

# 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_himachal pradesh.csv")
df
```

Out[2]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV
0	1587	HIMACHAL PRADESH	1901	137.8	174.5	75.0	19.2	89.6	32.7	280.5	459.7	53.0	3.9	0.0
1	1588	HIMACHAL PRADESH	1902	6.5	27.0	104.4	76.2	61.3	78.8	258.6	199.3	113.4	23.6	2.5
2	1589	HIMACHAL PRADESH	1903	76.5	21.4	213.7	25.4	54.7	32.2	157.7	256.5	107.9	5.8	0.2
3	1590	HIMACHAL PRADESH	1904	79.3	22.4	131.7	48.0	90.3	33.1	241.1	184.3	56.4	51.6	17.3
4	1591	HIMACHAL PRADESH	1905	81.3	76.8	160.2	39.3	50.4	43.6	191.1	132.8	119.1	0.3	0.9
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
110	1697	HIMACHAL PRADESH	2011	43.9	97.4	49.7	62.4	45.1	118.3	177.7	380.2	120.3	6.0	0.3
111	1698	HIMACHAL PRADESH	2012	92.3	51.3	28.4	55.9	9.4	31.1	241.5	280.6	133.1	3.1	3.2
112	1699	HIMACHAL PRADESH	2013	79.9	182.6	76.6	28.9	32.6	233.6	208.8	240.0	65.8	21.8	16.6
113	1700	HIMACHAL PRADESH	2014	69.6	124.9	125.2	60.6	68.9	51.7	203.6	146.7	84.6	19.3	4.5
114	1701	HIMACHAL PRADESH	2015	67.2	156.6	192.5	84.9	45.0	85.8	249.9	195.9	75.5	17.7	14.5

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')
```

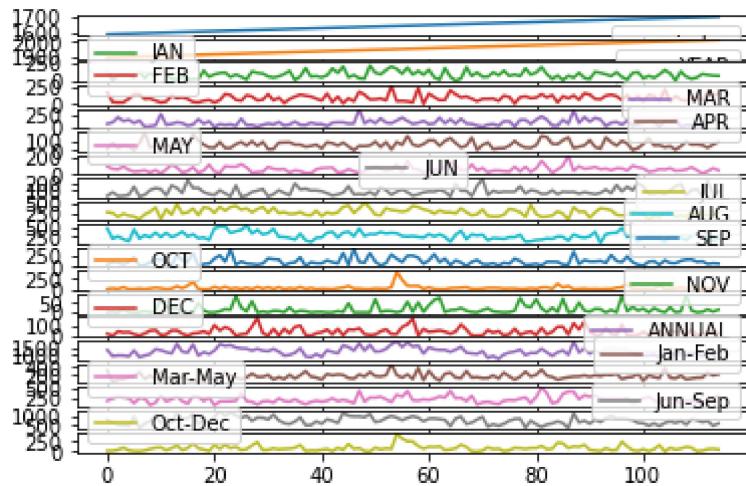
```
In [5]: df.info()
```

```
<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)
```

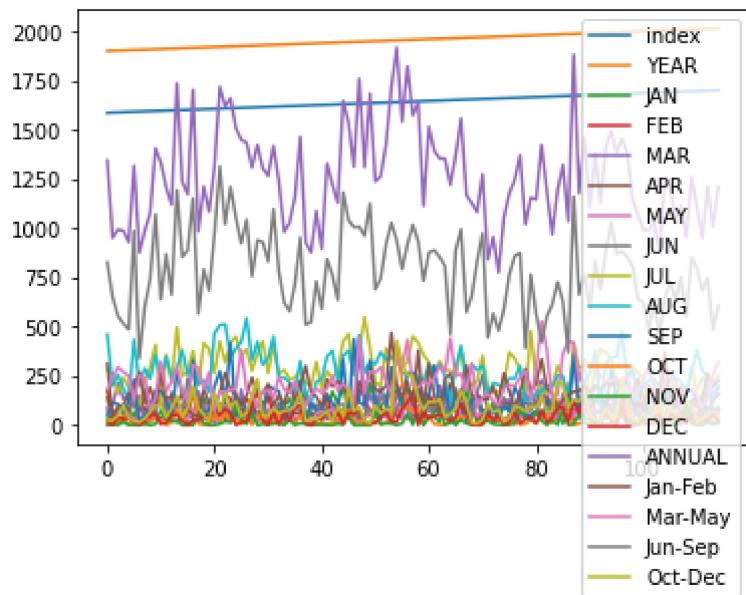


## Line chart

In [7]:

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

Out[7]: &lt;AxesSubplot:&gt;

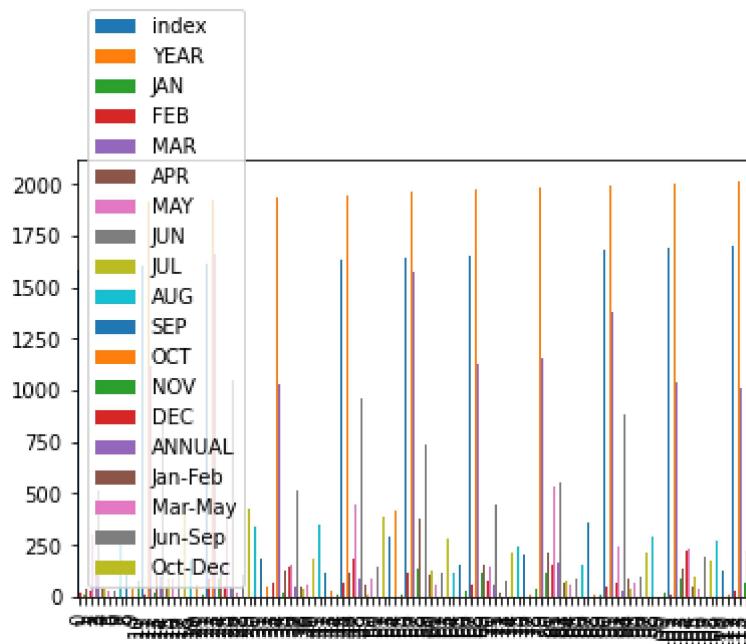


## Bar chart

In [8]:

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

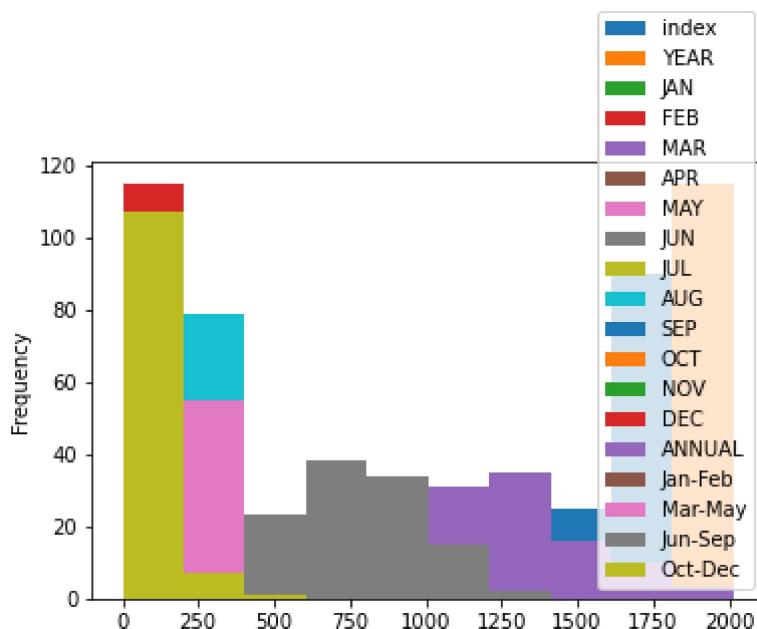
Out[8]: &lt;AxesSubplot:&gt;



## Histogram

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

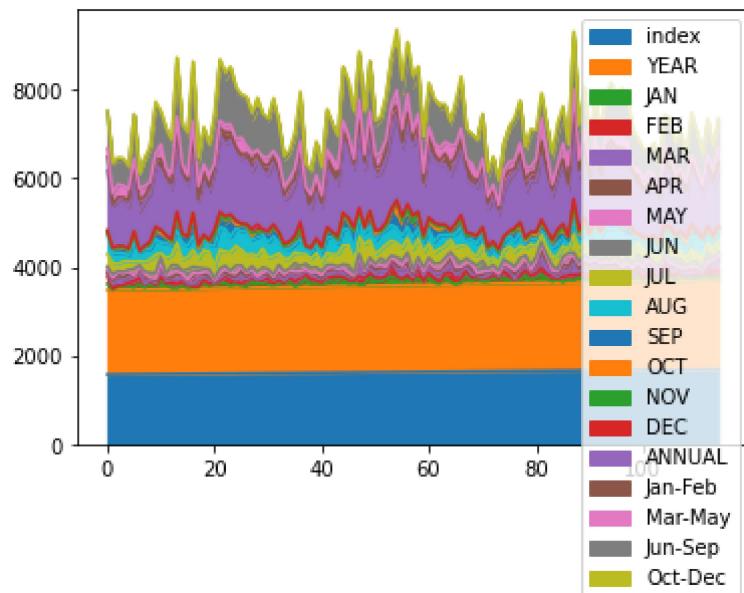
```
Out[9]: <AxesSubplot:ylabel='Frequency'>
```



## Area chart

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

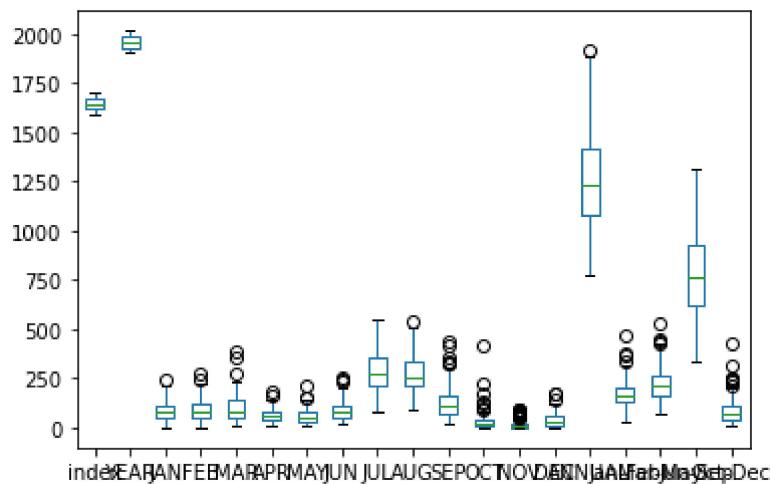
```
Out[10]: <AxesSubplot:>
```



## Box chart

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

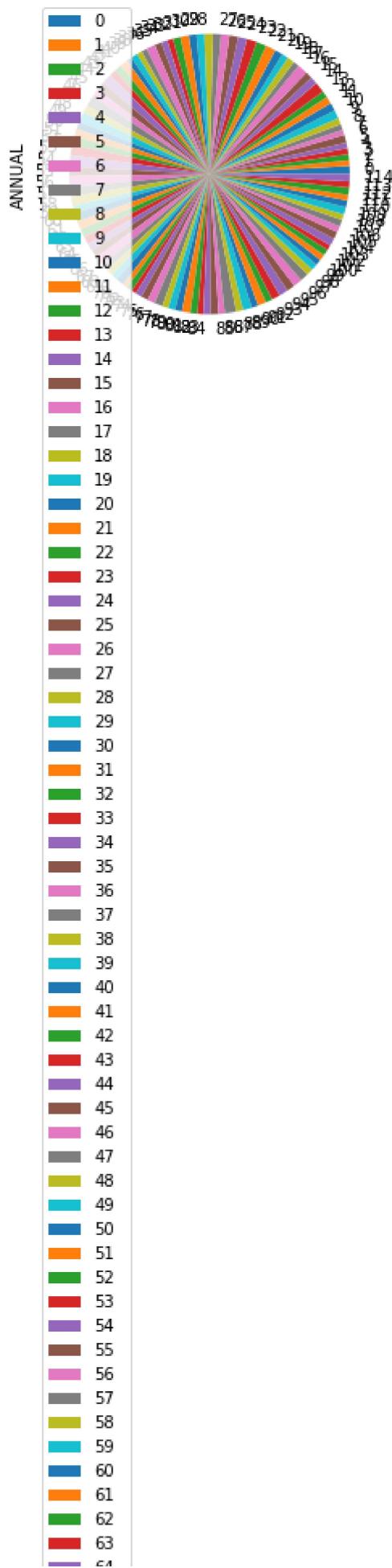
```
Out[11]: <AxesSubplot:>
```

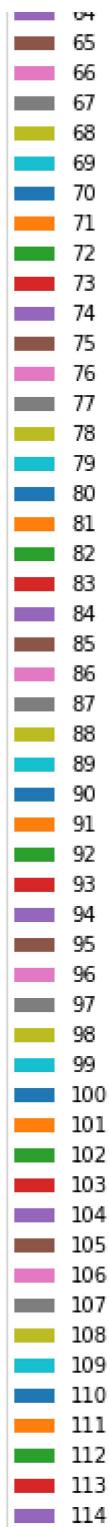


## Pie chart

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

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

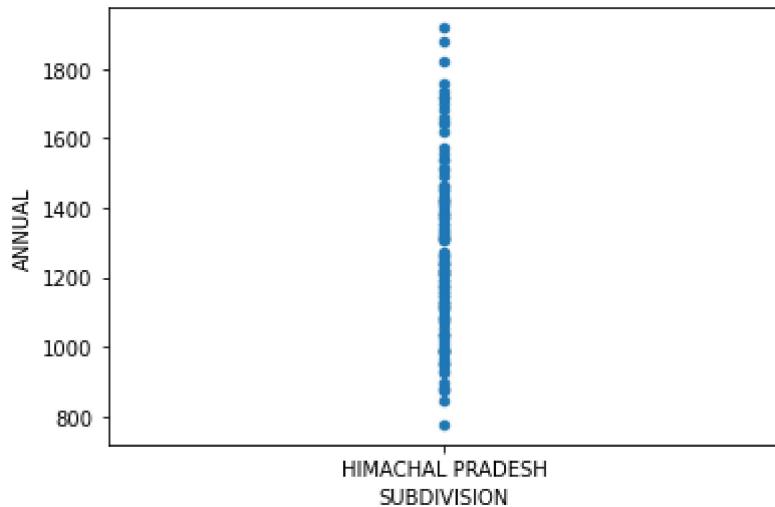




## Scatter chart

```
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]:

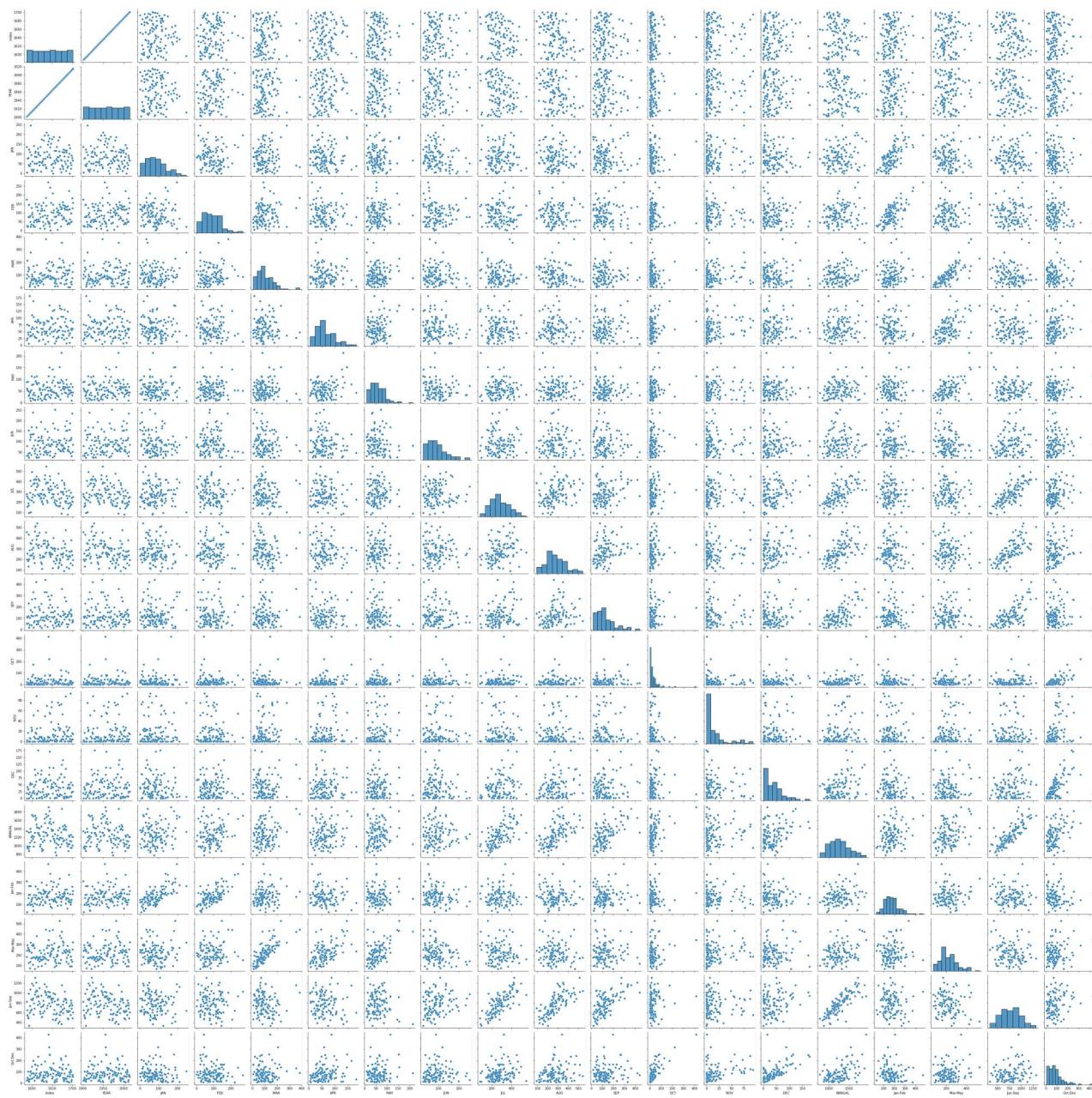
	index	YEAR	JAN	FEB	MAR	APR	MAY	JUN
<b>count</b>	115.000000	115.000000	115.000000	115.000000	115.000000	115.000000	115.000000	115.000000
<b>mean</b>	1644.000000	1958.000000	84.189565	90.894783	101.146087	62.428696	58.156522	91.220870
<b>std</b>	33.341666	33.341666	51.919380	52.257987	66.508952	35.885632	33.972489	49.014057
<b>min</b>	1587.000000	1901.000000	0.300000	0.700000	5.900000	4.500000	8.800000	23.700000
<b>25%</b>	1615.500000	1929.500000	45.100000	50.350000	54.350000	35.350000	34.650000	53.550000

	index	YEAR	JAN	FEB	MAR	APR	MAY	JUN
<b>50%</b>	1644.000000	1958.000000	78.000000	82.800000	83.600000	55.900000	54.300000	84.000000
<b>75%</b>	1672.500000	1986.500000	113.950000	124.800000	137.100000	84.750000	78.600000	111.550000
<b>max</b>	1701.000000	2015.000000	246.300000	271.800000	382.000000	181.700000	214.200000	252.700000

## EDA AND VISUALIZATION

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

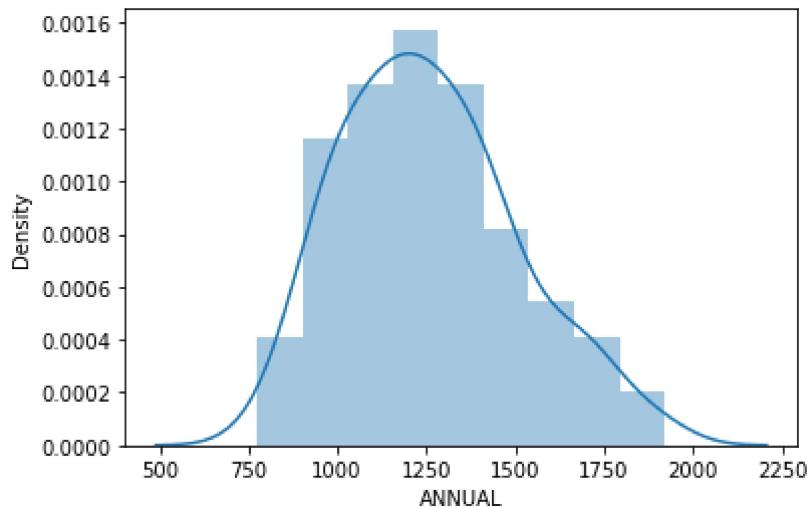
Out[16]: <seaborn.axisgrid.PairGrid at 0x23265badf70>



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:>
```

