```python
In [1]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import seaborn as sns
```

```python
In [2]: data=pd.read_csv(r"C:\Users\user\Desktop\vicky\rainfall\rainfall in india 1901-2015.csv")[2854:296
```

```python
In [3]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 113 entries, 2854 to 2966
Data columns (total 20 columns):
 #   Column       Non-Null Count   Dtype
---  ------       --------------   -----
 0   index        113 non-null     int64
 1   SUBDIVISION  113 non-null     object
 2   YEAR         113 non-null     int64
 3   JAN          113 non-null     float64
 4   FEB          113 non-null     float64
 5   MAR          113 non-null     float64
 6   APR          113 non-null     float64
 7   MAY          113 non-null     float64
 8   JUN          113 non-null     float64
 9   JUL          113 non-null     float64
 10  AUG          113 non-null     float64
 11  SEP          113 non-null     float64
 12  OCT          113 non-null     float64
 13  NOV          113 non-null     float64
 14  DEC          113 non-null     float64
 15  ANNUAL       113 non-null     float64
 16  Jan-Feb      113 non-null     float64
 17  Mar-May      113 non-null     float64
 18  Jun-Sep      113 non-null     float64
 19  Oct-Dec      113 non-null     float64
dtypes: float64(17), int64(2), object(1)
memory usage: 17.8+ KB
```

```python
In [4]: data.head()
```

Out[4]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL
2854	2854	VIDARBHA	1903	5.2	4.0	0.1	2.5	37.8	121.2	475.5	325.5	154.8	100.8	2.0	0.0	1229.4
2855	2855	VIDARBHA	1904	4.3	2.4	12.9	0.2	14.8	148.9	158.3	151.8	196.9	61.7	0.0	0.9	753.2
2856	2856	VIDARBHA	1905	7.3	12.7	12.4	16.2	14.0	81.0	254.5	216.3	321.3	6.0	0.2	0.0	941.8
2857	2857	VIDARBHA	1906	12.9	3.3	11.9	0.2	6.1	328.1	350.4	325.8	113.3	6.9	14.3	18.1	1191.2
2858	2858	VIDARBHA	1907	3.6	52.9	3.7	45.3	1.6	181.4	309.7	332.6	59.7	0.2	15.7	4.3	1010.8

In [5]:
```python
data.tail()
```

Out[5]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL
2962	2962	VIDARBHA	2011	0.0	1.2	0.1	7.7	0.6	137.9	247.1	302.8	191.0	4.7	0.0	0.0	893.2
2963	2963	VIDARBHA	2012	3.1	0.1	0.0	0.6	0.2	125.5	370.5	316.2	249.4	34.9	7.3	0.0	1107.7
2964	2964	VIDARBHA	2013	6.6	13.0	3.8	2.8	0.5	366.7	535.5	326.1	131.7	133.5	0.0	0.0	1520.0
2965	2965	VIDARBHA	2014	1.2	18.3	49.6	2.6	4.0	63.3	337.6	191.7	224.9	17.3	6.2	2.3	919.1
2966	2966	VIDARBHA	2015	26.3	4.7	66.3	28.1	12.8	254.6	137.2	288.9	167.5	7.0	0.0	0.2	993.8

In [6]:
```python
data.shape
```

Out[6]: (113, 20)

In [7]:
```python
new_data=data.fillna(value=1)
new_data
```

Out[7]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL
2854	2854	VIDARBHA	1903	5.2	4.0	0.1	2.5	37.8	121.2	475.5	325.5	154.8	100.8	2.0	0.0	1229.4
2855	2855	VIDARBHA	1904	4.3	2.4	12.9	0.2	14.8	148.9	158.3	151.8	196.9	61.7	0.0	0.9	753.2
2856	2856	VIDARBHA	1905	7.3	12.7	12.4	16.2	14.0	81.0	254.5	216.3	321.3	6.0	0.2	0.0	941.8
2857	2857	VIDARBHA	1906	12.9	3.3	11.9	0.2	6.1	328.1	350.4	325.8	113.3	6.9	14.3	18.1	1191.2
2858	2858	VIDARBHA	1907	3.6	52.9	3.7	45.3	1.6	181.4	309.7	332.6	59.7	0.2	15.7	4.3	1010.8
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2962	2962	VIDARBHA	2011	0.0	1.2	0.1	7.7	0.6	137.9	247.1	302.8	191.0	4.7	0.0	0.0	893.2
2963	2963	VIDARBHA	2012	3.1	0.1	0.0	0.6	0.2	125.5	370.5	316.2	249.4	34.9	7.3	0.0	1107.7
2964	2964	VIDARBHA	2013	6.6	13.0	3.8	2.8	0.5	366.7	535.5	326.1	131.7	133.5	0.0	0.0	1520.0
2965	2965	VIDARBHA	2014	1.2	18.3	49.6	2.6	4.0	63.3	337.6	191.7	224.9	17.3	6.2	2.3	919.1
2966	2966	VIDARBHA	2015	26.3	4.7	66.3	28.1	12.8	254.6	137.2	288.9	167.5	7.0	0.0	0.2	993.8

113 rows × 20 columns

In [8]:
```python
new_data.index
```

Out[8]: RangeIndex(start=2854, stop=2967, step=1)

In [9]:
```python
new_data.columns
```

Out[9]:
```
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 [10]: new_data.plot.line()
```

Out[10]: <AxesSubplot:>

```
In [11]: new_data.plot.bar()
```

Out[11]: <AxesSubplot:>

In [12]: `new_data.plot.area()`

Out[12]: `<AxesSubplot:>`

In [13]: `new_data.plot.hist()`

Out[13]: `<AxesSubplot:ylabel='Frequency'>`

In [14]:
```
new_data.plot.pie(y='ANNUAL')
```

Out[14]: <AxesSubplot:ylabel='ANNUAL'>

In [14]:
```
new_data.plot.pie(y='ANNUAL')
```

Out[14]: <AxesSubplot:ylabel='ANNUAL'>

In [15]: `new_data.plot.scatter(x='YEAR',y='ANNUAL')`

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

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

Out[16]:   <seaborn.axisgrid.PairGrid at 0x1f433542580>

In [17]: `sns.distplot(data['YEAR'])`

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2557: FutureWarning: `distplo
t` is a deprecated function and will be removed in a future version. Please adapt your code to us
e either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-leve
l function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[17]: `<AxesSubplot:xlabel='YEAR', ylabel='Density'>`

In [18]: `sns.heatmap(new_data.corr())`

Out[18]: `<AxesSubplot:>`