

# 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\ORISSA.csv')  
df
```

Out[2]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OC
0	667	ORISSA	1901	39.5	65.1	16.1	51.6	79.0	78.2	288.4	307.7	185.3	76.
1	668	ORISSA	1902	3.4	0.2	14.2	101.1	56.7	108.3	437.4	349.1	202.7	33.
2	669	ORISSA	1903	19.7	18.9	10.5	34.6	73.3	154.3	410.4	295.2	265.6	228.
3	670	ORISSA	1904	0.2	12.2	20.6	10.1	100.2	342.9	336.7	350.4	227.8	111.
4	671	ORISSA	1905	24.3	17.2	66.3	56.9	107.5	92.0	330.1	281.4	344.1	36.
...	...	...	...	...	...	...	...	...	...	...	...	...	...
110	777	ORISSA	2011	3.7	16.2	4.9	58.2	75.6	210.1	199.6	358.6	398.7	20.
111	778	ORISSA	2012	50.8	3.6	0.9	34.8	21.3	169.6	324.3	417.0	242.4	66.
112	779	ORISSA	2013	3.3	7.8	2.1	53.6	57.7	272.6	380.0	254.9	208.1	391.
113	780	ORISSA	2014	0.0	17.6	25.1	11.7	111.9	92.2	496.2	386.3	281.1	111.
114	781	ORISSA	2015	15.1	3.3	10.5	67.6	32.6	238.6	294.8	264.0	237.0	24.

115 rows × 20 columns

# Data Cleaning and Data Preprocessing

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

Out[3]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OC
0	667	ORISSA	1901	39.5	65.1	16.1	51.6	79.0	78.2	288.4	307.7	185.3	76.
1	668	ORISSA	1902	3.4	0.2	14.2	101.1	56.7	108.3	437.4	349.1	202.7	33.
2	669	ORISSA	1903	19.7	18.9	10.5	34.6	73.3	154.3	410.4	295.2	265.6	228.
3	670	ORISSA	1904	0.2	12.2	20.6	10.1	100.2	342.9	336.7	350.4	227.8	111.
4	671	ORISSA	1905	24.3	17.2	66.3	56.9	107.5	92.0	330.1	281.4	344.1	36.
...	...	...	...	...	...	...	...	...	...	...	...	...	...
110	777	ORISSA	2011	3.7	16.2	4.9	58.2	75.6	210.1	199.6	358.6	398.7	20.
111	778	ORISSA	2012	50.8	3.6	0.9	34.8	21.3	169.6	324.3	417.0	242.4	66.
112	779	ORISSA	2013	3.3	7.8	2.1	53.6	57.7	272.6	380.0	254.9	208.1	391.
113	780	ORISSA	2014	0.0	17.6	25.1	11.7	111.9	92.2	496.2	386.3	281.1	111.
114	781	ORISSA	2015	15.1	3.3	10.5	67.6	32.6	238.6	294.8	264.0	237.0	24.

115 rows × 20 columns



In [4]: 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')

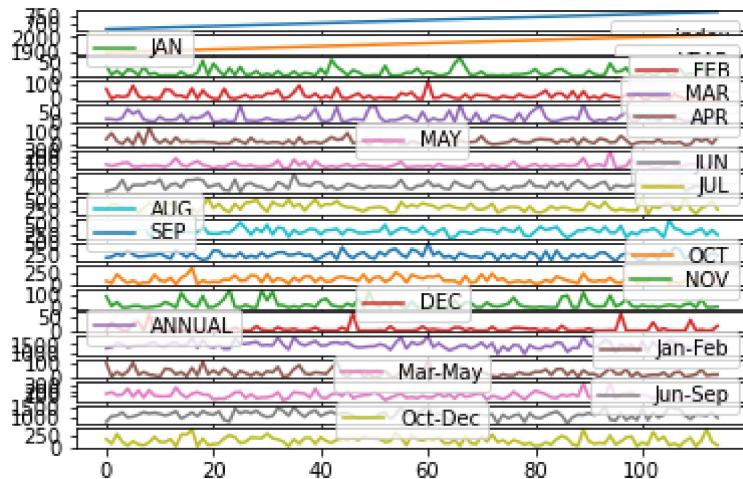
In [5]: df.info()

```
<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

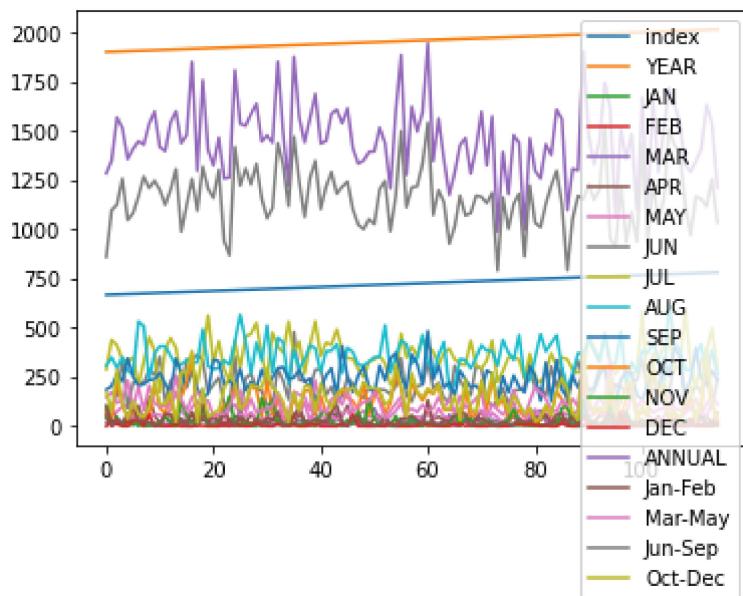
In [6]: `df.plot.line(subplots=True)`

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



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

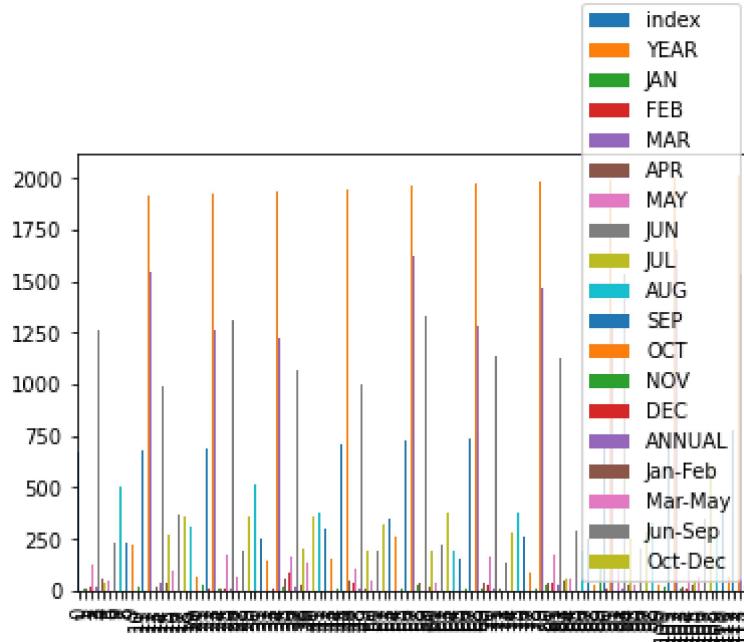
Out[7]: `<AxesSubplot:>`



## Bar Chart

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

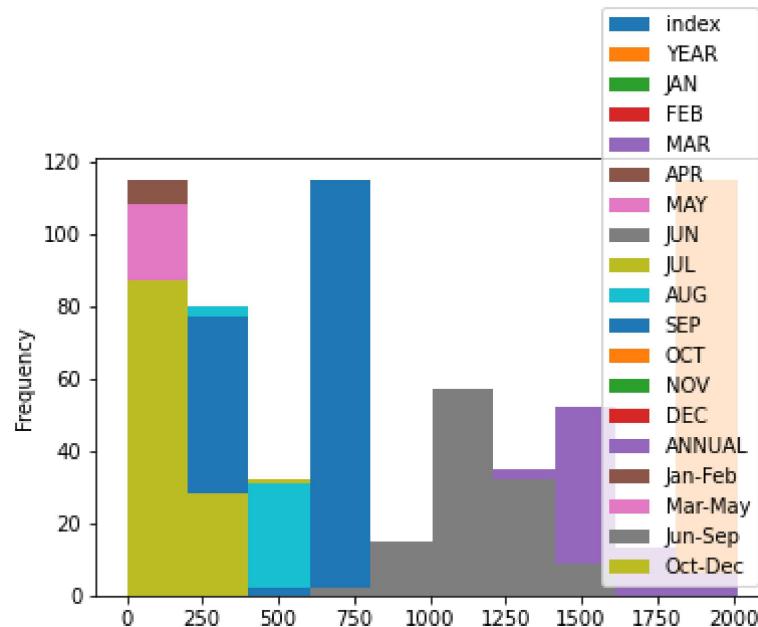
Out[8]: <AxesSubplot:>



## Histogram

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

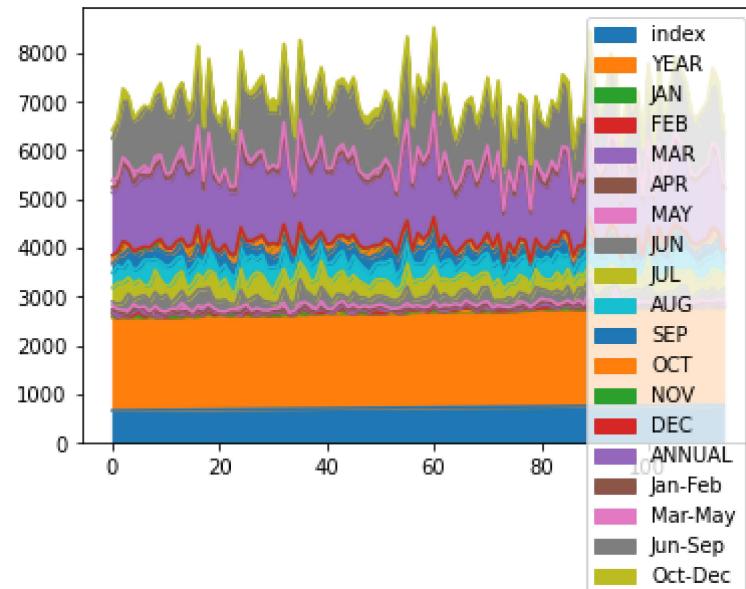
Out[9]: <AxesSubplot:ylabel='Frequency'>



## Area Chart

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

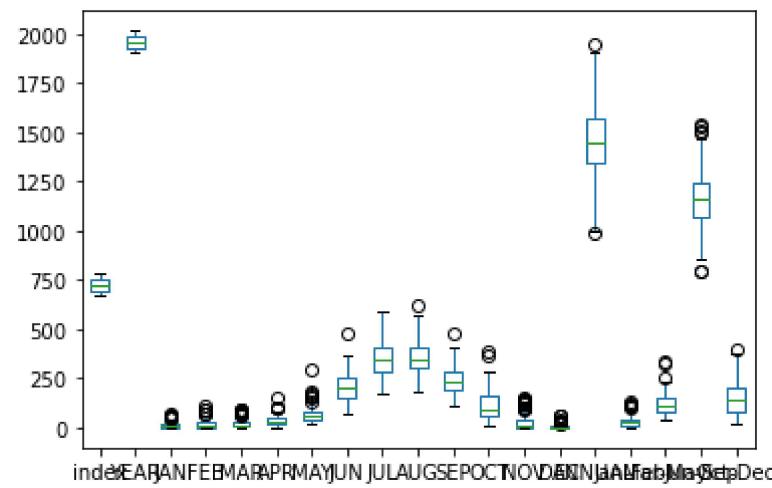
```
Out[10]: <AxesSubplot:>
```



## Box Chart

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

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

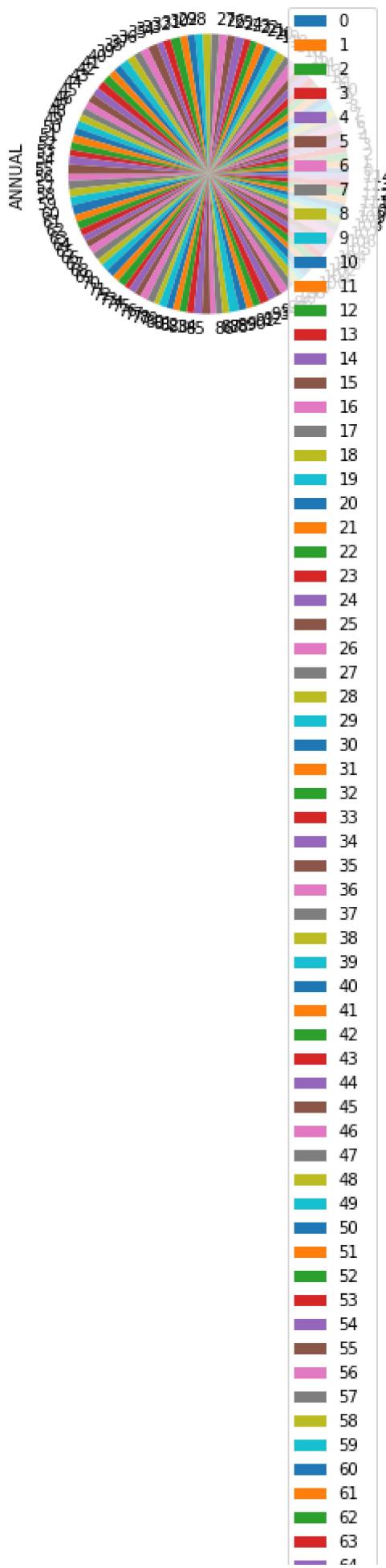


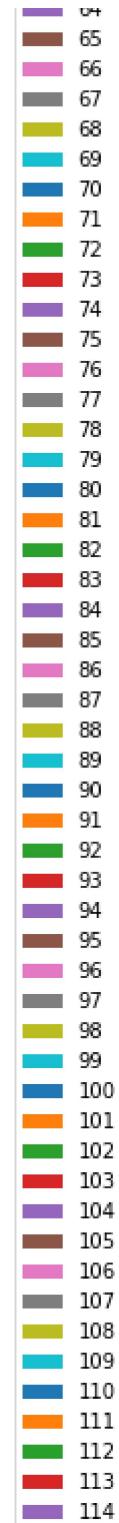
## Pie Chart

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

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



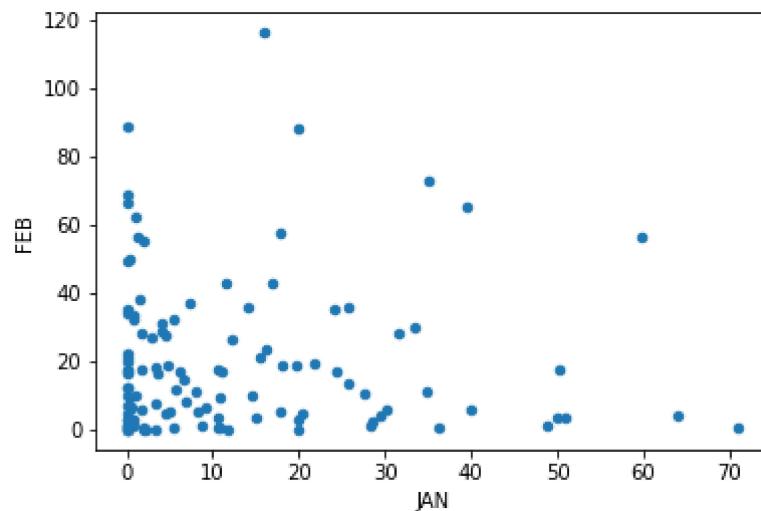




# Scatter Plot

```
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()

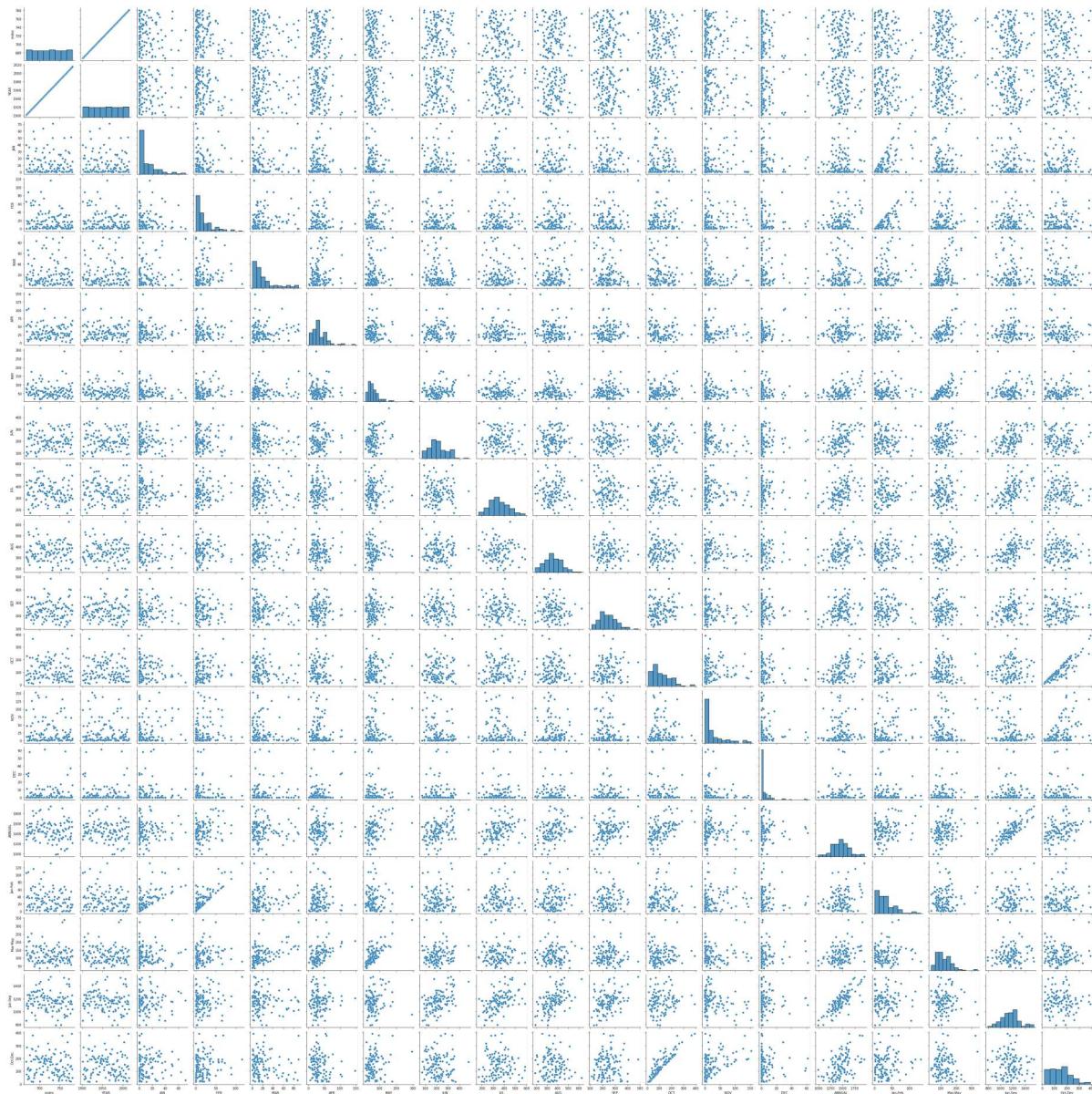
Out[15]:

	index	YEAR	JAN	FEB	MAR	APR	MAY
<b>count</b>	115.000000	115.000000	115.000000	115.000000	115.000000	115.000000	115.000000
<b>mean</b>	724.000000	1958.000000	12.329565	19.71913	21.134783	34.160000	64.886087
<b>std</b>	33.341666	33.341666	15.620842	22.23721	22.259542	23.243417	39.851027
<b>min</b>	667.000000	1901.000000	0.000000	0.000000	0.100000	1.100000	16.200000
<b>25%</b>	695.500000	1929.500000	0.500000	3.45000	5.500000	18.400000	38.350000
<b>50%</b>	724.000000	1958.000000	5.500000	12.10000	12.600000	28.500000	57.700000
<b>75%</b>	752.500000	1986.500000	18.850000	28.45000	26.800000	46.850000	77.300000
<b>max</b>	781.000000	2015.000000	70.900000	116.20000	89.800000	148.400000	293.000000

## EDA And Visualization

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

```
Out[16]: <seaborn.axisgrid.PairGrid at 0x1639a30dbb0>
```

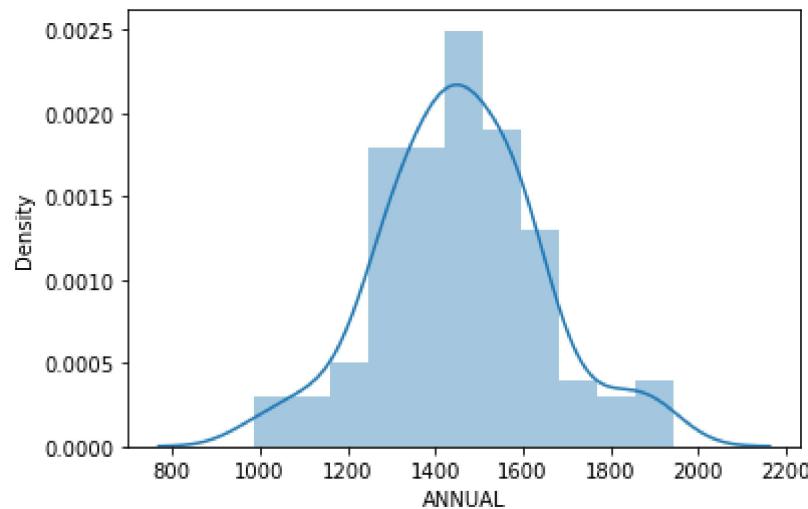


In [17]: `sns.distplot(df['ANNUAL'])`

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

Out[18]: <AxesSubplot:>

