

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

```
In [2]: data=pd.read_csv(r"C:\Users\user\Desktop\vicky\rainfall\rainfall in india 1901-2015.csv")[1474:1587]
```

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

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 114 entries, 1474 to 1587
Data columns (total 20 columns):
#   Column          Non-Null Count  Dtype  
---  -
0   index           114 non-null   int64  
1   SUBDIVISION     114 non-null   object  
2   YEAR            114 non-null   int64  
3   JAN             114 non-null   float64 
4   FEB             114 non-null   float64 
5   MAR             114 non-null   float64 
6   APR             114 non-null   float64 
7   MAY             114 non-null   float64 
8   JUN             114 non-null   float64 
9   JUL             114 non-null   float64 
10  AUG             114 non-null   float64 
11  SEP             114 non-null   float64 
12  OCT             114 non-null   float64 
13  NOV             114 non-null   float64 
14  DEC             114 non-null   float64 
15  ANNUAL          114 non-null   float64 
16  Jan-Feb         114 non-null   float64 
17  Mar-May         114 non-null   float64 
18  Jun-Sep         114 non-null   float64 
19  Oct-Dec         114 non-null   float64 
dtypes: float64(17), int64(2), object(1)
memory usage: 17.9+ KB
```

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

Out[4]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL	
1474	1474	PUNJAB	1903	29.5	0.5	45.0	1.3	9.2	5.2	212.2	119.1	132.5	6.9	0.0	9.5	571.0	2
1475	1475	PUNJAB	1904	24.2	1.7	87.8	1.2	13.8	22.0	59.9	124.0	73.8	7.4	9.8	25.9	451.5	2
1476	1476	PUNJAB	1905	53.0	40.3	24.3	0.5	2.2	19.2	122.6	50.3	111.1	1.2	0.0	9.4	434.3	3
1477	1477	PUNJAB	1906	5.3	83.1	50.5	1.5	4.7	38.3	99.4	190.5	181.5	0.5	0.0	22.2	677.5	8
1478	1478	PUNJAB	1907	27.3	95.4	53.1	50.7	3.3	20.2	82.8	249.0	6.8	0.4	0.0	0.0	589.2	12

```
In [5]: data.shape
```

Out[5]: (114, 20)

```
In [6]: new_data=data.fillna(value=1)
new_data
```

Out[6]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL
1474	1474	PUNJAB	1903	29.5	0.5	45.0	1.3	9.2	5.2	212.2	119.1	132.5	6.9	0.0	9.5	571.0
1475	1475	PUNJAB	1904	24.2	1.7	87.8	1.2	13.8	22.0	59.9	124.0	73.8	7.4	9.8	25.9	451.5
1476	1476	PUNJAB	1905	53.0	40.3	24.3	0.5	2.2	19.2	122.6	50.3	111.1	1.2	0.0	9.4	434.3
1477	1477	PUNJAB	1906	5.3	83.1	50.5	1.5	4.7	38.3	99.4	190.5	181.5	0.5	0.0	22.2	677.5
1478	1478	PUNJAB	1907	27.3	95.4	53.1	50.7	3.3	20.2	82.8	249.0	6.8	0.4	0.0	0.0	589.2
...
1583	1583	PUNJAB	2012	62.6	3.2	1.9	31.1	1.6	11.9	120.2	135.1	112.3	2.2	0.4	11.0	493.6
1584	1584	PUNJAB	2013	9.3	50.1	11.6	3.4	3.6	120.3	117.9	217.1	24.4	16.2	6.1	6.6	586.6
1585	1585	PUNJAB	2014	21.8	20.1	30.3	24.5	20.8	20.6	76.3	41.9	105.8	6.0	0.7	14.1	382.7
1586	1586	PUNJAB	2015	17.7	31.3	68.5	29.8	16.7	48.3	130.2	88.6	69.2	9.0	0.8	0.7	510.8
1587	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	19.1	1345.1

114 rows × 20 columns

```
In [7]: new_data.index
```

Out[7]: RangeIndex(start=1474, stop=1588, step=1)

```
In [8]: new_data.columns
```

Out[8]: 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 [9]: new_data.plot.line()
```

Out[9]: <AxesSubplot:>

```
In [10]: new_data.plot.bar()
```

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

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

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

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

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

```
In [13]: new_data.plot.pie(y='ANNUAL')
```

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



```
In [14]: new_data.plot.scatter(x='YEAR',y='ANNUAL')
```

```
Out[14]: <AxesSubplot:xlabel='YEAR', ylabel='ANNUAL'>
```



```
In [15]: sns.pairplot(new_data)
```

```
Out[15]: <seaborn.axisgrid.PairGrid at 0x2baec6c1700>
```

```
In [16]: sns.distplot(data['YEAR'])
```

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

```
In [17]: sns.heatmap(new_data.corr())
```

```
Out[17]: <AxesSubplot:>
```