zmgvnuyhi

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
```

[3]:

```
from google.colab import drive
drive.mount('/content/drive')
df=pd.read_csv("/content/drive/MyDrive/mydatasets/rainfall/rainfall_andaman &␣
 ↪nicobar islands.csv")
df
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call
drive.mount("/content/drive", force_remount=True).
```

[3]:

```
     index                SUBDIVISION  YEAR    JAN    FEB    MAR    APR  \
0        0  ANDAMAN & NICOBAR ISLANDS  1901   49.2   87.1   29.2    2.3
1        1  ANDAMAN & NICOBAR ISLANDS  1902    0.0  159.8   12.2    0.0
2        2  ANDAMAN & NICOBAR ISLANDS  1903   12.7  144.0    0.0    1.0
3        3  ANDAMAN & NICOBAR ISLANDS  1904    9.4   14.7    0.0  202.4
4        4  ANDAMAN & NICOBAR ISLANDS  1905    1.3    0.0    3.3   26.9
..     …                        …   …    …    …    …    …
105    105  ANDAMAN & NICOBAR ISLANDS  2011  265.9   84.8  272.8  111.4
106    106  ANDAMAN & NICOBAR ISLANDS  2012  119.9   45.6   30.9   55.8
107    107  ANDAMAN & NICOBAR ISLANDS  2013   67.1   37.6   43.0   46.3
108    108  ANDAMAN & NICOBAR ISLANDS  2014   41.9    8.6    0.0   11.1
109    109  ANDAMAN & NICOBAR ISLANDS  2015  126.8    7.6    3.1  138.2

       MAY    JUN    JUL    AUG    SEP    OCT    NOV    DEC  ANNUAL  Jan-Feb  \
0    528.8  517.5  365.1  481.1  332.6  388.5  558.2   33.6  3373.2    136.3
1    446.1  537.1  228.9  753.7  666.2  197.2  359.0  160.5  3520.7    159.8
2    235.1  479.9  728.4  326.7  339.0  181.2  284.4  225.0  2957.4    156.7
3    304.5  495.1  502.0  160.1  820.4  222.2  308.7   40.1  3079.6     24.1
4    279.5  628.7  368.7  330.5  297.0  260.7   25.4  344.7  2566.7      1.3
```

```
..     …    …    …    …    …    …    …    …     …      …
105  326.5  383.2  583.2  441.5  757.1  212.3  150.8  238.5  3828.0     350.7
106  533.9  458.2  317.3  369.6  868.9  209.7  300.5  187.3  3497.6     165.6
107  509.3  777.0  564.8  336.7  473.6  455.8  354.2   92.3  3757.8     104.7
108  238.0  416.6  467.6  321.6  412.9  402.6  201.2  100.4  2622.4      50.5
109  331.9  346.4  328.9  480.0  523.3  252.1  236.3  129.9  2904.6     134.4

     Mar-May  Jun-Sep  Oct-Dec
0      560.3   1696.3    980.3
1      458.3   2185.9    716.7
2      236.1   1874.0    690.6
3      506.9   1977.6    571.0
4      309.7   1624.9    630.8
..       …      …      …
105    710.7   2165.0    601.6
106    620.5   2014.0    697.5
107    598.6   2152.1    902.4
108    249.1   1618.7    704.2
109    473.2   1678.6    618.4

[110 rows x 20 columns]
```

3 Data Cleaning and Data Preprocessing

[4]:

```
df=df.dropna()
```

[5]:

```
df.columns
```

[5]:

```
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')
```

[6]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 104 entries, 0 to 109
Data columns (total 20 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   index        104 non-null    int64
 1   SUBDIVISION  104 non-null    object
 2   YEAR         104 non-null    int64
 3   JAN          104 non-null    float64
 4   FEB          104 non-null    float64
 5   MAR          104 non-null    float64
 6   APR          104 non-null    float64
 7   MAY          104 non-null    float64
 8   JUN          104 non-null    float64
 9   JUL          104 non-null    float64
 10  AUG          104 non-null    float64
 11  SEP          104 non-null    float64
 12  OCT          104 non-null    float64
 13  NOV          104 non-null    float64
 14  DEC          104 non-null    float64
 15  ANNUAL       104 non-null    float64
 16  Jan-Feb      104 non-null    float64
 17  Mar-May      104 non-null    float64
 18  Jun-Sep      104 non-null    float64
 19  Oct-Dec      104 non-null    float64
dtypes: float64(17), int64(2), object(1)
memory usage: 17.1+ KB
```

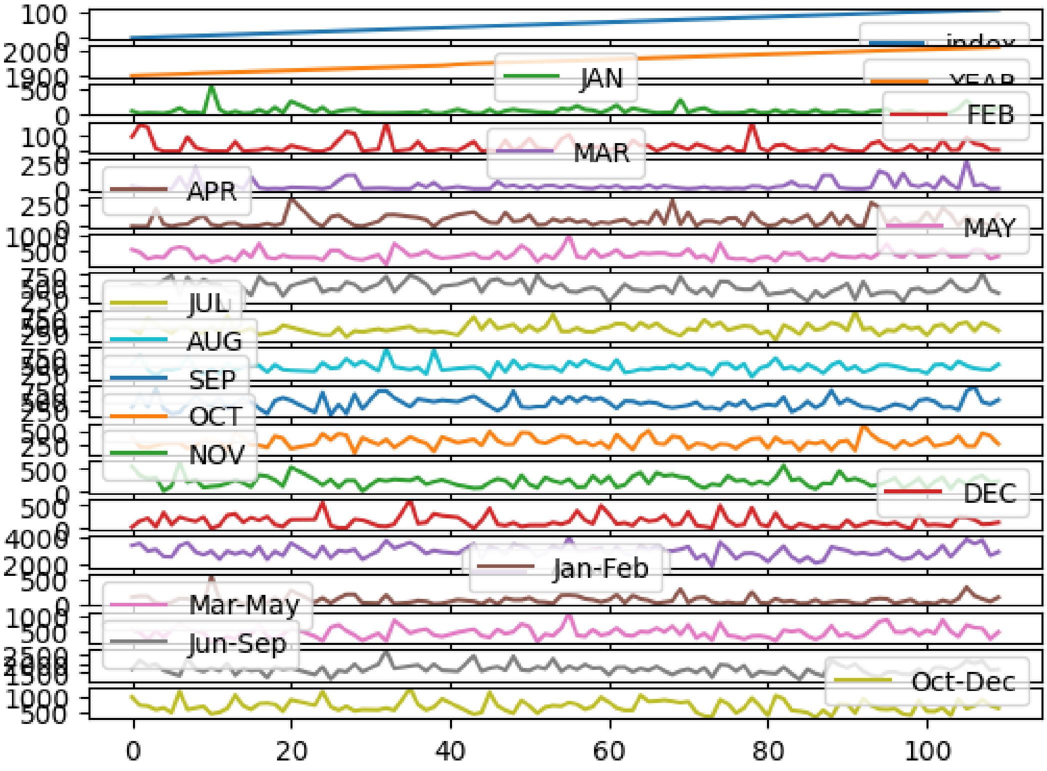
4 Line chart

[7]:

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

[7]:

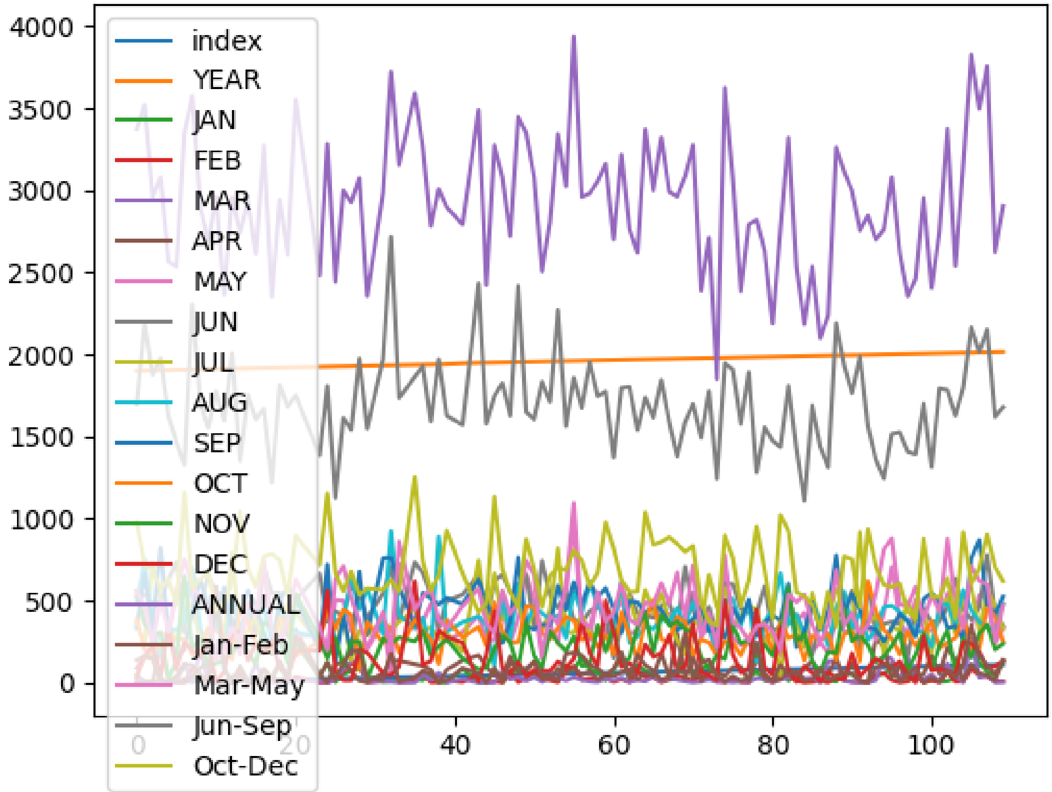
```
array([<Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
       <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
       <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >, <Axes: >,
       <Axes: >], dtype=object)
```



5 Line chart

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

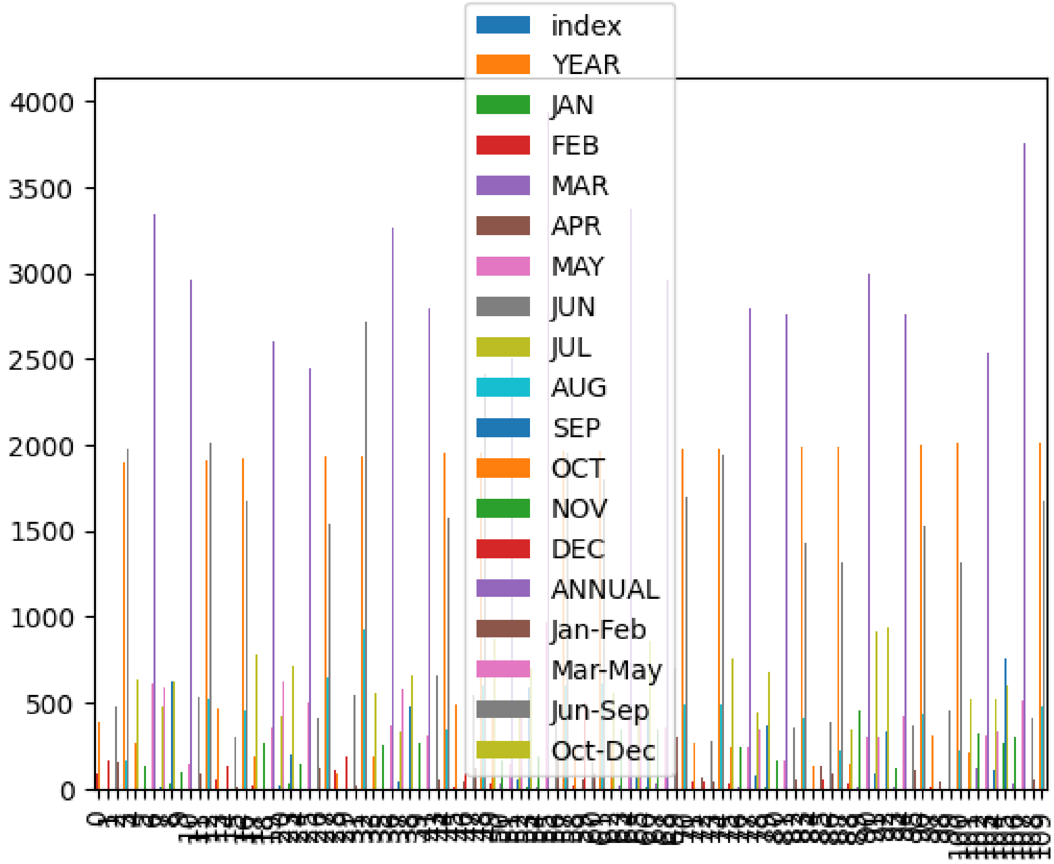
[8]:
```
<Axes: >
```



6 Bar chart

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

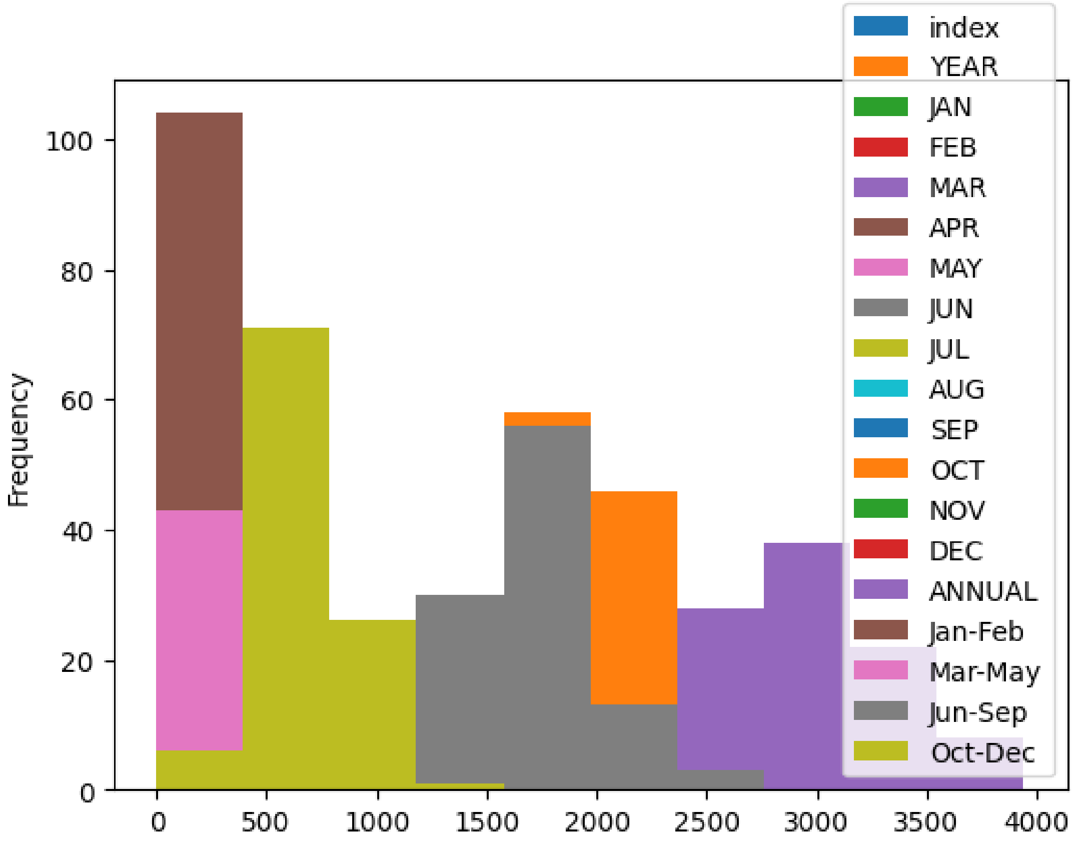
[9]:
```
<Axes: >
```



7 Histogram

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

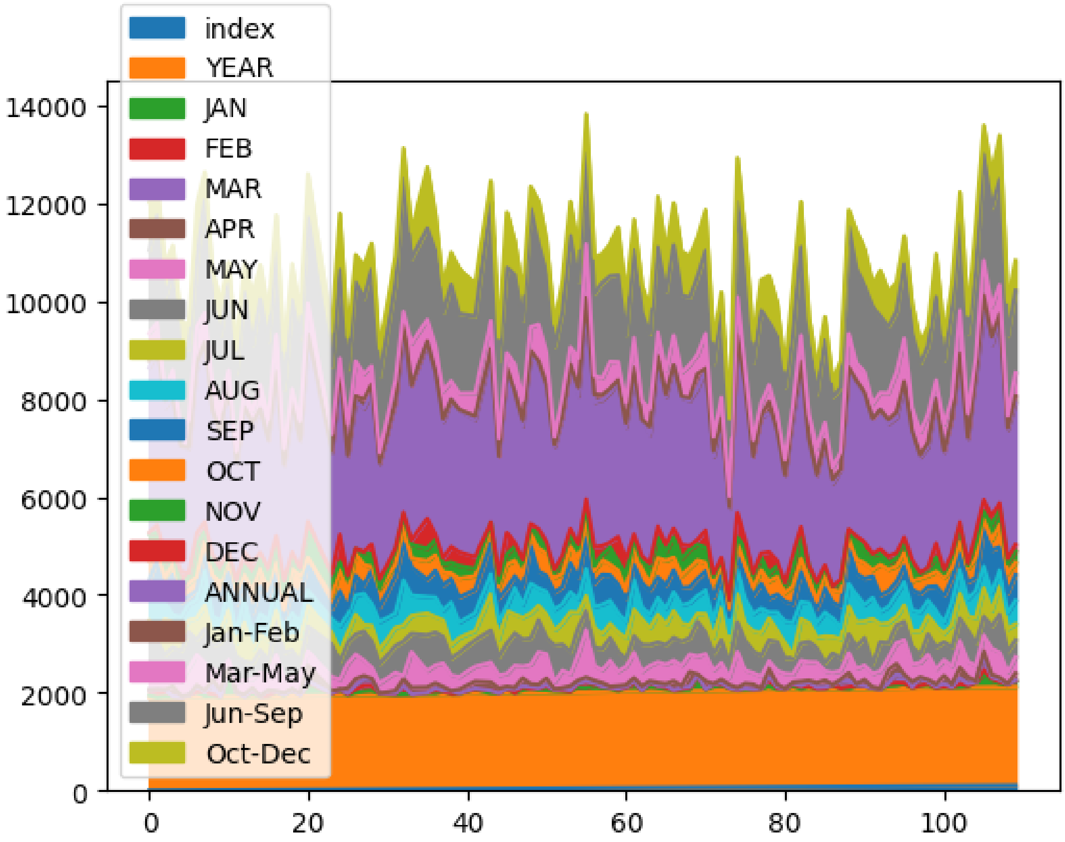
```
[10]: <Axes: ylabel='Frequency'>
```



8 Area chart

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

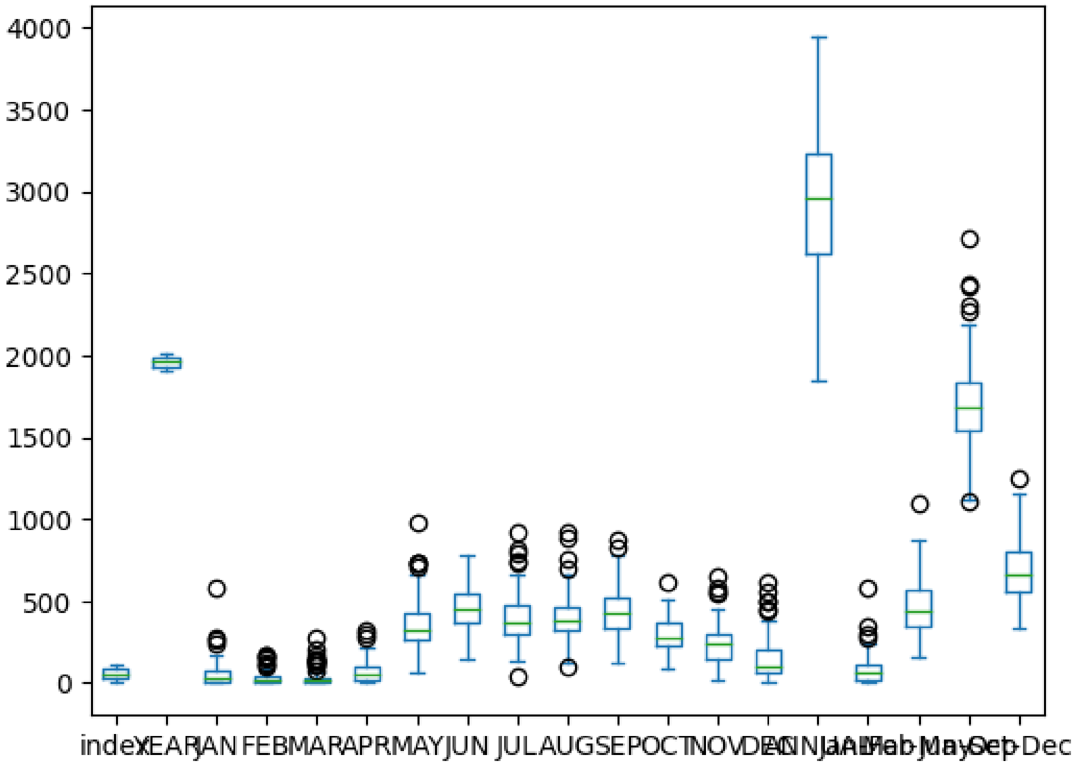
[11]:
```
<Axes: >
```



9 Box chart

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

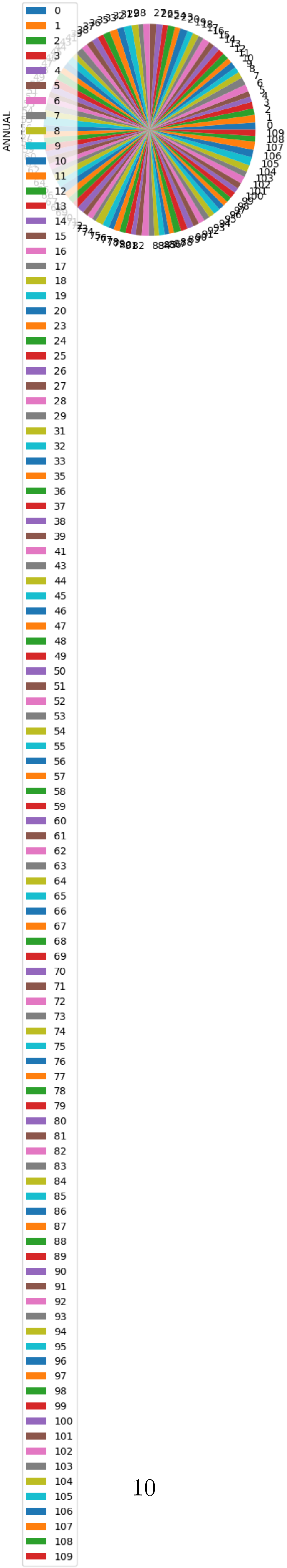
```
[12]: <Axes: >
```



10 Pie chart

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

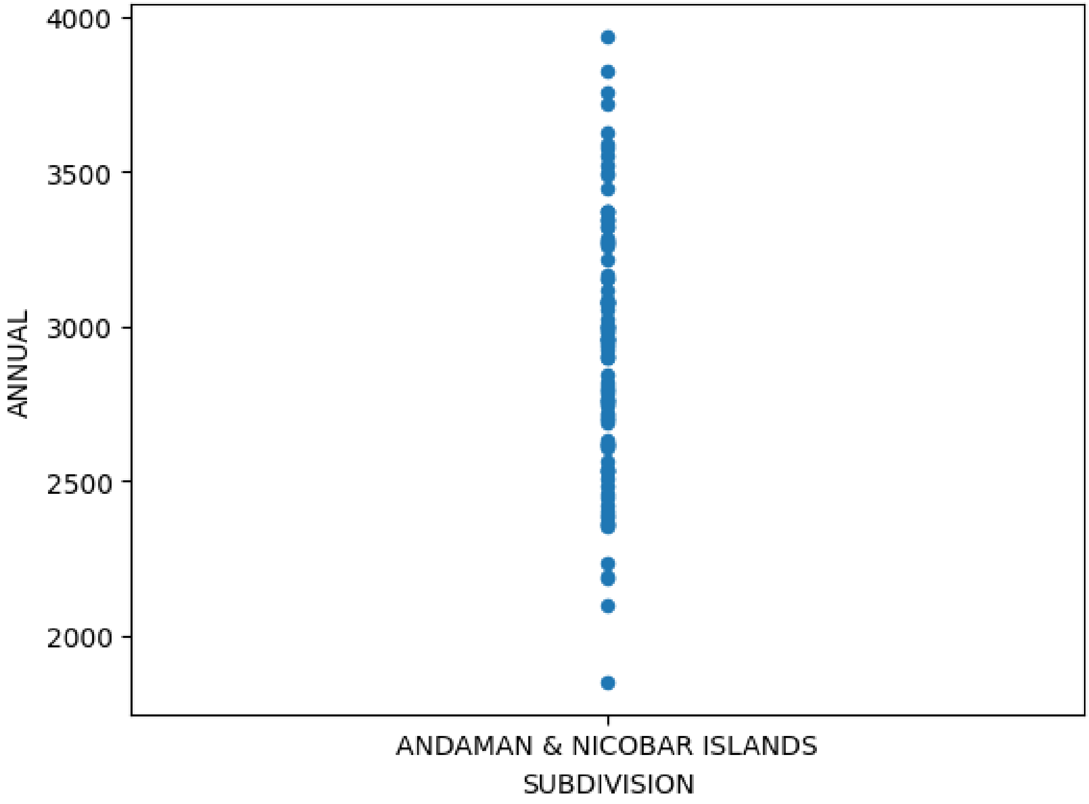
```
[14]: <Axes: ylabel='ANNUAL'>
```


ANNUAL
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
23
24
25
26
27
28
29
31
32
33
35
36
37
38
39
41
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

11 Scatter chart

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

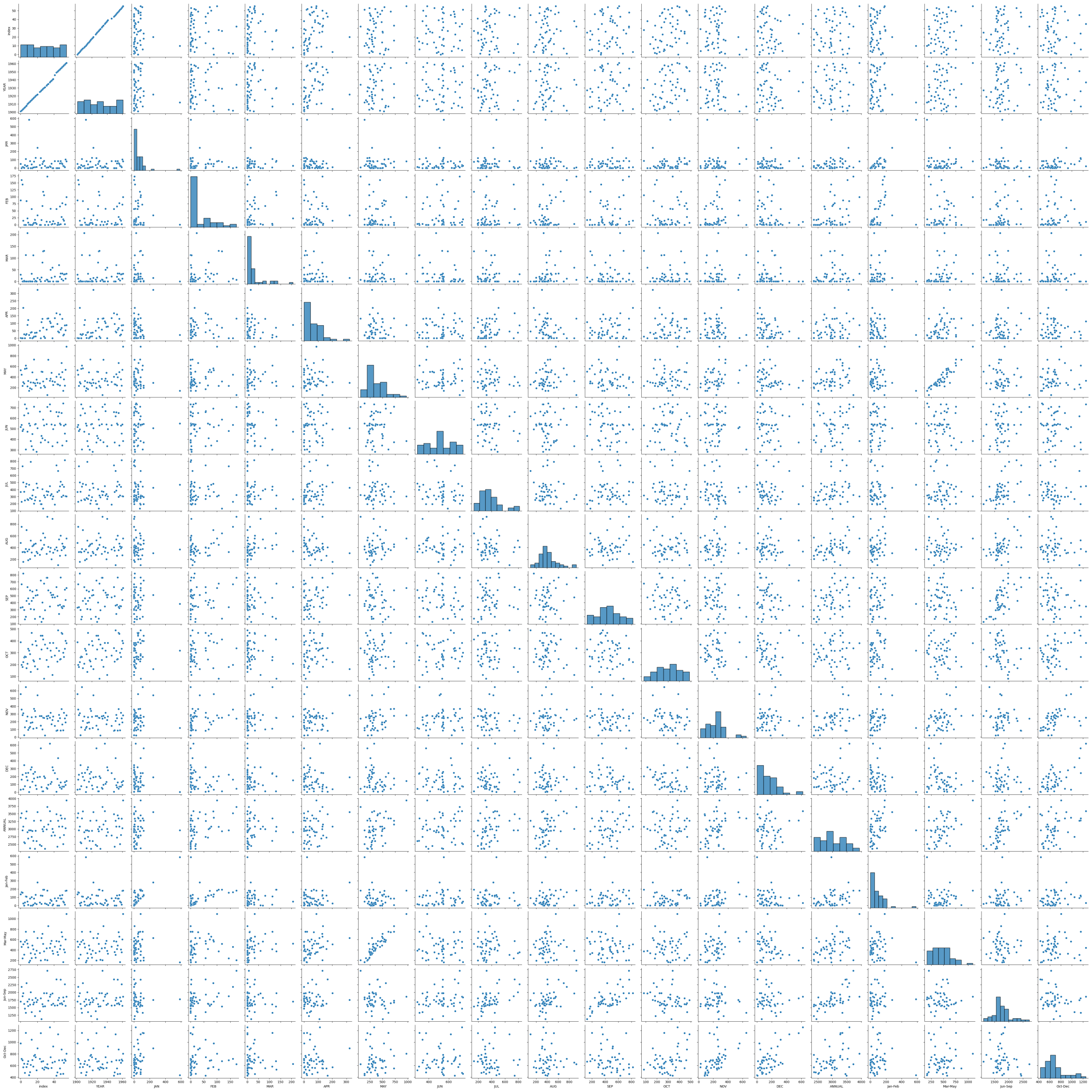
```
[17]: <Axes: xlabel='SUBDIVISION', ylabel='ANNUAL'>
```



12 Seaborn

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

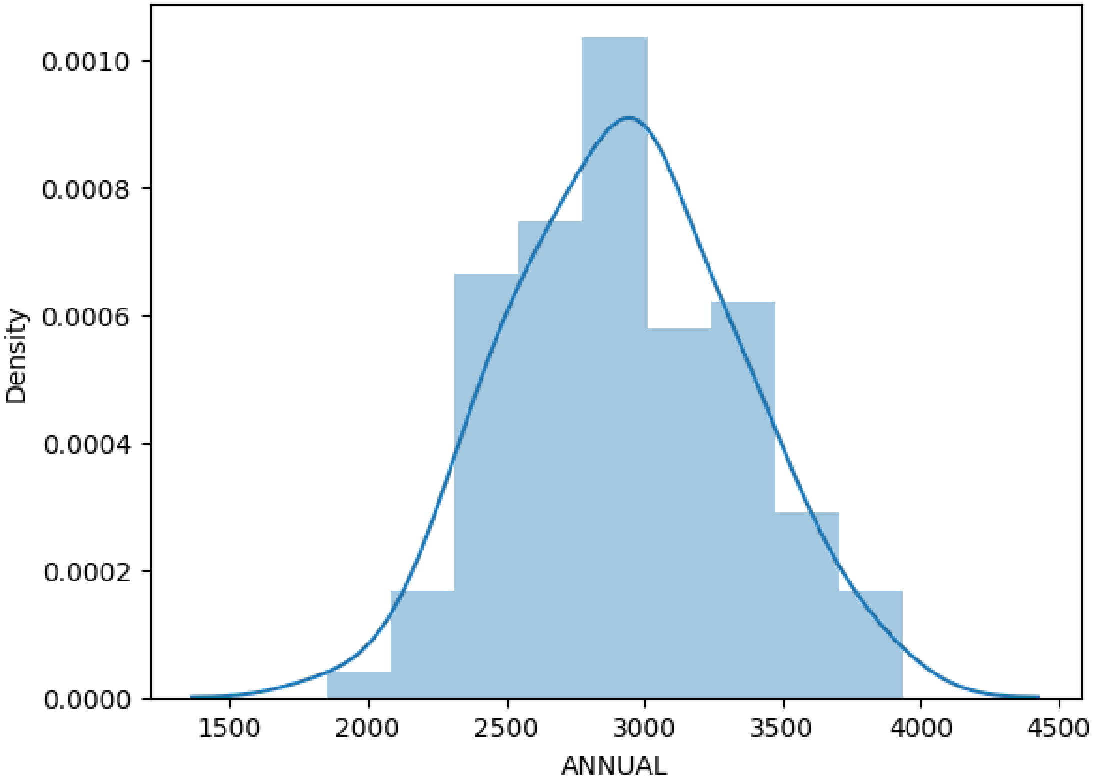
```
[21]: <seaborn.axisgrid.PairGrid at 0x7f72275de2c0>
```



[22]:
```
sns.distplot(df['ANNUAL'])
```

```
<ipython-input-22-5daa97052ca5>: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'])
```

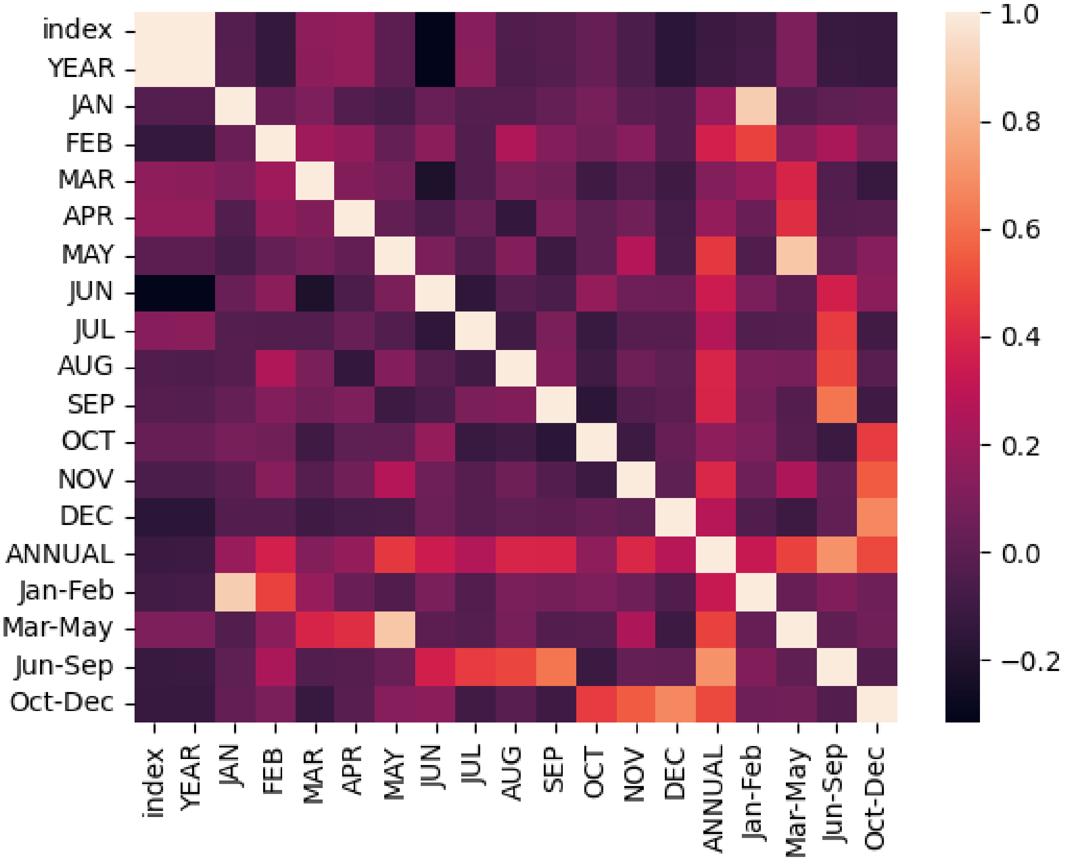
```
[22]: <Axes: xlabel='ANNUAL', ylabel='Density'>
```



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

```
<ipython-input-23-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())
```

```
[23]: <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