

## Importing Libraries

In [1]:

```
1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4 import seaborn as sns
```

## Importing Datasets

In [2]:

```

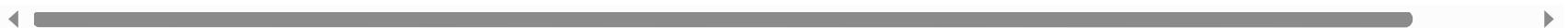
1 df=pd.read_csv('COASTAL ANDHRA PRADESH.csv')
2 df

```

Out[2]:

		index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL	Jan-Feb	Mar-May	June
0	3083		COASTAL ANDHRA PRADESH	1902	2.0	0.0	2.8	23.9	37.6	72.6	144.5	236.1	204.5	262.0	50.4	27.1	1063.6	2.0	64.4	6
1	3084		COASTAL ANDHRA PRADESH	1903	0.8	13.3	0.2	6.2	73.4	154.0	248.6	258.0	216.5	159.1	173.9	12.1	1316.2	14.2	79.8	8
2	3085		COASTAL ANDHRA PRADESH	1904	1.3	0.0	5.4	3.0	136.3	107.8	120.2	117.7	116.8	240.9	0.0	10.7	860.2	1.3	144.7	4
3	3086		COASTAL ANDHRA PRADESH	1905	1.1	16.7	68.0	37.0	68.8	84.4	64.6	210.8	170.2	66.0	7.4	0.0	795.2	17.8	173.8	5
4	3087		COASTAL ANDHRA PRADESH	1906	3.9	23.5	9.9	2.3	11.0	252.6	155.8	241.1	126.9	92.1	21.5	80.8	1021.5	27.4	23.2	7
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
109	3192		COASTAL ANDHRA PRADESH	2011	0.0	17.9	0.9	62.3	67.9	86.8	196.0	215.8	129.7	74.6	4.9	5.0	861.9	17.9	131.2	6
110	3193		COASTAL ANDHRA PRADESH	2012	37.6	0.0	2.7	24.0	39.3	95.4	221.9	221.2	246.5	140.0	289.7	0.0	1318.4	37.6	66.1	7
111	3194		COASTAL ANDHRA PRADESH	2013	2.0	29.6	0.2	48.0	28.2	127.5	162.4	123.1	132.0	411.5	53.1	2.8	1120.5	31.7	76.4	5
112	3195		COASTAL ANDHRA PRADESH	2014	0.4	1.2	9.1	6.0	112.9	45.7	151.8	177.8	144.5	195.6	23.7	6.4	874.9	1.5	128.0	5
113	3196		COASTAL ANDHRA PRADESH	2015	2.0	0.6	5.5	32.3	34.1	283.8	116.0	192.0	201.8	59.7	81.2	2.0	1010.9	2.5	71.9	7

114 rows × 20 columns



# **Data Cleaning and Data Preprocessing**

In [3]:

```

1 df=df.dropna()
2 df

```

Out[3]:

		index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL	Jan-Feb	Mar-May	Jun-Aug
0	3083		COASTAL ANDHRA PRADESH	1902	2.0	0.0	2.8	23.9	37.6	72.6	144.5	236.1	204.5	262.0	50.4	27.1	1063.6	2.0	64.4	6
1	3084		COASTAL ANDHRA PRADESH	1903	0.8	13.3	0.2	6.2	73.4	154.0	248.6	258.0	216.5	159.1	173.9	12.1	1316.2	14.2	79.8	8
2	3085		COASTAL ANDHRA PRADESH	1904	1.3	0.0	5.4	3.0	136.3	107.8	120.2	117.7	116.8	240.9	0.0	10.7	860.2	1.3	144.7	4
3	3086		COASTAL ANDHRA PRADESH	1905	1.1	16.7	68.0	37.0	68.8	84.4	64.6	210.8	170.2	66.0	7.4	0.0	795.2	17.8	173.8	5
4	3087		COASTAL ANDHRA PRADESH	1906	3.9	23.5	9.9	2.3	11.0	252.6	155.8	241.1	126.9	92.1	21.5	80.8	1021.5	27.4	23.2	7
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
109	3192		COASTAL ANDHRA PRADESH	2011	0.0	17.9	0.9	62.3	67.9	86.8	196.0	215.8	129.7	74.6	4.9	5.0	861.9	17.9	131.2	6
110	3193		COASTAL ANDHRA PRADESH	2012	37.6	0.0	2.7	24.0	39.3	95.4	221.9	221.2	246.5	140.0	289.7	0.0	1318.4	37.6	66.1	7
111	3194		COASTAL ANDHRA PRADESH	2013	2.0	29.6	0.2	48.0	28.2	127.5	162.4	123.1	132.0	411.5	53.1	2.8	1120.5	31.7	76.4	5
112	3195		COASTAL ANDHRA PRADESH	2014	0.4	1.2	9.1	6.0	112.9	45.7	151.8	177.8	144.5	195.6	23.7	6.4	874.9	1.5	128.0	5
113	3196		COASTAL ANDHRA PRADESH	2015	2.0	0.6	5.5	32.3	34.1	283.8	116.0	192.0	201.8	59.7	81.2	2.0	1010.9	2.5	71.9	7

114 rows × 20 columns

```
In [4]: 1 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')
```

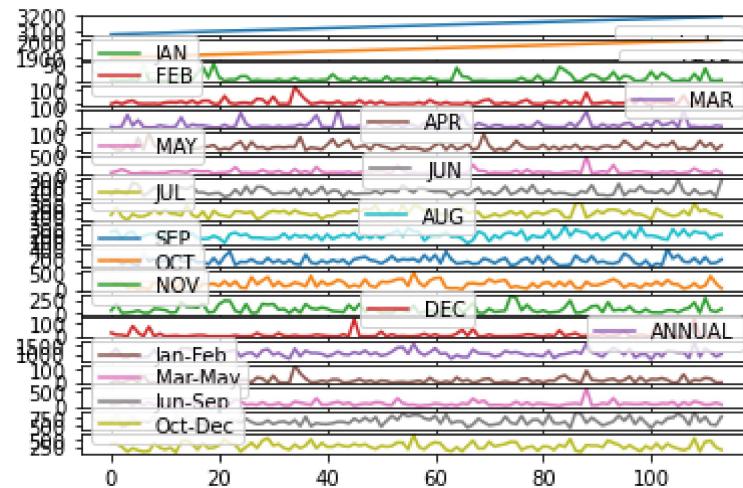
```
In [5]: 1 df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 114 entries, 0 to 113
Data columns (total 20 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   index       114 non-null    int64  
 1   SUBDIVISION 114 non-null    object  
 2   YEAR        114 non-null    int64  
 3   JAN         114 non-null    float64 
 4   FEB         114 non-null    float64 
 5   MAR         114 non-null    float64 
 6   APR         114 non-null    float64 
 7   MAY         114 non-null    float64 
 8   JUN         114 non-null    float64 
 9   JUL         114 non-null    float64 
 10  AUG         114 non-null    float64 
 11  SEP         114 non-null    float64 
 12  OCT         114 non-null    float64 
 13  NOV         114 non-null    float64 
 14  DEC         114 non-null    float64 
 15  ANNUAL      114 non-null    float64 
 16  Jan-Feb     114 non-null    float64 
 17  Mar-May     114 non-null    float64 
 18  Jun-Sep     114 non-null    float64 
 19  Oct-Dec     114 non-null    float64 
dtypes: float64(17), int64(2), object(1)
memory usage: 18.7+ KB
```

## Line Chart

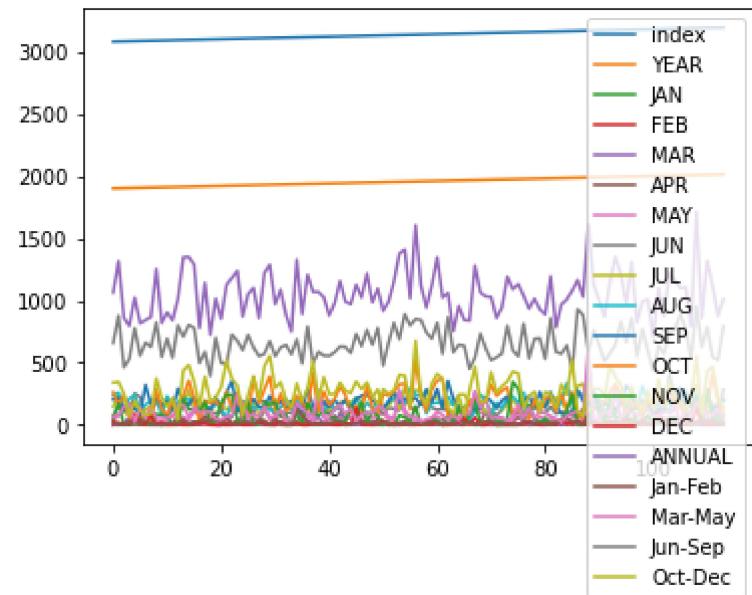
```
In [6]: 1 df.plot.line(subplots=True)
```

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



In [7]: 1 df.plot.line()

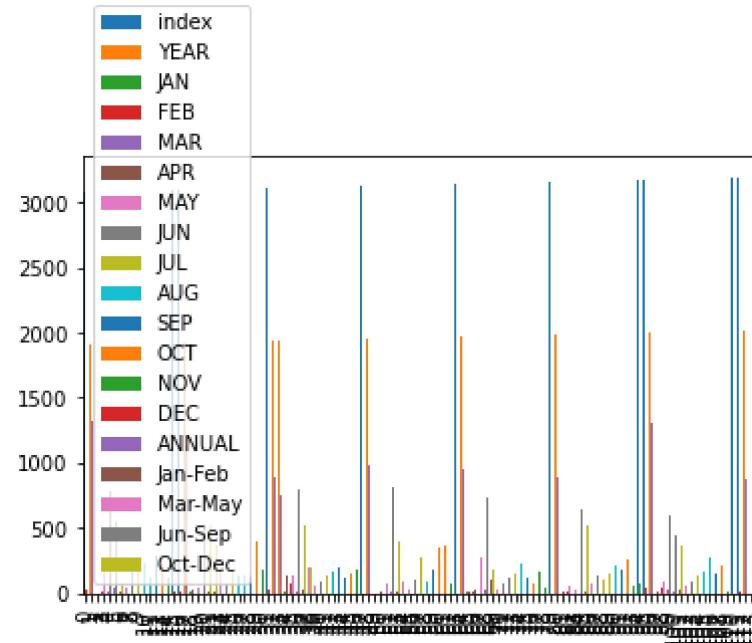
Out[7]: <AxesSubplot:>



## Bar Chart

In [8]: 1 df.plot.bar()

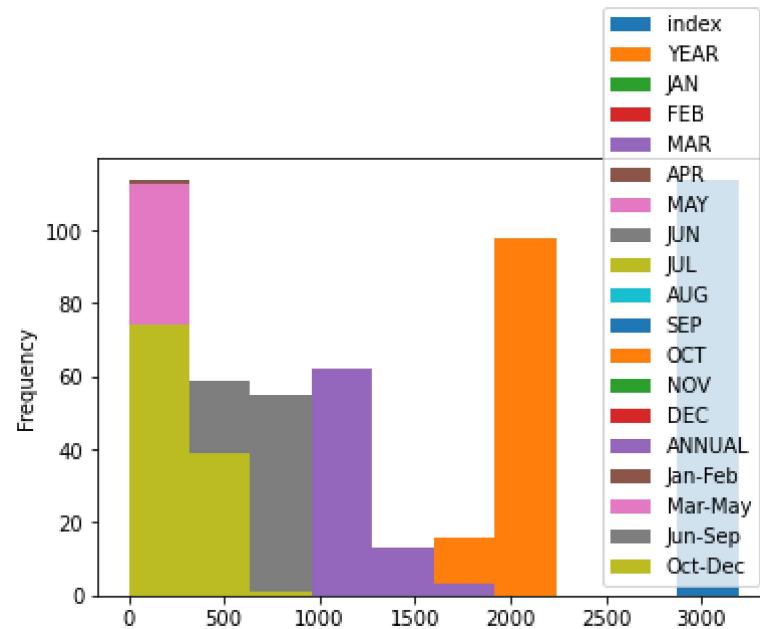
Out[8]: <AxesSubplot:>



## Histogram

In [9]: 1 df.plot.hist()

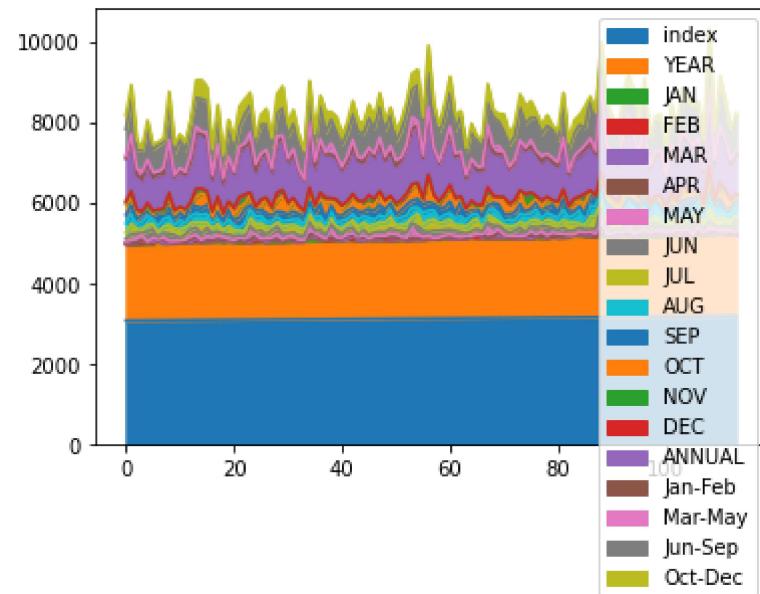
Out[9]: <AxesSubplot:ylabel='Frequency'>



## Area Chart

In [10]: 1 df.plot.area()

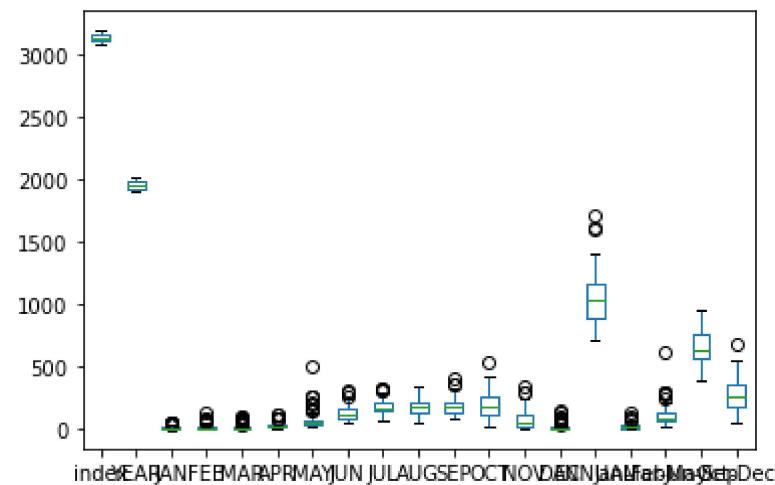
Out[10]: <AxesSubplot:>



## Box Chart

```
In [11]: 1 df.plot.box()
```

```
Out[11]: <AxesSubplot:>
```



## Pie Chart

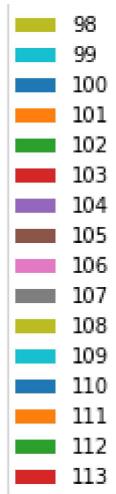
```
In [12]: 1 df.plot.pie(y='ANNUAL')
```

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





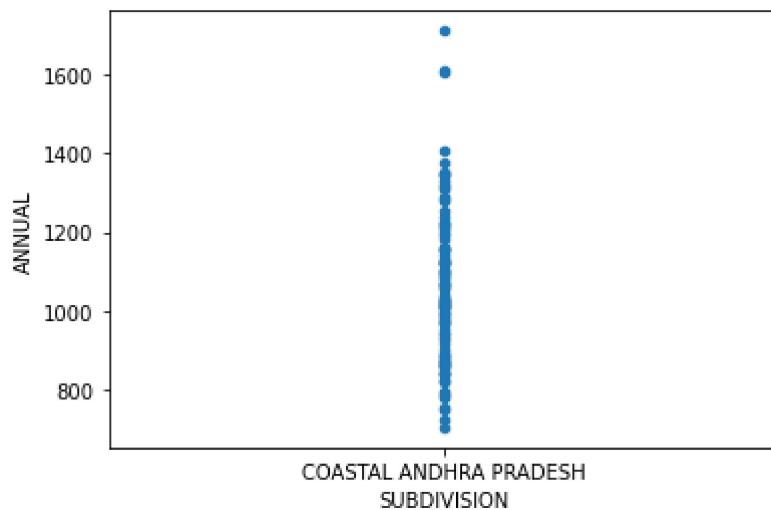
40
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97



## Scatter Plot

```
In [13]: 1 df.plot.scatter(x='SUBDIVISION',y='ANNUAL')
```

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



In [14]: 1 df.info()

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 114 entries, 0 to 113
Data columns (total 20 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   index       114 non-null    int64  
 1   SUBDIVISION 114 non-null    object  
 2   YEAR        114 non-null    int64  
 3   JAN         114 non-null    float64 
 4   FEB         114 non-null    float64 
 5   MAR         114 non-null    float64 
 6   APR         114 non-null    float64 
 7   MAY         114 non-null    float64 
 8   JUN         114 non-null    float64 
 9   JUL         114 non-null    float64 
 10  AUG         114 non-null    float64 
 11  SEP         114 non-null    float64 
 12  OCT         114 non-null    float64 
 13  NOV         114 non-null    float64 
 14  DEC         114 non-null    float64 
 15  ANNUAL      114 non-null    float64 
 16  Jan-Feb     114 non-null    float64 
 17  Mar-May     114 non-null    float64 
 18  Jun-Sep     114 non-null    float64 
 19  Oct-Dec     114 non-null    float64 
dtypes: float64(17), int64(2), object(1)
memory usage: 18.7+ KB
```

```
In [15]: 1 df.describe()
```

Out[15]:

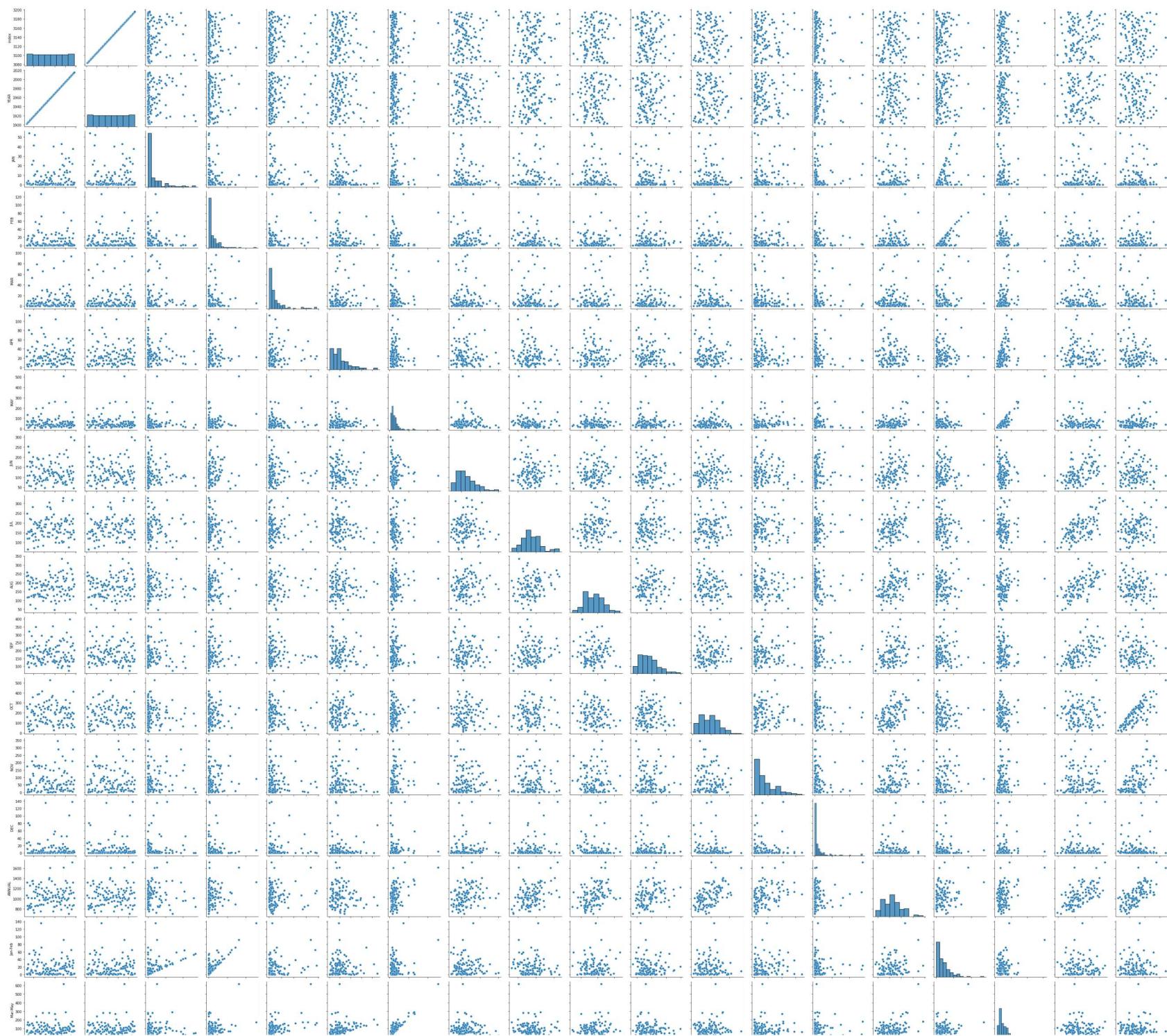
	index	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG
count	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000
mean	3139.500000	1958.500000	7.384211	12.327193	13.274561	26.723684	62.495614	124.097368	174.357895	176.293860
std	33.052988	33.052988	11.526142	19.030773	20.116633	21.404161	63.998420	50.638587	53.735786	55.461064
min	3083.000000	1902.000000	0.000000	0.000000	0.000000	1.100000	10.500000	44.300000	64.600000	45.500000
25%	3111.250000	1930.250000	0.200000	0.425000	1.525000	12.925000	31.275000	86.950000	139.250000	132.375000
50%	3139.500000	1958.500000	2.000000	5.100000	5.650000	21.150000	44.250000	116.050000	168.100000	176.100000
75%	3167.750000	1986.750000	10.125000	17.375000	14.575000	35.175000	69.900000	152.300000	204.275000	215.025000
max	3196.000000	2015.000000	54.100000	127.100000	96.600000	112.200000	507.700000	300.100000	329.600000	336.200000

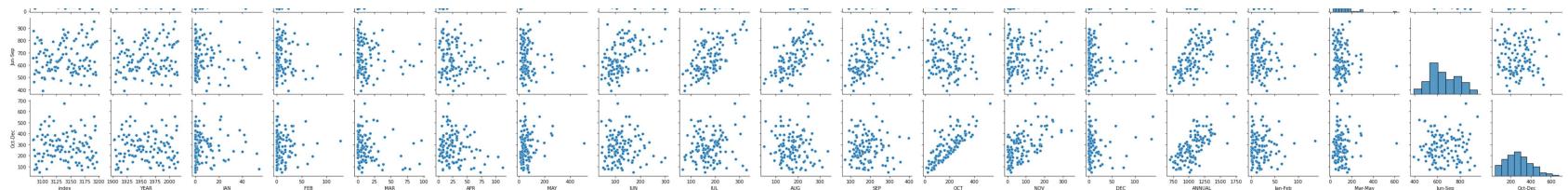
## EDA And Visualization

```
In [16]: 1 sns.pairplot(df)
```

```
Out[16]: <seaborn.axisgrid.PairGrid at 0x2ce8adc40d0>
```



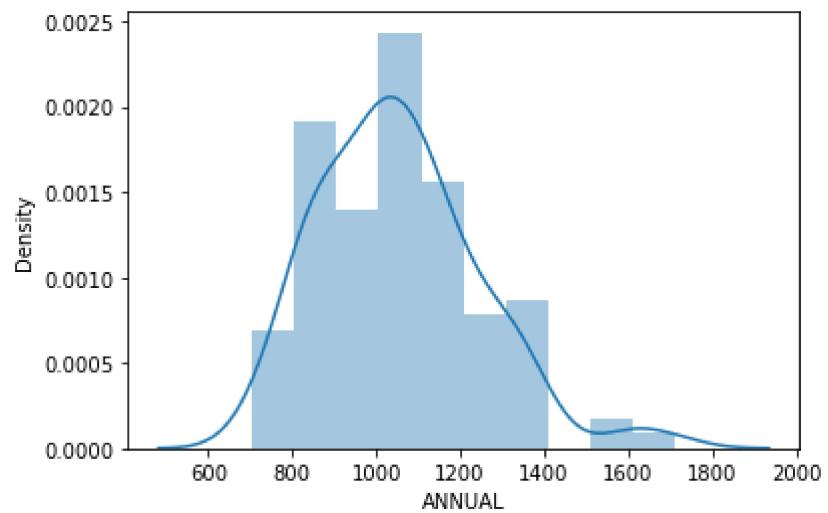




```
In [17]: 1 sns.distplot(df['ANNUAL'])
```

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

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



```
In [18]: 1 sns.heatmap(df.corr())
```

```
Out[18]: <AxesSubplot:>
```

