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("jammu _ kashmir.csv")
df
```

Out[2]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT
0	1702	JAMMU & KASHMIR	1901	66.4	69.3	69.6	132.2	105.8	53.4	171.7	181.3	101.8	24.1
1	1703	JAMMU & KASHMIR	1902	6.5	9.7	91.3	100.5	70.7	113.3	108.4	136.9	62.2	15.1
2	1704	JAMMU & KASHMIR	1903	96.2	21.5	238.6	58.7	57.3	18.9	332.5	218.6	176.9	10.7
3	1705	JAMMU & KASHMIR	1904	110.6	17.3	145.2	64.5	67.8	25.9	182.3	132.2	62.3	50.0
4	1706	JAMMU & KASHMIR	1905	146.7	76.3	161.4	71.7	65.2	43.3	145.2	111.5	239.7	5.8
...	...	...	...	...	...	...	...	...	...	...	...	...	...
110	1812	JAMMU & KASHMIR	2011	43.4	211.6	97.8	89.0	32.4	72.5	81.6	131.2	72.0	19.4
111	1813	JAMMU & KASHMIR	2012	150.9	95.8	45.2	86.6	48.9	32.6	118.8	264.9	106.7	15.7
112	1814	JAMMU & KASHMIR	2013	52.2	136.4	41.9	47.4	47.4	80.5	125.1	219.1	41.2	34.4
113	1815	JAMMU & KASHMIR	2014	75.8	64.0	153.1	76.1	52.7	25.3	100.5	134.6	362.8	32.2
114	1816	JAMMU & KASHMIR	2015	27.9	187.2	341.4	173.3	64.6	121.4	233.2	129.2	130.2	87.1

115 rows × 20 columns

Data Cleaning and Data Preprocessing

In [3]:
```
df=df.dropna()
df
```

Out[3]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT
0	1702	JAMMU & KASHMIR	1901	66.4	69.3	69.6	132.2	105.8	53.4	171.7	181.3	101.8	24.1
1	1703	JAMMU & KASHMIR	1902	6.5	9.7	91.3	100.5	70.7	113.3	108.4	136.9	62.2	15.1
2	1704	JAMMU & KASHMIR	1903	96.2	21.5	238.6	58.7	57.3	18.9	332.5	218.6	176.9	10.7
3	1705	JAMMU & KASHMIR	1904	110.6	17.3	145.2	64.5	67.8	25.9	182.3	132.2	62.3	50.0
4	1706	JAMMU & KASHMIR	1905	146.7	76.3	161.4	71.7	65.2	43.3	145.2	111.5	239.7	5.8
...	...	...	...	...	...	...	...	...	...	...	...	...	...
110	1812	JAMMU & KASHMIR	2011	43.4	211.6	97.8	89.0	32.4	72.5	81.6	131.2	72.0	19.4
111	1813	JAMMU & KASHMIR	2012	150.9	95.8	45.2	86.6	48.9	32.6	118.8	264.9	106.7	15.7
112	1814	JAMMU & KASHMIR	2013	52.2	136.4	41.9	47.4	47.4	80.5	125.1	219.1	41.2	34.4
113	1815	JAMMU & KASHMIR	2014	75.8	64.0	153.1	76.1	52.7	25.3	100.5	134.6	362.8	32.2
114	1816	JAMMU & KASHMIR	2015	27.9	187.2	341.4	173.3	64.6	121.4	233.2	129.2	130.2	87.1

114 rows × 20 columns

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()
```

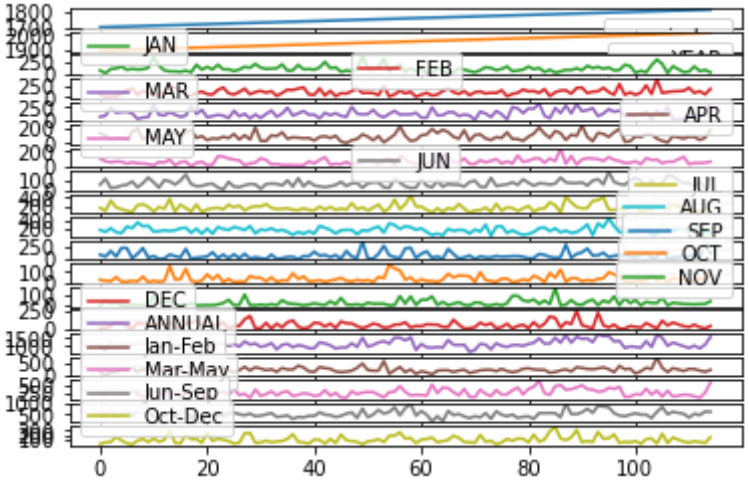
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 114 entries, 0 to 114
Data columns (total 20 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   index        114 non-null    int64  
 1   SUBDIVISION  114 non-null    object 
 2   YEAR         114 non-null    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: 18.7+ KB
```

Line chart

In [6]:
```
df.plot.line(subplots=True)
```

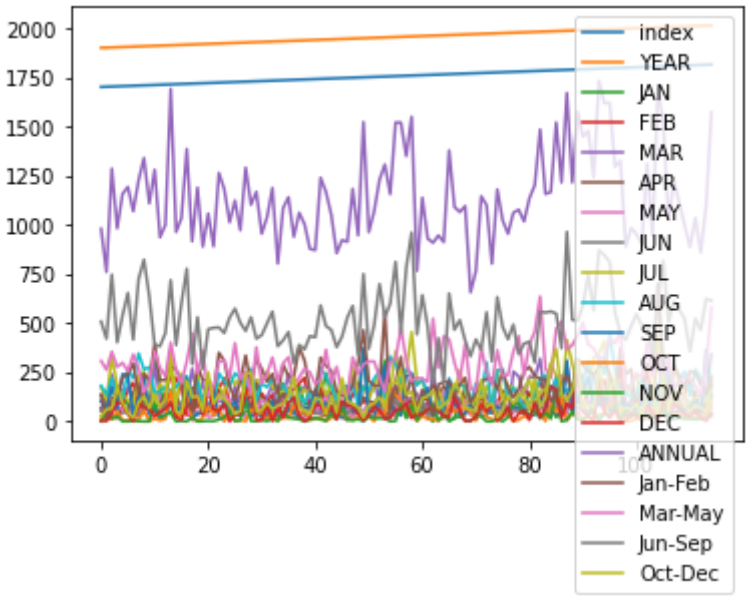
Out[6]:
```
array([<AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
       <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
       <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
       <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>, <AxesSubplot:>,
    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



Line chart

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

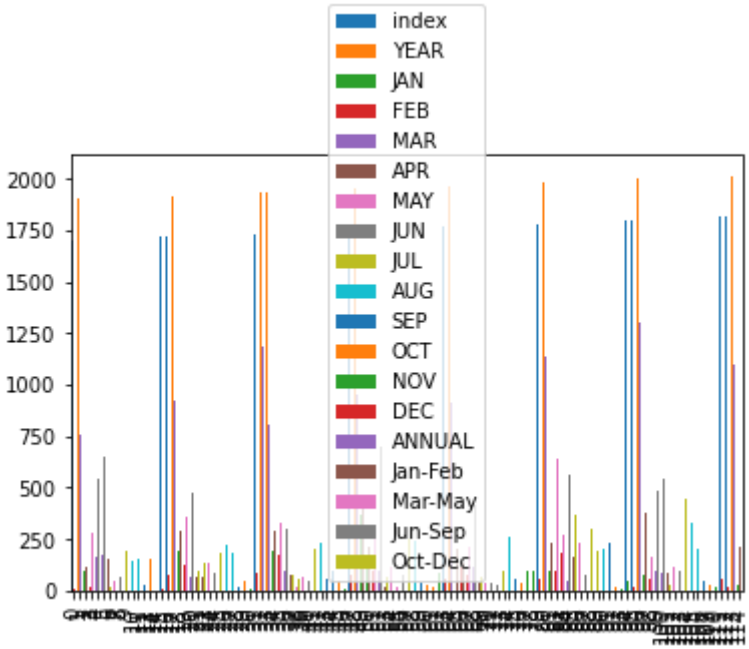
Out[7]: <AxesSubplot:>



Bar chart

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

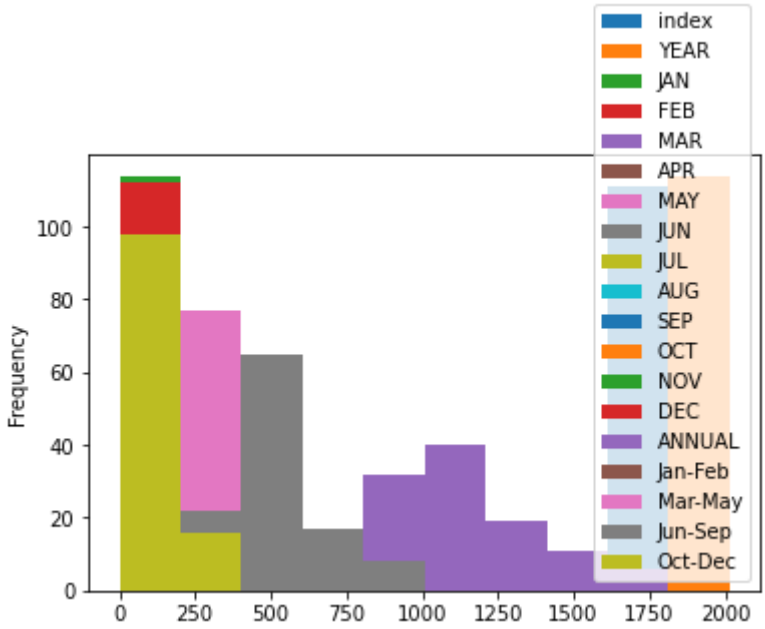
Out[8]: <AxesSubplot:>



Histogram

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

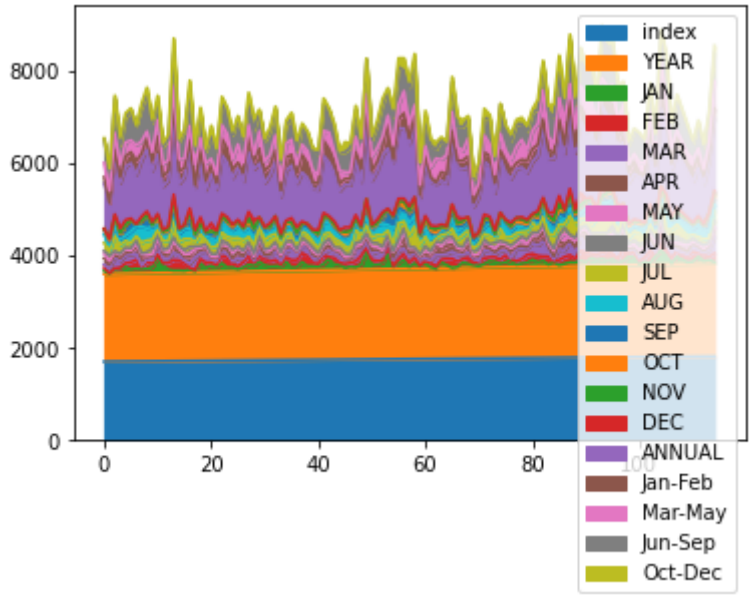
Out[9]: <AxesSubplot:ylabel='Frequency'>



Area chart

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

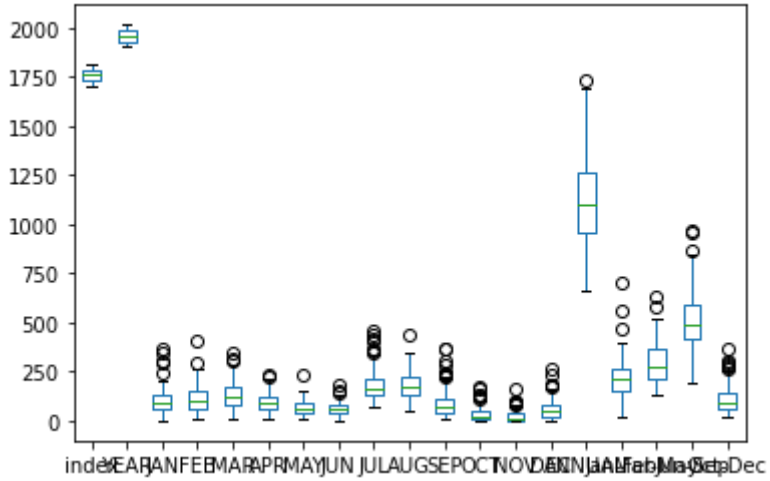
Out[10]: <AxesSubplot:>



Box chart

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

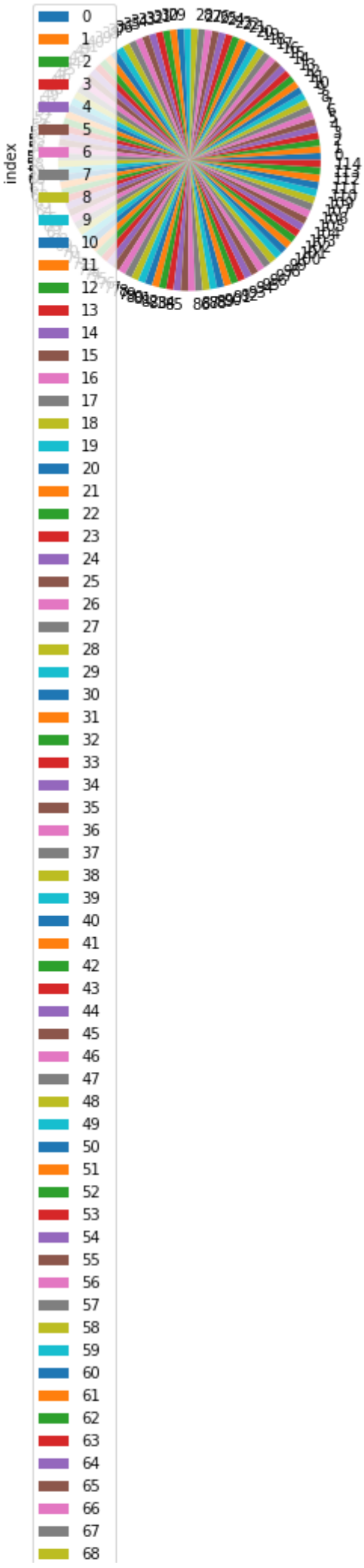
Out[11]: <AxesSubplot:>

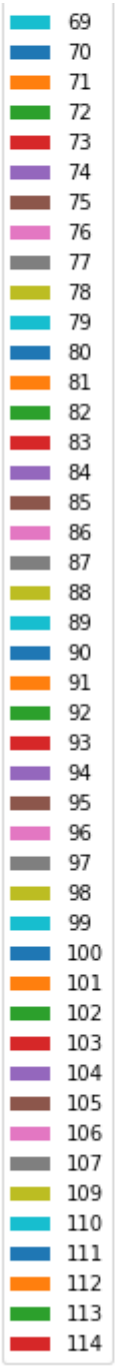


Pie chart

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

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

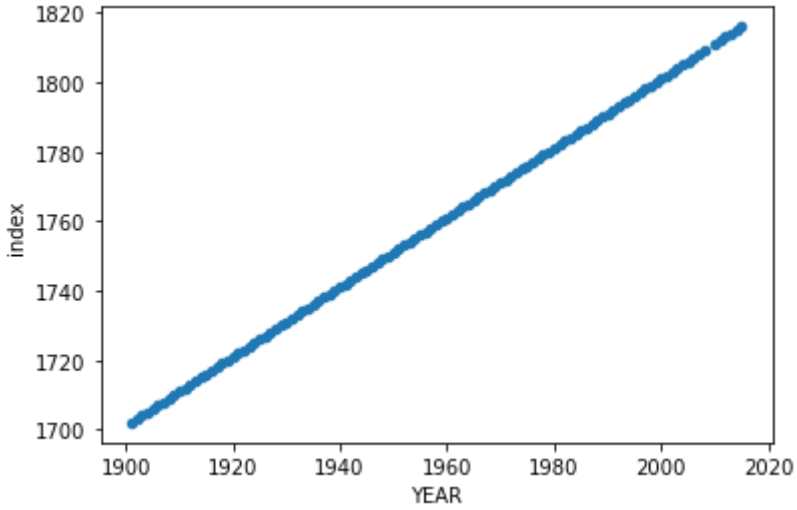

index
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68



Scatter chart

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

Out[13]:
```
<AxesSubplot:xlabel='YEAR', ylabel='index'>
```



In [14]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 114 entries, 0 to 114
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: 18.7+ KB
```

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

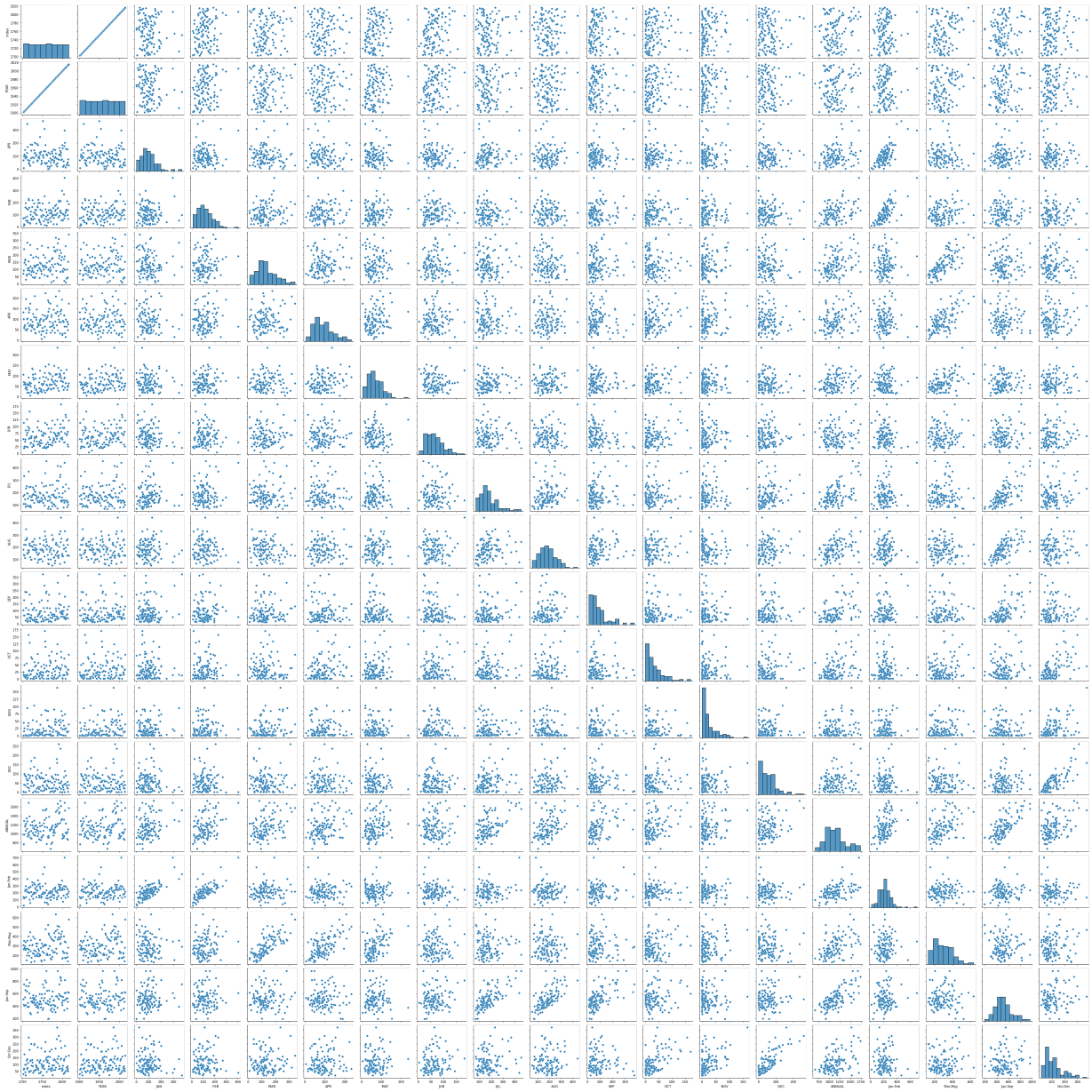
Out[15]:

	index	YEAR	JAN	FEB	MAR	APR	MAY	
count	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.00(
mean	1758.552632	1957.552632	101.870175	115.428947	131.971930	93.528947	67.504386	64.23
std	33.140380	33.140380	66.496789	67.229113	71.742153	48.559036	37.603315	33.29
min	1702.000000	1901.000000	0.000000	9.700000	9.900000	5.700000	4.700000	4.00(
25%	1730.250000	1929.250000	59.000000	64.600000	85.675000	59.475000	40.025000	39.30(
50%	1758.500000	1957.500000	89.800000	101.950000	116.400000	86.700000	61.200000	59.75(
75%	1786.750000	1985.750000	130.700000	149.900000	174.800000	119.450000	91.825000	81.07
max	1816.000000	2015.000000	367.800000	403.500000	341.400000	233.200000	234.400000	182.00(

EDA AND VISUALIZATION

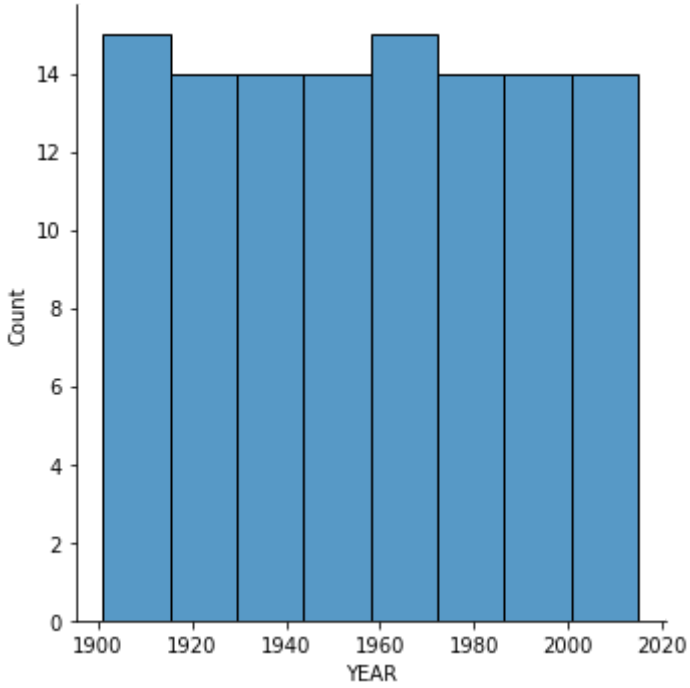
In [16]:
```
sns.pairplot(df)
```

Out[16]:
```
<seaborn.axisgrid.PairGrid at 0x1f98f3a9760>
```



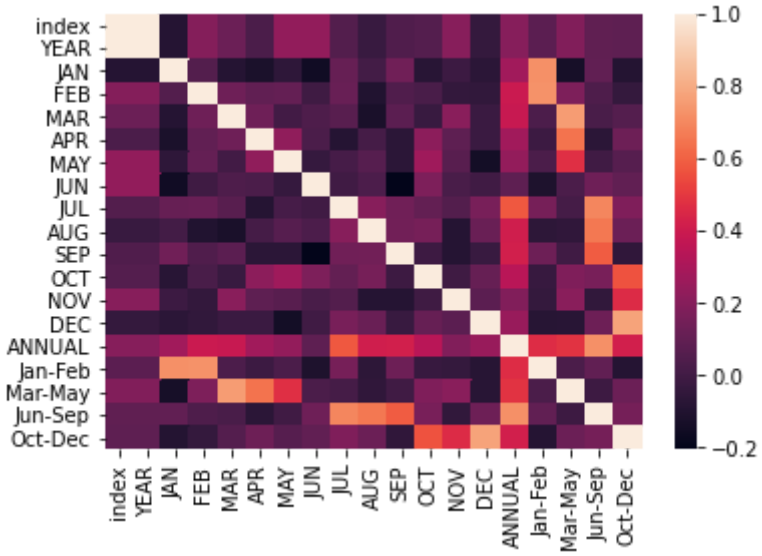
In [17]:
```
sns.displot(df['YEAR'])
```

Out[17]:
```
<seaborn.axisgrid.FacetGrid at 0x1f99a90e100>
```



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

Out[18]: <AxesSubplot:>



In [ ]: