

# 3. CSVファイルのデータを 可視化してみよう

# CSVとは

- Comma-Separated Values (CSV) は、複数の項目をカンマ「,」で区切ったテキストデータです

- CSVファイルの例と、そのファイルで表現されている情報の対応

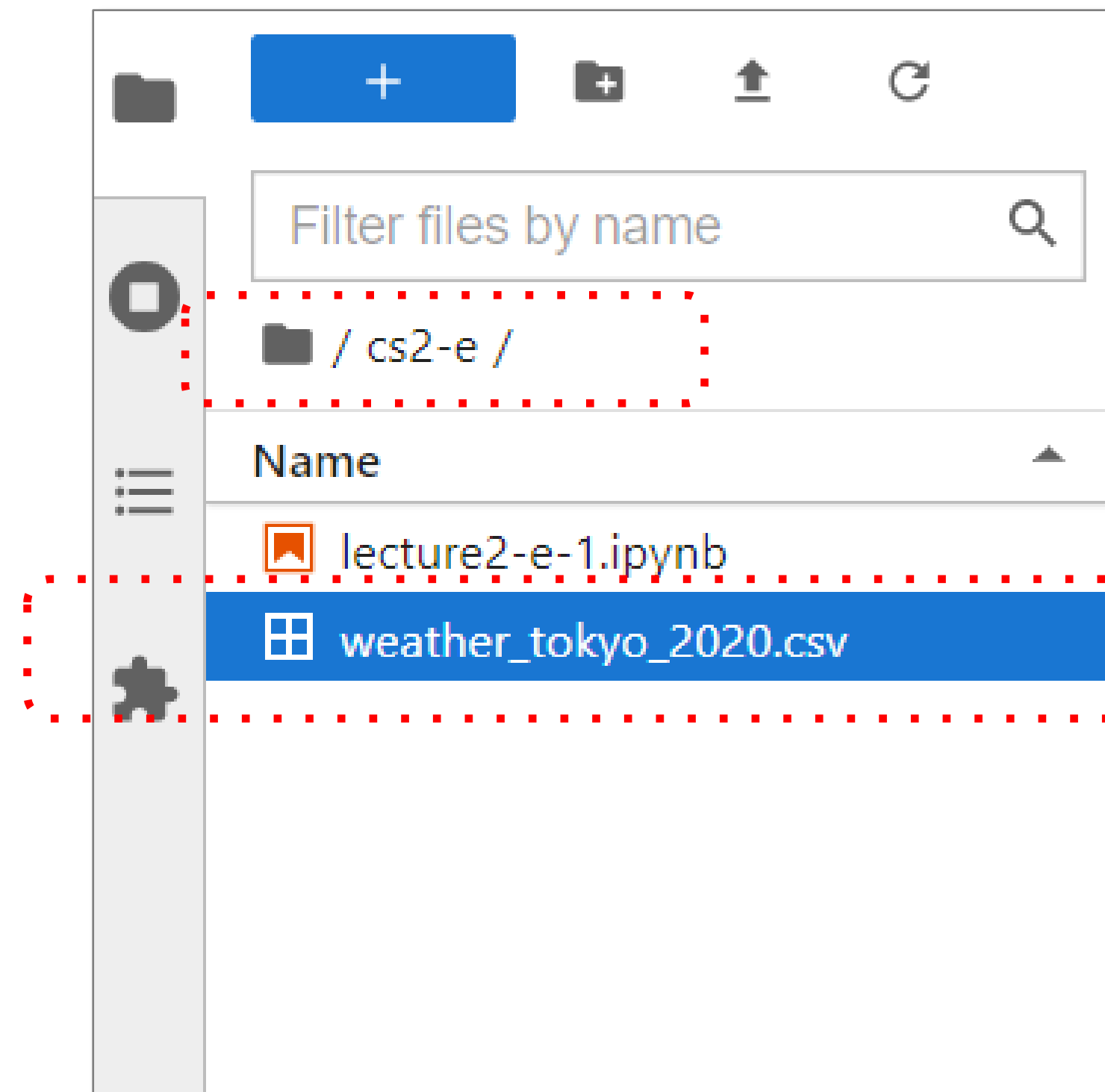
日付,平均気温,最低気温,最高気温
2021/9/1,23.5,18.3,20.7
2021/9/2,19.4,18.4,20.6
2021/9/3,19.7,18.7,21.4

日付	平均気温	最低気温	最高気温
2021/9/1	23.5	18.3	20.7
2021/9/2	19.4	18.4	20.6
2021/9/3	19.7	18.7	21.4

- 拡張子は「.csv」
- コンピュータで複数の情報を簡便に扱うために広く使われている

# データファイルの準備

- データファイル「weather\_tokyo\_2020.csv」をダウンロードし、「cs2-e」フォルダの中に移動してください
  - <https://drive.google.com/file/d/10pprKQqINvPqX2DRibnQO6gQ9ddj0uw->

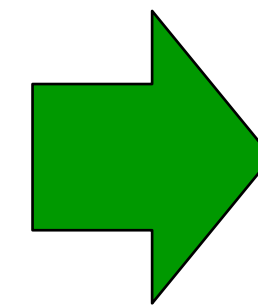
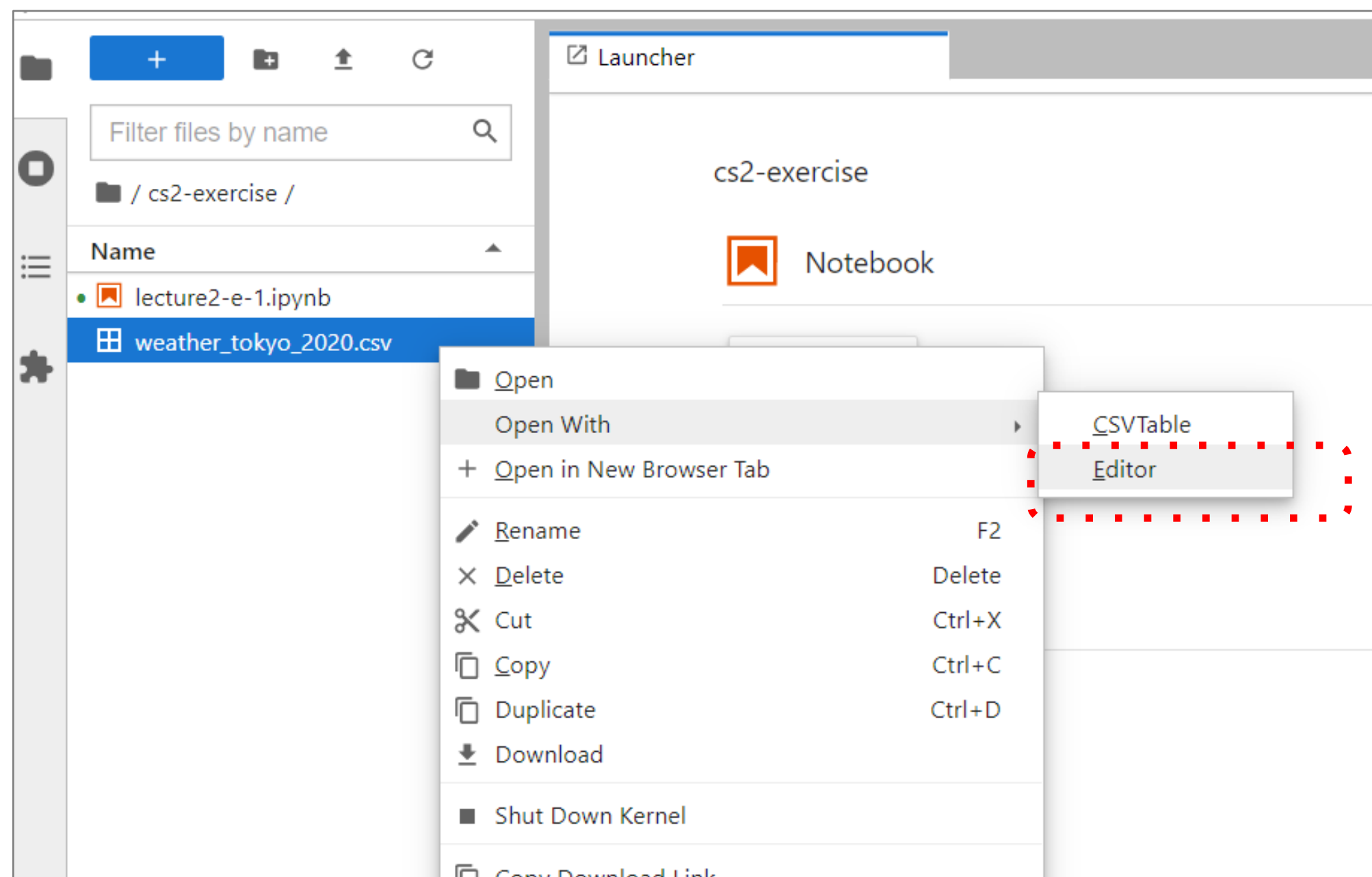


1. 現在ディレクトリを確認

2. ダウンロードしたファイルを  
ドラッグ & ドロップでここに移動

# CSVファイルの確認

- 「weather\_tokyo\_2020.csv」を右クリックし、Open With > Editor で開いてCSVファイルの内容を確認してみましょう
  - 各行には、2020年の東京の月ごとの最高気温、最低気温、平均気温、降水量合計が格納されています

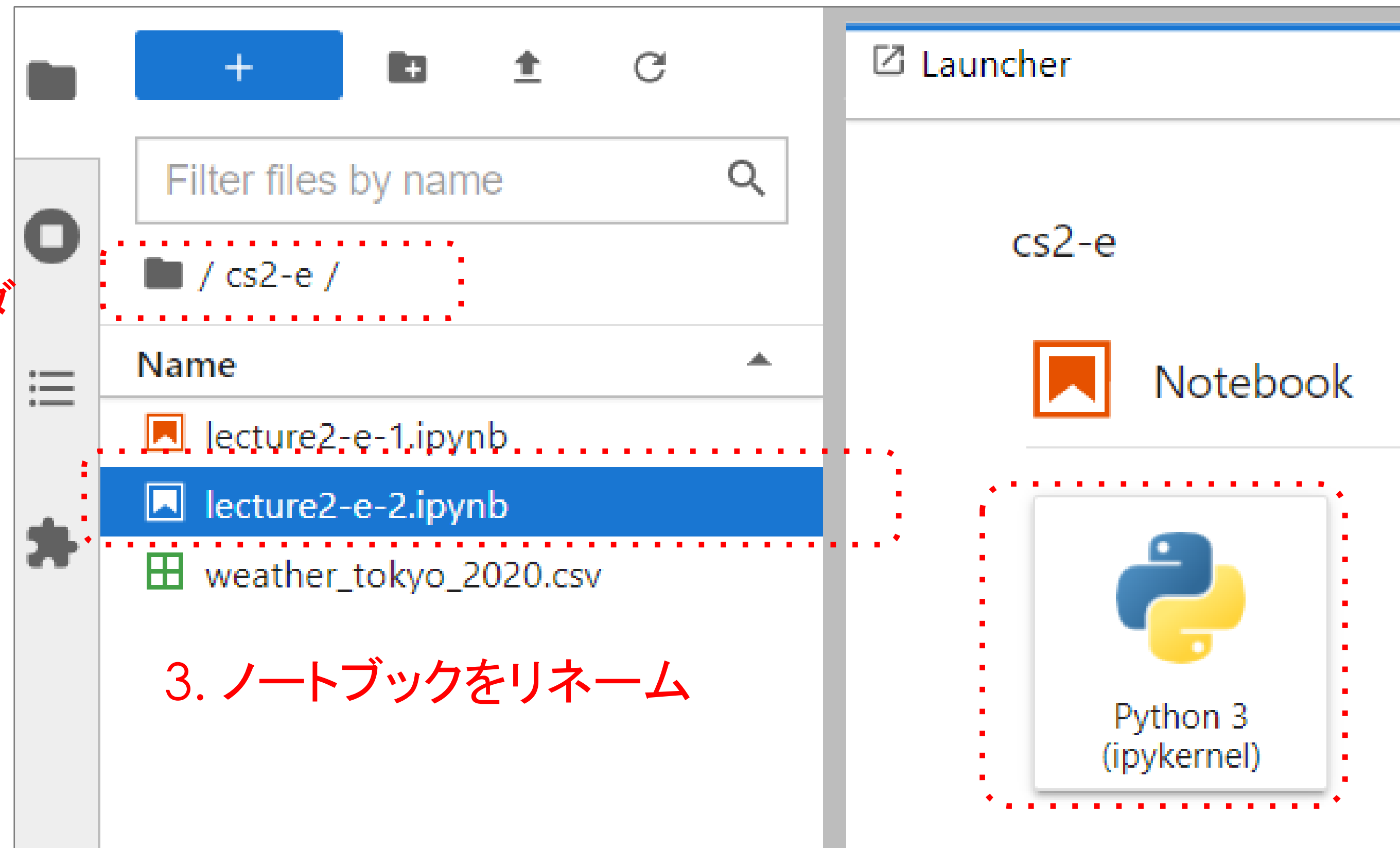


weather_tokyo_2020.csv					
1	1	11.1	3.7	7.1	135.0
2	2	13.3	4.0	8.3	15.0
3	3	16.0	6.2	10.7	131.0
4	4	18.2	7.9	12.8	296.5
5	5	24.0	15.6	19.5	118.0
6	6	27.5	19.8	23.2	212.5
7	7	27.7	21.8	24.3	270.5
8	8	34.1	25.3	29.1	61.5
9	9	28.1	21.5	24.2	117.5
10	10	21.4	14.4	17.5	205.0
11	11	18.6	10.1	14.0	14.5
12	12	12.3	3.7	7.7	13.0

# 演習用のノートブックの作成

- 「cs2-e」の下に、lecture2-e-2.ipynb を作成しましょう

1. cs2-eフォルダに移動



2. ノートブックを作成

# Matplotlib と csvモジュールをインポートする

- lecture2-e-2.ipynbに以下のセルを作成します
  - Markdownセル
  - Codeセルを追加し、以下を行います
    - matplotlib.pyplot を plt という別名でインポート
    - csvファイルを扱うための、csvモジュールをインポート

Markdownセル



```
### CSVファイルのデータを可視化してみよう
```

Codeセル



```
import matplotlib.pyplot as plt  
import csv
```

# データの読み込みと確認

- Codeセルを作成し、以下のプログラムを書いて実行してみましょう

- 「csv」モジュールを利用してCSVファイルのデータをリストとして取得しています

```
with open('weather_tokyo_2020.csv', 'r') as f:
    reader = csv.reader(f)
    for row in reader:
        print(row)
```

→ CSVファイルを読み込みモードで開く

→ 「csv.reader()」の第1引数に open() で開いたファイルオブジェクト「f」を指定し、for 文で反復処理させることによって1行ずつリストとして取得することができる

```
['1', '11.1', '3.7', '7.1', '135.0']
['2', '13.3', '4.0', '8.3', '15.0']
['3', '16.0', '6.2', '10.7', '131.0']
['4', '18.2', '7.9', '12.8', '296.5']
['5', '24.0', '15.6', '19.5', '118.0']
['6', '27.5', '19.8', '23.2', '212.5']
['7', '27.7', '21.8', '24.3', '270.5']
['8', '34.1', '25.3', '29.1', '61.5']
['9', '28.1', '21.5', '24.2', '117.5']
['10', '21.4', '14.4', '17.5', '205.0']
['11', '18.6', '10.1', '14.0', '14.5']
['12', '12.3', '3.7', '7.7', '13.0']
```

# 平均気温のデータを抽出してみよう

- 前のページで作成した Codeセルを以下のように変更し、実行してみましょう
  - 各行の0列目(月)と3列目(平均気温)のデータを取得し、それぞれ別のリストに格納する
  - データは文字列として格納されていることから、数値データは適切な型に変換する

```
month = []
temperature = []
with open('weather_tokyo_2020.csv', 'r') as f:
    reader = csv.reader(f)
    for row in reader:
        month.append(row[0])
        temperature.append(float(row[3]))

print(month)
print(temperature)
```

```
['1', '2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12']
[7.1, 8.3, 10.7, 12.8, 19.5, 23.2, 24.3, 29.1, 24.2, 17.5, 14.0, 7.7]
```

気温データはfloat  
型に変換

['1',	'11.1',	'3.7',	'7.1',	'135.0']
['2',	'13.3',	'4.0',	'8.3',	'15.0']
['3',	'16.0',	'6.2',	'10.7',	'131.0']
['4',	'18.2',	'7.9',	'12.8',	'296.5']
['5',	'24.0',	'15.6',	'19.5',	'118.0']
['6',	'27.5',	'19.8',	'23.2',	'212.5']
['7',	'27.7',	'21.8',	'24.3',	'270.5']
['8',	'34.1',	'25.3',	'29.1',	'61.5']
['9',	'28.1',	'21.5',	'24.2',	'117.5']
['10',	'21.4',	'14.4',	'17.5',	'205.0']
['11',	'18.6',	'10.1',	'14.0',	'14.5']
['12',	'12.3',	'3.7',	'7.7',	'13.0']

月

平均気温