

## 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_KARNATAKA.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
0	3543		COASTAL KARNATAKA	1902	3.2	0.3	4.9	10.2	54.6	698.4	1401.6	454.2	708.4	180.4	50.8	132.2	3699.2	3.5	69.
1	3544		COASTAL KARNATAKA	1903	0.7	0.0	0.0	4.1	202.8	536.5	1405.5	593.8	304.4	185.0	79.3	5.3	3317.4	0.7	206.
2	3545		COASTAL KARNATAKA	1904	2.4	0.0	4.8	23.7	93.2	1108.2	1070.0	465.6	245.3	127.2	0.7	0.0	3141.1	2.4	121.
3	3546		COASTAL KARNATAKA	1905	0.0	0.2	0.0	6.4	83.1	767.3	777.3	586.9	172.9	222.2	36.1	0.0	2652.3	0.2	89.
4	3547		COASTAL KARNATAKA	1906	23.0	0.0	0.0	0.5	29.8	593.6	1173.4	535.0	273.3	128.5	22.1	56.8	2836.0	23.0	30.
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
109	3652		COASTAL KARNATAKA	2011	4.8	3.8	8.7	66.1	49.3	1018.4	1080.5	861.3	545.2	178.8	81.5	10.2	3908.6	8.6	124.
110	3653		COASTAL KARNATAKA	2012	NaN	11.4	5.1	77.0	22.9	650.9	754.6	1027.6	382.0	115.1	68.0	3.6	NaN	NaN	104.
111	3654		COASTAL KARNATAKA	2013	2.4	19.6	19.0	28.5	100.4	1153.0	1515.3	680.2	379.1	265.1	56.9	10.0	4229.4	21.9	147.
112	3655		COASTAL KARNATAKA	2014	0.0	0.3	1.9	40.5	181.9	507.0	1155.4	1121.0	379.3	226.4	40.0	30.8	3684.4	0.3	224.
113	3656		COASTAL KARNATAKA	2015	1.4	1.0	32.3	72.2	150.3	735.3	930.9	575.2	260.3	208.5	124.2	14.3	3106.0	2.4	254.

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
0	3543		COASTAL KARNATAKA	1902	3.2	0.3	4.9	10.2	54.6	698.4	1401.6	454.2	708.4	180.4	50.8	132.2	3699.2	3.5	69.1
1	3544		COASTAL KARNATAKA	1903	0.7	0.0	0.0	4.1	202.8	536.5	1405.5	593.8	304.4	185.0	79.3	5.3	3317.4	0.7	206.1
2	3545		COASTAL KARNATAKA	1904	2.4	0.0	4.8	23.7	93.2	1108.2	1070.0	465.6	245.3	127.2	0.7	0.0	3141.1	2.4	121.1
3	3546		COASTAL KARNATAKA	1905	0.0	0.2	0.0	6.4	83.1	767.3	777.3	586.9	172.9	222.2	36.1	0.0	2652.3	0.2	89.1
4	3547		COASTAL KARNATAKA	1906	23.0	0.0	0.0	0.5	29.8	593.6	1173.4	535.0	273.3	128.5	22.1	56.8	2836.0	23.0	30.1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
108	3651		COASTAL KARNATAKA	2010	14.4	0.4	3.5	62.2	80.2	682.7	1200.2	637.5	468.4	294.7	231.5	11.0	3686.8	14.8	145.1
109	3652		COASTAL KARNATAKA	2011	4.8	3.8	8.7	66.1	49.3	1018.4	1080.5	861.3	545.2	178.8	81.5	10.2	3908.6	8.6	124.1
111	3654		COASTAL KARNATAKA	2013	2.4	19.6	19.0	28.5	100.4	1153.0	1515.3	680.2	379.1	265.1	56.9	10.0	4229.4	21.9	147.1
112	3655		COASTAL KARNATAKA	2014	0.0	0.3	1.9	40.5	181.9	507.0	1155.4	1121.0	379.3	226.4	40.0	30.8	3684.4	0.3	224.1
113	3656		COASTAL KARNATAKA	2015	1.4	1.0	32.3	72.2	150.3	735.3	930.9	575.2	260.3	208.5	124.2	14.3	3106.0	2.4	254.1

113 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')
```

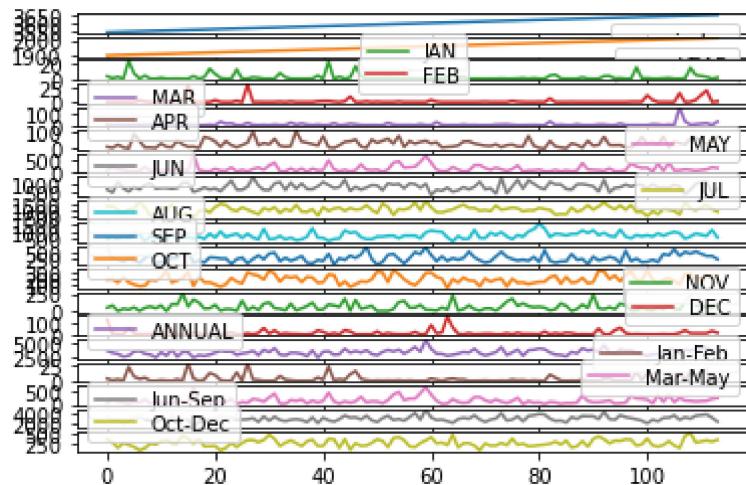
In [5]: 1 df.info()

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 113 entries, 0 to 113
Data columns (total 20 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   index       113 non-null    int64  
 1   SUBDIVISION 113 non-null    object  
 2   YEAR        113 non-null    int64  
 3   JAN         113 non-null    float64 
 4   FEB         113 non-null    float64 
 5   MAR         113 non-null    float64 
 6   APR         113 non-null    float64 
 7   MAY         113 non-null    float64 
 8   JUN         113 non-null    float64 
 9   JUL         113 non-null    float64 
 10  AUG         113 non-null    float64 
 11  SEP         113 non-null    float64 
 12  OCT         113 non-null    float64 
 13  NOV         113 non-null    float64 
 14  DEC         113 non-null    float64 
 15  ANNUAL      113 non-null    float64 
 16  Jan-Feb     113 non-null    float64 
 17  Mar-May     113 non-null    float64 
 18  Jun-Sep     113 non-null    float64 
 19  Oct-Dec     113 non-null    float64 
dtypes: float64(17), int64(2), object(1)
memory usage: 18.5+ KB
```

## Line Chart

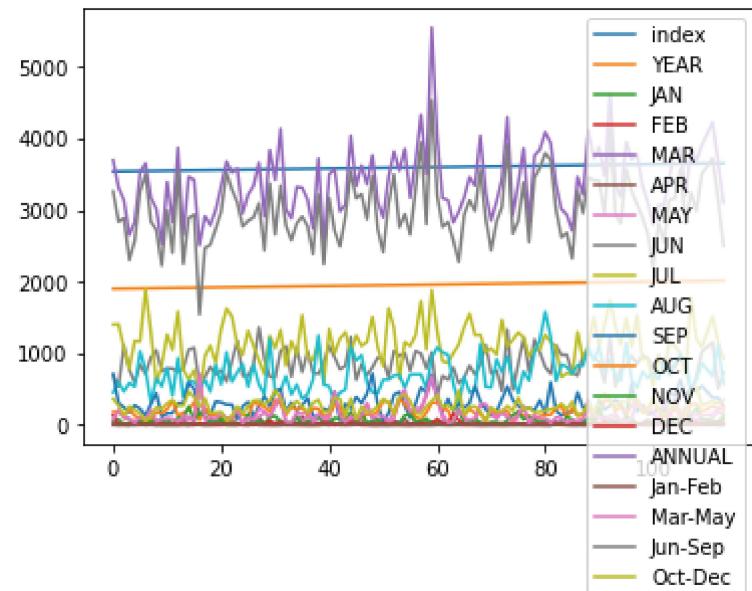
```
In [6]: 1 df.plot.line(subplots=True)
```

```
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()

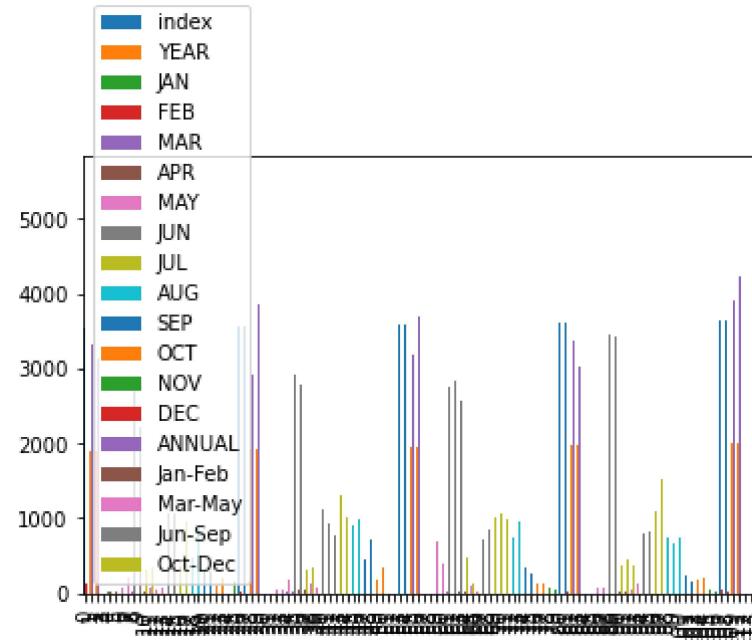
Out[7]: <AxesSubplot:>



## Bar Chart

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

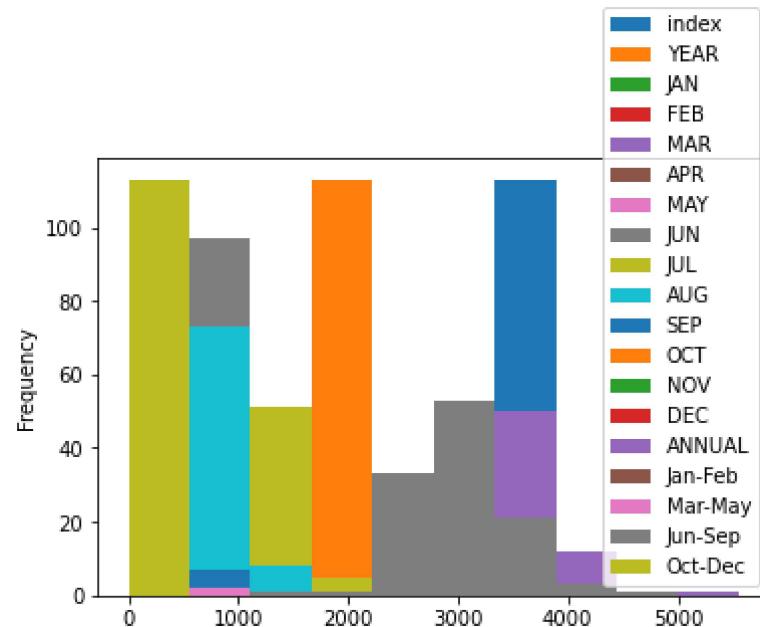
Out[8]: <AxesSubplot:>



## Histogram

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

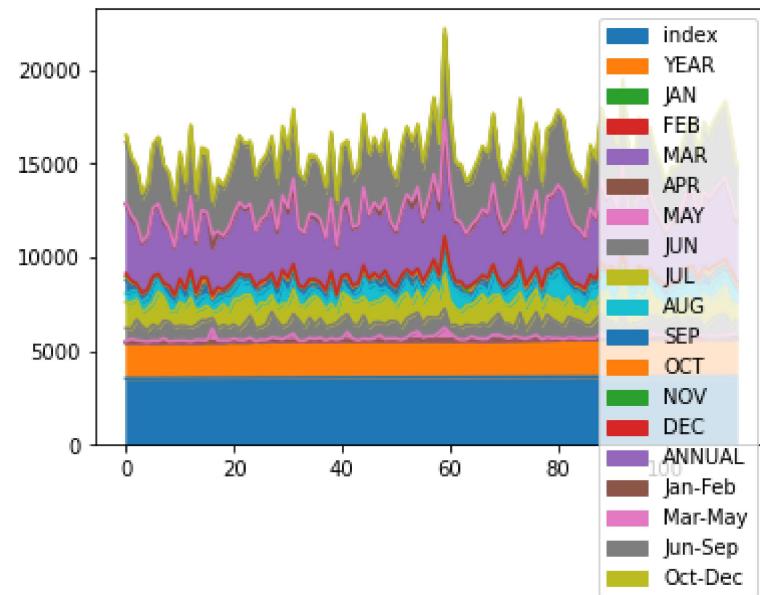
Out[9]: <AxesSubplot:ylabel='Frequency'>



## Area Chart

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

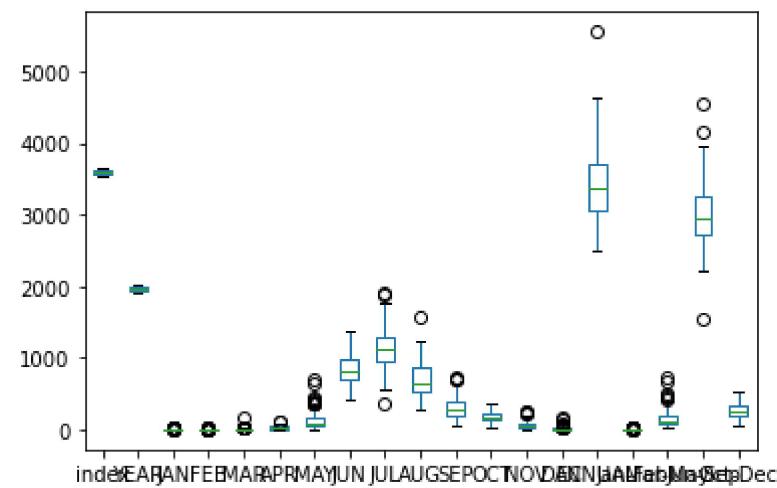
Out[10]: <AxesSubplot:>



## Box Chart

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

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

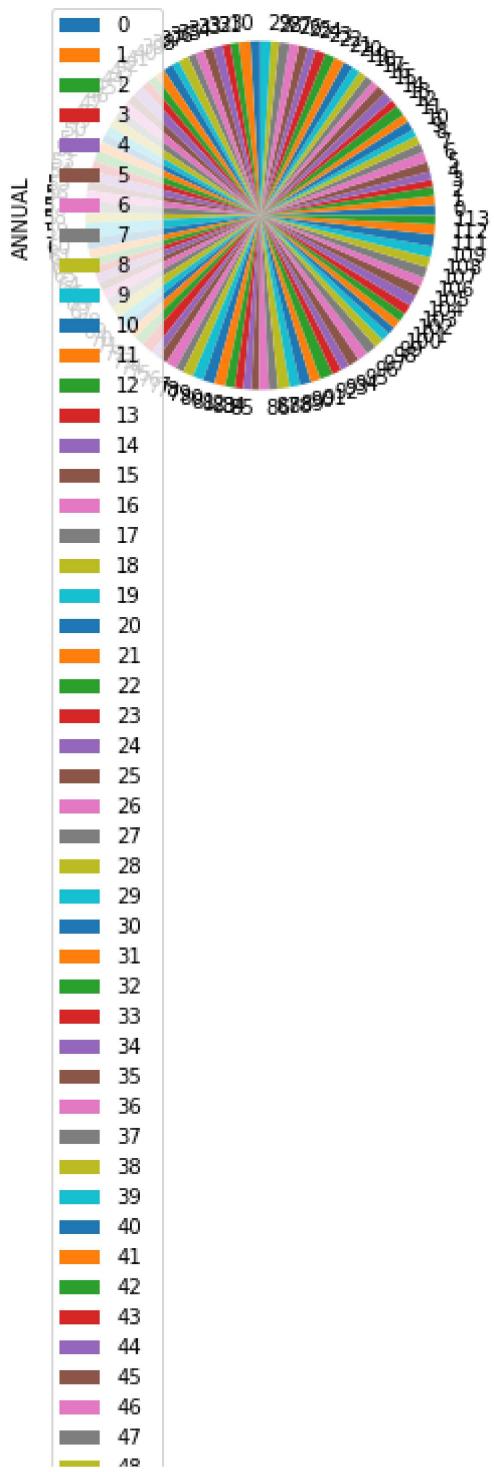


## Pie Chart

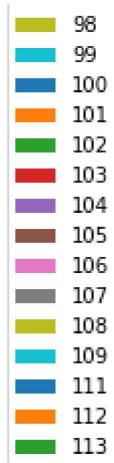
```
In [12]: 1 df.plot.pie(y='ANNUAL')
```

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





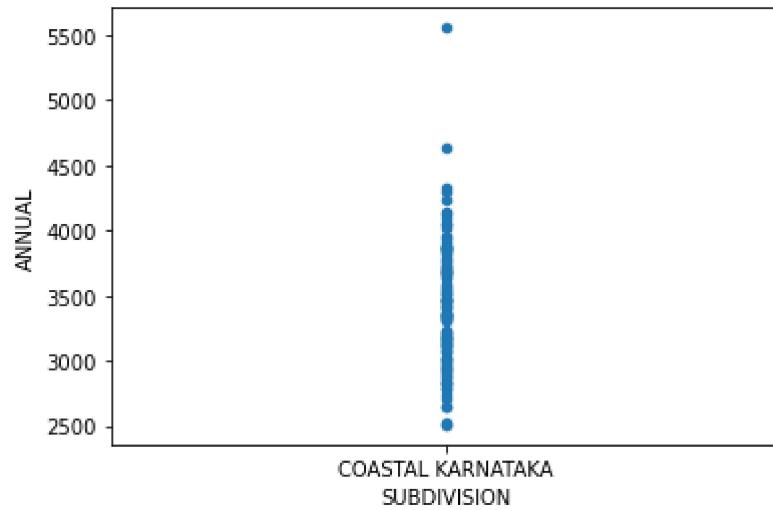
40
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97



## 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: 113 entries, 0 to 113
Data columns (total 20 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   index       113 non-null    int64  
 1   SUBDIVISION 113 non-null    object  
 2   YEAR        113 non-null    int64  
 3   JAN         113 non-null    float64 
 4   FEB         113 non-null    float64 
 5   MAR         113 non-null    float64 
 6   APR         113 non-null    float64 
 7   MAY         113 non-null    float64 
 8   JUN         113 non-null    float64 
 9   JUL         113 non-null    float64 
 10  AUG         113 non-null    float64 
 11  SEP         113 non-null    float64 
 12  OCT         113 non-null    float64 
 13  NOV         113 non-null    float64 
 14  DEC         113 non-null    float64 
 15  ANNUAL      113 non-null    float64 
 16  Jan-Feb     113 non-null    float64 
 17  Mar-May     113 non-null    float64 
 18  Jun-Sep     113 non-null    float64 
 19  Oct-Dec     113 non-null    float64 
dtypes: float64(17), int64(2), object(1)
memory usage: 18.5+ KB
```

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

Out[15]:

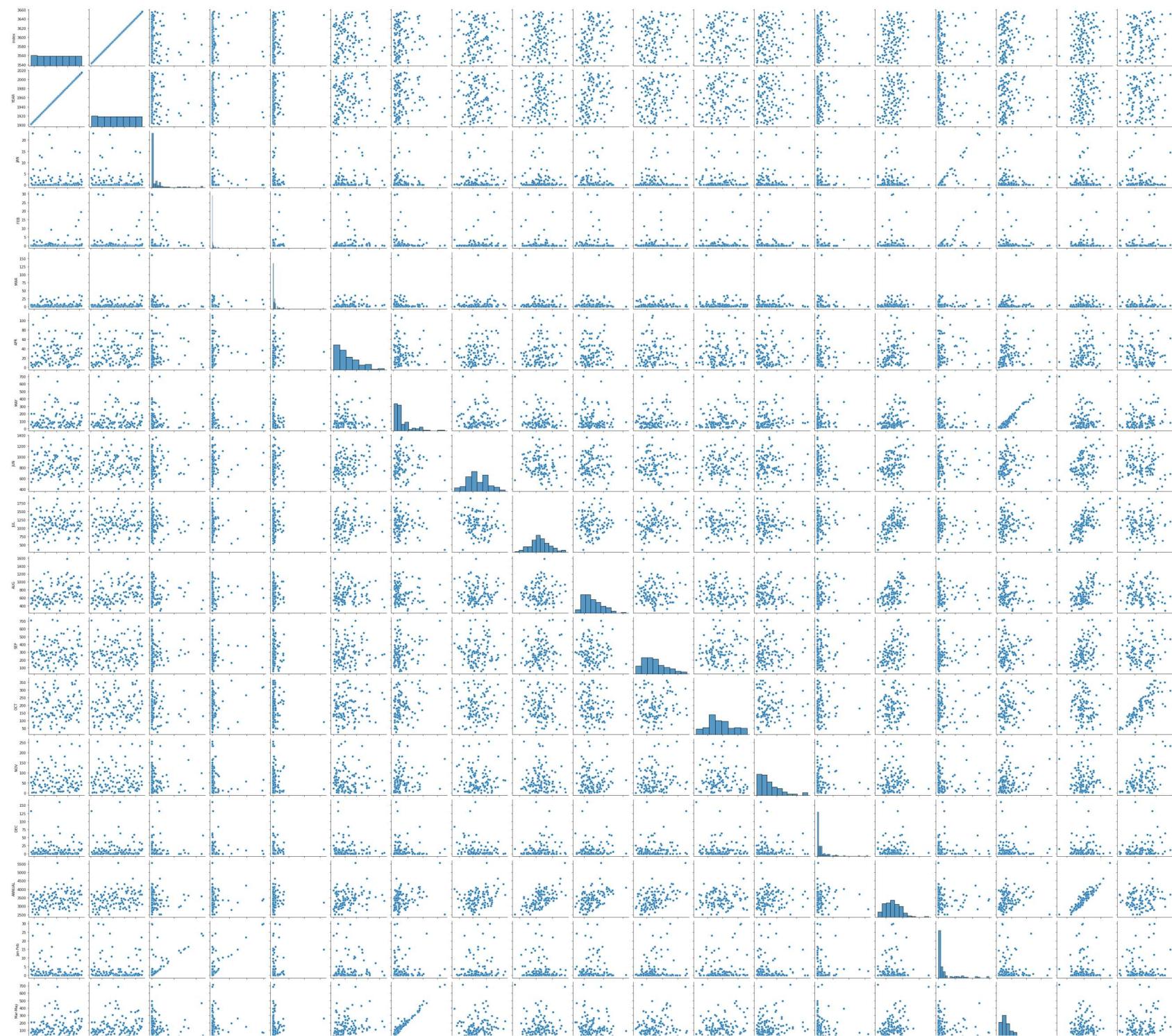
	index	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	
count	113.000000	113.000000	113.000000	113.000000	113.000000	113.000000	113.000000	113.000000	113.000000	113.000000	113.000000
mean	3599.026549	1958.026549	1.938938	1.438938	6.330088	30.318584	124.036283	841.953097	1131.526549	711.788496	301.500000
std	32.809640	32.809640	4.237132	4.677557	16.715176	23.861623	126.089062	194.076712	294.769389	240.735070	14.200000
min	3543.000000	1902.000000	0.000000	0.000000	0.000000	0.000000	8.400000	405.400000	360.600000	278.300000	5.000000
25%	3571.000000	1930.000000	0.000000	0.000000	0.200000	11.300000	44.100000	704.800000	960.900000	535.000000	18.000000
50%	3599.000000	1958.000000	0.100000	0.000000	1.400000	24.700000	80.200000	813.300000	1108.800000	654.400000	27.000000
75%	3627.000000	1986.000000	2.000000	0.500000	5.700000	44.800000	164.000000	977.900000	1296.100000	861.300000	38.000000
max	3656.000000	2015.000000	23.000000	29.800000	161.400000	110.100000	699.500000	1361.600000	1904.900000	1579.100000	7.000000

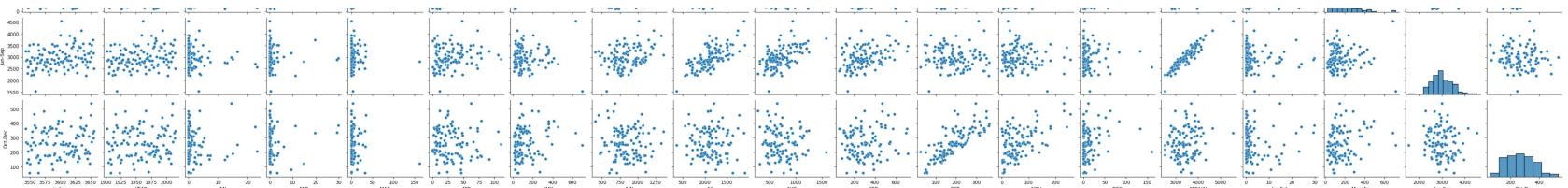
## EDA And Visualization

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

```
Out[16]: <seaborn.axisgrid.PairGrid at 0x1d965ddbd00>
```



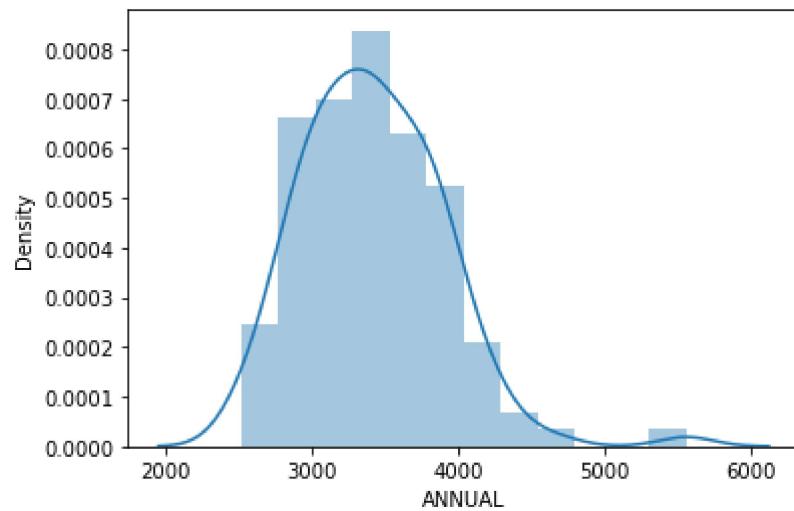




```
In [17]: 1 sns.distplot(df['ANNUAL'])
```

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:>
```

