

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_east rajasthan.csv")
df
```

Out[2]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT
0	1932	EAST RAJASTHAN	1901	21.6	8.9	2.9	0.7	5.0	15.0	164.8	175.6	7.5	9.8
1	1933	EAST RAJASTHAN	1902	4.1	0.7	0.0	1.8	9.9	34.6	247.6	116.7	145.6	14.4
2	1934	EAST RAJASTHAN	1903	1.9	0.7	1.3	0.1	12.9	15.6	238.2	229.1	168.5	17.8
3	1935	EAST RAJASTHAN	1904	4.3	5.5	21.7	0.2	27.5	49.9	289.7	223.5	50.2	1.5
4	1936	EAST RAJASTHAN	1905	4.1	8.8	3.2	1.6	2.0	14.4	130.5	30.9	83.8	0.0
...
110	2042	EAST RAJASTHAN	2011	0.0	11.2	0.2	0.5	5.1	140.9	193.6	284.1	166.4	0.0
111	2043	EAST RAJASTHAN	2012	1.9	0.0	0.0	3.6	9.5	11.2	170.5	365.0	131.3	0.5
112	2044	EAST RAJASTHAN	2013	1.4	21.7	0.4	3.2	1.0	90.6	319.0	278.5	88.0	30.6
113	2045	EAST RAJASTHAN	2014	28.4	10.0	6.4	7.3	8.4	23.5	197.1	261.0	136.9	3.2
114	2046	EAST RAJASTHAN	2015	12.1	0.1	55.9	15.9	3.5	96.4	297.6	142.8	20.1	5.0

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')
```

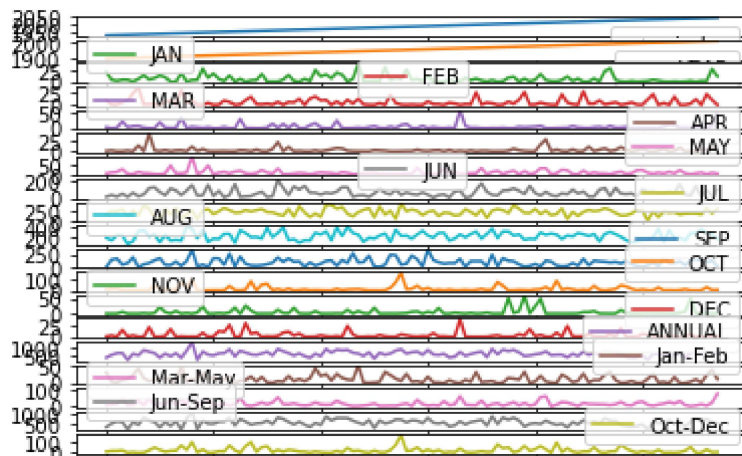
```
In [5]: df.info()
```

```
<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)`

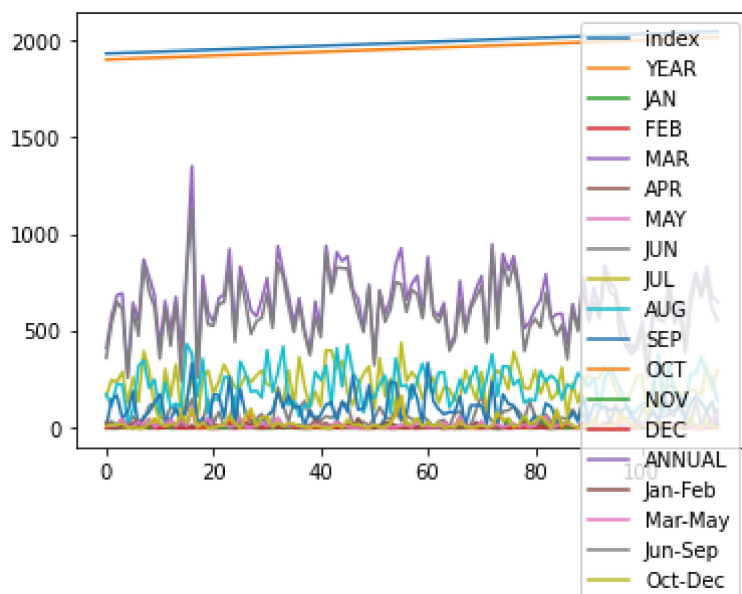
Out[6]: array([<AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>], dtype=object)



Line chart

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

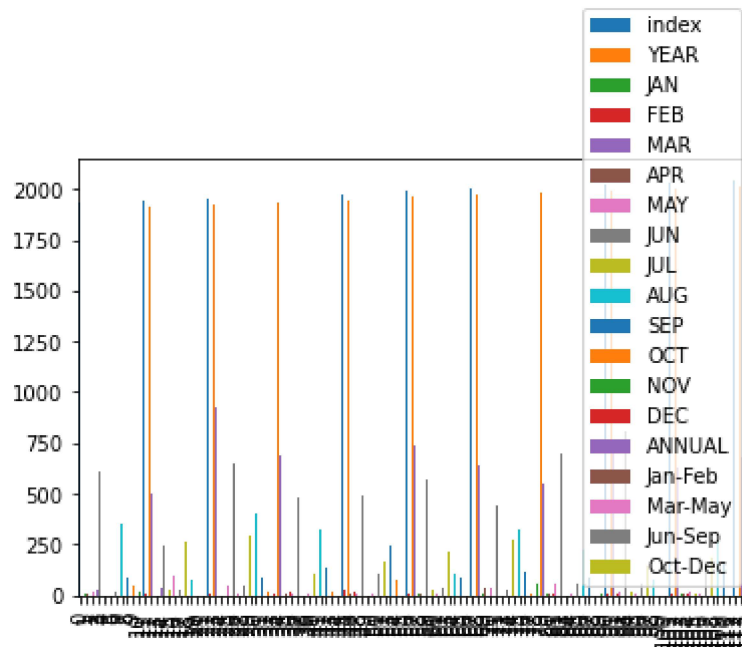
Out[7]: <AxesSubplot:>



Bar chart

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

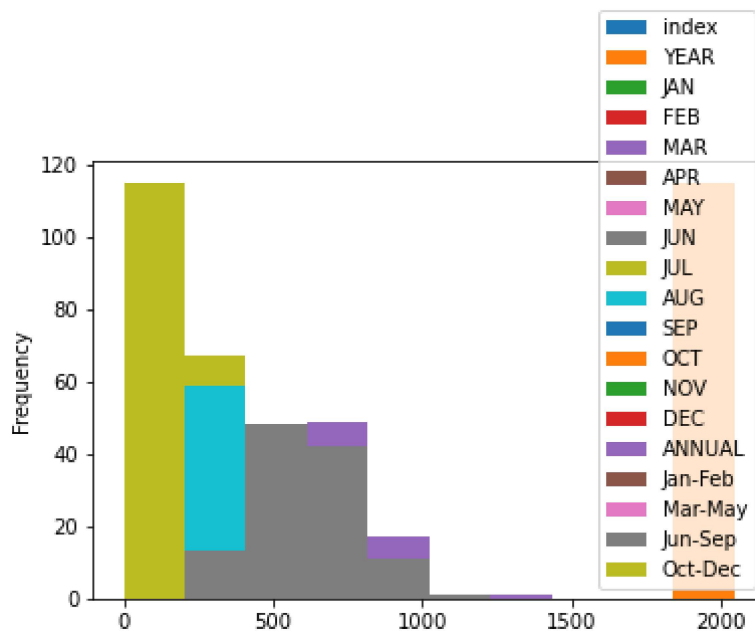
```
Out[8]: <AxesSubplot:>
```



Histogram

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

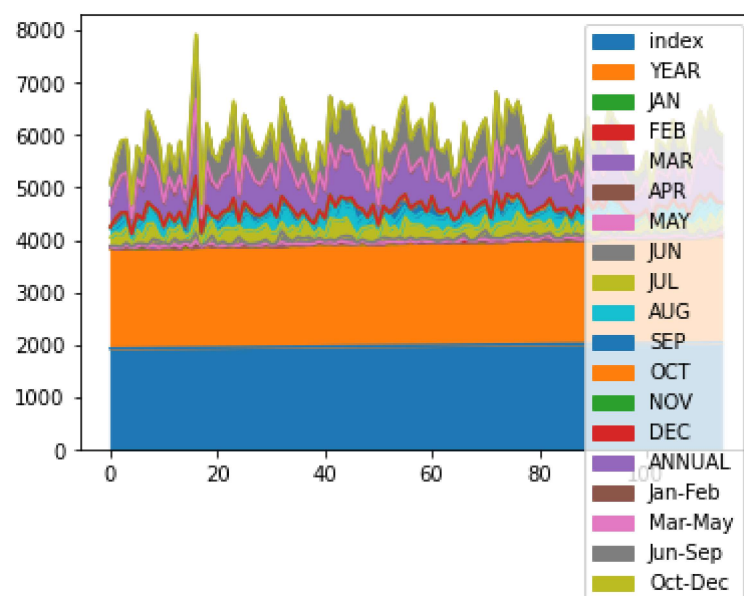
```
Out[9]: <AxesSubplot:ylabel='Frequency'>
```



Area chart

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

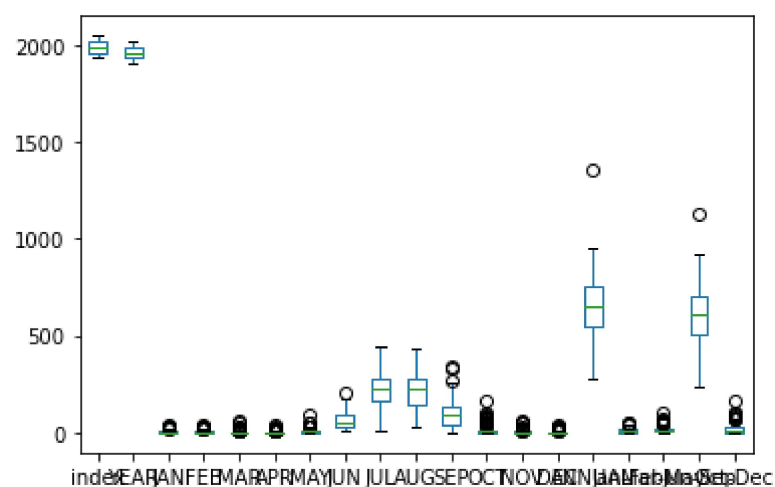
```
Out[10]: <AxesSubplot:>
```



Box chart

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

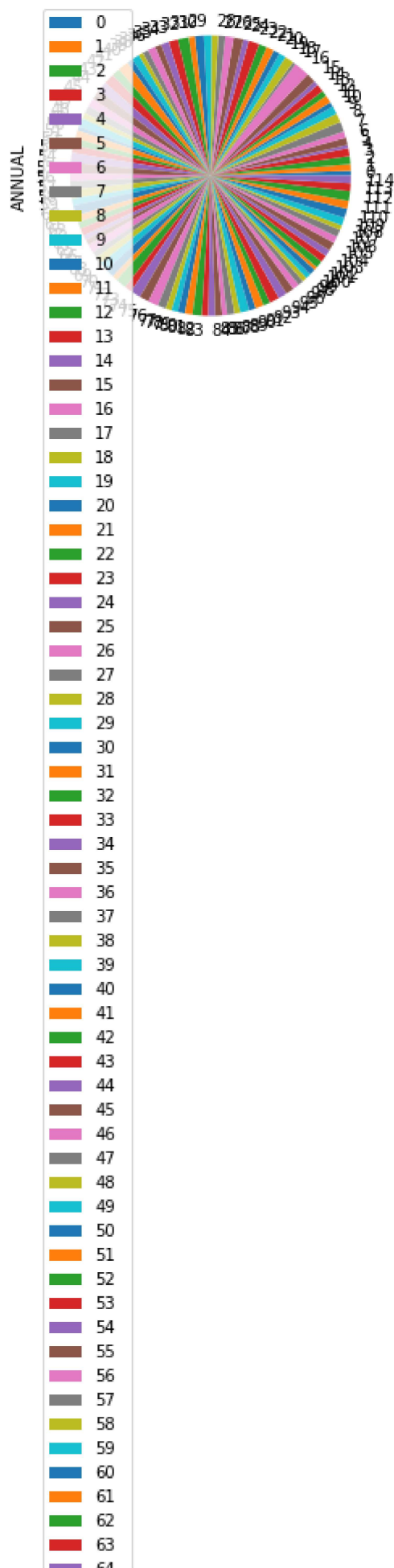
```
Out[11]: <AxesSubplot:>
```

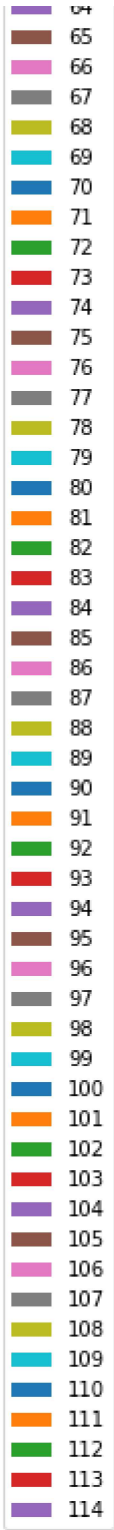


Pie chart

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

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

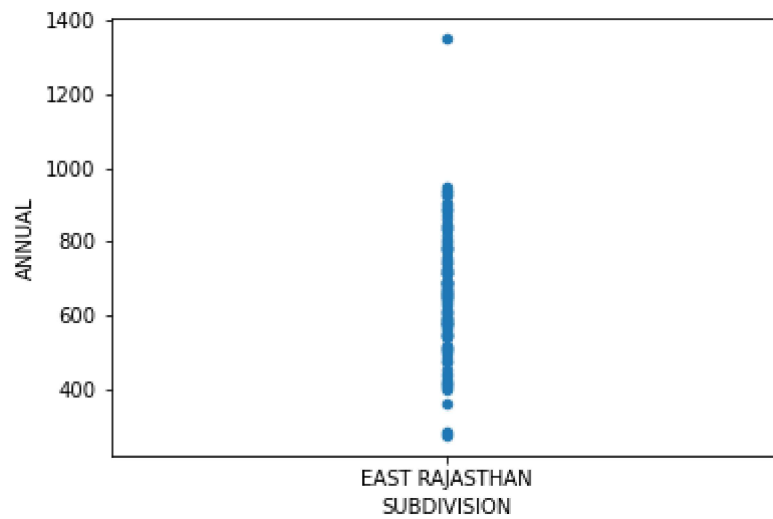





Scatter chart

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

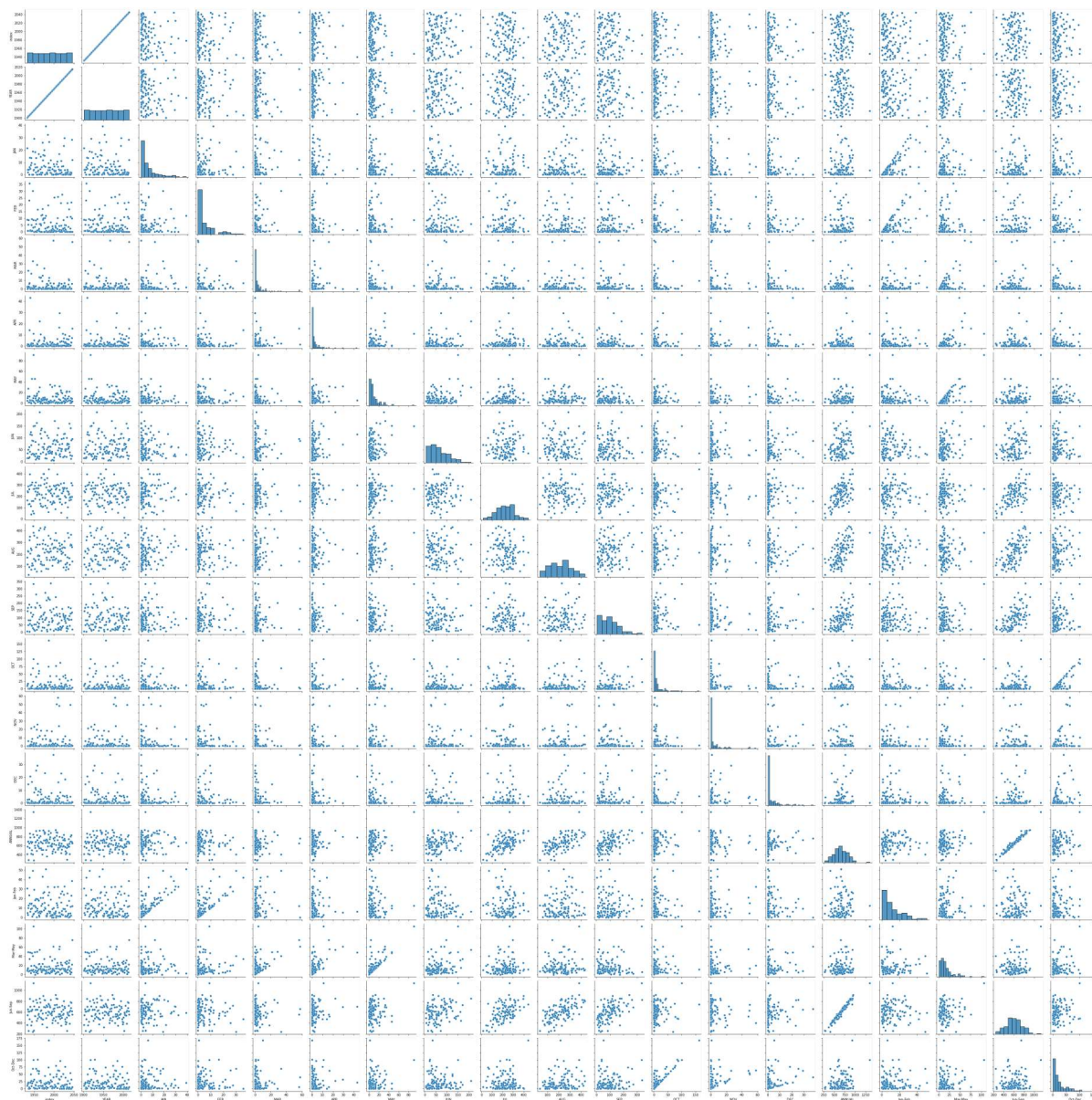
```
Out[13]: <AxesSubplot:xlabel='SUBDIVISION', ylabel='ANNUAL'>
```



Seaborn

```
In [14]: sns.pairplot(df)
```

```
Out[14]: <seaborn.axisgrid.PairGrid at 0x1b5a702fbe0>
```

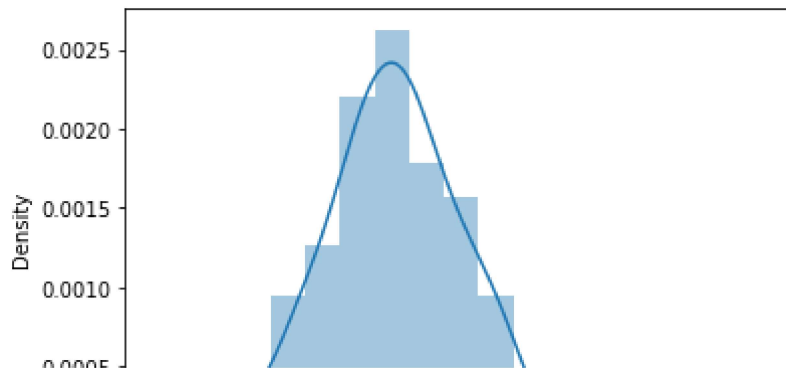


In [15]: `sns.distplot(df['ANNUAL'])`

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



In [16]: `sns.heatmap(df.corr())`

Out[16]: `<AxesSubplot:>`

