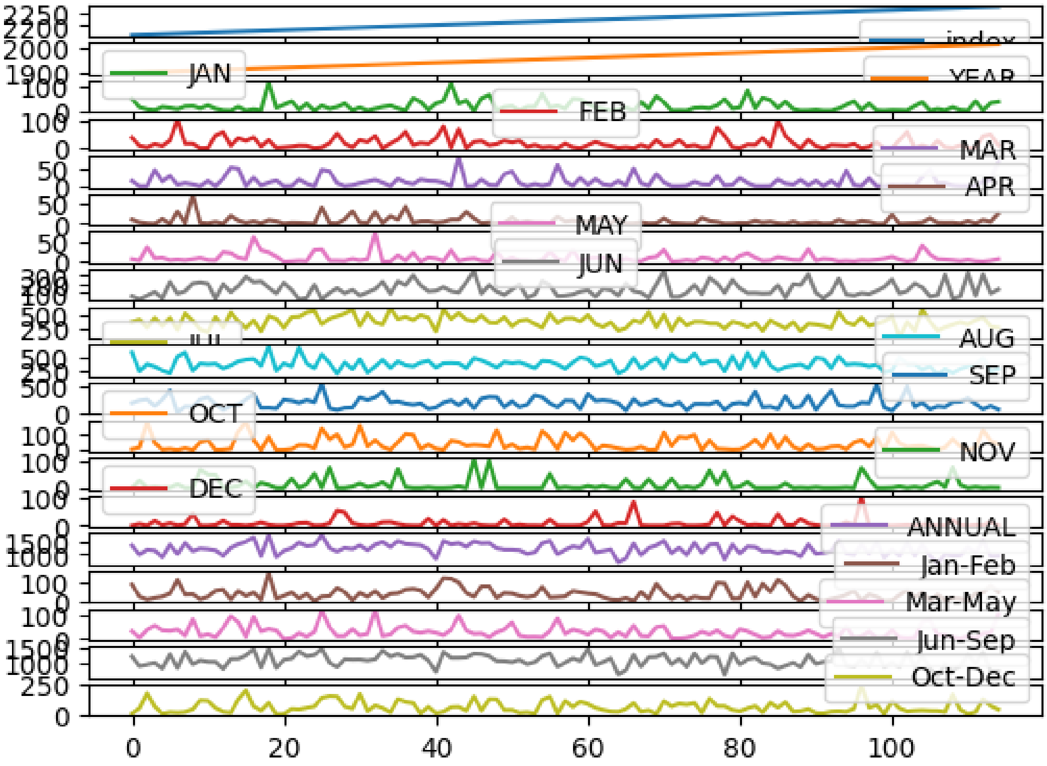
4 Line chart

[6]:

```
df.plot.line(subplots=True)
```

[6]:

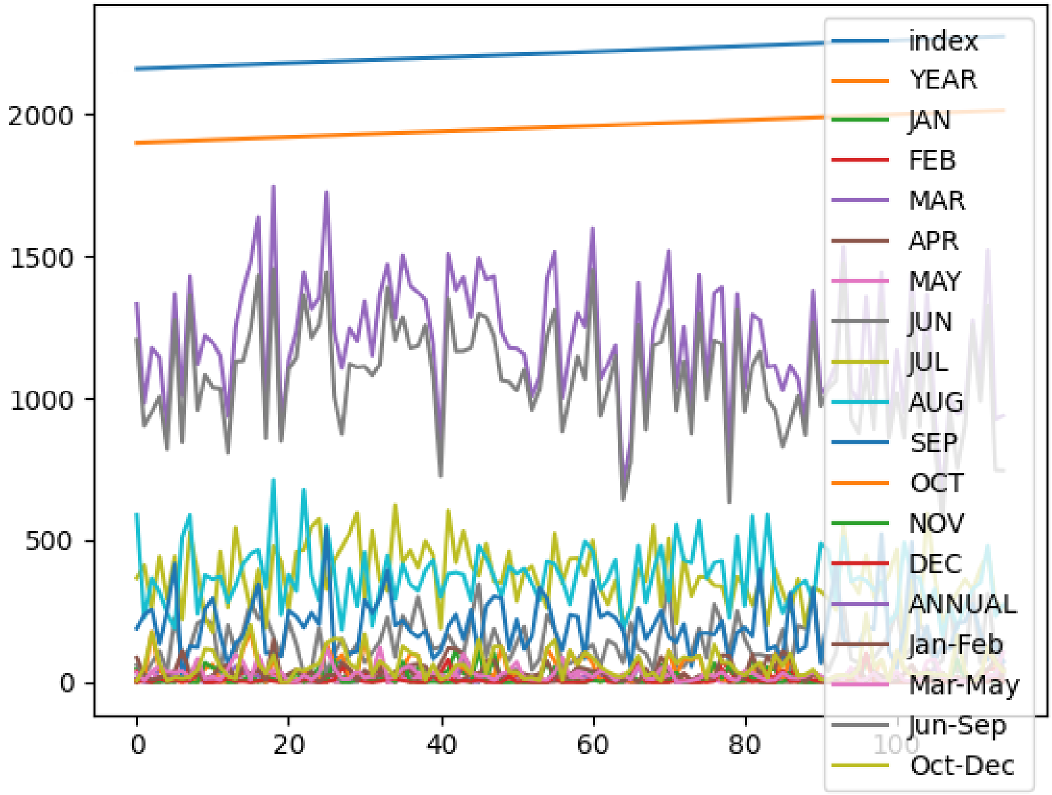
```
array([<Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
       <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
       <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
       <Axes: >], dtype=object)
```



5 Line chart

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

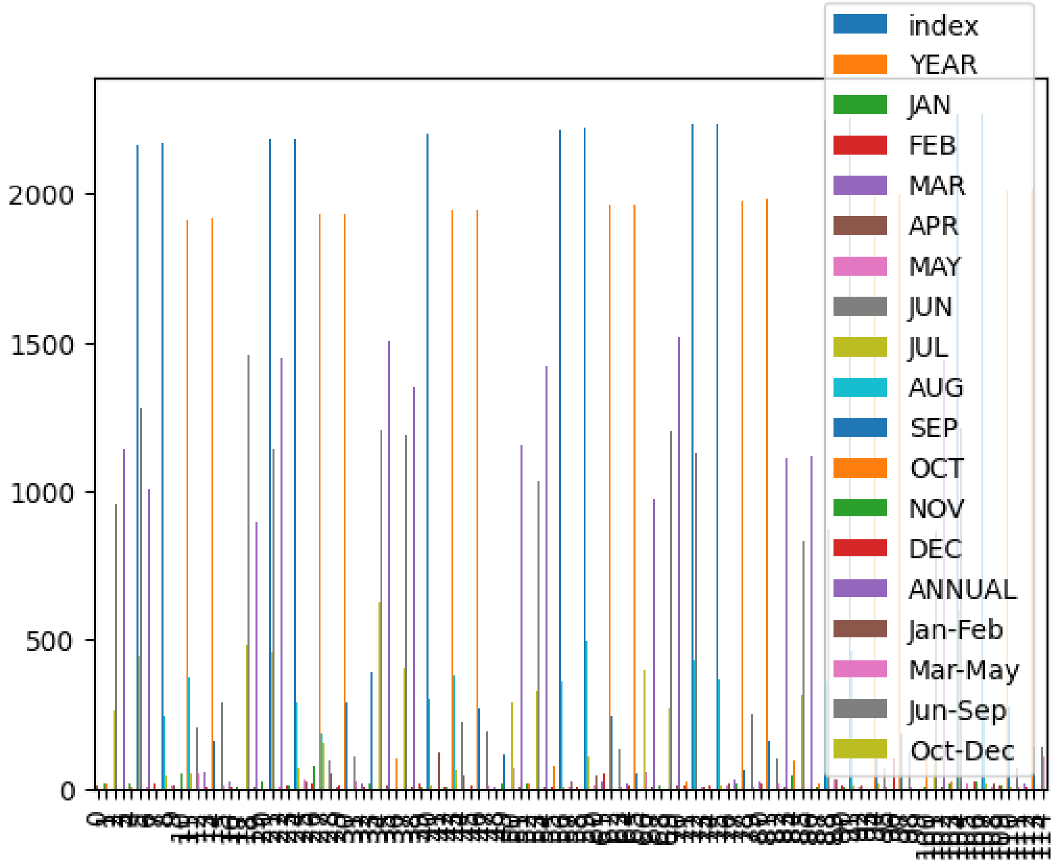
[7]:
```
<Axes: >
```



6 Bar chart

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

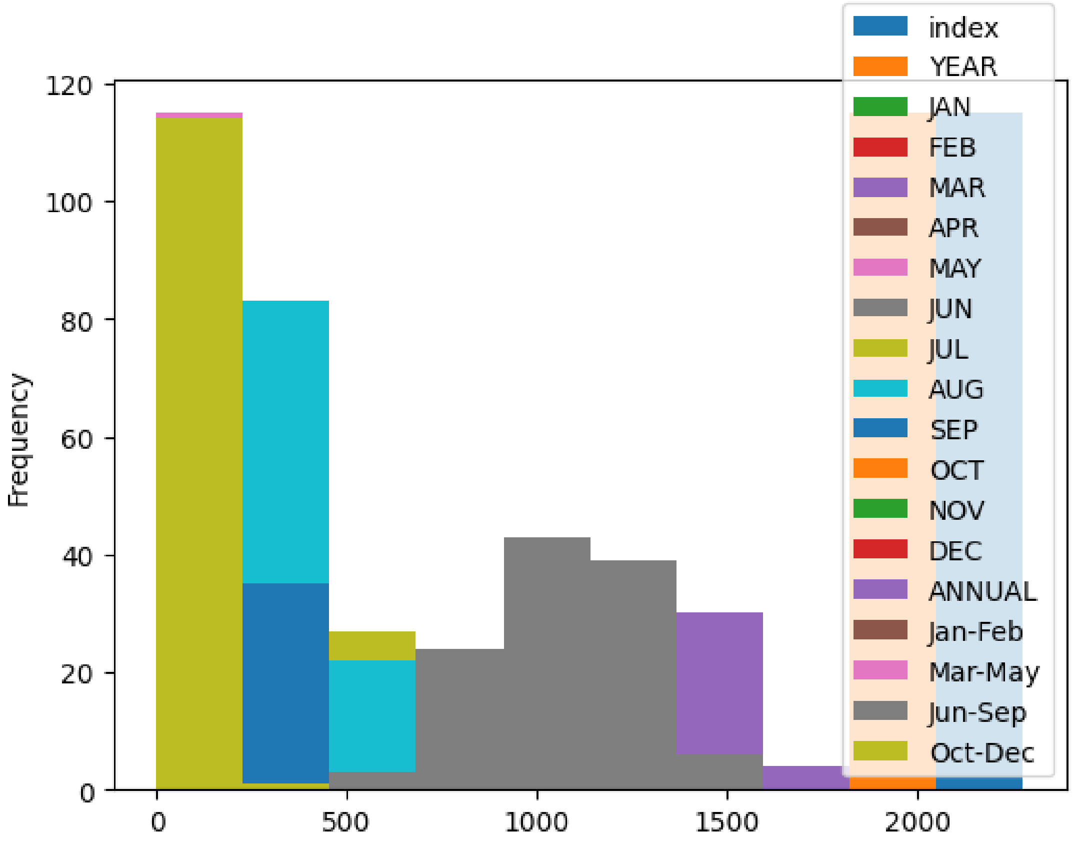
[8]:
```
<Axes: >
```



7 Histogram

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

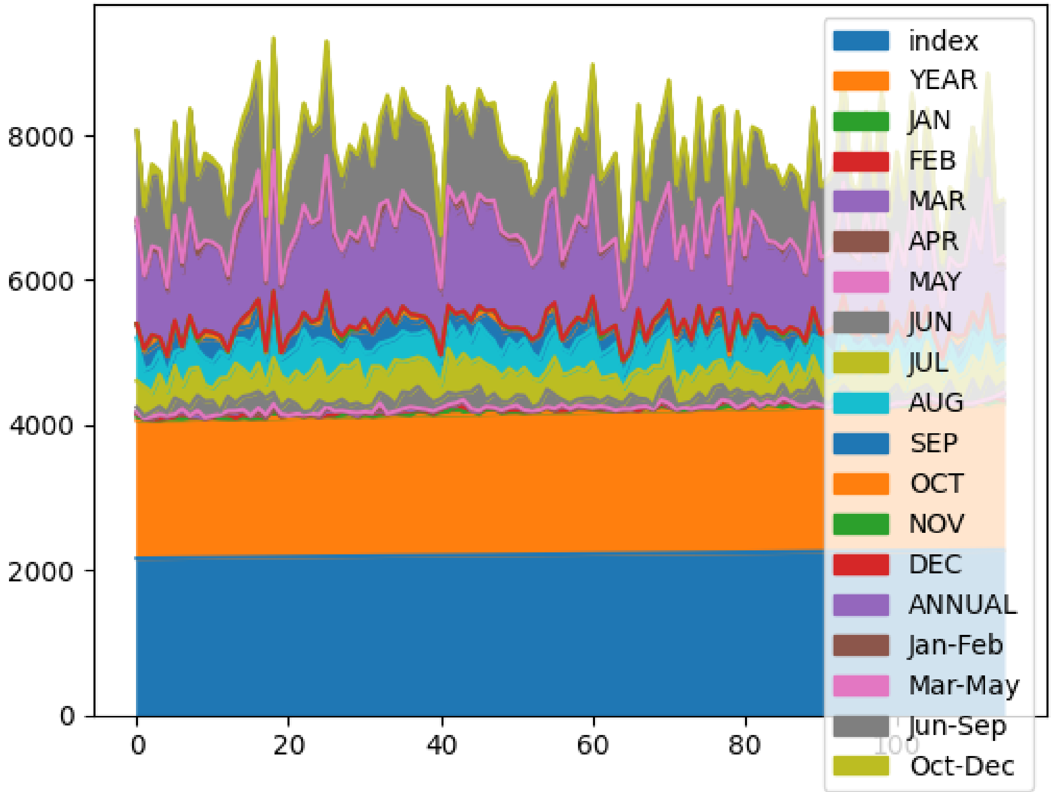
```
[9]: <Axes: ylabel='Frequency'>
```



8 Area chart

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

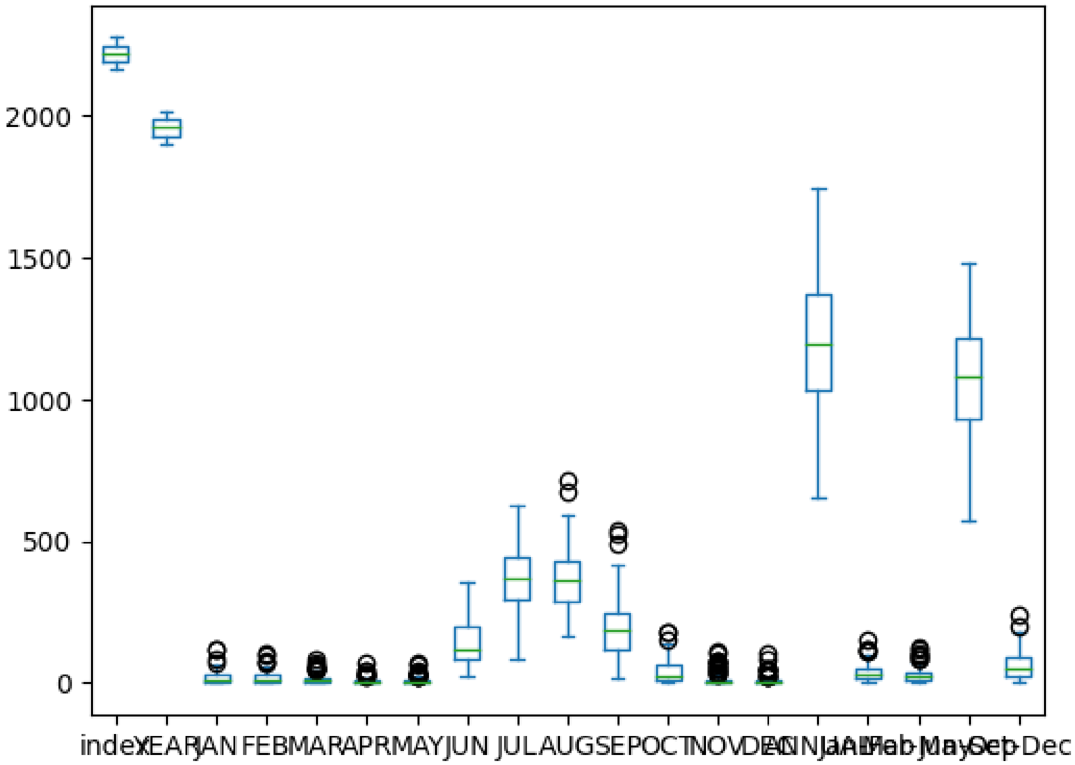
[10]:
```
<Axes: >
```



9 Box chart

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

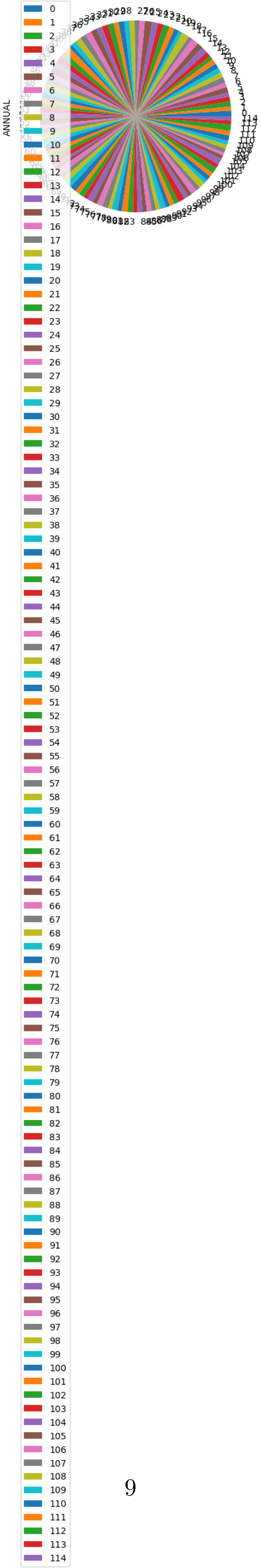
[11]:
```
<Axes: >
```



10 Pie chart

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

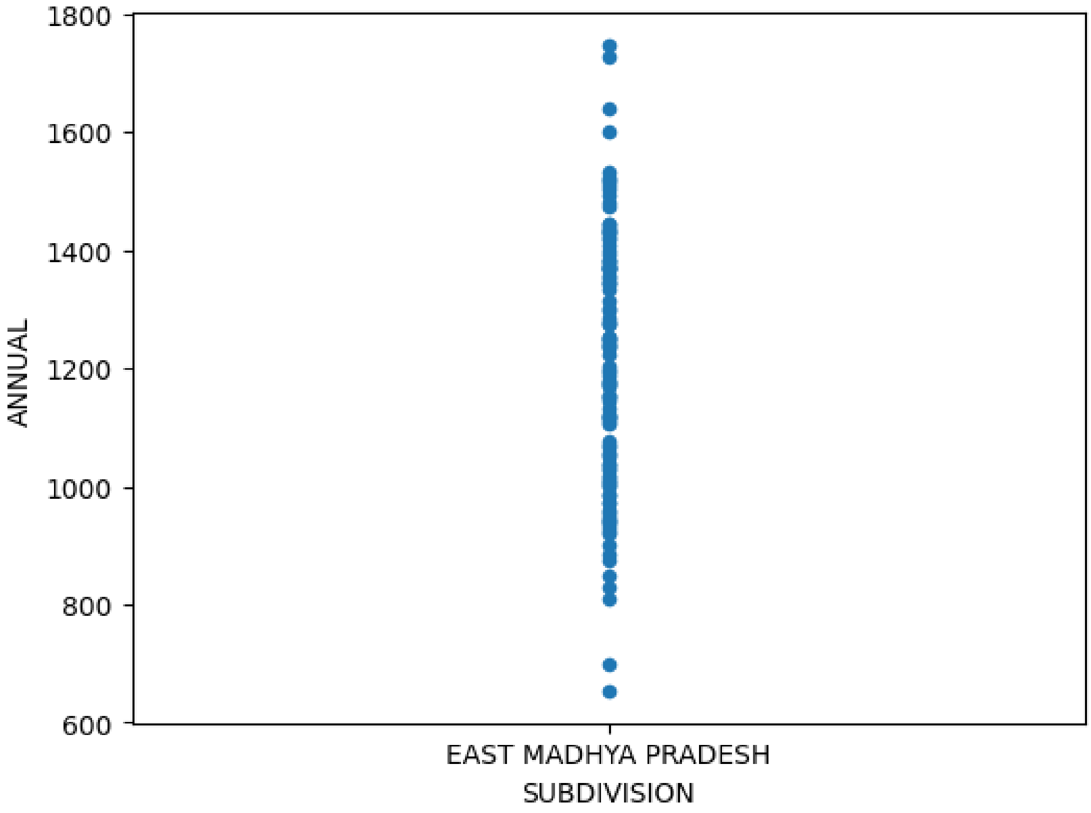
```
[12]: <Axes: ylabel='ANNUAL'>
```


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')
```

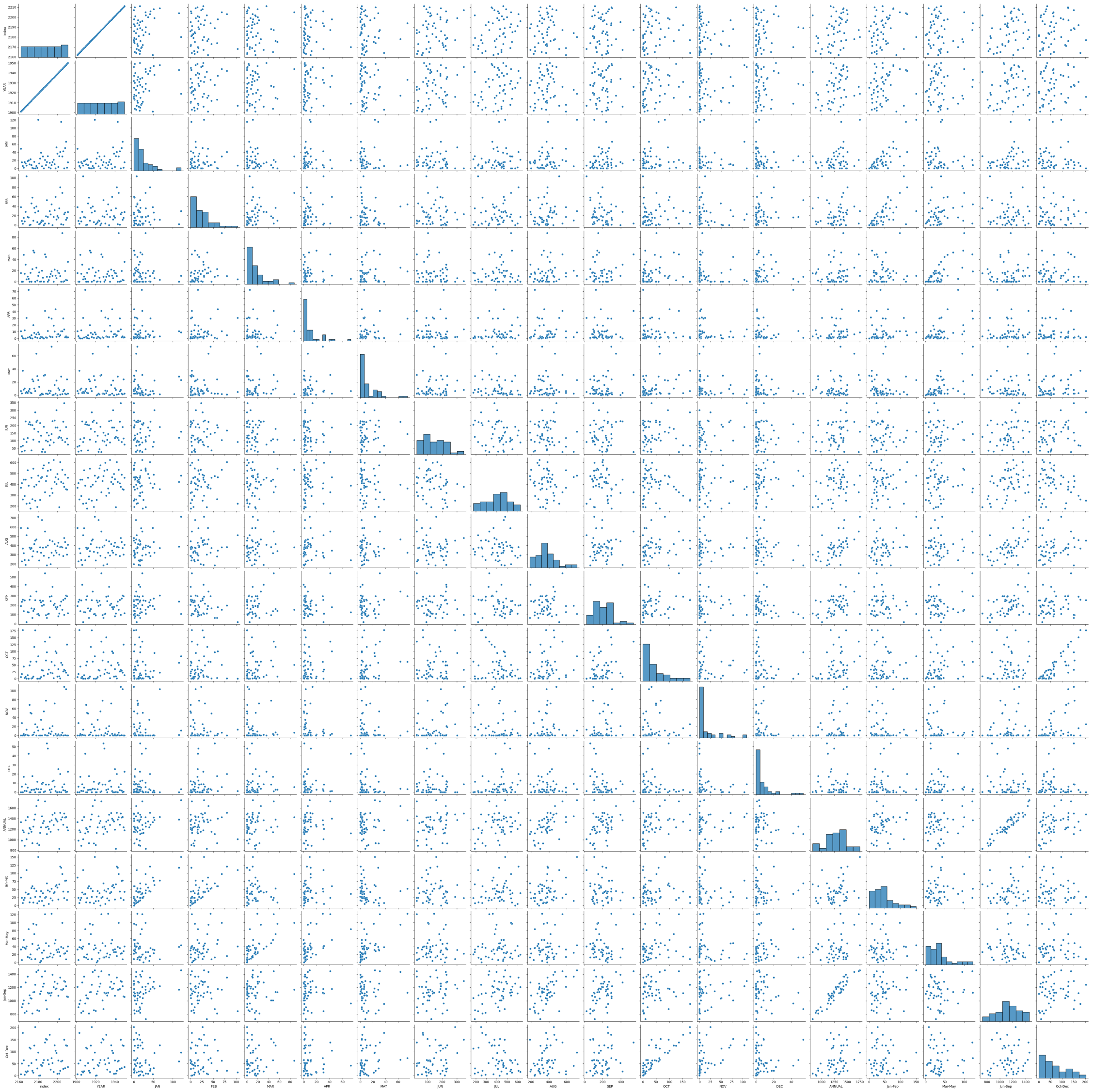
[13]:
```
<Axes: xlabel='SUBDIVISION', ylabel='ANNUAL'>
```



12 Seaborn

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

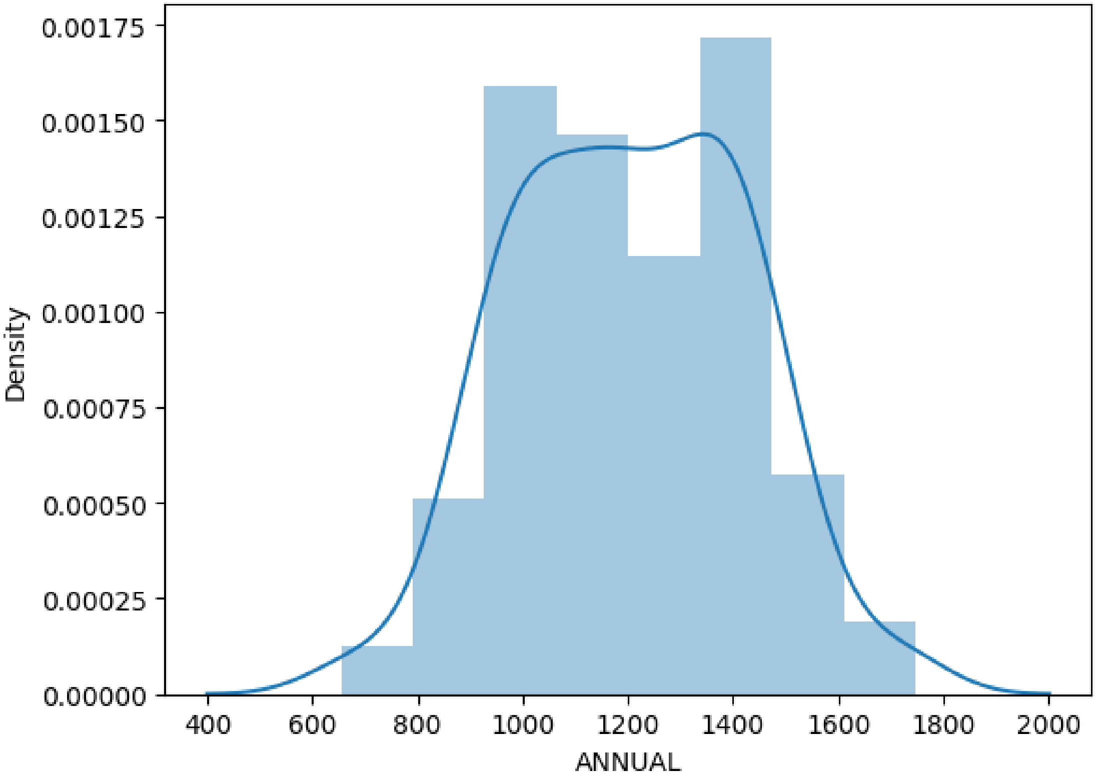
[14]:
```
<seaborn.axisgrid.PairGrid at 0x7c0cb3427b50>
```



```
[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'>

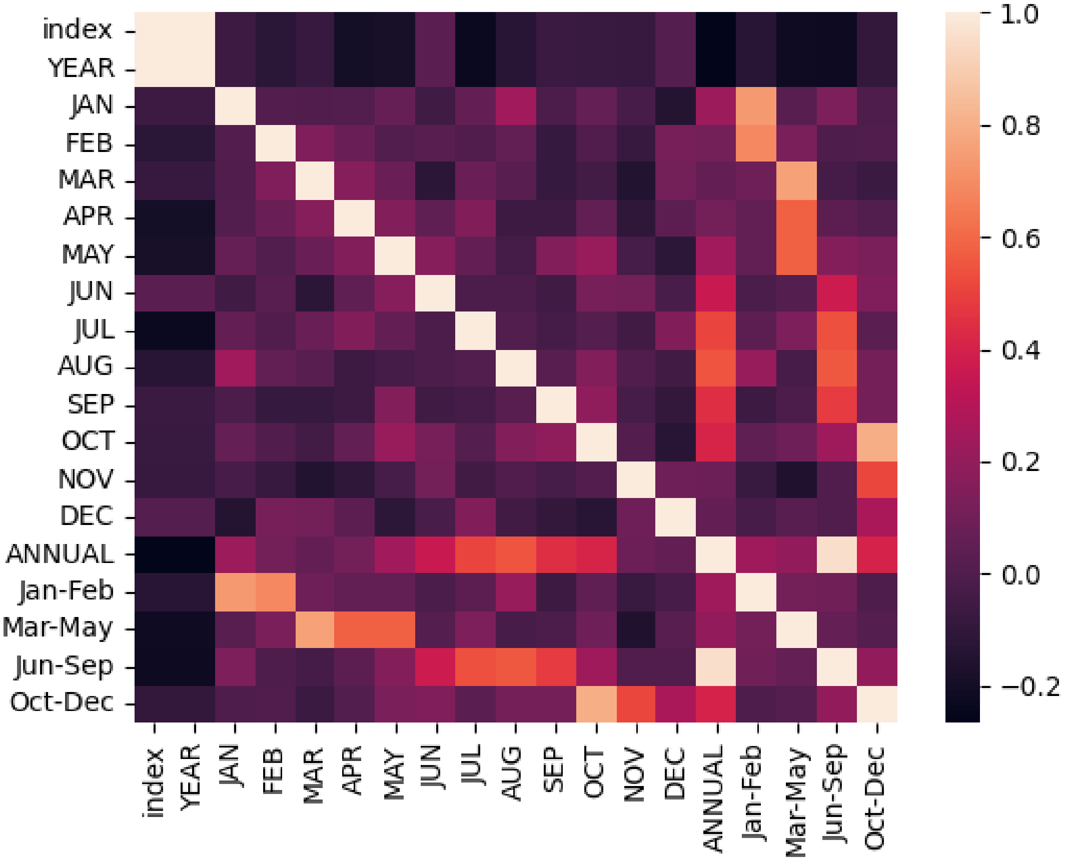


[16]:

```
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