Import Libraies

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

In [2]:
```
df=pd.read_csv(r"C:\Users\user\Downloads\FP2_RainFall\rainfall in india 1901-2015.csv")[437:551
df
```

Out[2]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNU
437	437	SUB HIMALAYAN WEST BENGAL & SIKKIM	1901	26.5	14.8	14.1	29.2	195.5	488.4	524.8	501.1	242.7	55.5	17.9	2.6	21
438	438	SUB HIMALAYAN WEST BENGAL & SIKKIM	1902	1.2	0.7	87.1	126.1	271.3	539.2	671.0	603.8	799.9	74.4	5.6	0.0	318
439	439	SUB HIMALAYAN WEST BENGAL & SIKKIM	1903	5.5	8.7	19.6	18.6	163.6	541.2	431.5	708.8	365.2	141.3	0.3	0.0	240
440	440	SUB HIMALAYAN WEST BENGAL & SIKKIM	1904	3.4	29.2	0.9	124.3	333.6	274.2	500.4	468.5	260.6	164.8	8.9	1.1	216
441	441	SUB HIMALAYAN WEST BENGAL & SIKKIM	1905	12.0	31.2	51.9	104.4	290.6	524.8	523.1	1036.6	321.1	87.9	2.7	18.7	300
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
546	546	SUB HIMALAYAN WEST BENGAL & SIKKIM	2010	5.6	19.6	77.6	176.6	335.9	558.1	593.4	461.3	308.1	66.2	7.9	2.2	26
547	547	SUB HIMALAYAN WEST BENGAL & SIKKIM	2011	8.5	19.9	71.2	135.0	247.8	419.8	612.3	470.3	356.3	46.7	26.7	4.3	24
548	548	SUB HIMALAYAN WEST BENGAL & SIKKIM	2012	15.3	13.9	45.5	159.8	202.4	604.2	684.5	332.7	434.7	119.4	12.5	7.4	263
549	549	SUB HIMALAYAN WEST BENGAL & SIKKIM	2013	3.0	23.6	32.1	114.7	296.5	404.9	588.4	416.3	308.0	199.8	16.1	2.7	240
550	550	SUB HIMALAYAN WEST BENGAL & SIKKIM	2014	0.2	26.6	37.7	47.9	308.6	543.2	384.6	563.3	371.5	31.2	5.3	2.4	232

114 rows × 20 columns

Data Cleaning and Preprocessing

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

Out[3]:

	index	SUBDIVISION	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNU
437	437	SUB HIMALAYAN WEST BENGAL & SIKKIM	1901	26.5	14.8	14.1	29.2	195.5	488.4	524.8	501.1	242.7	55.5	17.9	2.6	21
438	438	SUB HIMALAYAN WEST BENGAL & SIKKIM	1902	1.2	0.7	87.1	126.1	271.3	539.2	671.0	603.8	799.9	74.4	5.6	0.0	31
439	439	SUB HIMALAYAN WEST BENGAL & SIKKIM	1903	5.5	8.7	19.6	18.6	163.6	541.2	431.5	708.8	365.2	141.3	0.3	0.0	24
440	440	SUB HIMALAYAN WEST BENGAL & SIKKIM	1904	3.4	29.2	0.9	124.3	333.6	274.2	500.4	468.5	260.6	164.8	8.9	1.1	21
441	441	SUB HIMALAYAN WEST BENGAL & SIKKIM	1905	12.0	31.2	51.9	104.4	290.6	524.8	523.1	1036.6	321.1	87.9	2.7	18.7	30
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
546	546	SUB HIMALAYAN WEST BENGAL & SIKKIM	2010	5.6	19.6	77.6	176.6	335.9	558.1	593.4	461.3	308.1	66.2	7.9	2.2	26
547	547	SUB HIMALAYAN WEST BENGAL & SIKKIM	2011	8.5	19.9	71.2	135.0	247.8	419.8	612.3	470.3	356.3	46.7	26.7	4.3	24
548	548	SUB HIMALAYAN WEST BENGAL & SIKKIM	2012	15.3	13.9	45.5	159.8	202.4	604.2	684.5	332.7	434.7	119.4	12.5	7.4	26
549	549	SUB HIMALAYAN WEST BENGAL & SIKKIM	2013	3.0	23.6	32.1	114.7	296.5	404.9	588.4	416.3	308.0	199.8	16.1	2.7	24
550	550	SUB HIMALAYAN WEST BENGAL & SIKKIM	2014	0.2	26.6	37.7	47.9	308.6	543.2	384.6	563.3	371.5	31.2	5.3	2.4	23

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')
```

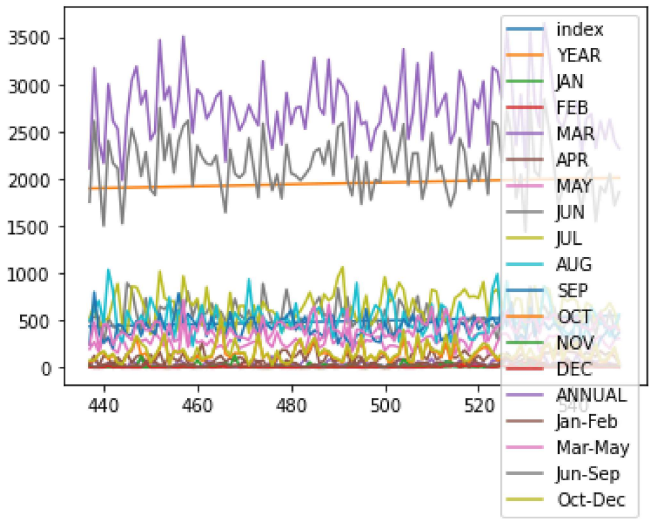
In [5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 114 entries, 437 to 550
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
```

Line chart

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

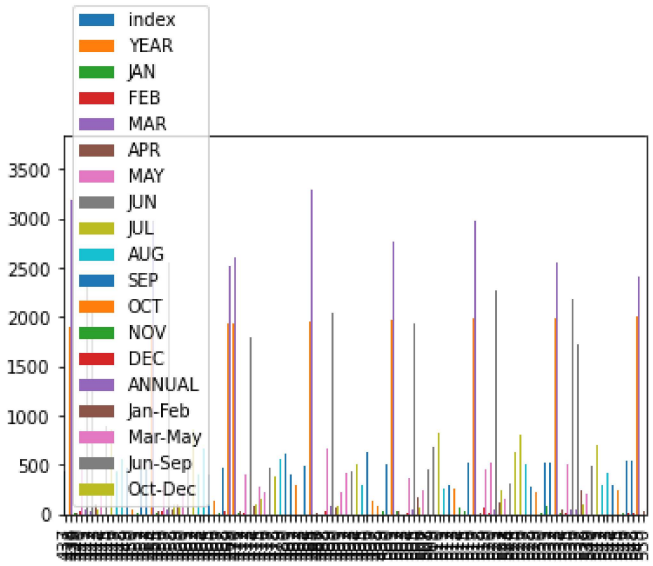
Out[6]: <AxesSubplot:>



Bar chart

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

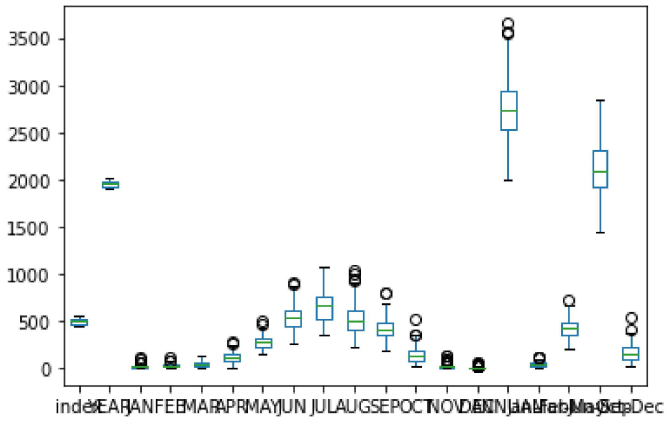
Out[7]: <AxesSubplot:>



Box chart

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

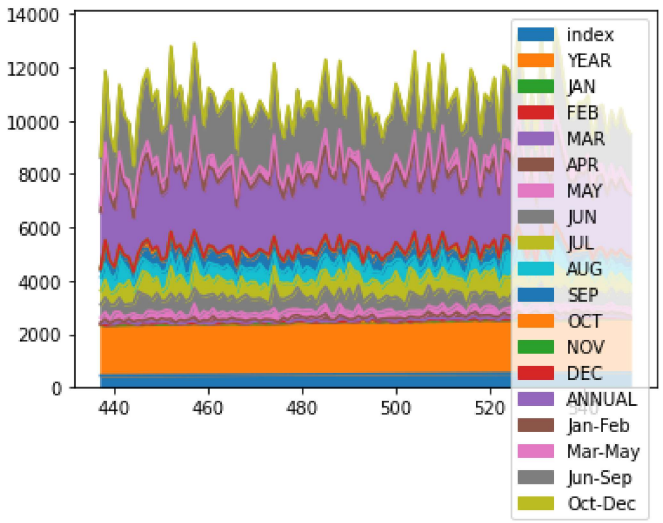
Out[8]: <AxesSubplot:>



Area chart

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

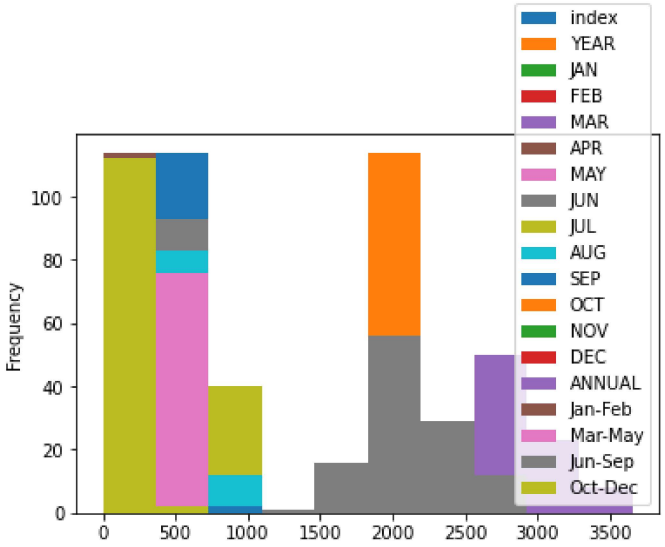
Out[9]: <AxesSubplot:>



Histogram

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

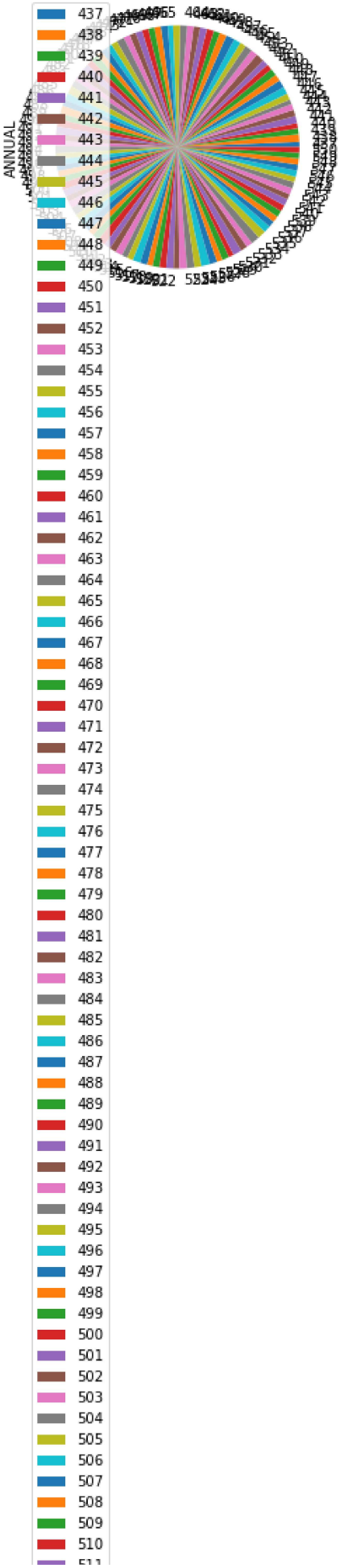
Out[10]: <AxesSubplot:ylabel='Frequency'>

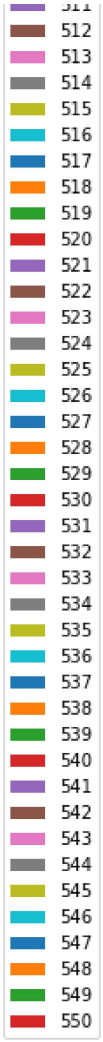


pie chart

In [11]:
```
df.plot.pie(y="ANNUAL")
```

Out[11]:
```
<AxesSubplot:ylabel='ANNUAL'>
```

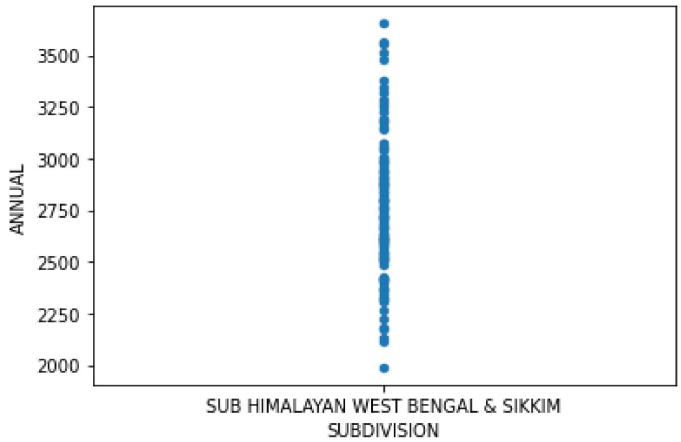

ANNUAL
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510



Scatter chart

In [12]:
```
df.plot.scatter(y='ANNUAL',x='SUBDIVISION')
```

Out[12]: <AxesSubplot:xlabel='SUBDIVISION', ylabel='ANNUAL'>



In [13]:
```
df.describe()
```

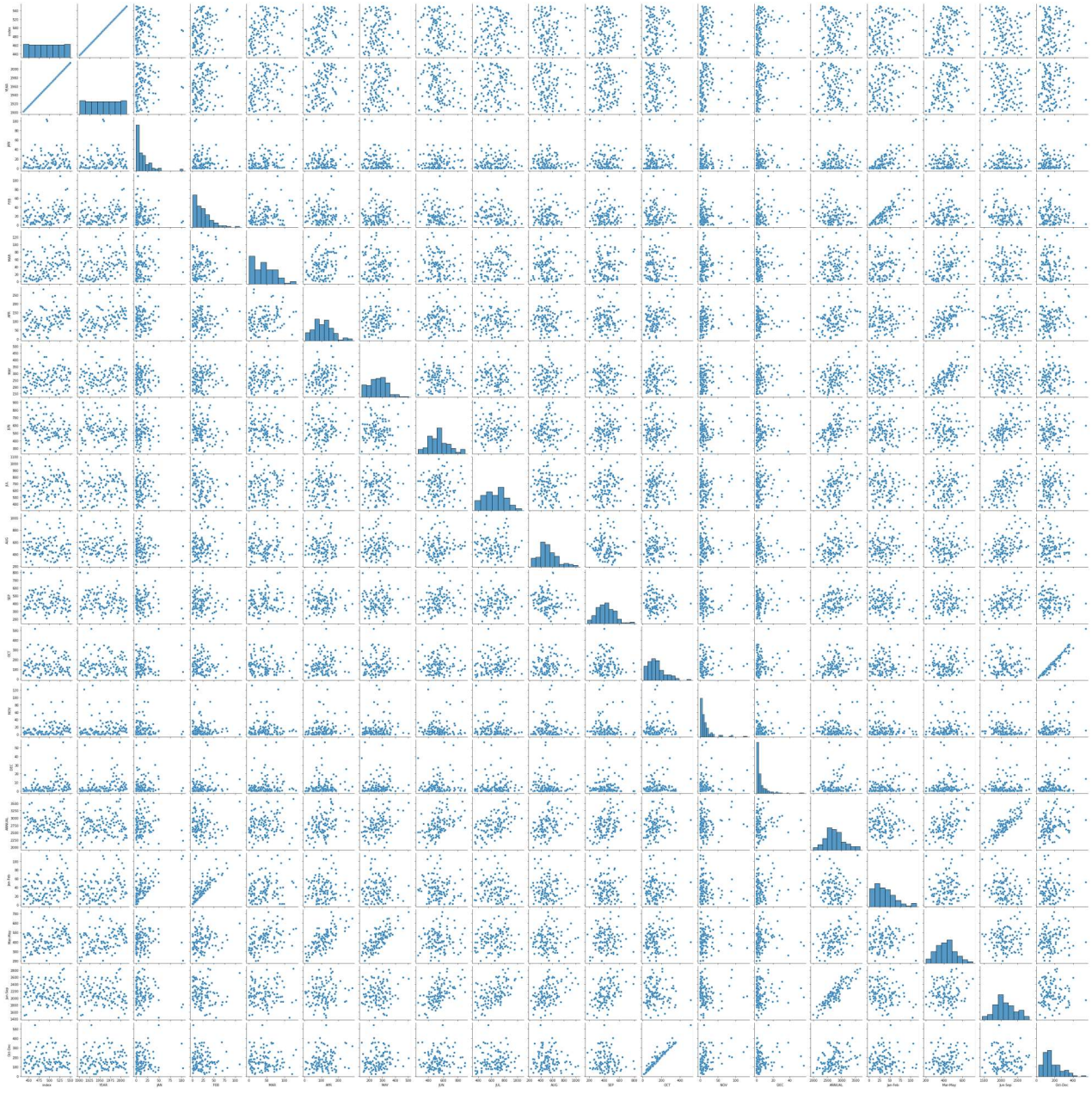
Out[13]:

	index	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL
count	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000	114.000000
mean	493.500000	1957.500000	14.069298	23.044737	42.945614	110.345614	268.832456	538.142105	648.622807
std	33.052988	33.052988	17.140756	19.655813	30.919848	55.817272	70.018954	134.683546	163.852083
min	437.000000	1901.000000	0.000000	0.100000	0.000000	4.800000	142.000000	261.700000	340.900000
25%	465.250000	1929.250000	2.225000	8.625000	15.100000	71.200000	217.100000	446.950000	523.525000
50%	493.500000	1957.500000	9.350000	19.700000	41.700000	109.850000	269.100000	530.250000	660.750000
75%	521.750000	1985.750000	19.575000	33.650000	62.125000	141.875000	312.200000	612.200000	753.775000
max	550.000000	2014.000000	103.000000	109.900000	132.100000	281.800000	503.100000	896.000000	1064.600000

EDA AND VISUALIZATION

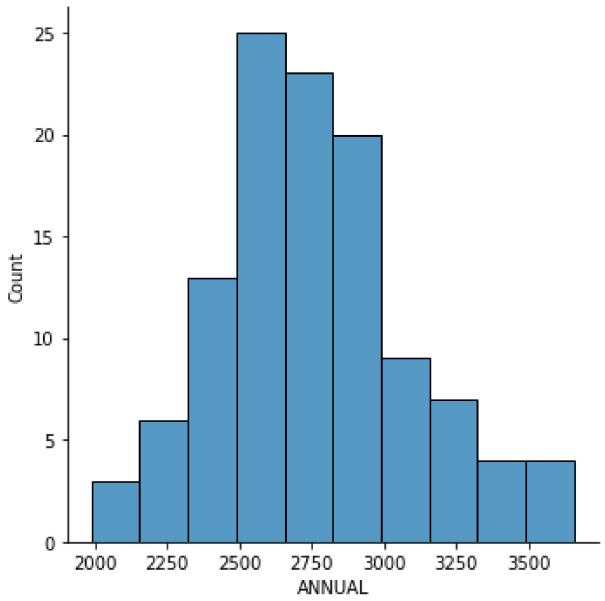
In [14]:
```
sns.pairplot(df)
```

Out[14]:
```
<seaborn.axisgrid.PairGrid at 0x2243d9909d0>
```



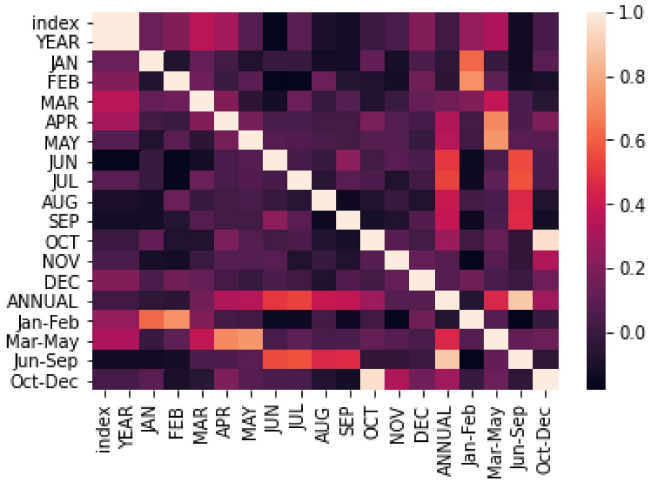
In [15]:
```
sns.displot(df['ANNUAL'])
```

Out[15]:
```
<seaborn.axisgrid.FacetGrid at 0x22449e67310>
```



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

Out[16]:
```
<AxesSubplot:>
```



In [ ]:
```

```