Importing Libraries ¶

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

Importing Datasets

In [2]:
```
df=pd.read_csv(r'C:\Users\user\Downloads\Rainfall\HARYANA DELHI   CHANDIGARH.csv
df
```

Out[2]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT
0	1357	HARYANA DELHI & CHANDIGARH	1901	35.4	28.9	11.1	0.0	5.1	13.2	126.4	151.5	10.5	2.0
1	1358	HARYANA DELHI & CHANDIGARH	1902	0.0	0.7	2.9	10.2	15.8	74.6	149.3	97.1	59.8	9.3
2	1359	HARYANA DELHI & CHANDIGARH	1903	14.7	0.5	2.3	0.5	8.5	8.6	151.6	138.2	97.7	4.0
3	1360	HARYANA DELHI & CHANDIGARH	1904	7.6	0.7	48.0	0.5	29.3	34.3	109.7	162.9	102.3	1.5
4	1361	HARYANA DELHI & CHANDIGARH	1905	44.8	20.8	14.0	1.3	7.4	20.1	93.6	23.1	92.6	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
110	1467	HARYANA DELHI & CHANDIGARH	2011	0.7	26.7	6.9	8.9	28.7	94.4	85.0	127.3	133.1	0.0
111	1468	HARYANA DELHI & CHANDIGARH	2012	8.2	0.2	0.1	11.8	3.8	5.3	68.1	196.6	90.7	2.4
112	1469	HARYANA DELHI & CHANDIGARH	2013	21.1	52.2	5.3	3.3	1.4	62.1	96.5	161.9	42.8	10.9
113	1470	HARYANA DELHI & CHANDIGARH	2014	13.0	17.3	26.8	7.5	20.3	25.9	72.3	34.8	67.3	10.5
114	1471	HARYANA DELHI & CHANDIGARH	2015	12.4	6.6	71.8	34.8	8.4	43.7	130.3	89.2	32.1	3.7

115 rows × 20 columns

Data Cleaning and Data Preprocessing

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

Out[3]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT
0	1357	HARYANA DELHI & CHANDIGARH	1901	35.4	28.9	11.1	0.0	5.1	13.2	126.4	151.5	10.5	2.0
1	1358	HARYANA DELHI & CHANDIGARH	1902	0.0	0.7	2.9	10.2	15.8	74.6	149.3	97.1	59.8	9.3
2	1359	HARYANA DELHI & CHANDIGARH	1903	14.7	0.5	2.3	0.5	8.5	8.6	151.6	138.2	97.7	4.0
3	1360	HARYANA DELHI & CHANDIGARH	1904	7.6	0.7	48.0	0.5	29.3	34.3	109.7	162.9	102.3	1.5
4	1361	HARYANA DELHI & CHANDIGARH	1905	44.8	20.8	14.0	1.3	7.4	20.1	93.6	23.1	92.6	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
110	1467	HARYANA DELHI & CHANDIGARH	2011	0.7	26.7	6.9	8.9	28.7	94.4	85.0	127.3	133.1	0.0
111	1468	HARYANA DELHI & CHANDIGARH	2012	8.2	0.2	0.1	11.8	3.8	5.3	68.1	196.6	90.7	2.4
112	1469	HARYANA DELHI & CHANDIGARH	2013	21.1	52.2	5.3	3.3	1.4	62.1	96.5	161.9	42.8	10.9
113	1470	HARYANA DELHI & CHANDIGARH	2014	13.0	17.3	26.8	7.5	20.3	25.9	72.3	34.8	67.3	10.5
114	1471	HARYANA DELHI & CHANDIGARH	2015	12.4	6.6	71.8	34.8	8.4	43.7	130.3	89.2	32.1	3.7

115 rows × 20 columns

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

Out[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')
```
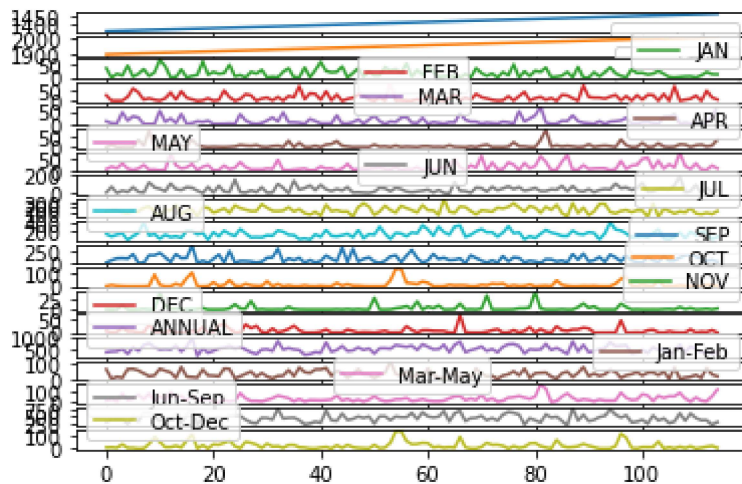
In [5]: `df.info()`

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 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: 18.9+ KB
```

Line Chart

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

```
Out[6]: array([<AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
                <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
                <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
                <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
                <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>], dtype=object)
```
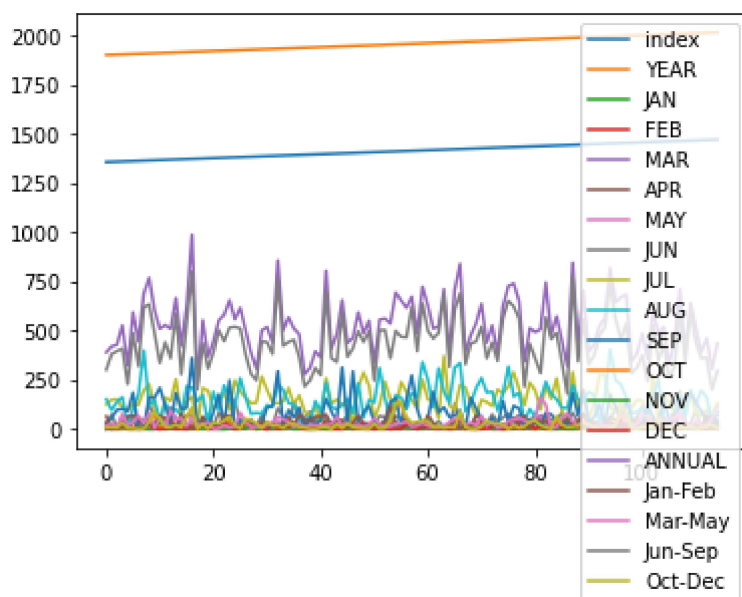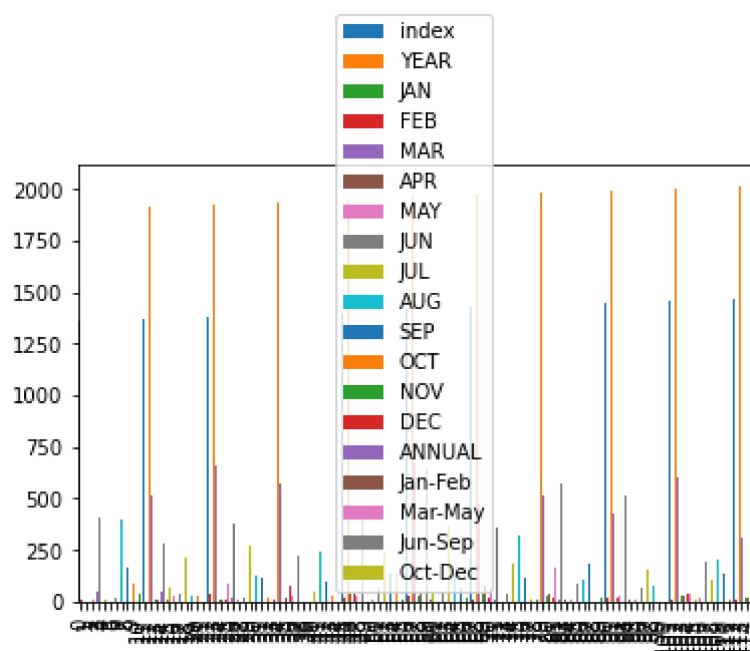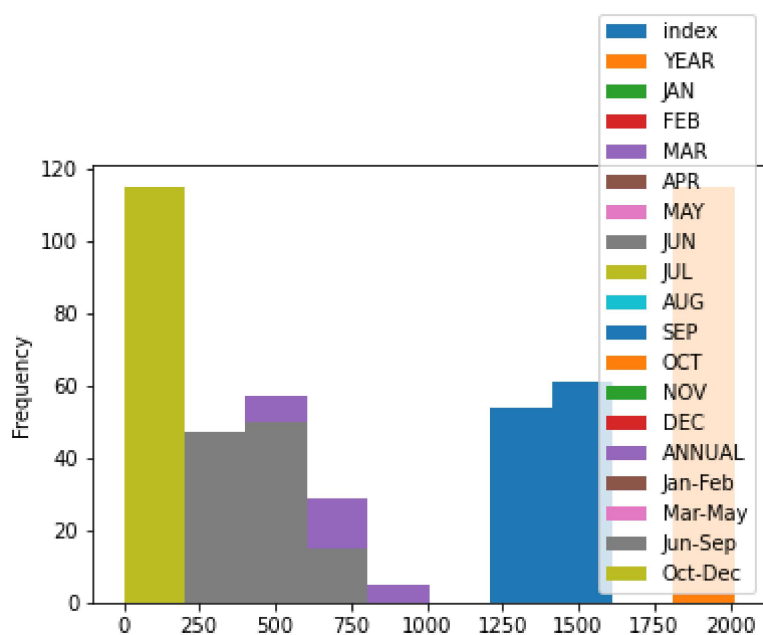
```
In [7]: df.plot.line()
```

```
Out[7]: <AxesSubplot:>
```

Bar Chart

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

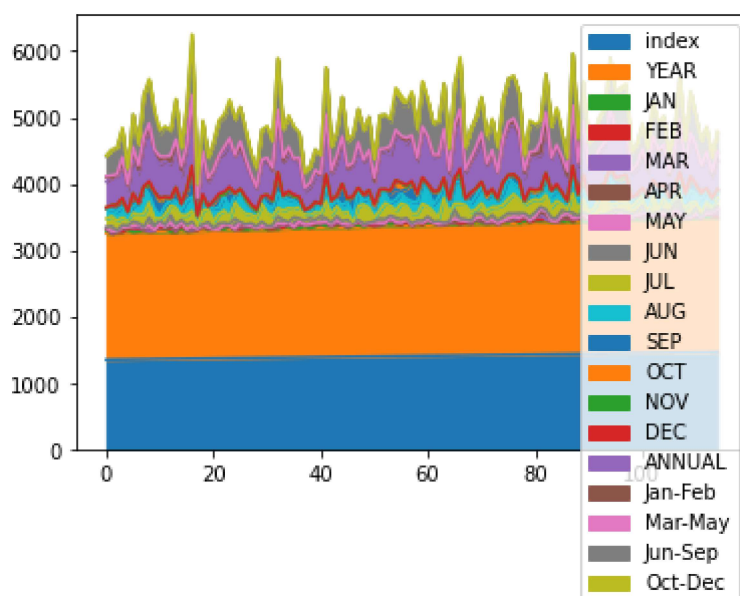Out[8]: `<AxesSubplot:>`

Histogram

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

Out[9]: `<AxesSubplot:ylabel='Frequency'>`
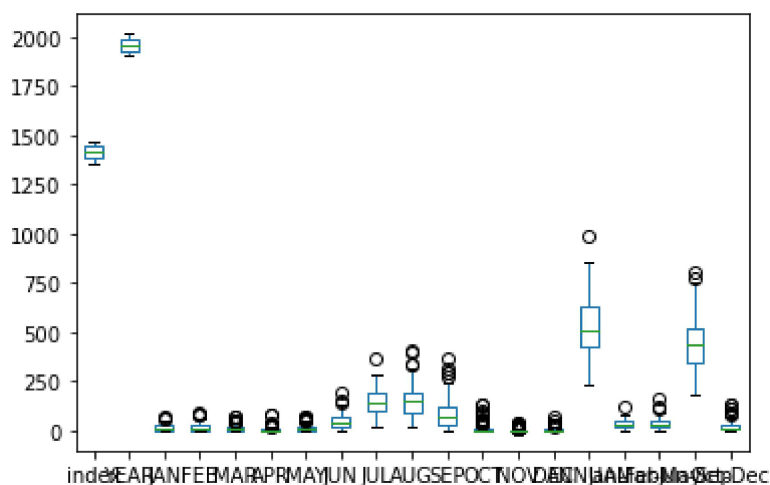

Area Chart

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

Out[10]: `<AxesSubplot:>`
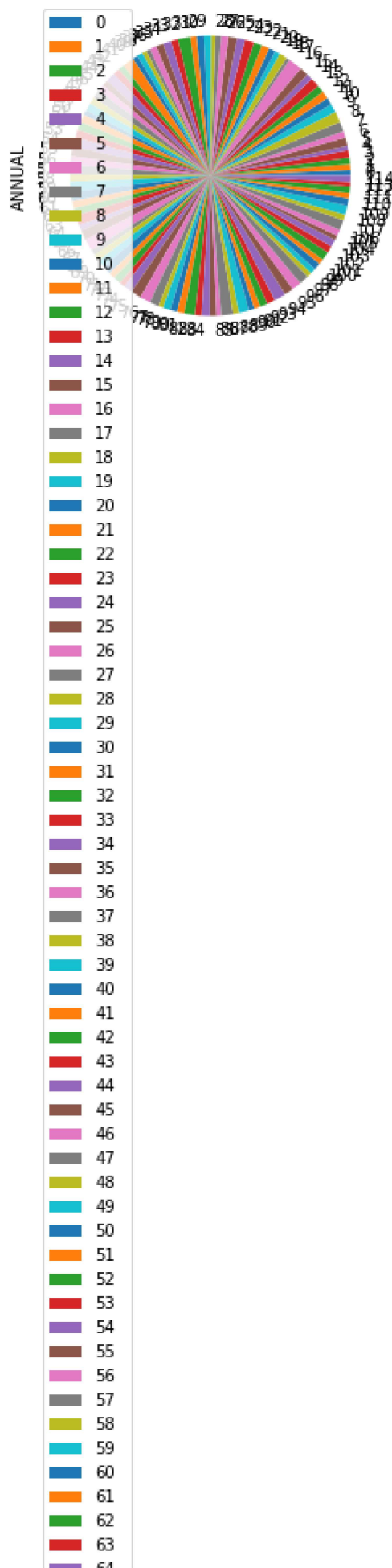

Box Chart

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

Out[11]: `<AxesSubplot:>`

Pie Chart

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

Out[12]: `<AxesSubplot:ylabel='ANNUAL'>`

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
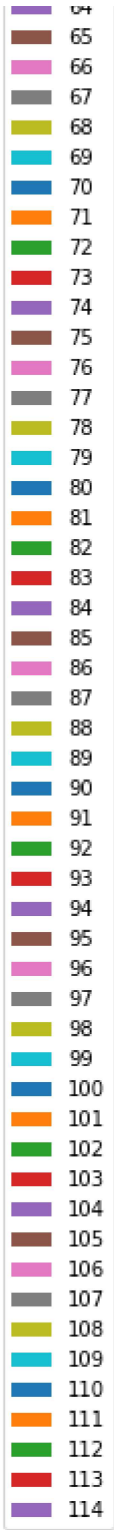
Scatter Plot

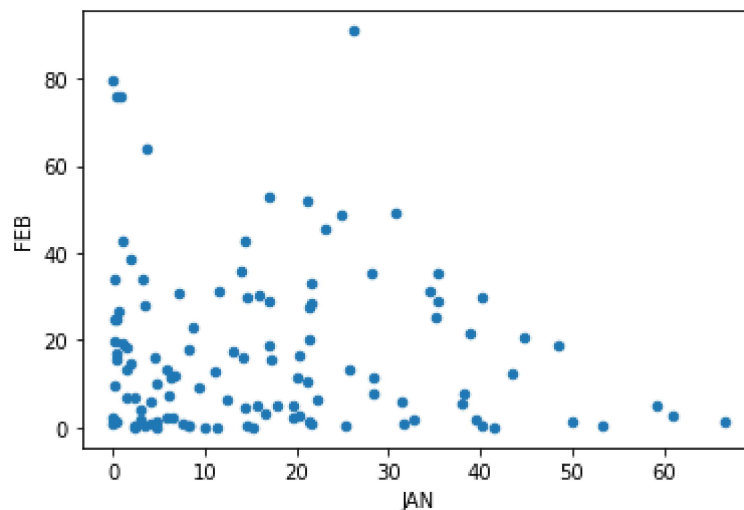In [13]: `df.plot.scatter(x='JAN',y='FEB')`

Out[13]: `<AxesSubplot:xlabel='JAN', ylabel='FEB'>`

In [14]: `df.info()`

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 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: 18.9+ KB
```

In [15]: `df.describe()`

Out[15]:

	index	YEAR	JAN	FEB	MAR	APR	MAY	11
count	115.000000	115.000000	115.000000	115.000000	115.000000	115.000000	115.000000	11!
mean	1414.000000	1958.000000	16.889565	17.433913	12.935652	7.633913	14.533913	4i
std	33.341666	33.341666	15.514478	18.893422	15.251840	12.847533	15.900347	3∢
min	1357.000000	1901.000000	0.000000	0.000000	0.000000	0.000000	0.000000	:
25%	1385.500000	1929.500000	3.550000	2.250000	2.100000	0.800000	3.700000	2∢
50%	1414.000000	1958.000000	14.300000	12.100000	7.200000	2.800000	7.900000	4:
75%	1442.500000	1986.500000	25.150000	27.850000	17.700000	8.750000	20.700000	6(
max	1471.000000	2015.000000	66.500000	91.000000	71.800000	82.500000	72.900000	19:

EDA And Visualization

In [16]: `sns.pairplot(df)`

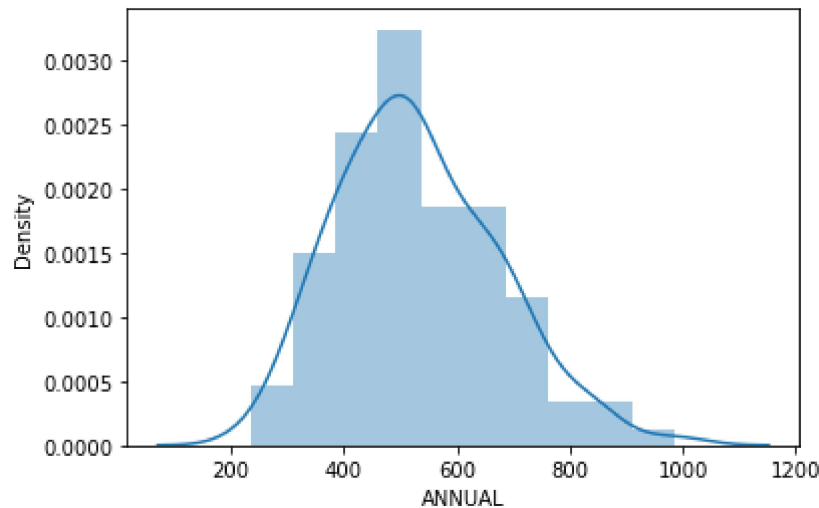Out[16]: `<seaborn.axisgrid.PairGrid at 0x170faad0eb0>`
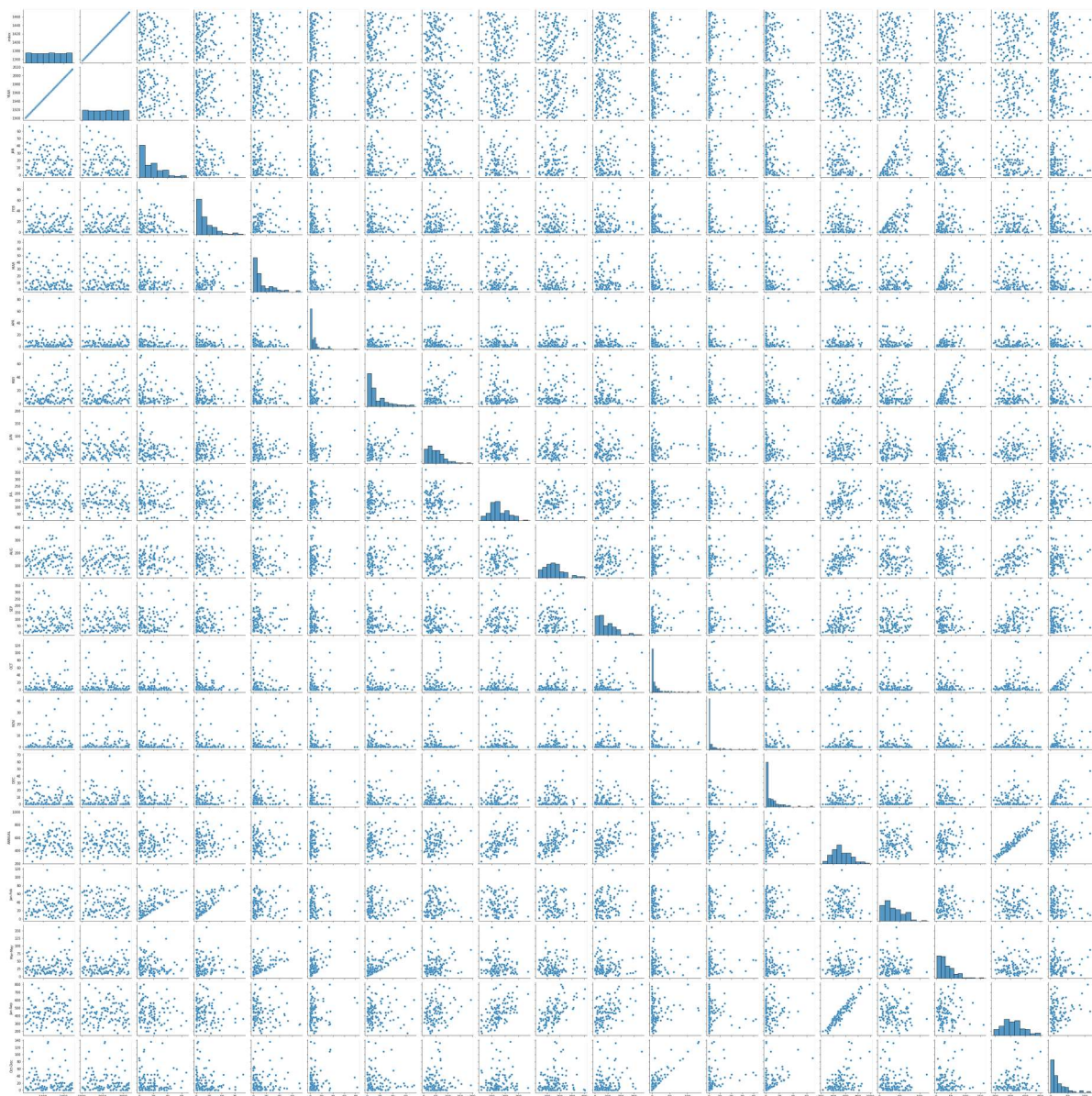
In [17]: ```python
sns.distplot(df['ANNUAL'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2557: Fut
ureWarning: `distplot` is a deprecated function and will be removed in a futu
re version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for hi
stograms).
  warnings.warn(msg, FutureWarning)
```

Out[17]: `<AxesSubplot:xlabel='ANNUAL', ylabel='Density'>`

In [18]: ```python
sns.heatmap(df.corr())
```

Out[18]: `<AxesSubplot:>`