

L4

```
In [1]: 1 %matplotlib inline
        2 import pandas as pd
        3 import matplotlib.pyplot as plt
```

Load "SP500.csv" file using 'pd.read_csv'. Make "year" column as the index by specifying 'index_col'.

```
In [2]: 1 df = pd.read_csv("../Data/SP500.csv", index_col='year')
        2 df
```

Out[2]:

	dividendYield	Peratio	ShillerPERatio	10yearTyield	SP500level
year					
1950	0.0744	7.47	11.90	0.0257	21.21
1951	0.0602	9.95	12.53	0.0268	24.19
1952	0.0541	10.86	13.01	0.0283	26.18
1953	0.0584	10.10	12.00	0.0248	25.46
1954	0.0440	12.58	15.99	0.0261	35.60
...
2015	0.0211	22.18	24.21	0.0209	1918.60
2016	0.0203	23.59	28.06	0.0243	2275.12
2017	0.0185	24.97	33.31	0.0258	2789.80
2018	0.0209	19.60	28.38	0.0271	2607.39
2019	0.0181	24.47	31.31	0.0183	3265.38

70 rows × 5 columns

Add 'annual_return' column. You can use the function `pct_change()` to calculate annual returns from the 'SP500level'.

```
In [3]: 1 df['annual_return'] = df['SP500level'].pct_change()
        2 df
```

Out[3]:

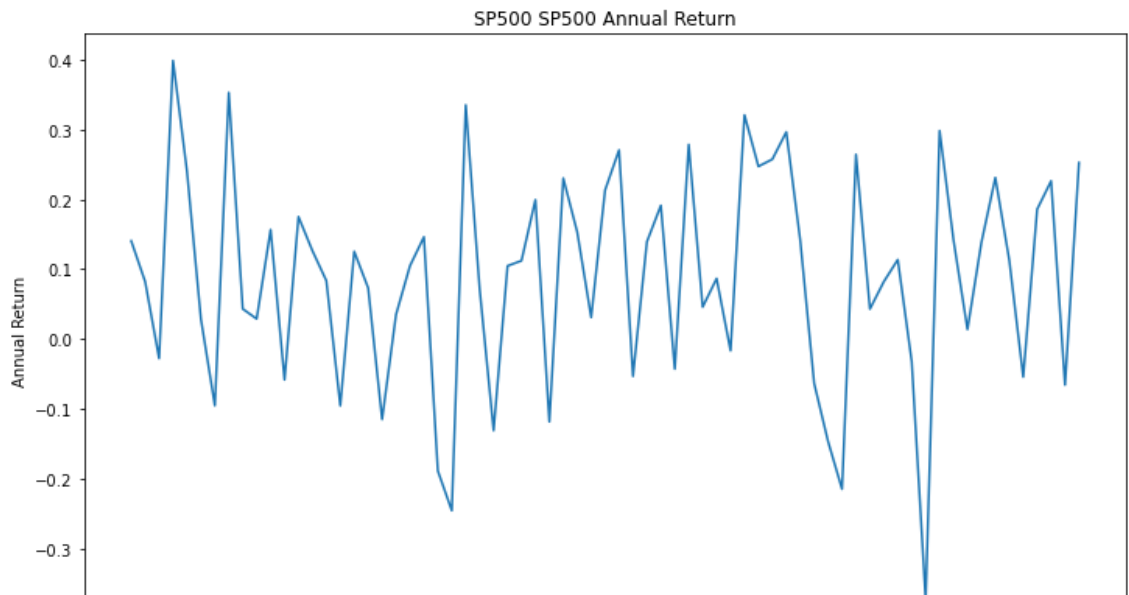
	dividendYield	Peratio	ShillerPERatio	10yearTyield	SP500level	annual_return
year						
1950	0.0744	7.47	11.90	0.0257	21.21	NaN
1951	0.0602	9.95	12.53	0.0268	24.19	0.140500
1952	0.0541	10.86	13.01	0.0283	26.18	0.082265
1953	0.0584	10.10	12.00	0.0248	25.46	-0.027502
1954	0.0440	12.58	15.99	0.0261	35.60	0.398272
...
2015	0.0211	22.18	24.21	0.0209	1918.60	-0.054029
2016	0.0203	23.59	28.06	0.0243	2275.12	0.185823
2017	0.0185	24.97	33.31	0.0258	2789.80	0.226221
2018	0.0209	19.60	28.38	0.0271	2607.39	-0.065385
2019	0.0181	24.47	31.31	0.0183	3265.38	0.252356

70 rows × 6 columns

Draw annual return graph using `plot()` method from pandas. 'set_title' function can specify the title of the graph.

```
In [4]: 1 ax = df['annual_return'].plot(figsize = (12, 7))
        2 ax.set_title("SP500 SP500 Annual Return")
        3 ax.set_xlabel="Year", ylabel="Annual Return")
```

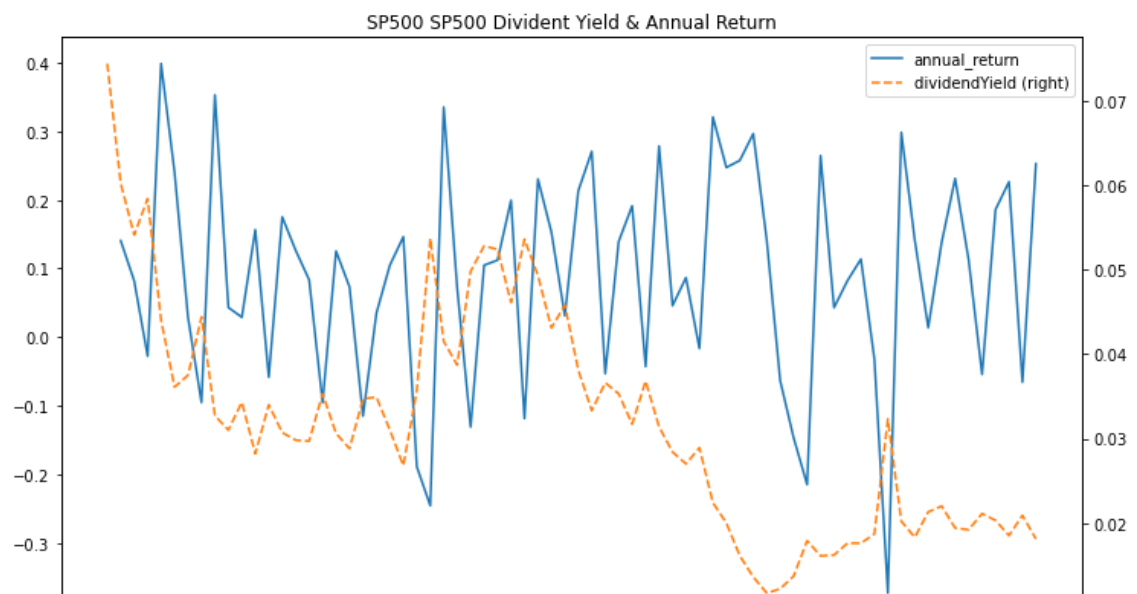
Out[4]: [Text(0, 0.5, 'Annual Return'), Text(0.5, 0, 'Year')]



Plot the dividend yield as well. To plot another column with different y-axis scale, we can use `secondary_y` parameter and then specify the column name you like to draw together. Change the style of the line with `style` parameter. `-` means solid line and `--` means dashed line.

```
In [5]: 1 columns_to_plot = ['annual_return', 'dividendYield']
        2 ax = df[columns_to_plot].plot(figsize = (12, 7), secondary_y = 'dividendYield')
        3 ax.set_title("SP500 SP500 Divident Yield & Annual Return")
```

Out[5]: Text(0.5, 1.0, 'SP500 SP500 Divident Yield & Annual Return')



We're going to shift the 'annual_return' column with `shift()` method from pandas. If the argument is a positive number, it will shift down. If the argument is negative number, it will shift one cell up. The default is to shift one down.

```
In [6]: 1 return_shifted = df.copy()
        2 return_shifted['annual_return'] = return_shifted['annual_return'].shift(
        3 return_shifted
```

Out[6]:

	dividendYield	Peratio	ShillerPEratio	10yearTyield	SP500level	annual_return
year						
1950	0.0744	7.47	11.90	0.0257	21.21	0.140500
1951	0.0602	9.95	12.53	0.0268	24.19	0.082265
1952	0.0541	10.86	13.01	0.0283	26.18	-0.027502
1953	0.0584	10.10	12.00	0.0248	25.46	0.398272
1954	0.0440	12.58	15.99	0.0261	35.60	0.240169
...
2015	0.0211	22.18	24.21	0.0209	1918.60	0.185823
2016	0.0203	23.59	28.06	0.0243	2275.12	0.226221
2017	0.0185	24.97	33.31	0.0258	2789.80	-0.065385
2018	0.0209	19.60	28.38	0.0271	2607.39	0.252356
2019	0.0181	24.47	31.31	0.0183	3265.38	NaN

70 rows × 6 columns

Use the `corr()` method from pandas.

```
In [7]: 1 return_shifted['annual_return'].corr(return_shifted['dividendYield'])
```

Out[7]: 0.23975697555928685

```
In [8]: 1 return_shifted['annual_return'].corr(return_shifted['Peratio'])
```

Out[8]: -0.05627754214493599

```
In [9]: 1 return_shifted['annual_return'].corr(return_shifted['ShillerPEratio'])
```

Out[9]: -0.19116434171142993

```
In [10]: 1 yield_shifted = return_shifted.copy()
        2 yield_shifted['10yearTyield'] = yield_shifted['10yearTyield'].shift(-1)
        3 yield_shifted['annual_return'].corr(yield_shifted['10yearTyield'])
```

Out[10]: -0.03346616937676412

Save the DataFrame to a csv file.

```
In [11]: 1 df.to_csv("../Data/SP500_data.csv")
```

```
In [ ]: 1
```