

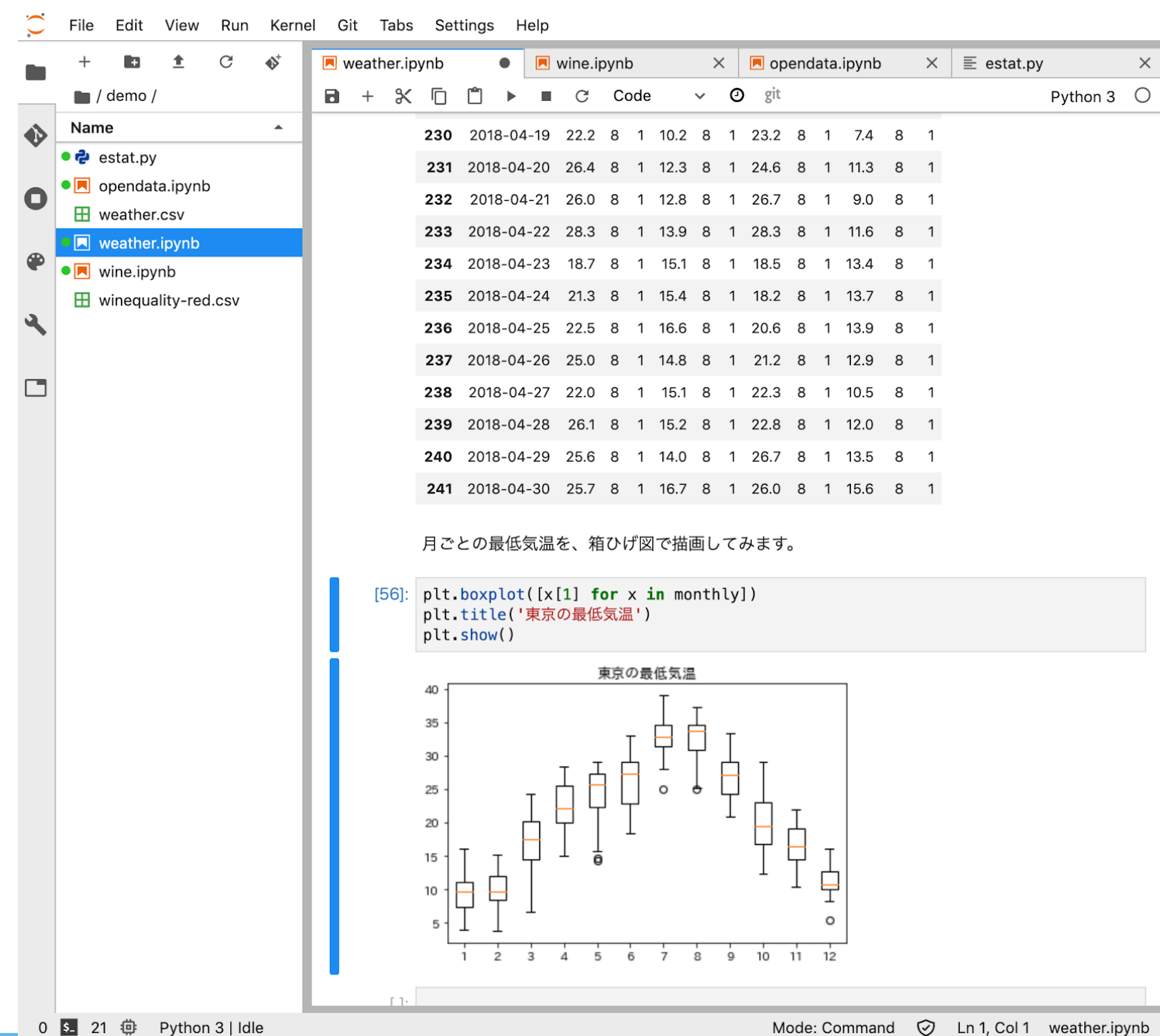
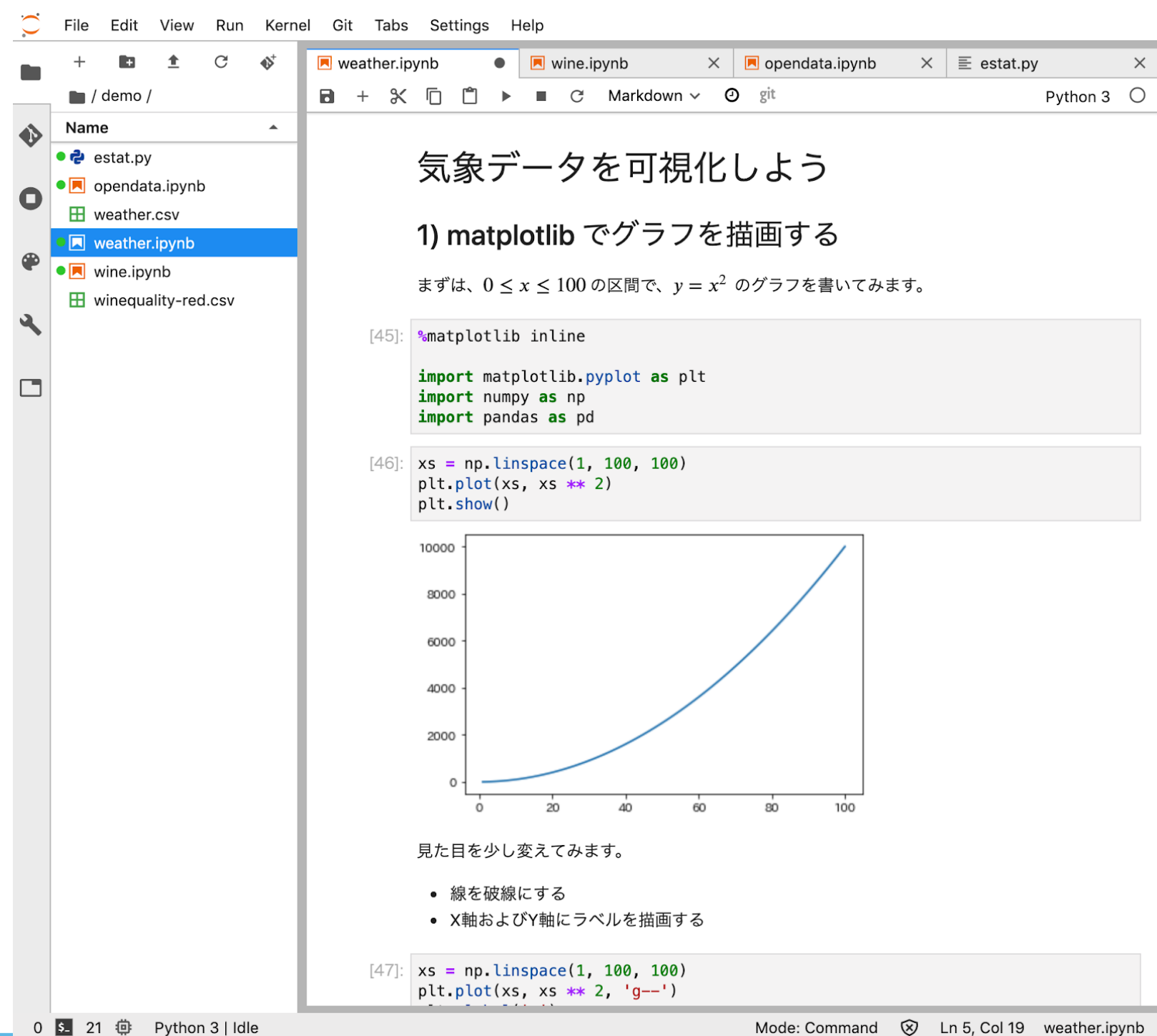
③

JupyterLab

CS概論の前半(アルゴリズムパート)では、当面 JupyterLab を使います。まずは、こちらの環境に慣れていきましょう。

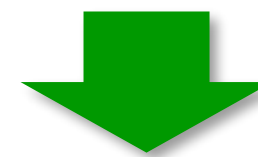
JupyterLab とは

- Pythonを利用してデータ分析を行うためのツール
 - プログラムだけでなく、説明や実行結果なども1枚のシートで纏めて管理できる「Python版実験ノート」



なぜJupyterを使うのか？

- Jupyterは、特にデータサイエンスの分野で、広く使われているツールの1つです
 - データサイエンス：コンピュータ・サイエンスや統計学を用いて、様々なデータから役に立つ知識を見出すことを目指す学問・研究分野です
(CS概論Ⅲで本格的に学習するので、お楽しみに！)
- データサイエンスでは、様々なデータを分かりやすいように可視化したり、パラメータを変えて試行錯誤したりといったことが必要です

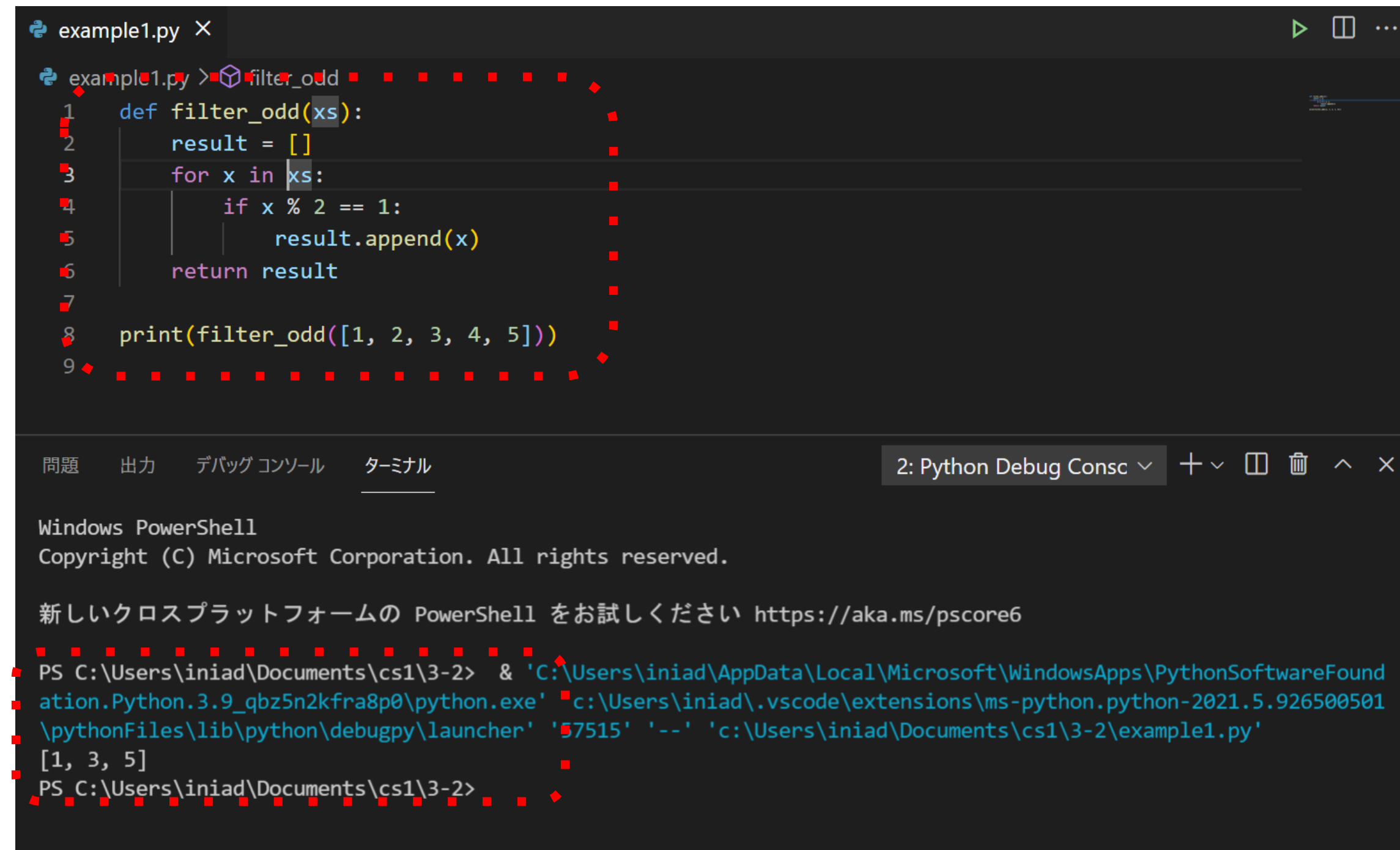


✓ Jupyterは、そういった目的に非常に適したソフトウェアです

これまで学習してきたPythonでは...

- CS概論 I で学習してきたPythonの環境では、プログラムと実行結果は別々に表示されています
 - プログラムを書いて、それを実行すると結果が表示されます

Pythonプログラム



The screenshot shows a code editor with a file named 'example1.py'. The code defines a function 'filter_odd' that takes a list 'xs' and returns a list of odd numbers. It then prints the result of 'filter_odd([1, 2, 3, 4, 5])'. Below the code editor is a terminal window titled '2: Python Debug Consc'. The terminal shows the command prompt 'PS C:\Users\iniad\Documents\cs1\3-2>' followed by the execution of the Python script. The output of the script is '[1, 3, 5]'. The terminal also displays some system information and a link to a PowerShell page.

```
example1.py X
example1.py > filter_odd
1 def filter_odd(xs):
2     result = []
3     for x in xs:
4         if x % 2 == 1:
5             result.append(x)
6     return result
7
8 print(filter_odd([1, 2, 3, 4, 5]))
9

問題 出力 デバッグ コンソール ターミナル
2: Python Debug Consc
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

新しいクロスプラットフォームの PowerShell をお試しください https://aka.ms/pscore6

PS C:\Users\iniad\Documents\cs1\3-2> & 'C:\Users\iniad\AppData\Local\Microsoft\WindowsApps\PythonSoftwareFound
ation.Python.3.9_qbz5n2kfra8p0\python.exe' 'c:\Users\iniad\.vscode\extensions\ms-python.python-2021.5.926500501
\pythonFiles\lib\python\debugpy\launcher' '57515' '--' 'c:\Users\iniad\Documents\cs1\3-2\example1.py'
[1, 3, 5]
PS C:\Users\iniad\Documents\cs1\3-2>
```

実行結果

JupyterLabでは...

- それに対して、Jupyterの場合はユーザがインタラクティブにプログラミングをすることができます
 - ドキュメント、プログラム、結果が全て1枚のシート内で管理できます
 - プログラムを書き換えて、再実行することも簡単です

ドキュメント

気象データを可視化する

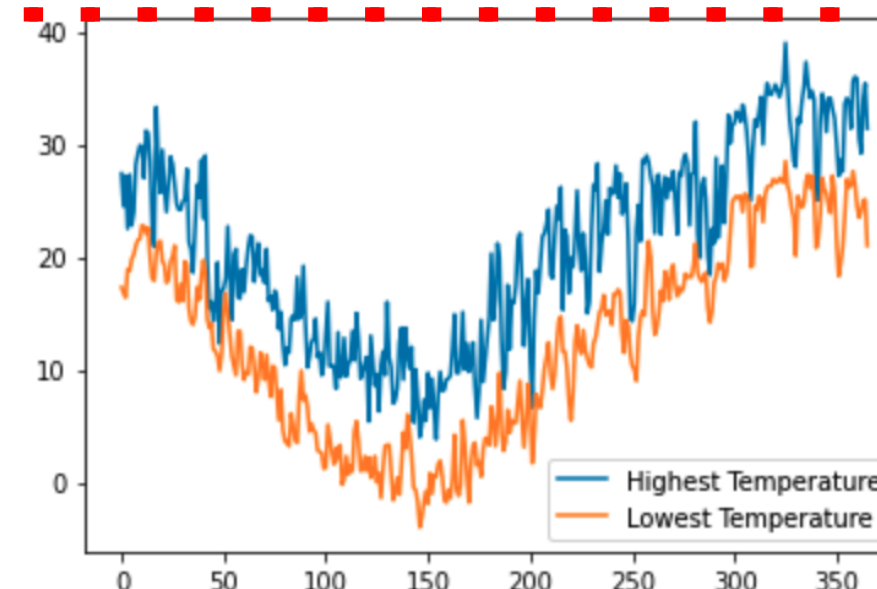
気象庁が公開しているCSV形式の気象データを可視化します。

```
[8]: import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

[13]: csv = pd.read_csv('weather.csv', skiprows=6, header=None)
x = np.arange(0, 366)
plt.plot(x, csv[1], label='Highest Temperature')
plt.plot(x, csv[4], label='Lowest Temperature')
plt.legend()
plt.show()
```

Pythonプログラム

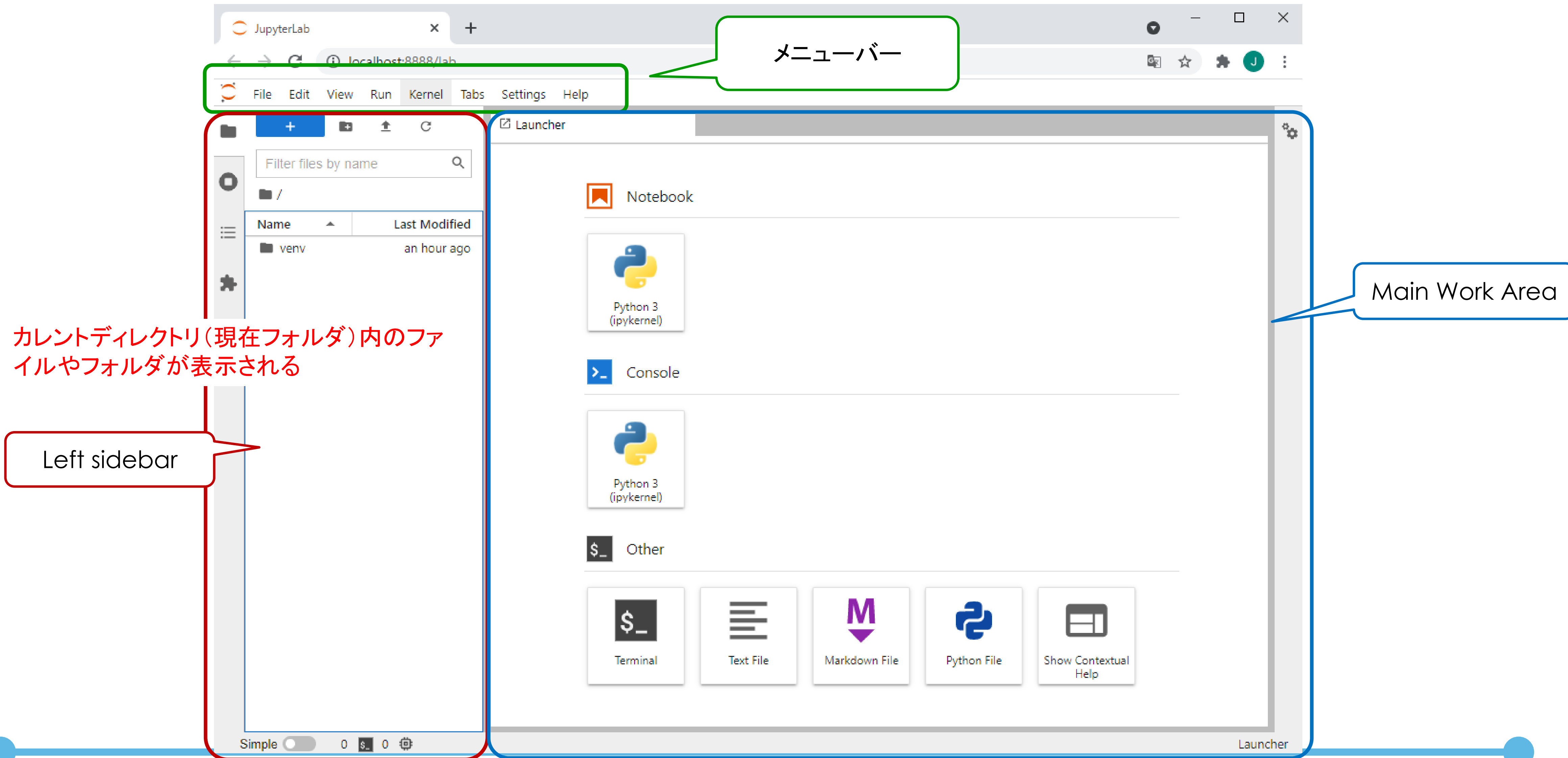
実行結果



JupyterLabの起動方法

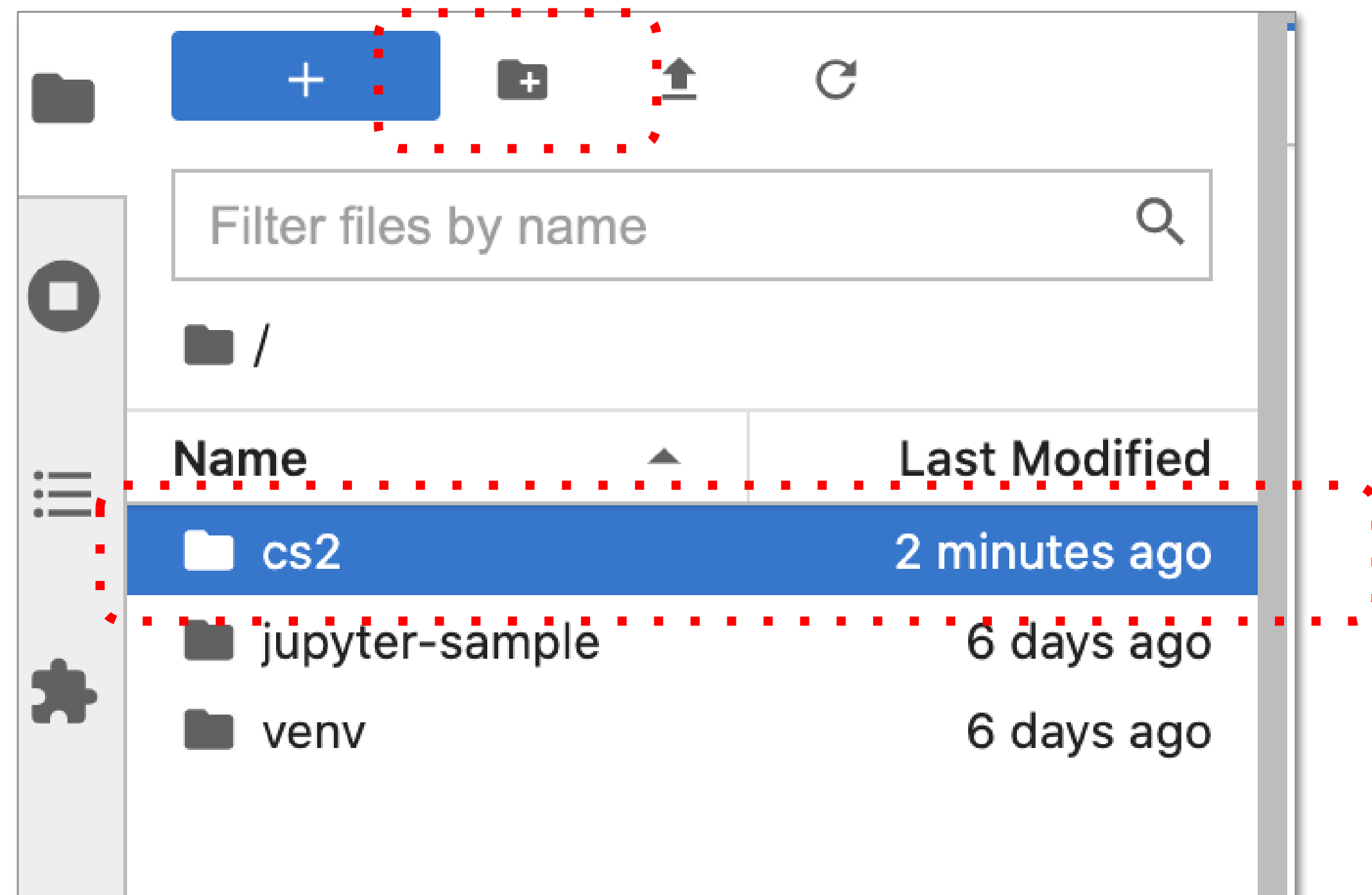
- JupyterLabを起動する際には、jupyterlab をインストールしたpython仮想で、以下のコマンドを実行します
 - `jupyter lab`
- 今回の講義では「情報連携基礎演習Ⅱ」の手順を完了後、Windows Power Shellで以下を実行します
 - `cd .\Documents\cs_exercise\cs2024_algo`
 - `.\venv\Scripts\activate`
 - `jupyter lab`
- このようにすると、ブラウザでJupyterLabが起動します
 - Power Shellは閉じないこと！

JupyterLab : 画面の概要



講義用フォルダの用意

- コンピュータ・サイエンス概論Ⅱの講義用に、cs2フォルダを作成しましょう
 - 細かい手順は省きます(わからない場合は、演習の手順を復習しましょう)

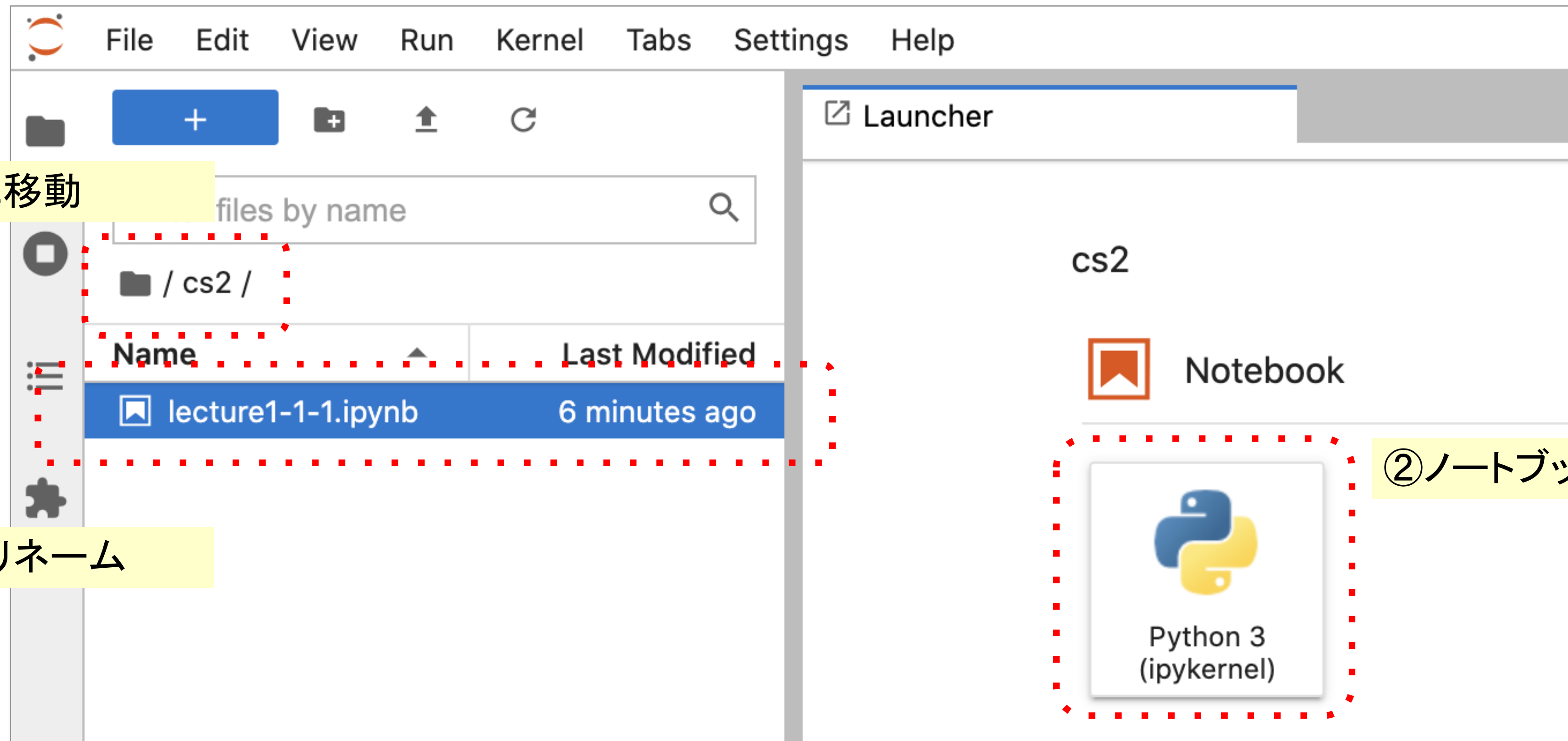


①フォルダを作成

②フォルダをリネーム

講義用ノートブックの用意

- cs2の下に、lecture1-1-1.ipynb を作成しましょう
 - 細かい手順は省きます(わからない場合は、演習の手順を復習しましょう)



①cs2フォルダに移動

③ノートブックをリネーム

②ノートブックを作成

JupyterLabの基本的な使い方

- JupyterLabでは、ノートブック内に複数のセルを並べていきます
- 次の2種類のセルを、自分で作れるようになりましょう(詳細は「情報連携基礎演習Ⅱ」の資料を確認すること)
 - Markdownセル：説明等を見やすく書くためのセルです
 - Codeセル：Pythonのプログラムを書くためのセルです
- どちらの場合も、Shift + Enterで実行できます

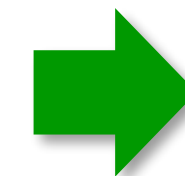
Markdown

```
## 1-1-1 JupyterLabの練習
```

```
JupyterLabでPythonの復習をしましょう。
```

Code

```
[ ]: s = 'INIAD'  
      print(s * 3)
```



1-1-1 JupyterLabの練習

JupyterLabでPythonの復習をしましょう。

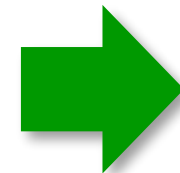
```
[1]: s = 'INIAD'  
      print(s * 3)
```

INIADINIADINIAD

JupyterLab 利用の際の留意点

- JupyterLabでは、プログラムはセル単位で実行されます
 - 通常のPythonプログラムは、ファイルの上から順に(まとめて)実行しましたが、JupyterLabの場合は明示的にセルを実行します
- どのセルを実行したかを把握しておかないと、挙動が分からなくなってしまうので、注意しましょう

```
[2]: x = 0
[3]: x += 1
[5]: x
[5]: 1
```



```
[2]: x = 0
[7]: x += 1
[8]: x
[8]: 3
```

同じセルを複数回実行すると、毎回変数が書き換わります！