20104016

DEENA

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_punjab.csv")
df
```

Out[2]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT
0	1472	PUNJAB	1901	55.7	50.1	25.2	2.1	25.2	10.4	178.2	145.0	24.4	3.7
1	1473	PUNJAB	1902	0.0	0.8	9.9	10.9	29.6	49.9	125.6	94.9	67.2	9.0
2	1474	PUNJAB	1903	29.5	0.5	45.0	1.3	9.2	5.2	212.2	119.1	132.5	6.9
3	1475	PUNJAB	1904	24.2	1.7	87.8	1.2	13.8	22.0	59.9	124.0	73.8	7.4
4	1476	PUNJAB	1905	53.0	40.3	24.3	0.5	2.2	19.2	122.6	50.3	111.1	1.2
...	...	...	...	...	...	...	...	...	...	...	...	...	...
110	1582	PUNJAB	2011	3.5	35.6	8.2	17.8	18.9	162.9	120.9	193.5	140.2	0.0
111	1583	PUNJAB	2012	62.6	3.2	1.9	31.1	1.6	11.9	120.2	135.1	112.3	2.2
112	1584	PUNJAB	2013	9.3	50.1	11.6	3.4	3.6	120.3	117.9	217.1	24.4	16.2
113	1585	PUNJAB	2014	21.8	20.1	30.3	24.5	20.8	20.6	76.3	41.9	105.8	6.0
114	1586	PUNJAB	2015	17.7	31.3	68.5	29.8	16.7	48.3	130.2	88.6	69.2	9.0

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')
```

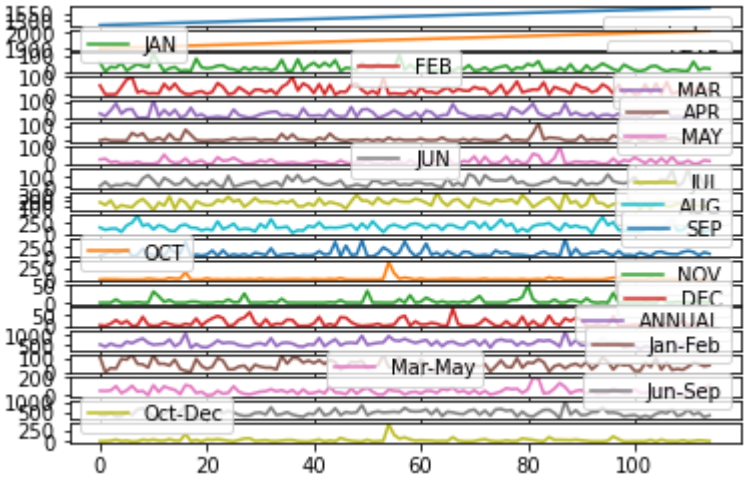
In [5]:
```
df.info()
```

```
<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)
```

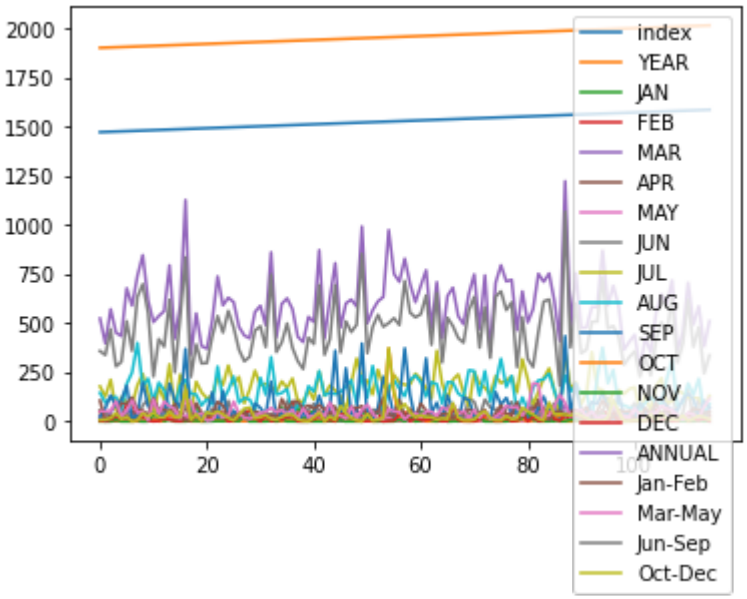
Out[6]:
```
array([<AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
       <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
       <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
       <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
    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



Line chart

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

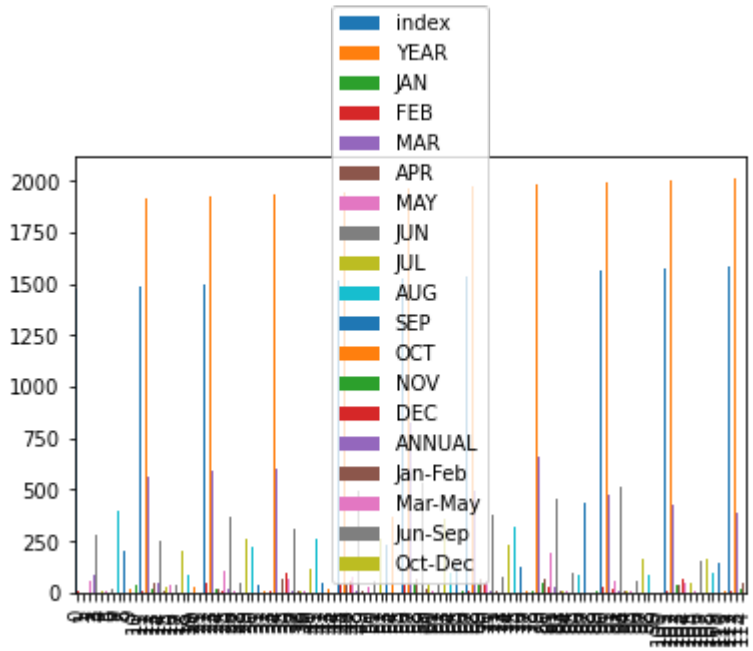
Out[7]:
```
<AxesSubplot:>
```



Bar chart

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

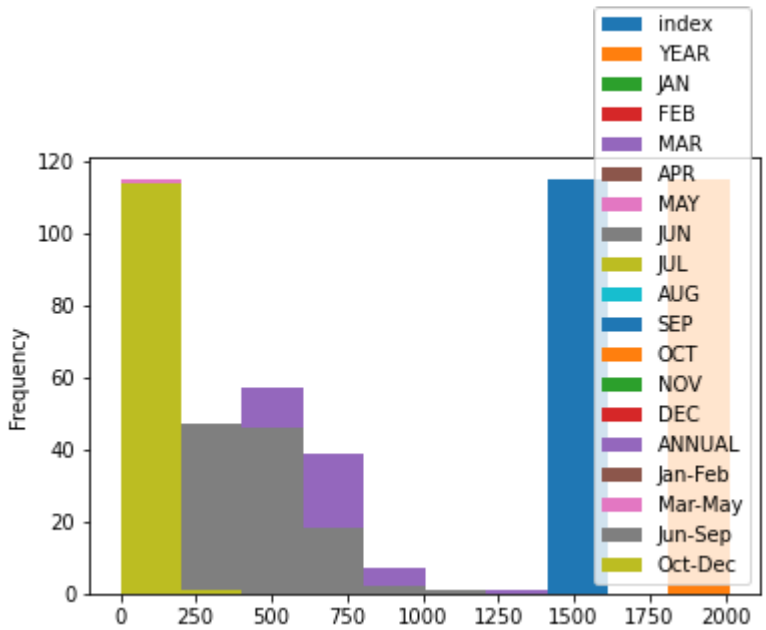
Out[8]: <AxesSubplot:>



Histogram

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

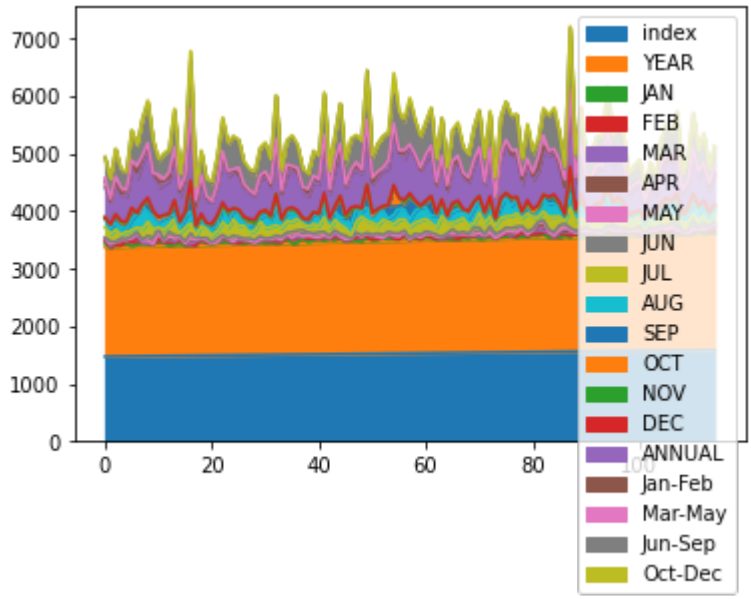
Out[9]: <AxesSubplot:ylabel='Frequency'>



Area chart

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

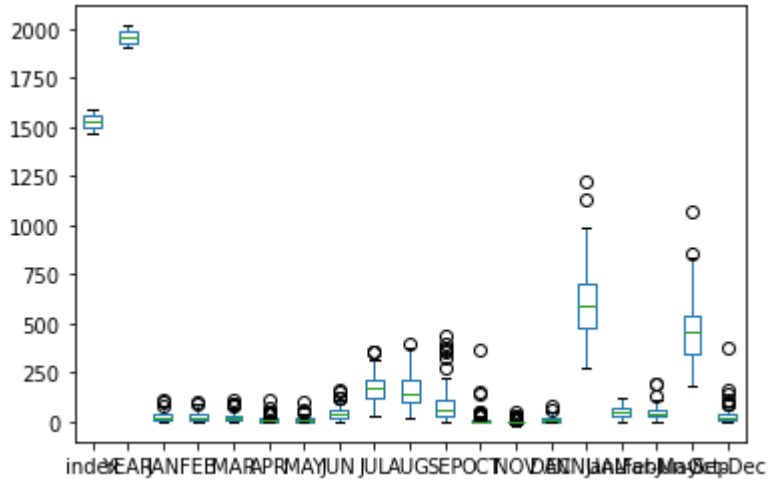
Out[10]: <AxesSubplot:>



Box chart

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

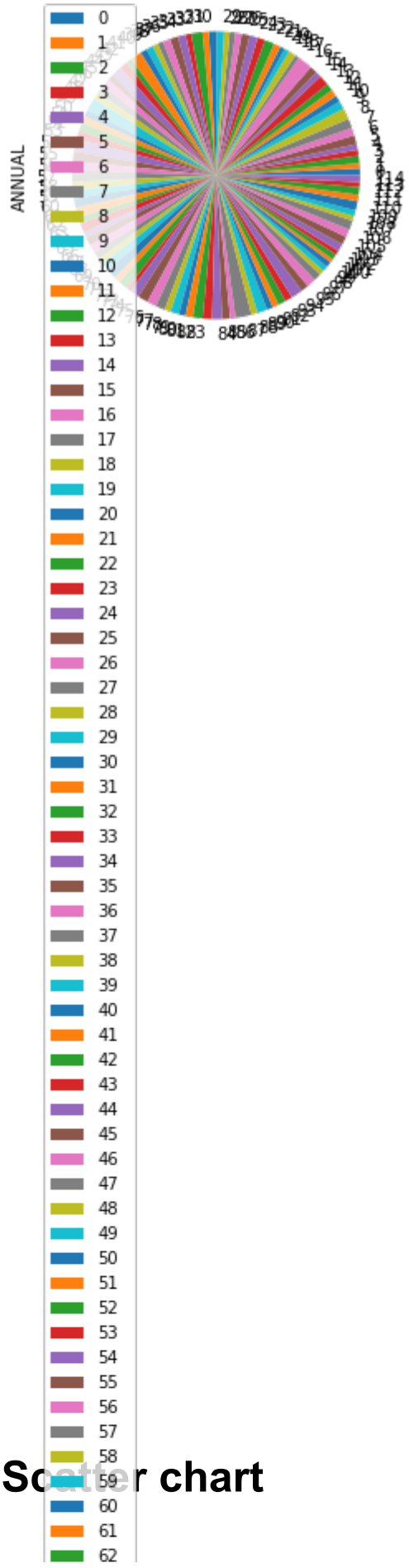
Out[11]: <AxesSubplot:>



Pie chart

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

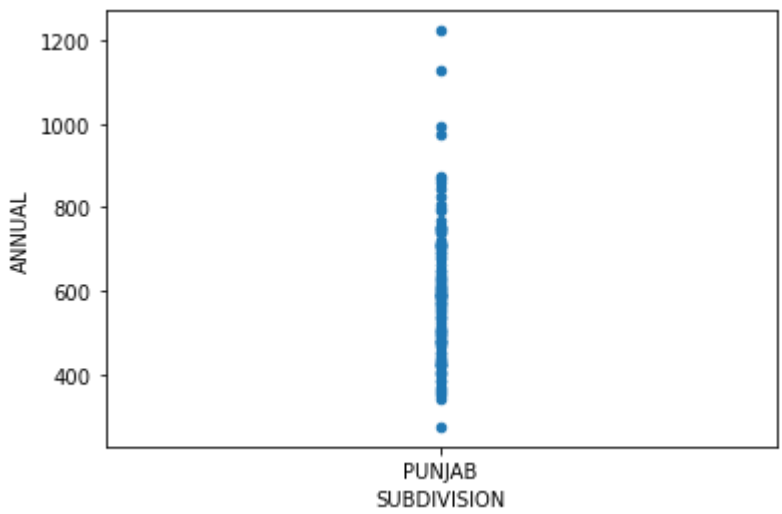
Out[12]:
```
<AxesSubplot:ylabel='ANNUAL'>
```



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
```

In [15]:
```
df.describe()
```

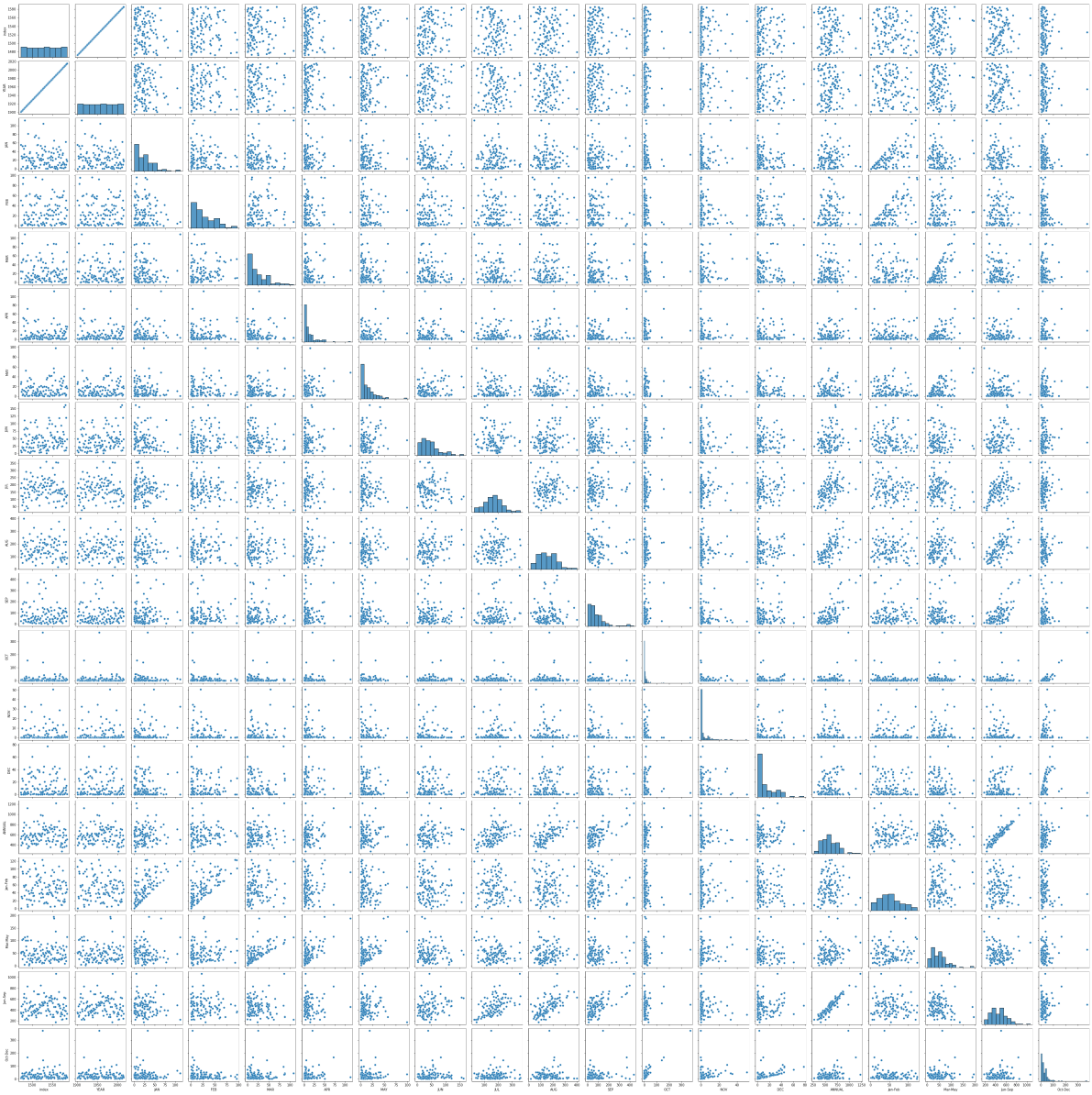
Out[15]:

	index	YEAR	JAN	FEB	MAR	APR	MAY	
count	115.000000	115.000000	115.000000	115.000000	115.000000	115.000000	115.000000	11
mean	1529.000000	1958.000000	25.246087	26.786957	23.651304	12.660000	14.136522	4
std	33.341666	33.341666	22.306656	23.473612	22.890109	16.751778	15.185232	3
min	1472.000000	1901.000000	0.000000	0.000000	0.000000	0.000000	0.100000	
25%	1500.500000	1929.500000	7.250000	5.650000	6.900000	2.550000	3.350000	2
50%	1529.000000	1958.000000	21.600000	21.300000	15.800000	6.700000	9.200000	4
75%	1557.500000	1986.500000	36.100000	40.600000	33.650000	15.700000	19.700000	6
max	1586.000000	2015.000000	112.100000	96.000000	108.500000	113.200000	98.300000	16

EDA AND VISUALIZATION

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

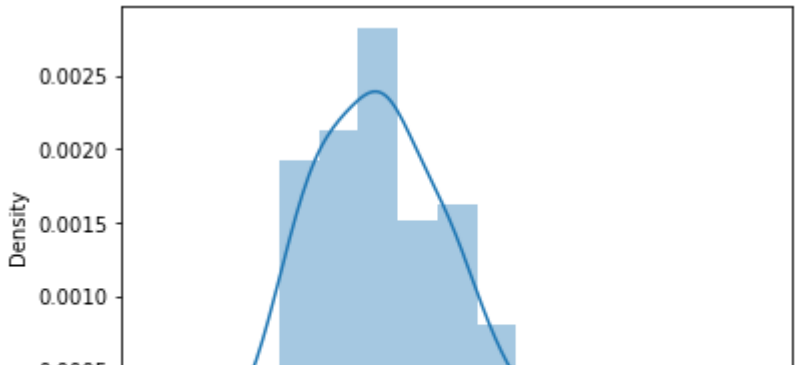
Out[16]: <seaborn.axisgrid.PairGrid at 0x2c94944ee50>



In [17]:
```
sns.distplot(df['ANNUAL'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2557: Fut
ureWarning: `distplot` is a deprecated function and will be removed in a futu
re version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for hi
stograms).
  warnings.warn(msg, FutureWarning)
```

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



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

Out[18]: <AxesSubplot:>