

20104028

Heamnath N

Importing Libraries

In [1]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

Importing Datasets

In [2]:

```
df=pd.read_csv("rainfall_kerala.csv")
df
```

Out[2]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
0	3887	KERALA	1901	28.7	44.7	51.6	160.0	174.7	824.6	743.0	357.5	197.7	266.9	350.0	
1	3888	KERALA	1902	6.7	2.6	57.3	83.9	134.5	390.9	1205.0	315.8	491.6	358.4	158.0	
2	3889	KERALA	1903	3.2	18.6	3.1	83.6	249.7	558.6	1022.5	420.2	341.8	354.1	157.0	
3	3890	KERALA	1904	23.7	3.0	32.2	71.5	235.7	1098.2	725.5	351.8	222.7	328.1	33.0	
4	3891	KERALA	1905	1.2	22.3	9.4	105.9	263.3	850.2	520.5	293.6	217.2	383.5	74.0	
...	
110	3997	KERALA	2011	20.5	45.7	24.1	165.2	124.2	788.5	536.8	492.7	391.2	227.2	169.0	
111	3998	KERALA	2012	7.4	11.0	21.0	171.1	95.3	430.3	362.6	501.6	241.1	187.5	112.0	
112	3999	KERALA	2013	3.9	40.1	49.9	49.3	119.3	1042.7	830.2	369.7	318.6	259.9	154.0	
113	4000	KERALA	2014	4.6	10.3	17.9	95.7	251.0	454.4	677.8	733.9	298.8	355.5	99.0	
114	4001	KERALA	2015	3.1	5.8	50.1	214.1	201.8	563.6	406.0	252.2	292.9	308.1	223.0	

115 rows × 20 columns



Data Cleaning and Data Preprocessing

In [3]:

```
df=df.dropna()
```

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')
```

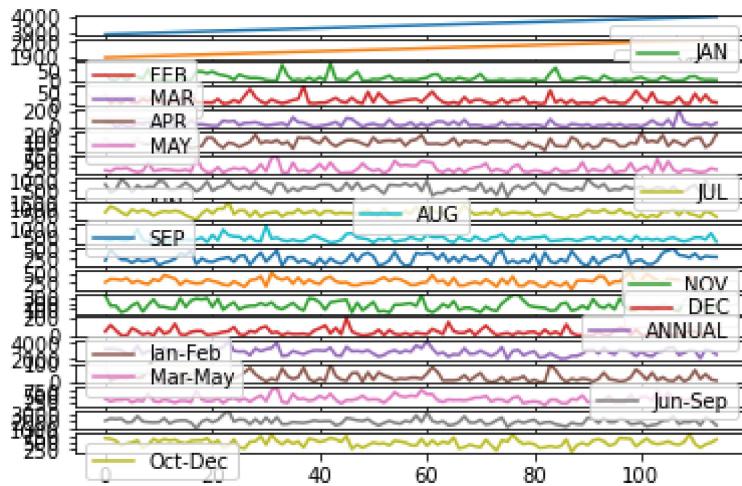
In [5]: df.info()

```
<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:>, <AxesSubplot:>,
   <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
   <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
   <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
   <AxesSubplot:>, <AxesSubplot:>], dtype=object)
```

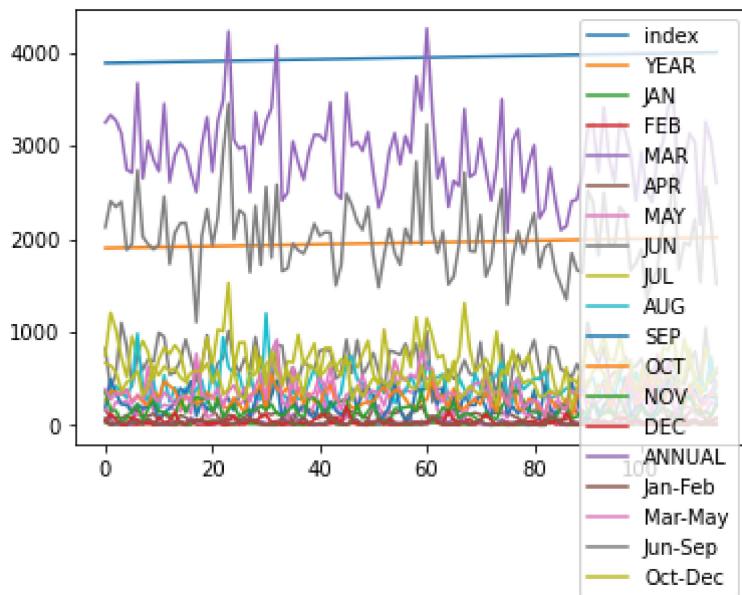


Line chart

In [7]:

```
df.plot.line()
```

Out[7]: <AxesSubplot:>

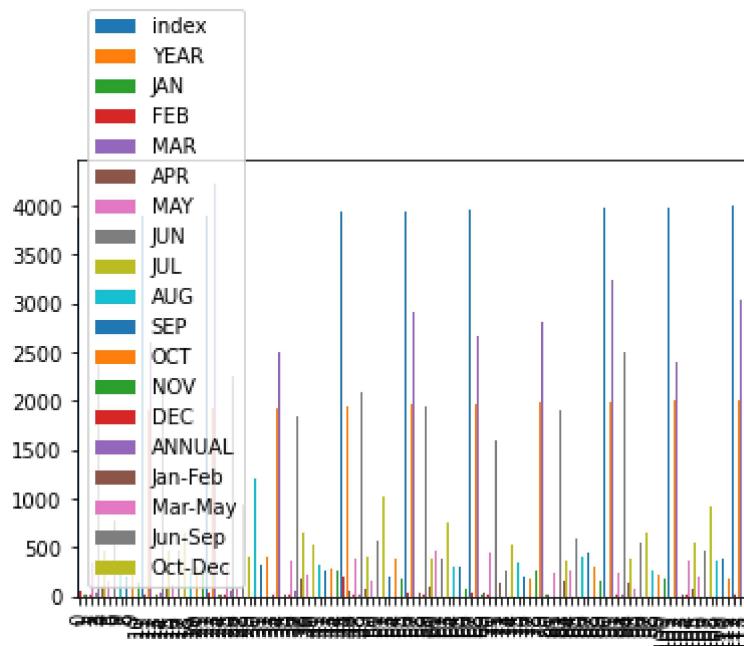


Bar chart

In [8]:

```
df.plot.bar()
```

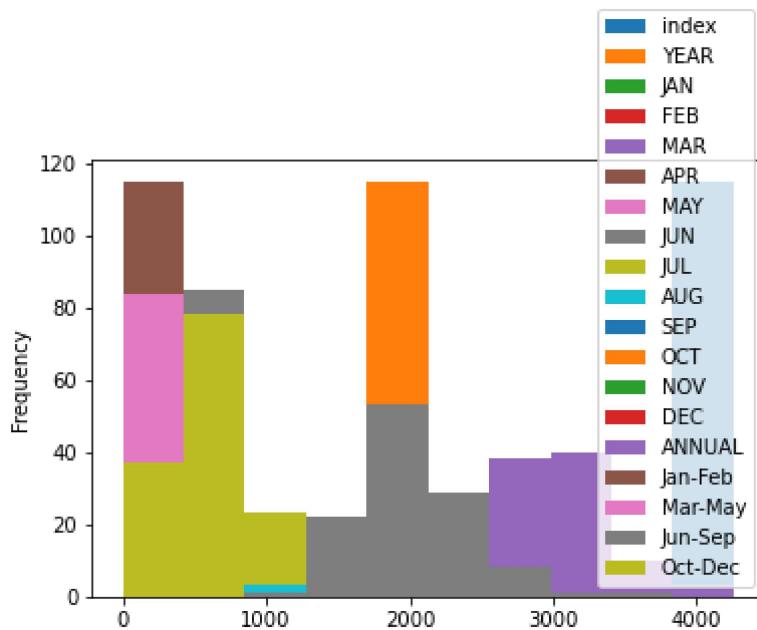
Out[8]: <AxesSubplot:>



Histogram

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

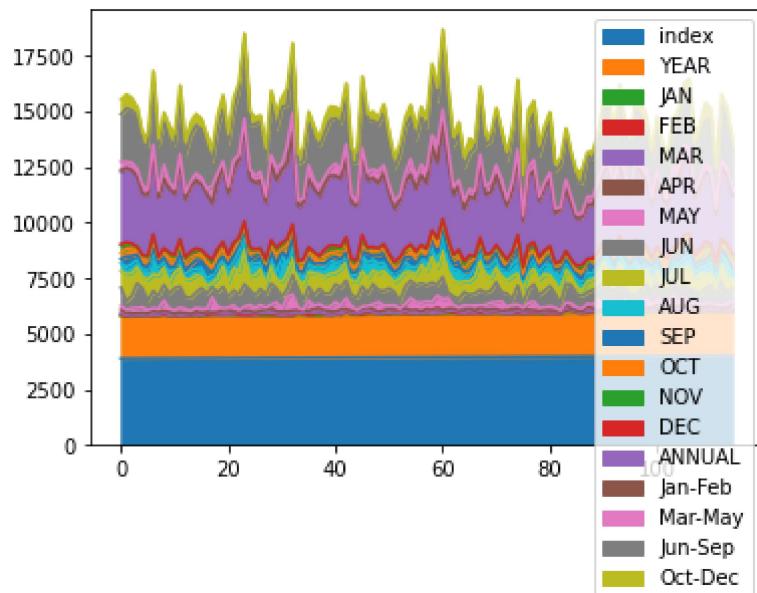
Out[9]: <AxesSubplot:ylabel='Frequency'>



Area chart

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

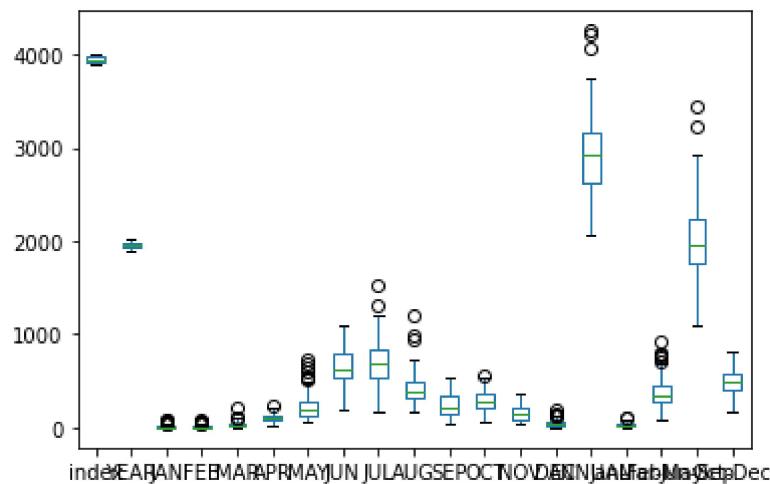
Out[10]: <AxesSubplot:>



Box chart

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

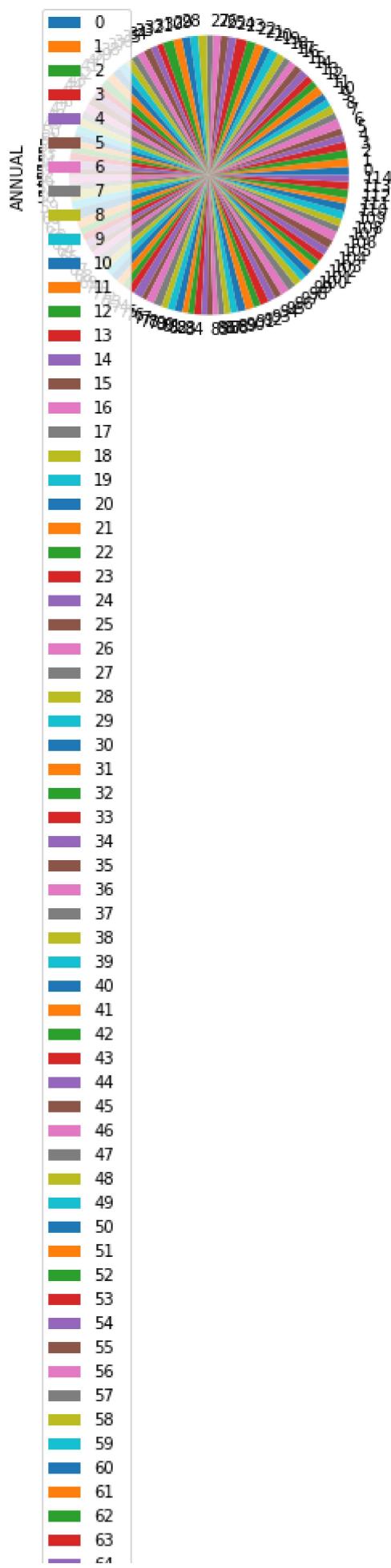
Out[11]: <AxesSubplot:>

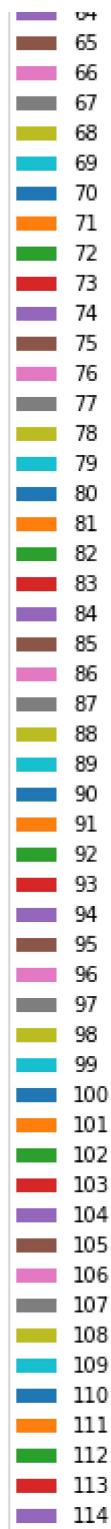


Pie chart

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

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

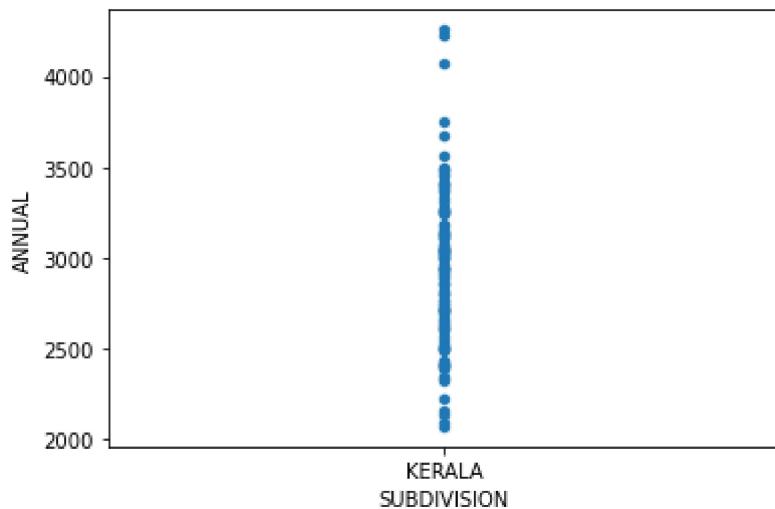




Scatter chart

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

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



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]:

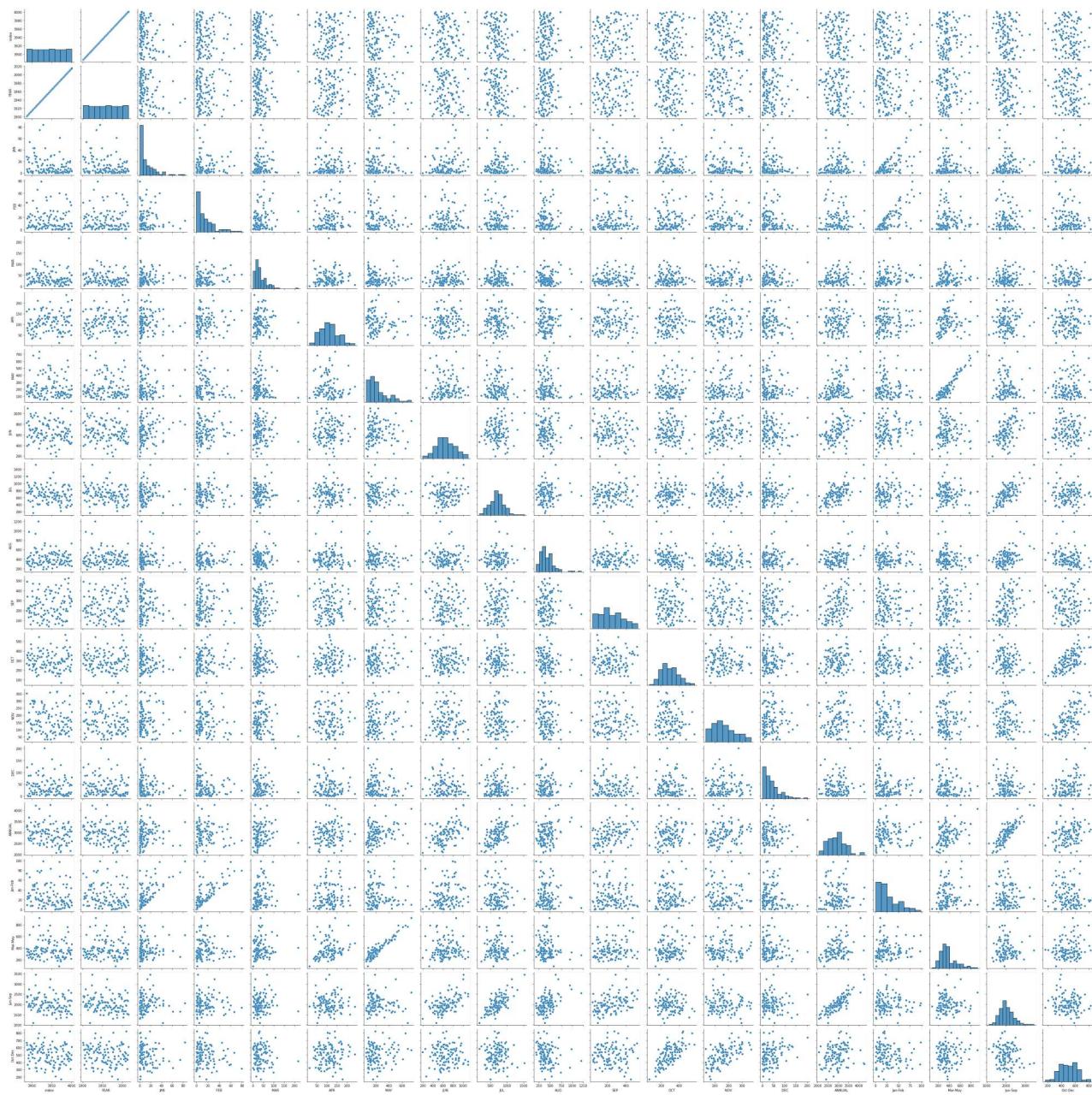
	index	YEAR	JAN	FEB	MAR	APR	MAY	JUL
count	115.000000	115.000000	115.000000	115.000000	115.000000	115.000000	115.000000	115.000000
mean	3944.000000	1958.000000	12.246957	15.496522	36.814783	110.573913	229.881739	654.30260
std	33.341666	33.341666	15.538923	16.206572	30.324601	44.673971	149.271697	187.64279
min	3887.000000	1901.000000	0.000000	0.000000	0.100000	13.100000	53.400000	196.80000
25%	3915.500000	1929.500000	2.250000	4.700000	18.100000	74.800000	124.350000	539.00000

	index	YEAR	JAN	FEB	MAR	APR	MAY	JUI
50%	3944.000000	1958.000000	6.000000	8.400000	28.300000	109.800000	185.400000	633.10000
75%	3972.500000	1986.500000	17.750000	21.400000	50.000000	136.000000	277.250000	791.50000
max	4001.000000	2015.000000	83.500000	79.000000	217.200000	238.000000	738.800000	1098.20000

EDA AND VISUALIZATION

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

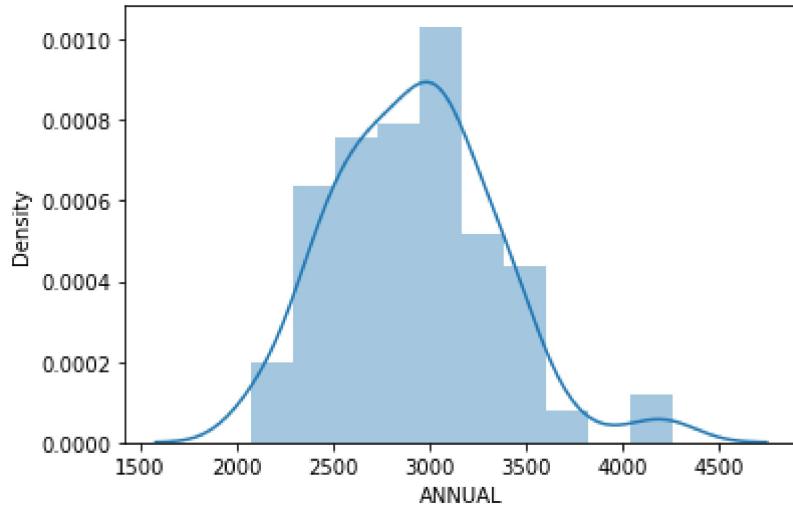
Out[16]: <seaborn.axisgrid.PairGrid at 0x2b09fd35fa0>



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) o  
r `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:>
```

