

# Importing Libraries

```
In [56]: import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns
```

# Importing Datasets

In [57]:

```
df=pd.read_csv(r'C:\Users\user\Downloads\Rainfall\Andaman_Nicobar.csv')
df
```

Out[57]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
0	0	ANDAMAN & NICOBAR ISLANDS	1901	49.2	87.1	29.2	2.3	528.8	517.5	365.1	481.1	332.6	3	1	1
1	1	ANDAMAN & NICOBAR ISLANDS	1902	0.0	159.8	12.2	0.0	446.1	537.1	228.9	753.7	666.2	1	1	1
2	2	ANDAMAN & NICOBAR ISLANDS	1903	12.7	144.0	0.0	1.0	235.1	479.9	728.4	326.7	339.0	1	1	1
3	3	ANDAMAN & NICOBAR ISLANDS	1904	9.4	14.7	0.0	202.4	304.5	495.1	502.0	160.1	820.4	2	2	2
4	4	ANDAMAN & NICOBAR ISLANDS	1905	1.3	0.0	3.3	26.9	279.5	628.7	368.7	330.5	297.0	2	2	2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
105	105	ANDAMAN & NICOBAR ISLANDS	2011	265.9	84.8	272.8	111.4	326.5	383.2	583.2	441.5	757.1	2	2	2
106	106	ANDAMAN & NICOBAR ISLANDS	2012	119.9	45.6	30.9	55.8	533.9	458.2	317.3	369.6	868.9	2	2	2
107	107	ANDAMAN & NICOBAR ISLANDS	2013	67.1	37.6	43.0	46.3	509.3	777.0	564.8	336.7	473.6	4	4	4
108	108	ANDAMAN & NICOBAR ISLANDS	2014	41.9	8.6	0.0	11.1	238.0	416.6	467.6	321.6	412.9	4	4	4
109	109	ANDAMAN & NICOBAR ISLANDS	2015	126.8	7.6	3.1	138.2	331.9	346.4	328.9	480.0	523.3	2	2	2

110 rows × 20 columns



## Data Cleaning and Data Preprocessing

In [58]: df=df.dropna()  
df

Out[58]:

		index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL	Jan-Feb	Mar-May	Jun-Sep	Oct-Dec
0	0	0	ANDAMAN & NICOBAR ISLANDS	1901	49.2	87.1	29.2	2.3	528.8	517.5	365.1	481.1	332.6	3	1	1	1	1	1	1	
1	1	1	ANDAMAN & NICOBAR ISLANDS	1902	0.0	159.8	12.2	0.0	446.1	537.1	228.9	753.7	666.2	1	1	1	1	1	1	1	
2	2	2	ANDAMAN & NICOBAR ISLANDS	1903	12.7	144.0	0.0	1.0	235.1	479.9	728.4	326.7	339.0	1	1	1	1	1	1	1	
3	3	3	ANDAMAN & NICOBAR ISLANDS	1904	9.4	14.7	0.0	202.4	304.5	495.1	502.0	160.1	820.4	2	2	2	2	2	2	2	
4	4	4	ANDAMAN & NICOBAR ISLANDS	1905	1.3	0.0	3.3	26.9	279.5	628.7	368.7	330.5	297.0	2	2	2	2	2	2	2	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
105	105	105	ANDAMAN & NICOBAR ISLANDS	2011	265.9	84.8	272.8	111.4	326.5	383.2	583.2	441.5	757.1	2	2	2	2	2	2	2	
106	106	106	ANDAMAN & NICOBAR ISLANDS	2012	119.9	45.6	30.9	55.8	533.9	458.2	317.3	369.6	868.9	2	2	2	2	2	2	2	
107	107	107	ANDAMAN & NICOBAR ISLANDS	2013	67.1	37.6	43.0	46.3	509.3	777.0	564.8	336.7	473.6	4	4	4	4	4	4	4	
108	108	108	ANDAMAN & NICOBAR ISLANDS	2014	41.9	8.6	0.0	11.1	238.0	416.6	467.6	321.6	412.9	4	4	4	4	4	4	4	
109	109	109	ANDAMAN & NICOBAR ISLANDS	2015	126.8	7.6	3.1	138.2	331.9	346.4	328.9	480.0	523.3	2	2	2	2	2	2	2	

104 rows × 20 columns



In [59]: df.columns

Out[59]: 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')

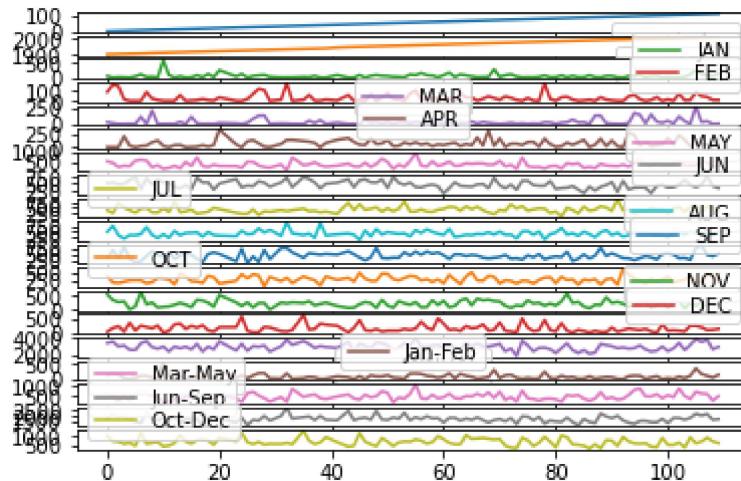
In [60]: df.info()

```
<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
```

## Line Chart

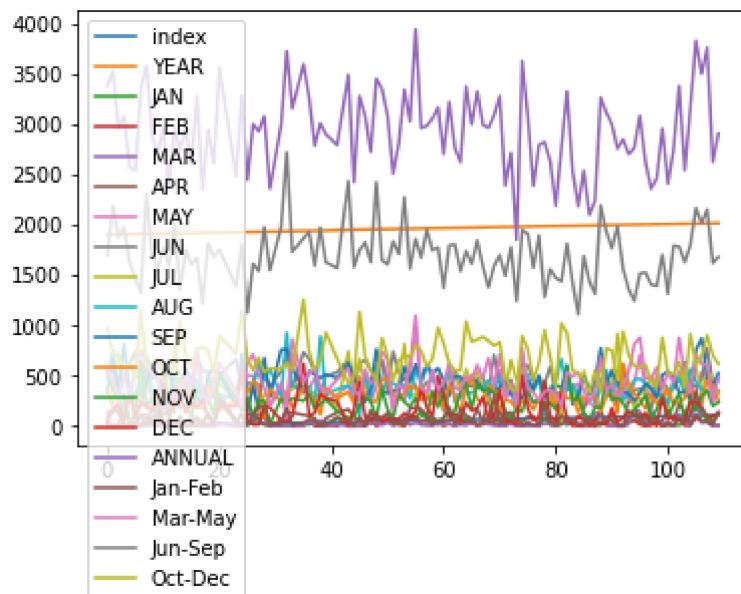
In [61]: `df.plot.line(subplots=True)`

Out[61]: `array([<AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>], dtype=object)`



In [62]: `df.plot.line()`

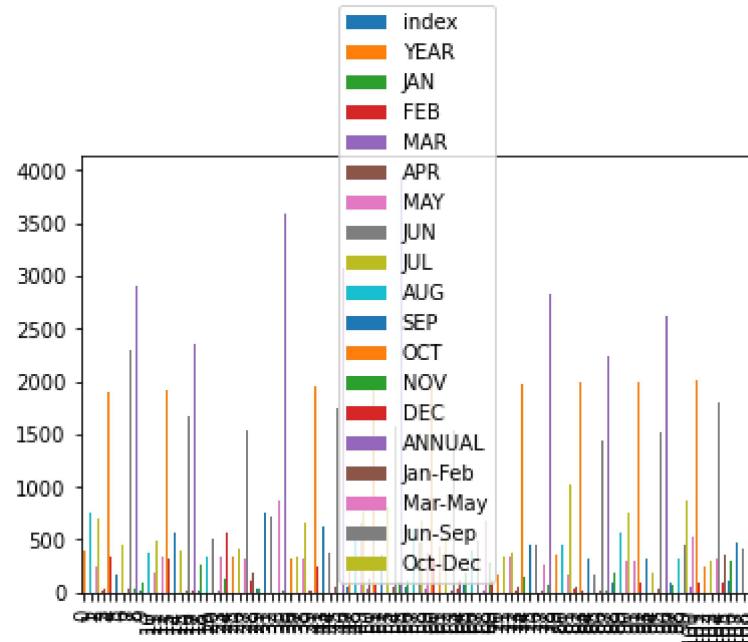
Out[62]: `<AxesSubplot:>`



## Bar Chart

In [63]: `df.plot.bar()`

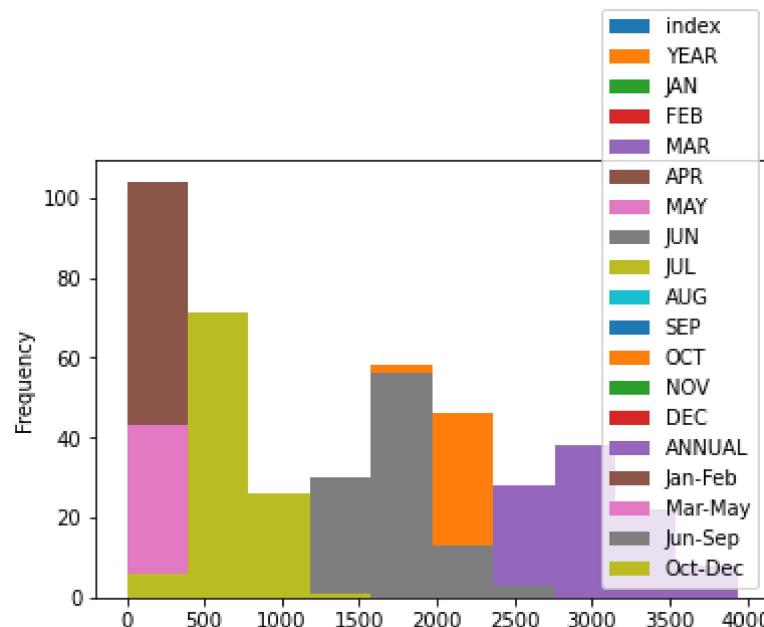
Out[63]: <AxesSubplot:>



## Histogram

In [64]: `df.plot.hist()`

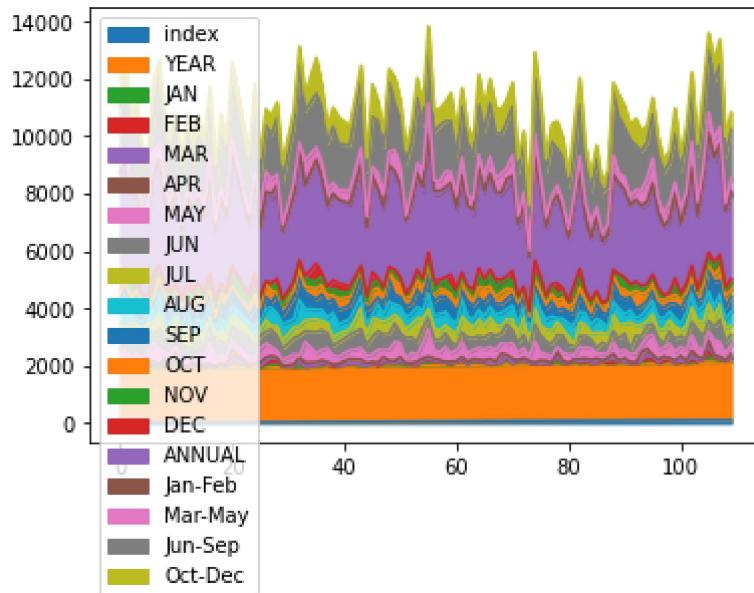
Out[64]: <AxesSubplot:ylabel='Frequency'>



## Area Chart

```
In [65]: df.plot.area()
```

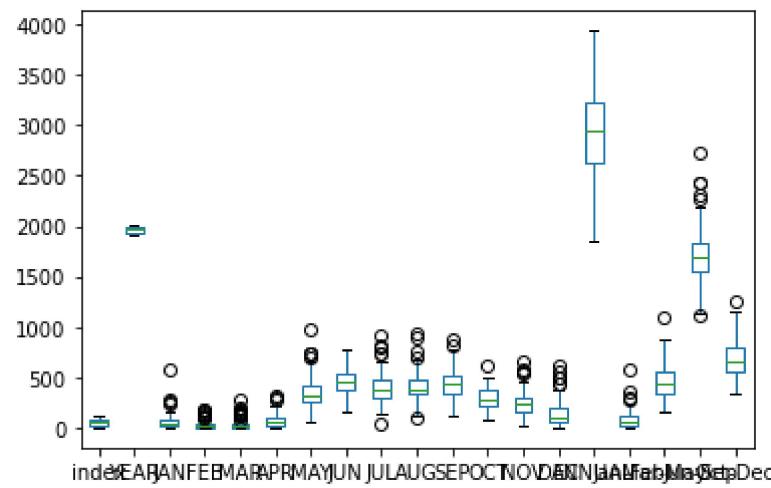
```
Out[65]: <AxesSubplot:>
```



## Box Chart

```
In [66]: df.plot.box()
```

```
Out[66]: <AxesSubplot:>
```

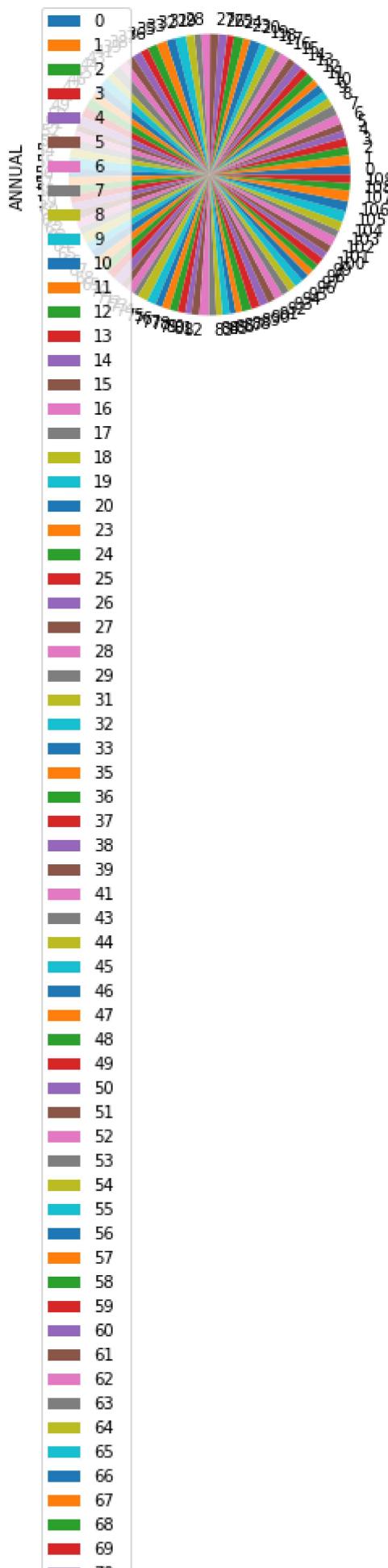


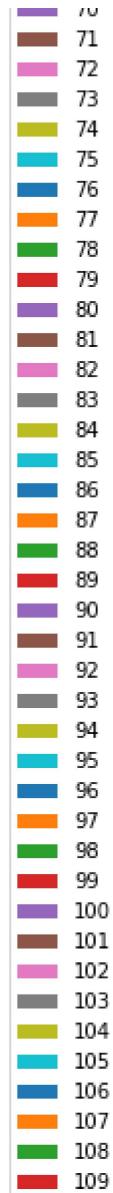
## Pie Chart

```
In [67]: df.plot.pie(y='ANNUAL')
```

```
Out[67]: <AxesSubplot:ylabel='ANNUAL'>
```



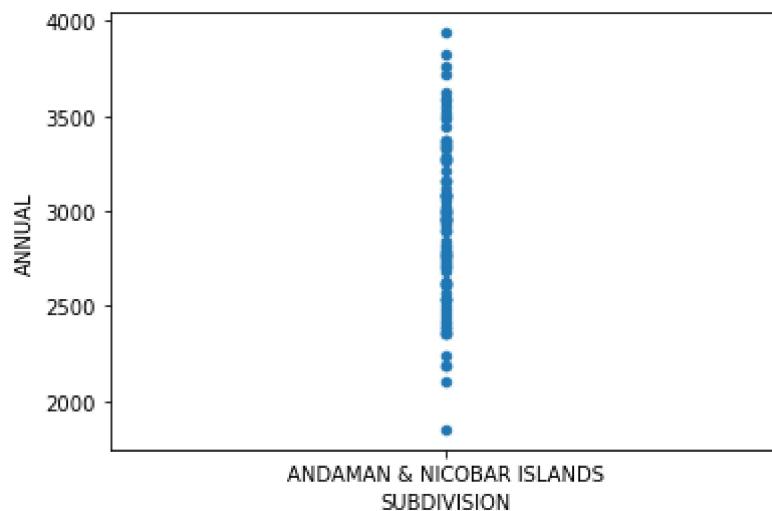




## Scatter Plot

```
In [68]: df.plot.scatter(x='SUBDIVISION',y='ANNUAL')
```

```
Out[68]: <AxesSubplot:xlabel='SUBDIVISION', ylabel='ANNUAL'>
```



```
In [69]: 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
```

In [70]: df.describe()

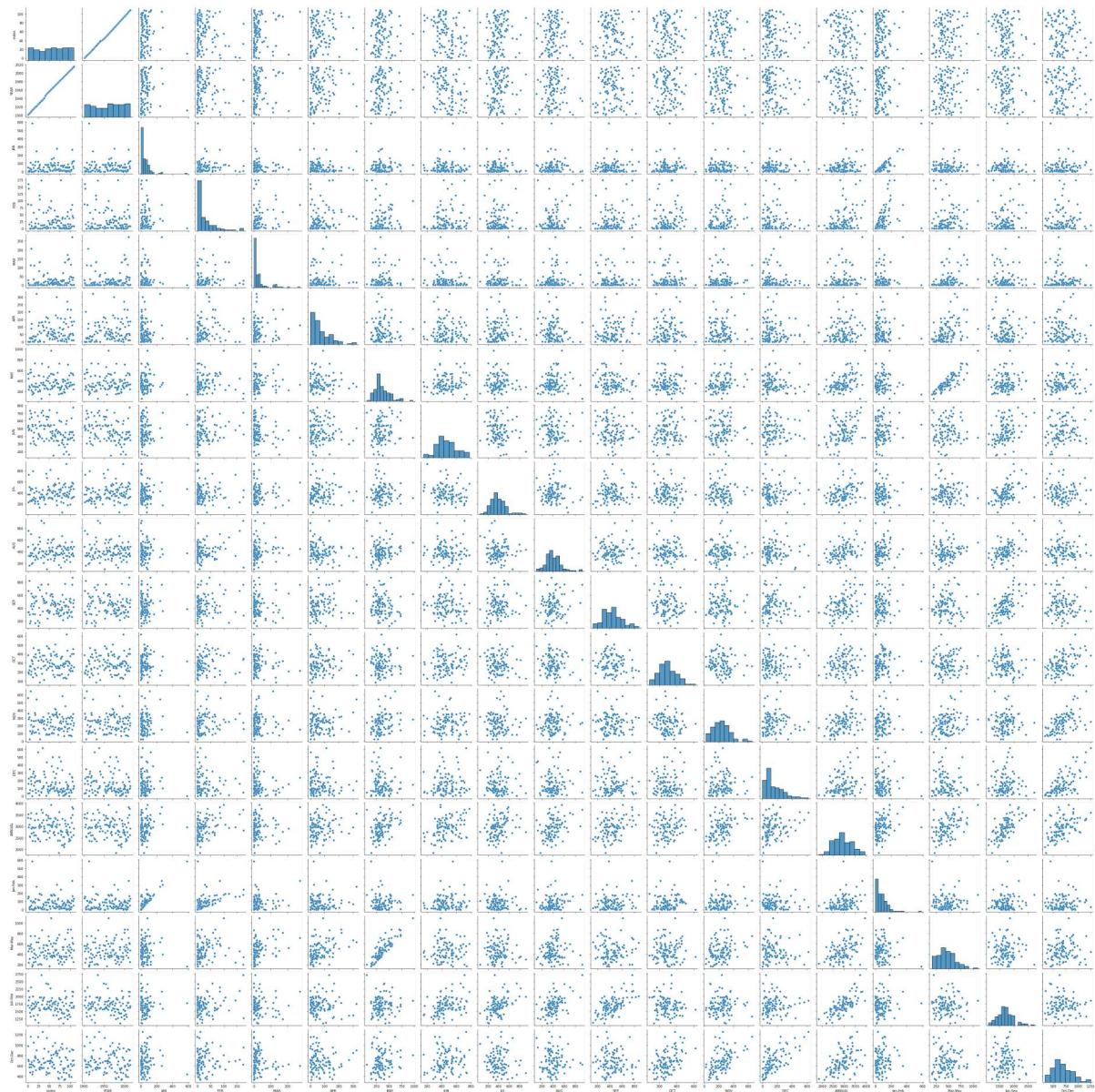
Out[70]:

	index	YEAR	JAN	FEB	MAR	APR	MAY
<b>count</b>	104.000000	104.000000	104.000000	104.000000	104.000000	104.000000	104.000000
<b>mean</b>	55.826923	1960.355769	53.829808	28.299038	31.080769	71.473077	361.098077
<b>std</b>	32.254884	34.010826	75.012392	38.286466	48.842153	66.908670	150.341139
<b>min</b>	0.000000	1901.000000	0.000000	0.000000	0.000000	0.000000	62.000000
<b>25%</b>	27.750000	1929.750000	10.200000	1.775000	2.300000	21.025000	263.125000
<b>50%</b>	57.500000	1963.500000	31.750000	12.800000	12.100000	52.300000	321.050000
<b>75%</b>	83.250000	1989.250000	76.275000	36.325000	31.775000	103.350000	425.325000
<b>max</b>	109.000000	2015.000000	583.700000	173.800000	272.800000	323.100000	973.100000

## EDA And Visualization

```
In [71]: sns.pairplot(df)
```

```
Out[71]: <seaborn.axisgrid.PairGrid at 0x1f31dc1e940>
```

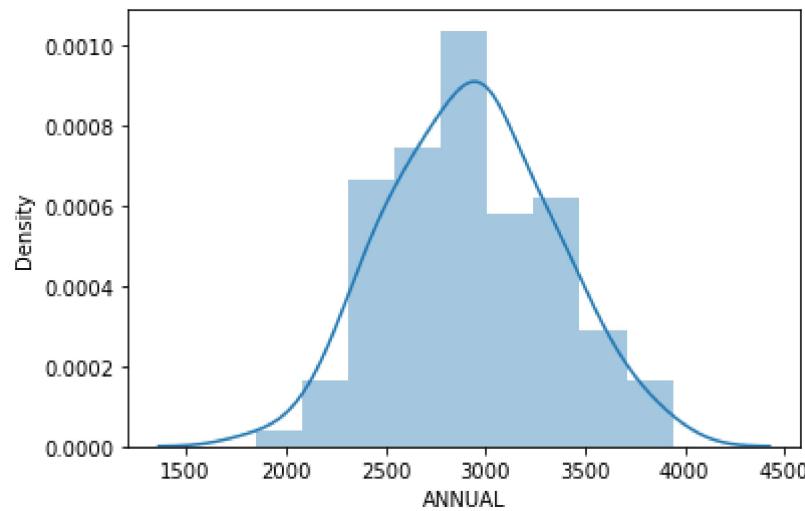


```
In [72]: sns.distplot(df[ 'ANNUAL' ])
```

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



```
In [73]: sns.heatmap(df.corr())
```

```
Out[73]: <AxesSubplot:>
```

