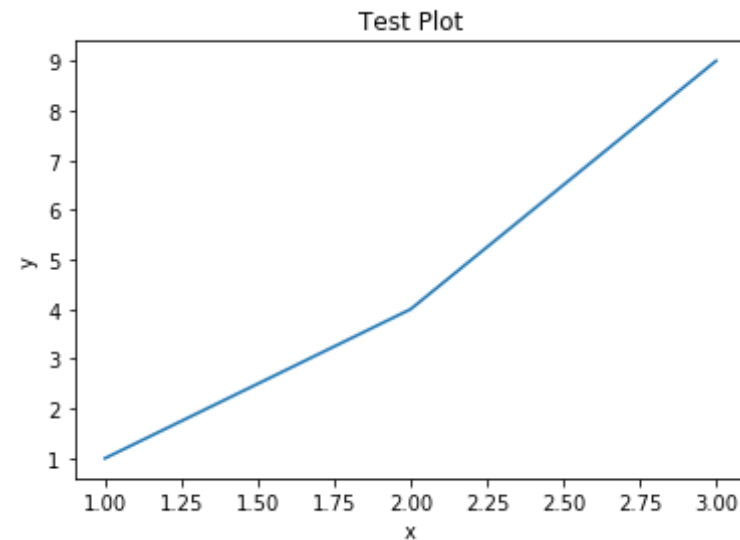


```
In [1]: import pandas as pd
```

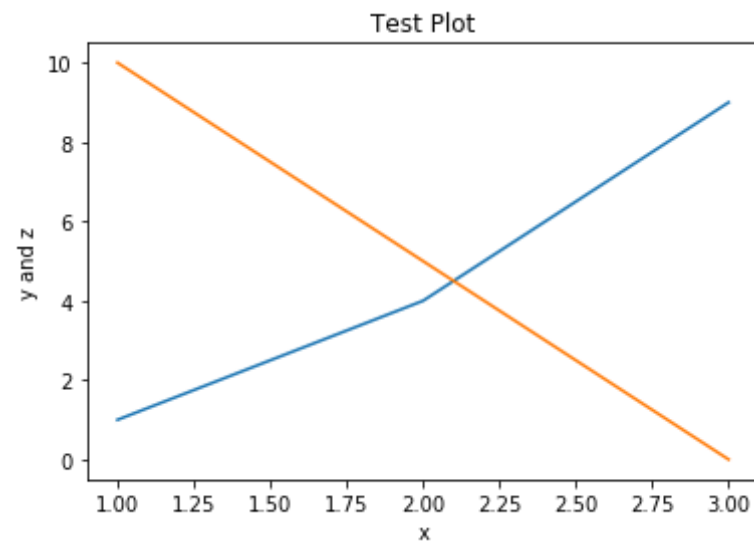
```
In [2]: from matplotlib import pyplot as plt
```

```
In [3]: x = [1, 2, 3]
y = [1, 4, 9]
plt.plot(x, y)
plt.title("Test Plot")
plt.xlabel("x")
plt.ylabel("y")
plt.show()
```

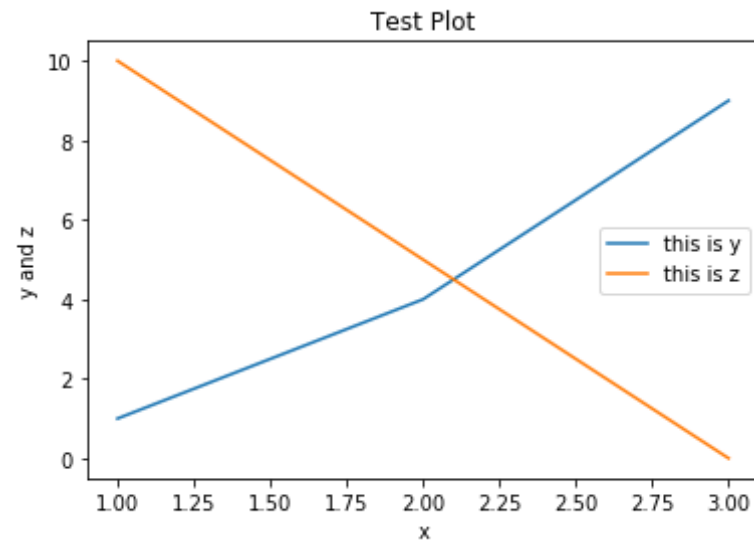


```
In [4]: x = [1, 2, 3]
y = [1, 4, 9]
z = [10, 5, 0]
plt.plot(x, y)
plt.plot(x, z)
plt.title("Test Plot")
```

```
plt.xlabel("x")  
plt.ylabel("y and z")  
plt.show()
```



```
In [5]: x = [1, 2, 3]  
y = [1, 4, 9]  
z = [10, 5, 0]  
plt.plot(x, y)  
plt.plot(x, z)  
plt.title("Test Plot")  
plt.xlabel("x")  
plt.ylabel("y and z")  
plt.legend(["this is y", "this is z"])  
plt.show()
```



```
In [6]: sample_data = pd.read_csv('sample_data.csv')
```

```
In [7]: sample_data
```

```
Out[7]:
```

	column_a	column_b	column_c
0	1	1	10
1	2	4	8
2	3	9	6
3	4	16	4
4	5	25	2

```
In [8]: type(sample_data)
```

```
Out[8]: pandas.core.frame.DataFrame
```

```
In [9]: sample_data.column_c
```

```
Out[9]: 0    10
        1     8
        2     6
        3     4
        4     2
        Name: column_c, dtype: int64
```

```
In [10]: type(sample_data.column_c)
```

```
Out[10]: pandas.core.series.Series
```

```
In [11]: #Print first value of column_c
```

```
In [12]: sample_data.column_c.iloc[0]
```

```
Out[12]: 10
```

```
In [13]: #Print second value of column_c
```

```
In [14]: sample_data.column_c.iloc[1]
```

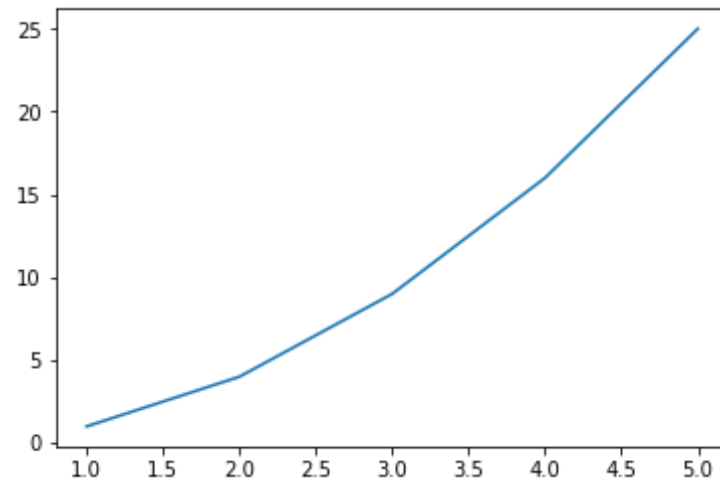
```
Out[14]: 8
```

```
In [15]: #Print third value of column_c
```

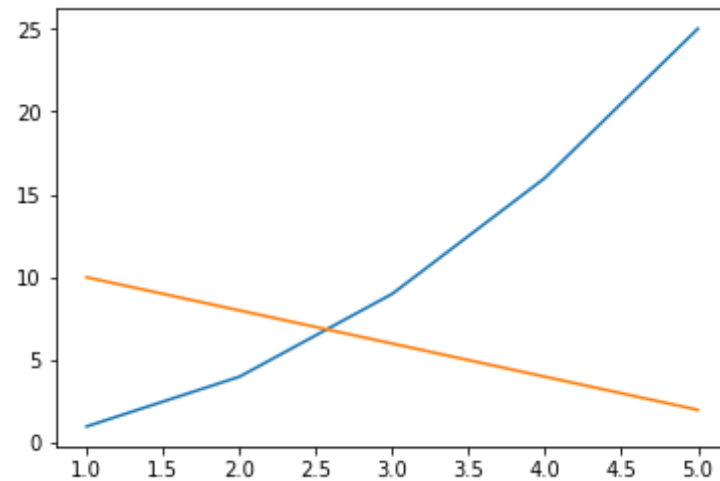
```
In [16]: sample_data.column_c.iloc[2]
```

```
Out[16]: 6
```

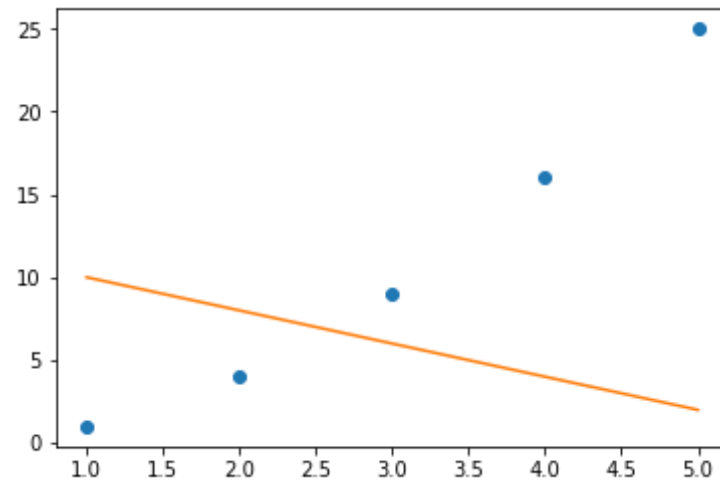
```
In [17]: plt.plot(sample_data.column_a, sample_data.column_b)
plt.show()
```



```
In [18]: plt.plot(sample_data.column_a, sample_data.column_b)
plt.plot(sample_data.column_a, sample_data.column_c)
plt.show()
```



```
In [19]: plt.plot(sample_data.column_a, sample_data.column_b, 'o')
plt.plot(sample_data.column_a, sample_data.column_c)
plt.show()
```



```
In [20]: data = pd.read_csv('countries.csv')
```

```
In [21]: data
```

```
Out[21]:
```

	country	year	population
0	Afghanistan	1952	8425333
1	Afghanistan	1957	9240934
2	Afghanistan	1962	10267083
3	Afghanistan	1967	11537966
4	Afghanistan	1972	13079460
5	Afghanistan	1977	14880372
6	Afghanistan	1982	12881816
7	Afghanistan	1987	13867957
8	Afghanistan	1992	16317921
9	Afghanistan	1997	22227415
10	Afghanistan	2002	25268405

	country	year	population
11	Afghanistan	2007	31889923
12	Albania	1952	1282697
13	Albania	1957	1476505
14	Albania	1962	1728137
15	Albania	1967	1984060
16	Albania	1972	2263554
17	Albania	1977	2509048
18	Albania	1982	2780097
19	Albania	1987	3075321
20	Albania	1992	3326498
21	Albania	1997	3428038
22	Albania	2002	3508512
23	Albania	2007	3600523
24	Algeria	1952	9279525
25	Algeria	1957	10270856
26	Algeria	1962	11000948
27	Algeria	1967	12760499
28	Algeria	1972	14760787
29	Algeria	1977	17152804
...
1674	Yemen, Rep.	1982	9657618
1675	Yemen, Rep.	1987	11219340
1676	Yemen, Rep.	1992	13367997
1677	Yemen, Rep.	1997	15826497
1678	Yemen, Rep.	2002	18701257

	country	year	population
1679	Yemen, Rep.	2007	22211743
1680	Zambia	1952	2672000
1681	Zambia	1957	3016000
1682	Zambia	1962	3421000
1683	Zambia	1967	3900000
1684	Zambia	1972	4506497
1685	Zambia	1977	5216550
1686	Zambia	1982	6100407
1687	Zambia	1987	7272406
1688	Zambia	1992	8381163
1689	Zambia	1997	9417789
1690	Zambia	2002	10595811
1691	Zambia	2007	11746035
1692	Zimbabwe	1952	3080907
1693	Zimbabwe	1957	3646340
1694	Zimbabwe	1962	4277736
1695	Zimbabwe	1967	4995432
1696	Zimbabwe	1972	5861135
1697	Zimbabwe	1977	6642107
1698	Zimbabwe	1982	7636524
1699	Zimbabwe	1987	9216418
1700	Zimbabwe	1992	10704340
1701	Zimbabwe	1997	11404948
1702	Zimbabwe	2002	11926563
1703	Zimbabwe	2007	12311143

1704 rows × 3 columns

```
In [22]: # Compare the population growth in the US and China
```

```
In [23]: data.country == 'United States'
```

```
Out[23]: 0      False
          1      False
          2      False
          3      False
          4      False
          5      False
          6      False
          7      False
          8      False
          9      False
         10      False
         11      False
         12      False
         13      False
         14      False
         15      False
         16      False
         17      False
         18      False
         19      False
         20      False
         21      False
         22      False
         23      False
         24      False
         25      False
         26      False
         27      False
         28      False
         29      False
          ...
        1674      False
        1675      False
```

```
1676 False
1677 False
1678 False
1679 False
1680 False
1681 False
1682 False
1683 False
1684 False
1685 False
1686 False
1687 False
1688 False
1689 False
1690 False
1691 False
1692 False
1693 False
1694 False
1695 False
1696 False
1697 False
1698 False
1699 False
1700 False
1701 False
1702 False
1703 False
Name: country, Length: 1704, dtype: bool
```

```
In [24]: data[data.country == 'United States']
```

```
Out[24]:
```

	country	year	population
1608	United States	1952	157553000
1609	United States	1957	171984000
1610	United States	1962	186538000

	country	year	population
1611	United States	1967	198712000
1612	United States	1972	209896000
1613	United States	1977	220239000
1614	United States	1982	232187835
1615	United States	1987	242803533
1616	United States	1992	256894189
1617	United States	1997	272911760
1618	United States	2002	287675526
1619	United States	2007	301139947

In [25]: `us = data[data.country == 'United States']`

In [26]: `us`

Out[26]:

	country	year	population
1608	United States	1952	157553000
1609	United States	1957	171984000
1610	United States	1962	186538000
1611	United States	1967	198712000
1612	United States	1972	209896000
1613	United States	1977	220239000
1614	United States	1982	232187835
1615	United States	1987	242803533
1616	United States	1992	256894189
1617	United States	1997	272911760

	country	year	population
1618	United States	2002	287675526
1619	United States	2007	301139947

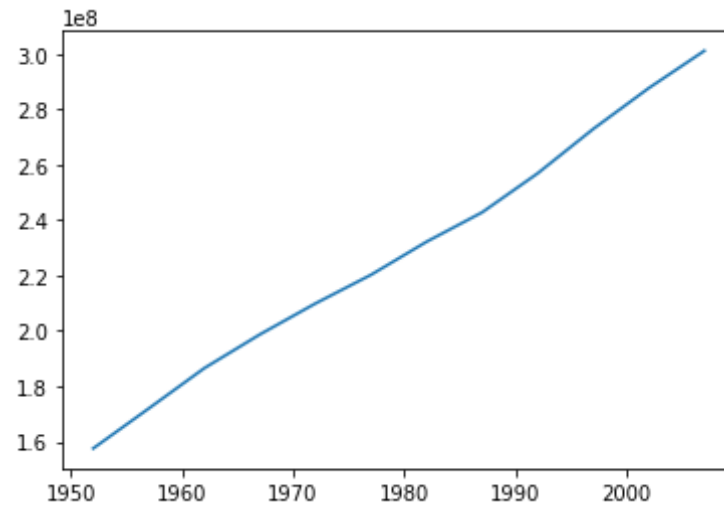
In [27]: `china = data[data.country == 'China']`

In [28]: `china`

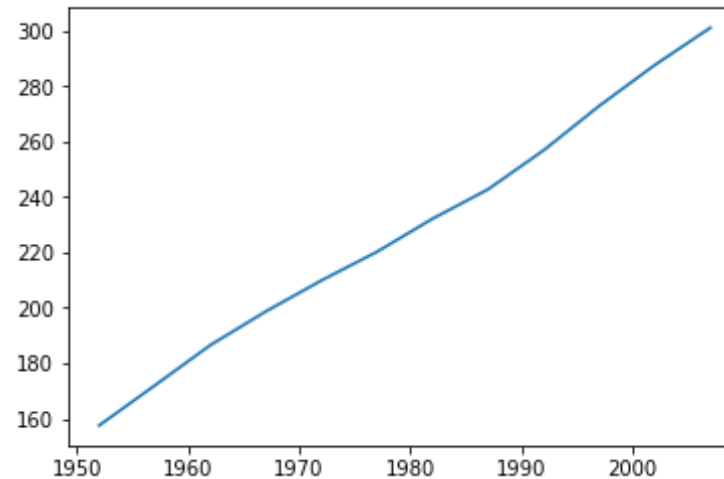
Out[28]:

	country	year	population
288	China	1952	556263527
289	China	1957	637408000
290	China	1962	665770000
291	China	1967	754550000
292	China	1972	862030000
293	China	1977	943455000
294	China	1982	1000281000
295	China	1987	1084035000
296	China	1992	1164970000
297	China	1997	1230075000
298	China	2002	1280400000
299	China	2007	1318683096

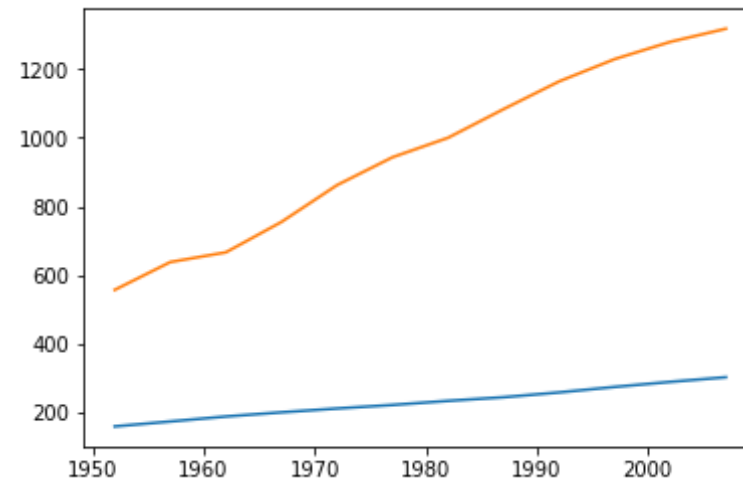
In [29]: `plt.plot(us.year, us.population)`
`plt.show()`



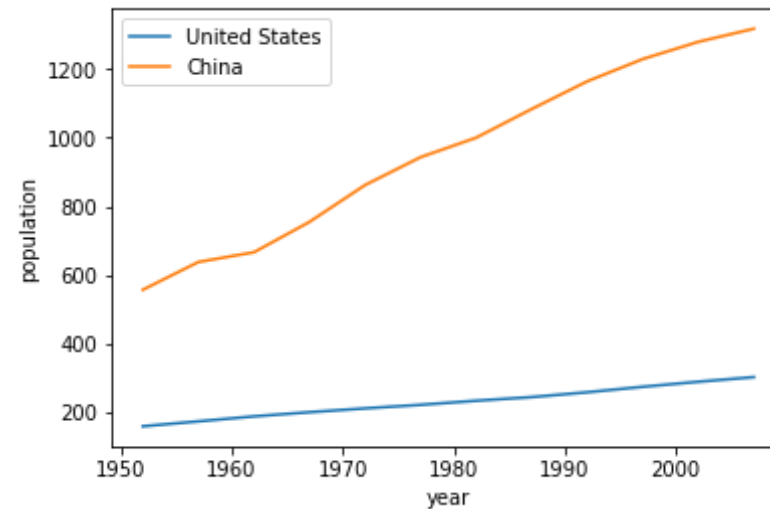
```
In [30]: plt.plot(us.year, us.population / 10**6)  
plt.show()
```



```
In [31]: plt.plot(us.year, us.population / 10**6)  
plt.plot(china.year, china.population / 10**6)  
plt.show()
```



```
In [32]: plt.plot(us.year, us.population / 10**6)
plt.plot(china.year, china.population / 10**6)
plt.legend(['United States', 'China'])
plt.xlabel('year')
plt.ylabel('population')
plt.show()
```



```
In [33]: us.population
```

```
Out[33]: 1608    157553000
          1609    171984000
          1610    186538000
          1611    198712000
          1612    209896000
          1613    220239000
          1614    232187835
          1615    242803533
          1616    256894189
          1617    272911760
          1618    287675526
          1619    301139947
          Name: population, dtype: int64
```

```
In [34]: us.population.iloc[0]
```

```
Out[34]: 157553000
```

```
In [35]: us.population / us.population.iloc[0]
```

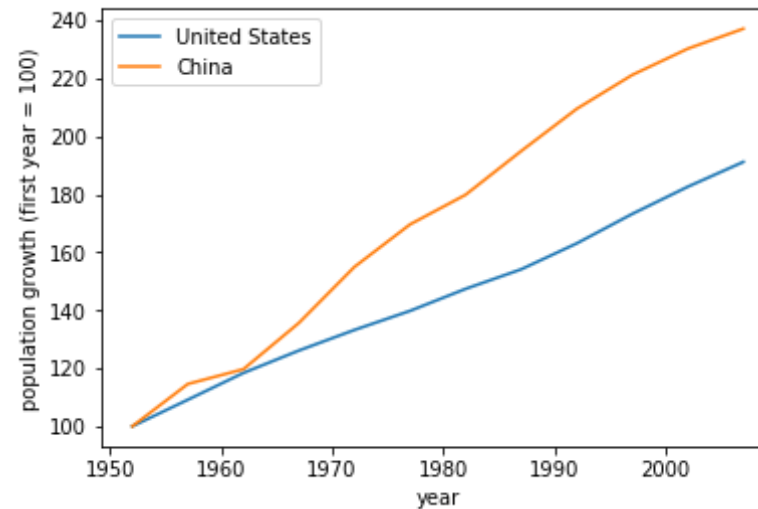
```
Out[35]: 1608    1.000000
          1609    1.091595
          1610    1.183970
          1611    1.261239
          1612    1.332225
          1613    1.397872
          1614    1.473713
          1615    1.541091
          1616    1.630526
          1617    1.732190
          1618    1.825897
          1619    1.911356
          Name: population, dtype: float64
```

```
In [36]: us.population / us.population.iloc[0] * 100
```

```
Out[36]: 1608    100.000000
```

```
1609    109.159457
1610    118.396984
1611    126.123908
1612    133.222471
1613    139.787246
1614    147.371256
1615    154.109114
1616    163.052553
1617    173.219018
1618    182.589685
1619    191.135648
Name: population, dtype: float64
```

```
In [37]: plt.plot(us.year, us.population / us.population.iloc[0] * 100)
plt.plot(china.year, china.population / china.population.iloc[0] * 100)
plt.legend(['United States', 'China'])
plt.xlabel('year')
plt.ylabel('population growth (first year = 100)')
plt.show()
```



```
In [ ]:
```