

2023年度第3回 k-unet パソコン研修会

プログラミング言語 Python の概要

令和5年12月14日
K-unet 技術支援グループ
本間 強

1. Python とは

Python は1991年にオランダ人のガイド・ヴァンロッサムというプログラマーによって開発され、オープンソースで運営されているプログラミング言語です。Python という名前の由来は開発者がイギリス BBC のコメディ番組『空飛ぶモンティ・パイソン』の大ファンだったからとのことです。

Python はデスクトップアプリ作成の他、AI（機械学習・深層学習）、Webサイト作成、ブロックチェーン開発、Web上の情報収集ツール、データ処理・分析・解析など高度なプログラム作成にも対応している。プログラミング言語人気度を測定するTIOBEインデックスによると、2023年10月のランキングでは「Python」が1位を維持しています。

ランキング：<https://resource-sharing.co.jp/tiobe-index-202310/>

Python 対応OS：Windows, Mac 及び Linux

(1) 特徴

高級言語の一種であり、オブジェクト指向でインタープリター形式の言語です。プログラム言語やシステムで用途が決まっている予約語が35ほどしか無いので言語がシンプルで初心者でも覚えやすい。

(次ページ(4)参照)

機械学習やデータ解析の分野などで注目を集めているが、汎用性が高くあらゆる分野で利用されている。

<https://www.dhw.ac.jp/now/list/howtobe/python/>

(2) Python ができること

- ・一般的なデスクトップアプリの作成
- ・機械学習・人工知能(AI)の開発
- ・データの自動収集
- ・データの自動処理
- ・データ解析・分析
- ・Webアプリの開発
- ・ブロックチェーンの開発

(3) Python ができないこと、苦手なこと

- ・ゲーム開発
- ・スマホアプリの開発
- ・Windows 系デスクトップアプリの開発

(4) プログラムコードの違いの例

1 から 100 までの整数の和を求めるプログラムコード

1) BASIC プログラムコード

```
10 S = 1
20 E = 100
30 SUM = 0
40 FOR I = S to E
50     SUM = SUM + I
60 NEXT
70 PRINT SUM
```

2) C プログラムコード

```
#include <stdio.h>
int main(void) {
    int sum = 0;
    for(int i =1; i<101; i++) {
        sum += i;
    }
    print ("%d, sum");
    return 0;
}
```

3) Python プログラムコード

```
s = 0
for i in range (1, 101):
    s += i
print(s)
```

(Python はプログラムコード文字数が少ない)

2. Python をインストールする方法

Pythonを入手してインストールする方法は2つあります。どちらも個人利用の場合は無料でダウンロード/インストールができます。

(注) Google Chrome でURLを開くと日本語表示に切り替え可能

(1)Python 公式サイトからダウンロードしてインストールする

Python公式サイト : <https://www.python.org/>

(注) 利用するライブラリーは別途ダウンロード/インストールが必要。

(2)Anaconda をダウンロードしてインストールする

Anaconda は Pythonの実行環境を提供し、データサイエンスに必要とされる各種ツールやライブラリを提供するプラットフォームです。Pythonの開発はライブラリによって補完される部分が非常に大きいため有用なライブラリをインストールする必要があるが、初めてPythonに触れる場合にはどのようなライブラリを導入すべきか分かりにくいので、その点を補ってくれる。

以下のURLからインストールファイルをダウンロードしてインストールする。

Anaconda ダウンロード先 : <https://www.anaconda.com/>

ファイル容量 : 約 1.07 GB (2023年11月21日ダウンロード時)

5. Anaconda によるプログラム作成

(1) Anacondaの起動 (Windows 11 の場合)

①タスクバーのスタートアイコンを左クリックし、表示画面右上の「全てのアプリ」をクリックする。表示されている「Anaconda3 (64bit)」をクリックし、Anaconda Navigator をクリックする。アプリケーションのメニュー画面が表示される。

②19種類のアイコンが表示されるが通常利用しているアイコンは以下のとおり。

1)CMD.exe Prompt : アプリケーションやライブラリのアップデート

2)Spyder : アプリケーション開発のための統合開発環境 (IDE)

3)JupyterNotenook : Webブラウザ上でプログラムを作成実行し、グラフ、作業メモや関連するドキュメントなどをファイル形式にまとめて一元的に管理することを目的としたオープンソースツール。

4)JupyterLab : JupyterNotebookと同様にWebブラウザ上で動作する。

JupyterNotebook の改良版であり、ユーザーインターフェースや機能が改善されている。JupyterLab はより柔軟で機能豊富なインターフェースを提供する。
<https://ai-inter1.com/jupyter-lab/>

Anacondaの起動 : ①から③を順次クリック

① スタートアイコンを左クリック



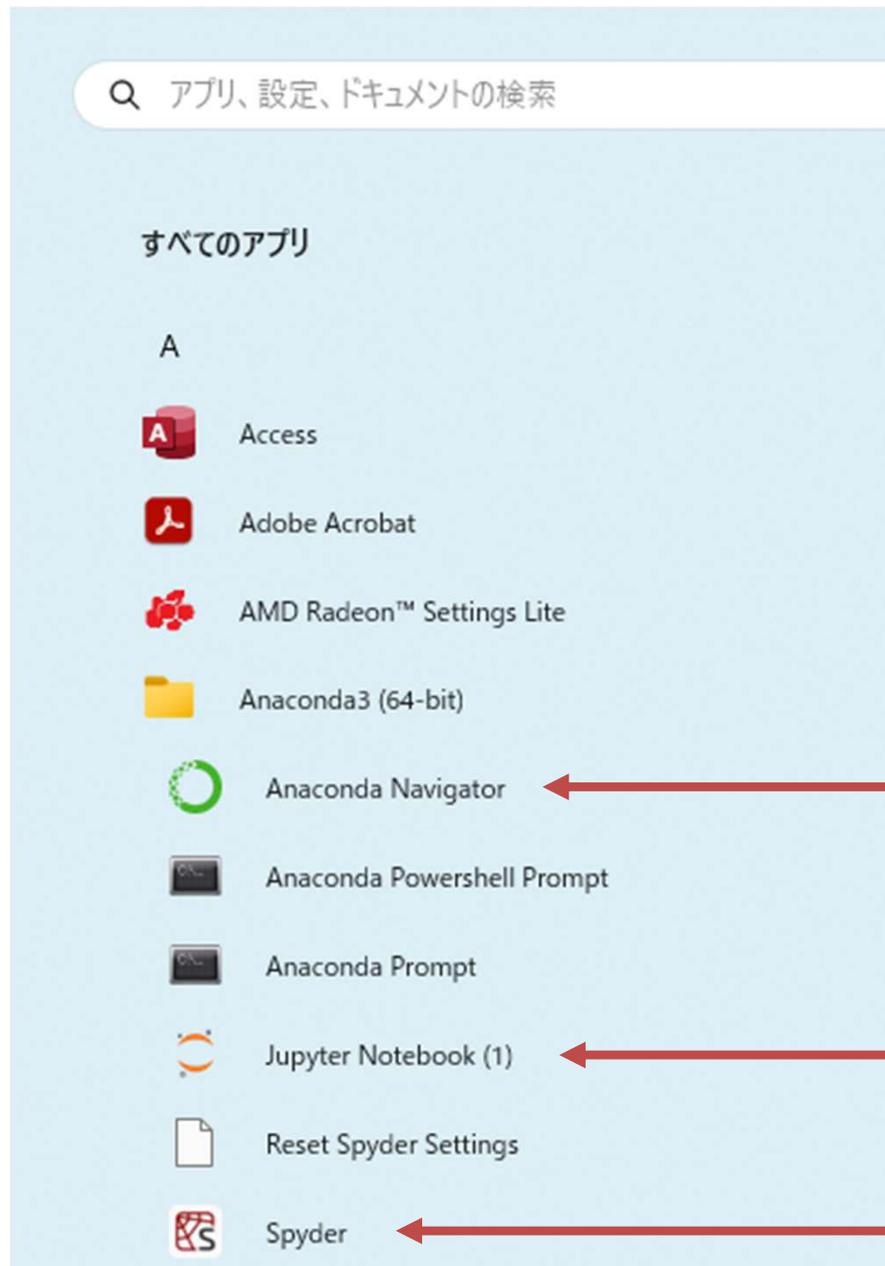
② 全てのアプリを左クリック



③ Anaconda3 を左クリック



Anaconda アプリの起動



④ Anaconda Navigator 起動

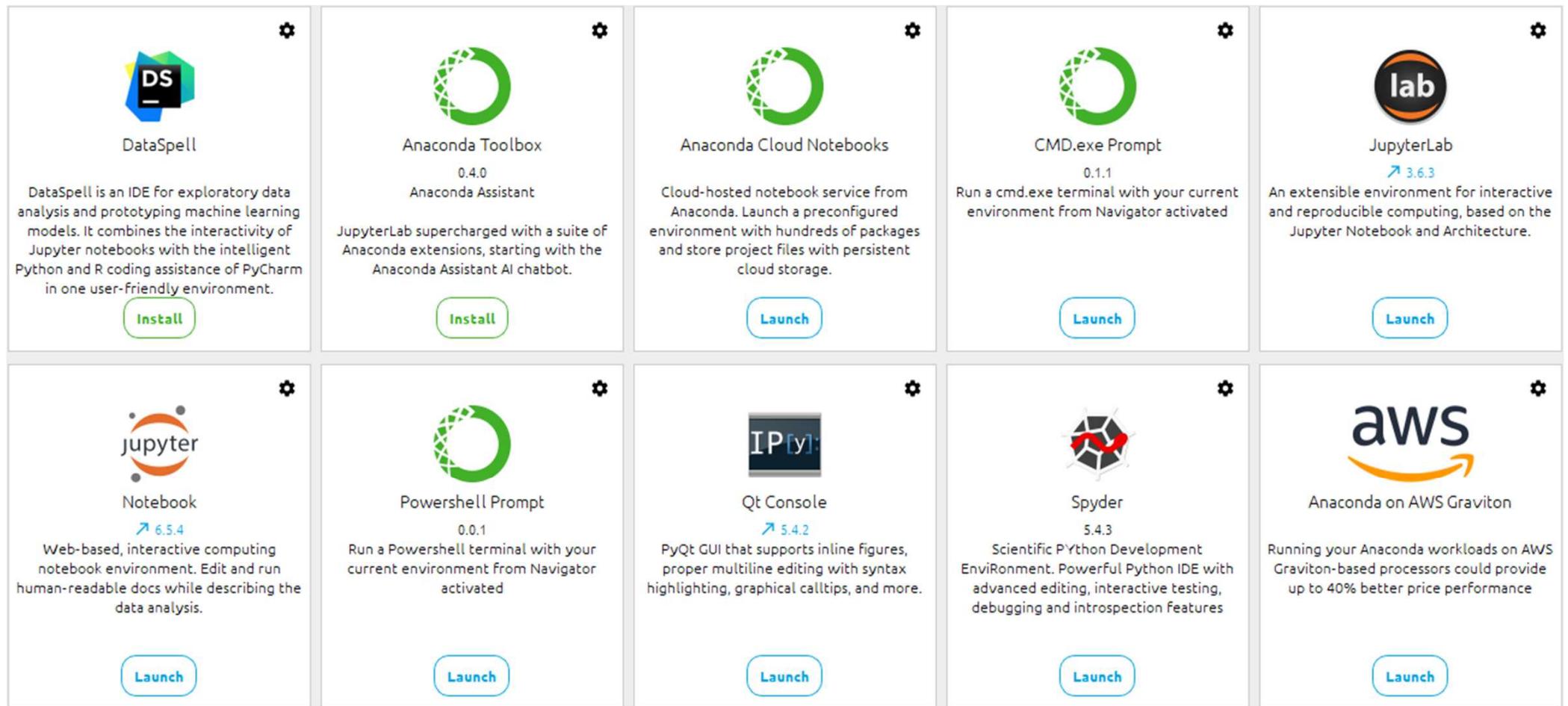
⑤ Jupyter Notebook 起動

⑥ Spyder 起動

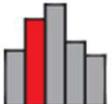
Anaconda Navigator の画面

19種類のアプリケーションのメニューアイコン画面が表示される。

Anaconda メニュー画面 1

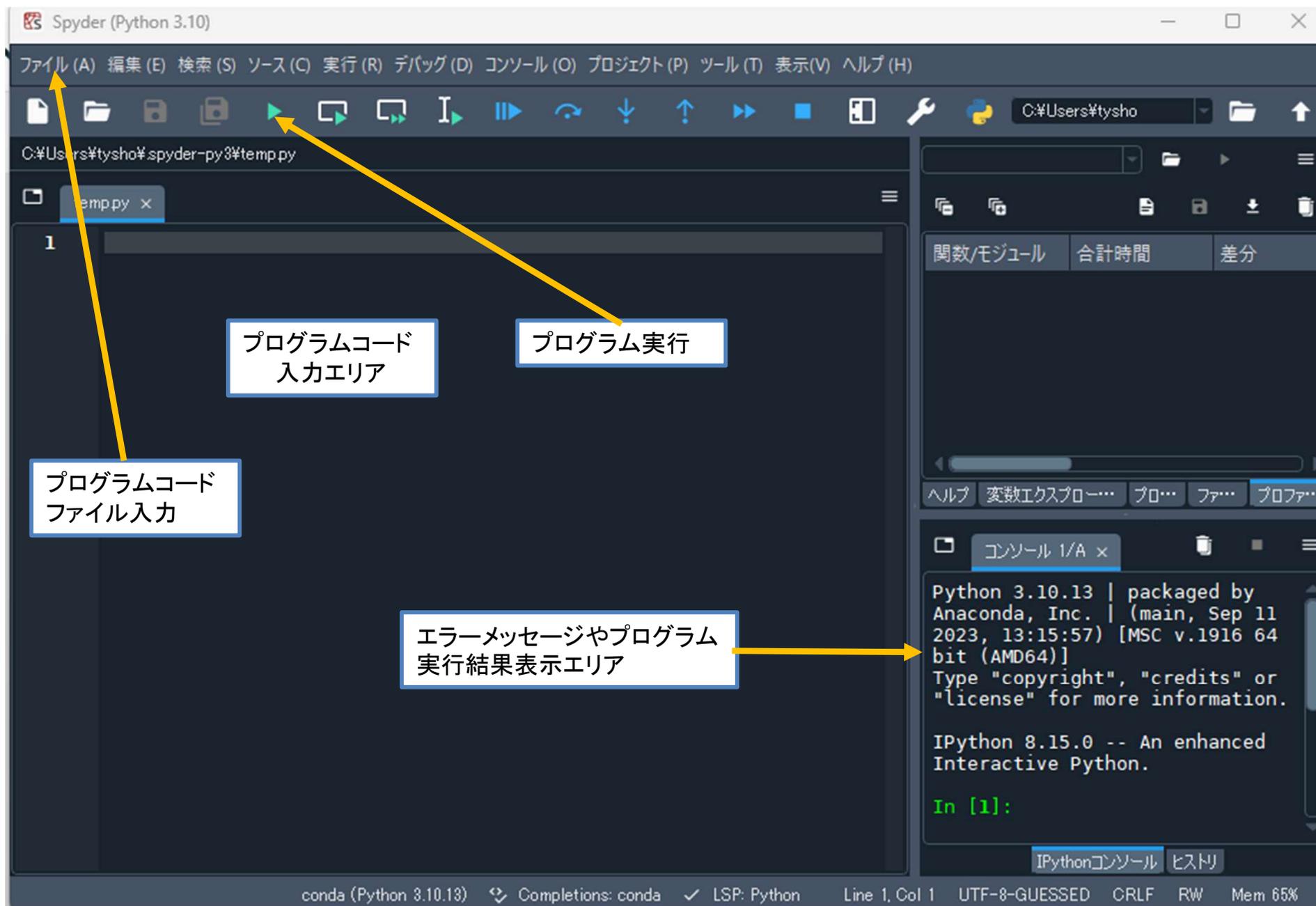


Anaconda メニュー画面 2

 <p>Datalore</p> <p>Kick-start your data science projects in seconds in a pre-configured environment. Enjoy coding assistance for Python, SQL, and R in Jupyter notebooks and benefit from no-code automations. Use Datalore online for free.</p> <p>Launch</p>	 <p>IBM watsonx</p> <p>IBM watsonx is an enterprise-ready AI platform including a data store, model builder, and AI model management and monitoring.</p> <p>Launch</p>	 <p>ORACLE Cloud Infrastructure</p> <p>Oracle Data Science Service</p> <p>OCI Data Science offers a machine learning platform to build, train, manage, and deploy your machine learning models on the cloud with your favorite open-source tools</p> <p>Launch</p>	 <p>console_shortcut_miniconda</p> <p>0.1.1 Anaconda Powershell Prompt</p> <p>Install</p>	 <p>Glueviz</p> <p>1.2.4 Multidimensional data visualization across files. Explore relationships within and among related datasets.</p> <p>Install</p>
 <p>Orange 3</p> <p>3.34.0 Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.</p> <p>Install</p>	 <p>powershell_shortcut_miniconda</p> <p>0.0.1 Anaconda Powershell Prompt</p> <p>Install</p>	 <p>PyCharm Professional</p> <p>A Full-fledged IDE by JetBrains for both Scientific and Web Python development. Supports HTML, JS, and SQL.</p> <p>Install</p>	 <p>RStudio</p> <p>1.1.456 A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.</p> <p>Install</p>	

(2) デスクトップアプリの作成と実行

1) Spyderを起動する



2) プログラム編集画面が表示されるので、プログラムコードを手入力するか、あるいはエディターで作成・保存したプログラムコードをプログラム編集画面に読み込む。

3) プログラムの実行手順

① プログラム実行結果が数値あるいは文字列のみの場合

上部メニューのスタートマーク  をクリックする。

② プログラム実行結果が図形を伴う場合

スタートマークをクリックすると Spyder 画面の下に表示されて見えない場合があるので、メニュー画面の「実行(Run)」、「プロファイラー(Profiler)」を順次クリックする。

4) デスクトップアプリの例

① 1 から 100 までの数の合計を求めるプログラム

プログラムコードを画面左側に入力してスタートマークをクリックする。

計算結果が画面右側最下部のウィンドウに表示される。

② タートルグラフィック (Tree1.py, Tree2.py)

作成済みのプログラムコードを読み込んでメニュー画面の「実行(Run)/プロファイラー(Profiler)」を順次クリックする。

③ スライドショー

作成済みのプログラムコードを読み込んでメニュー画面の「実行(Run)/プロファイラー(Profiler)」を順次クリックする。

④ Excelデータの処理 (詳細は別紙参照)

Sum100.py 計算結果

The screenshot shows the Spyder Python IDE interface. The main editor displays the following Python code:

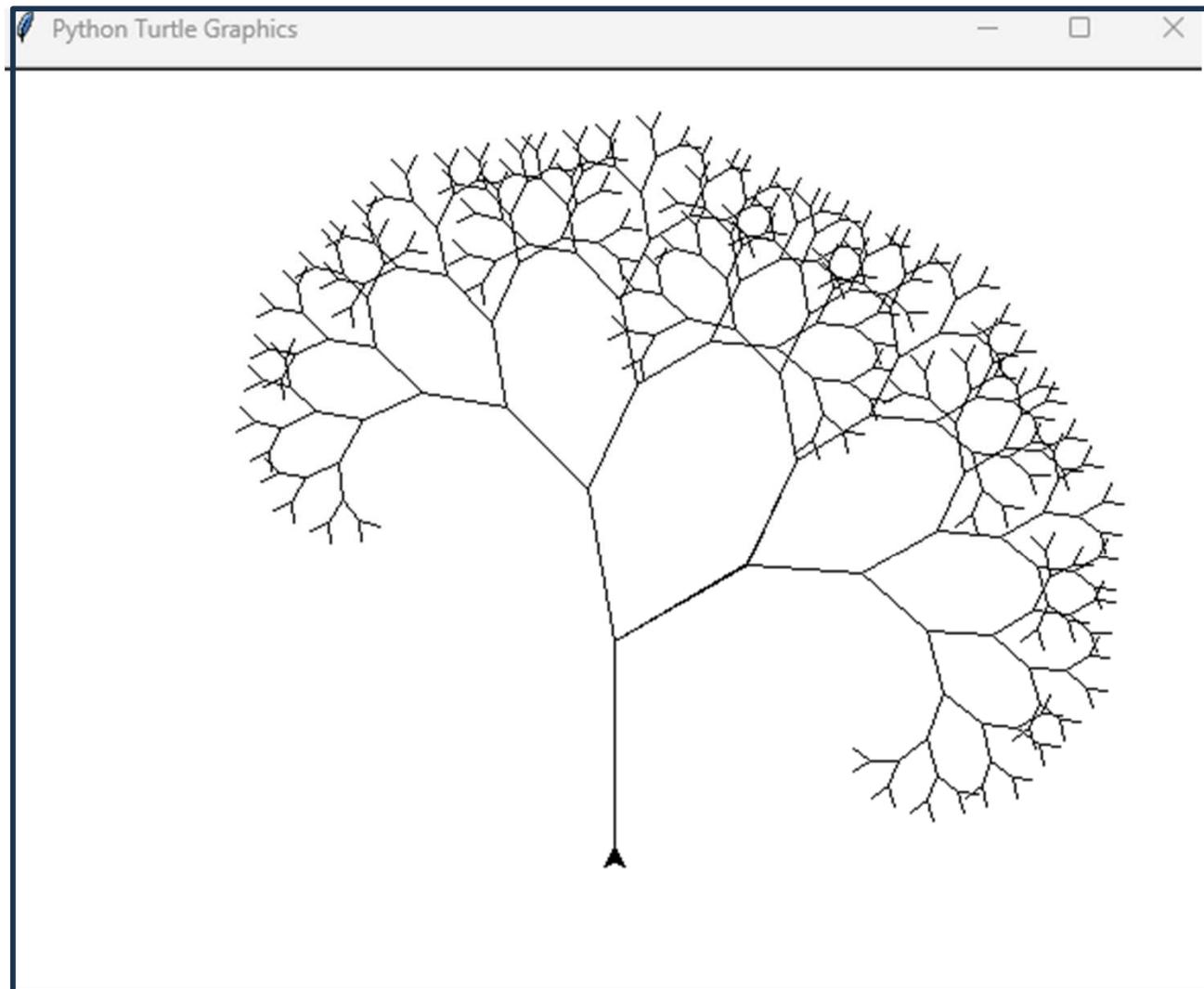
```
1 s = 0
2 for i in range(1, 101):
3     s += i
4 print(s)
5
```

The right-hand side of the IDE contains a console window showing the execution output:

```
Sum100.py , wd11 - C:/Users/tysho/OneDrive/Documents/PC研修会/2023年度研修会/第3回(20231214)/Pythonプログラム例/デスクトップアプリ)
5050
In [5]:
```

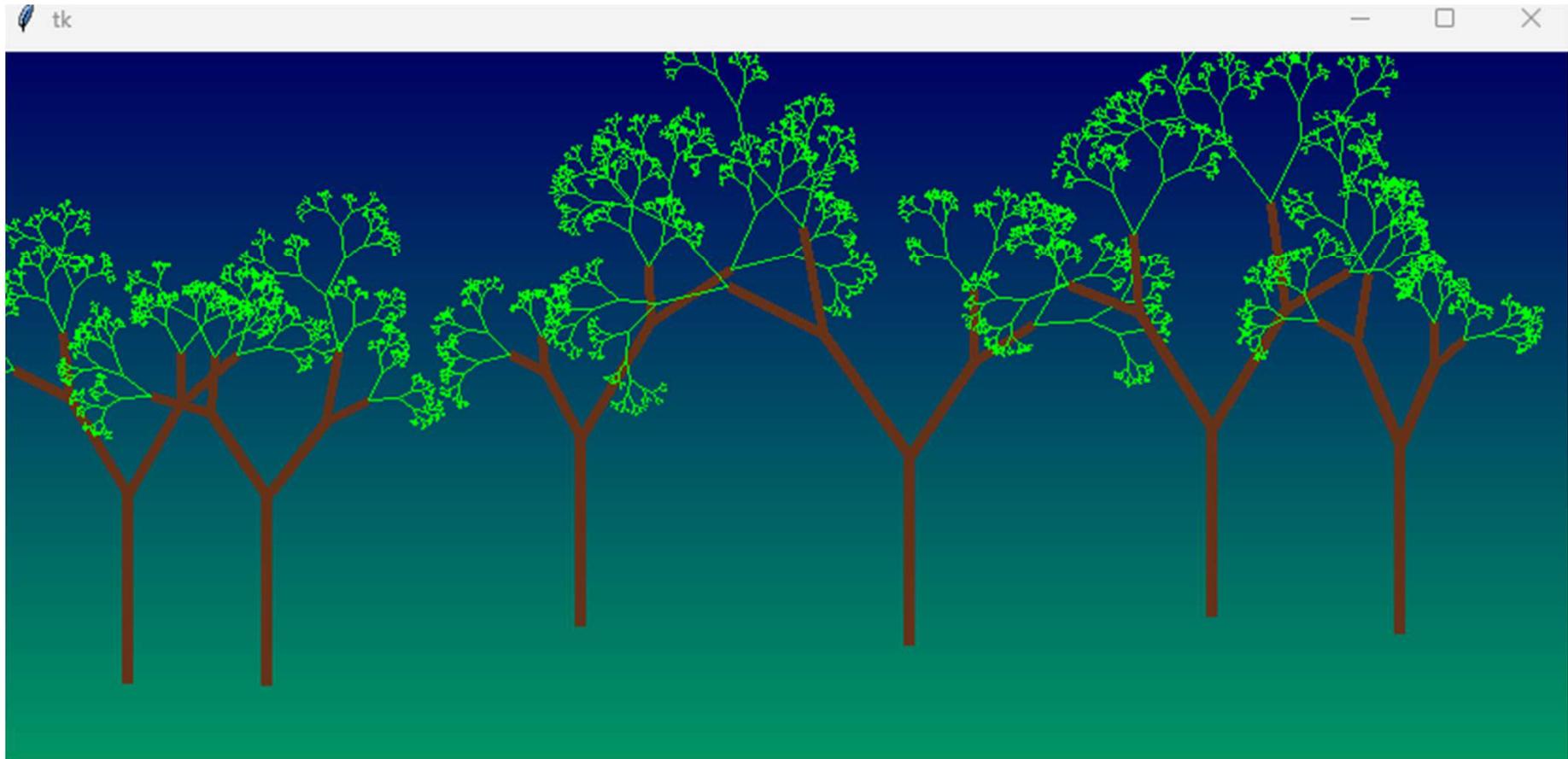
The status bar at the bottom indicates the environment is conda (Python 3.11.5) and shows various tool settings like Completions: conda, LSP: Python, and memory usage at 66%.

Tree1.py
実行画面



Tree2.py 実行画面

(マウスをクリックした部分に Tree 表示)



(プログラムコード出典 : 日経ソフトウェア誌)

Slideshow.py 実行画面



③ Excelデータ処理の例

Excel データを処理するライブラリー OpenPyXL を用いてExcelデータを
読み込み、それを加工してその結果を Excel 形式のデータファイルとして出力
できる。例として1年間の会計データを読み込み、勘定科目別にソートして
合計金額を出力するプログラム Account.py を作成してみた。

プログラムの概要は以下のとおり。

- 1) 所要のライブラリをインポート
- 2) 読み込む Excel ファイルを指定
- 3) 科目別のデータ収納リスト (IDリスト) の定義
- 4) データが記録されているExcelファイルを開く
- 5) データが記録されているExcelファイルのシート番号を指定
- 6) データが記録されているExcelファイルを先頭から最後まで読み込み、
指定した科目毎にデータを抜き取って合計額を算出しデータ収納
リスト (IDリスト) に順次書き込む。
- 7) IDリストをExcel形式のファイルで出力する。

出力されたExcel ファイルをエクセルで読み込み、目的どおりに科目別に
ソートされていることを確認する。

(参照図書：技術評論社「PythonでかなえるExcel作業効率化」)

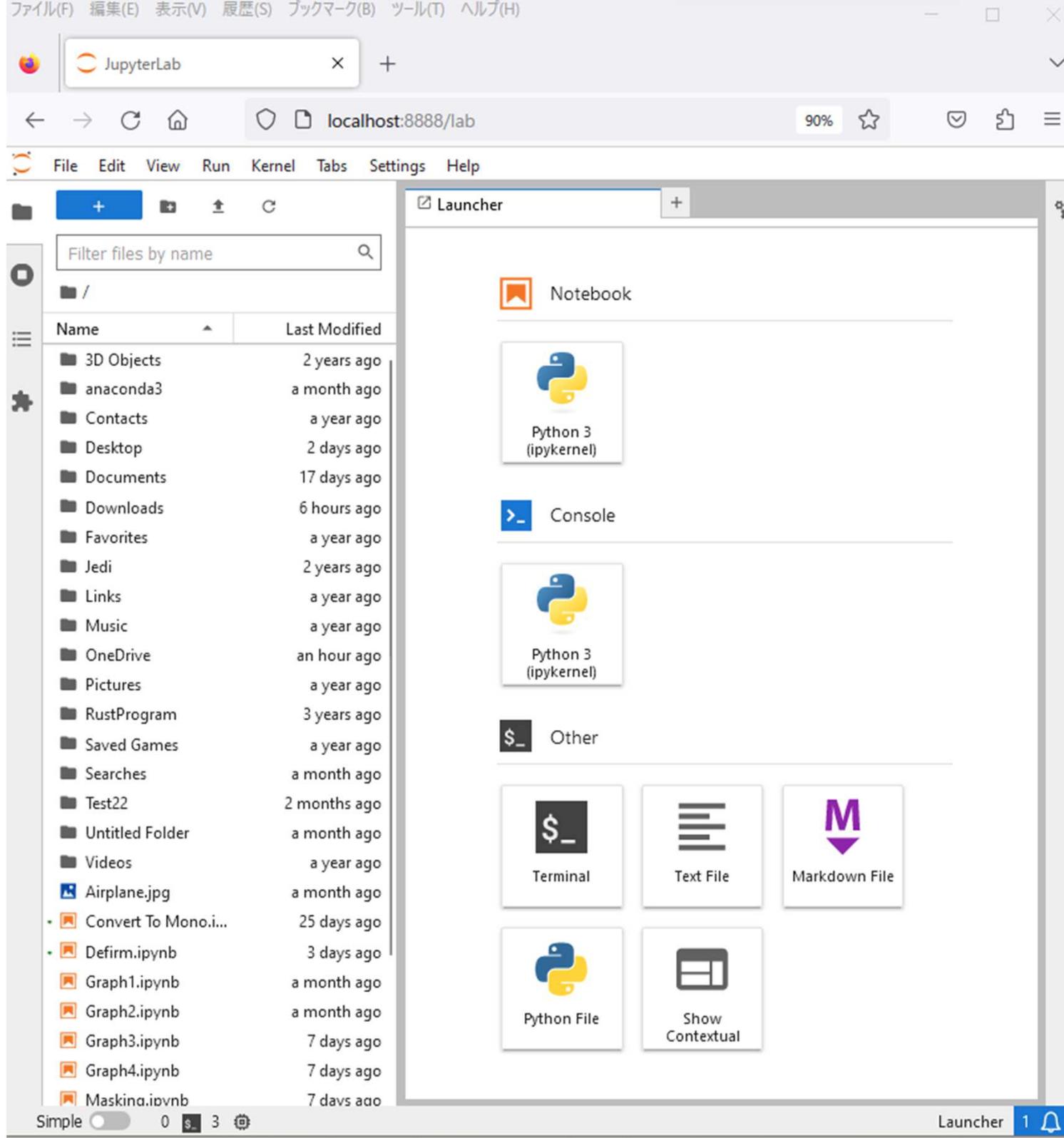
(3) JupyterLab によるプログラム作成

Webブラウザ上でプログラムを作成実行し、数式、グラフ、作業メモや関連するドキュメントなどをファイル形式にまとめて一元的に管理することを目的としたオープンソースツール。利用手順は以下のとおり。

- 1) メニュー画面 JupyterLab アイコンの Launch をクリックして起動する。
- 2) Lancher 「Notebook」 下部の 「Python 3(jpykernel)」 アイコンをクリックする。
- 3) 青色線で囲まれた部分に手動でプログラムコードを入力する。あるいはエディターで作成・保存したプログラムコードをコピーする。
- 4) 上部メニュー画面のスタートマーク  をクリックするとアプリが動作する。

JupyterLab 起動画面

右側 Launcher部
Notebook下部の
Python3(ipynkernel)
をクリックすると
JupyterLabのプログラ
ム作成画面が表示される。



JupyterLab プログラム 作成画面

The screenshot displays the JupyterLab web interface in a browser window. The browser's address bar shows the URL `localhost:8888/lab/tree/Untitled.ipynb`. The JupyterLab interface includes a top menu bar with options like File, Edit, View, Run, Kernel, Tabs, Settings, and Help. On the left side, there is a file browser pane with a search bar and a list of files and folders. The file list includes folders like Favorites, Jedi, Links, Music, OneDrive, Pictures, RustProgram, Saved Games, Searches, Test22, Untitled Folder, and Videos, as well as various image and notebook files. The file `Untitled.ipynb` is currently selected. The main area on the right is a code editor for `Untitled.ipynb`, showing a blank code cell with a Python 3 (ipykernel) kernel selected. The bottom status bar indicates the current mode is Command, and the cursor is at line 1, column 1.

Name	Last Modified
Favorites	a year ago
Jedi	2 years ago
Links	a year ago
Music	a year ago
OneDrive	8 minutes ago
Pictures	a year ago
RustProgram	3 years ago
Saved Games	a year ago
Searches	a month ago
Test22	2 months ago
Untitled Folder	a month ago
Videos	a year ago
Airplane.jpg	a month ago
Convert To M...	25 days ago
Defirm.ipynb	3 days ago
Graph1.ipynb	a month ago
Graph2.ipynb	a month ago
Graph3.ipynb	7 days ago
Graph4.ipynb	7 days ago
Masking.ipynb	7 days ago
Park2.jpg	6 months ago
Test1.ipynb	2 months ago
TurnSide.ipynb	a month ago
Untitled.ipynb	a minute ago
三次元曲線 1 ...	8 days ago
三次元曲線 2 ...	25 days ago

1 から100
までの合計
計算

The screenshot shows a web browser window displaying a JupyterLab interface. The browser's address bar shows the URL `localhost:8890/lab`. The JupyterLab interface includes a file browser on the left, a central code editor, and a bottom status bar.

The file browser on the left shows a list of files and folders with columns for Name and Last Modified. The code editor in the center displays a Python notebook with the following code:

```
[1]: # 1から100までの整数の和の計算
```

```
[3]: S = 0
for i in range(1, 101):
    S += i
print(S)
```

The output of the code is `5050`. The status bar at the bottom indicates the current mode is Command, the cursor is at Ln 1, Col 1, and the file name is `1~100までの和計算.ipynb`.

JupyterLab プログラム 実行例

The screenshot shows the JupyterLab interface. On the left is a file browser with a search bar and a list of files and folders. The file 'Graph4.ipynb' is selected. On the right is the code editor for 'Graph4.ipynb'. The code in the editor is as follows:

```
[2]: import numpy as np
import matplotlib.pyplot as plt
fig = plt.figure()
ax1 = fig.add_subplot(111)

x = np.linspace(0, 4*np.pi, 401)
line1=ax1.plot(x, np.sin(x))
line2=ax1.plot(x, np.cos(x))
ax1.set_xlabel('[rad]')
labes=["sin(θ)", "cos(θ)"]
ax1.legend(labes)
plt.show()
```

Below the code is a plot showing the sine and cosine functions. The x-axis is labeled '[rad]' and ranges from 0 to 12. The y-axis ranges from -1.00 to 1.00. The sine function is plotted in blue and the cosine function in orange. A legend in the bottom-left corner identifies the lines as 'sin(θ)' and 'cos(θ)'.

3次元 グラフ

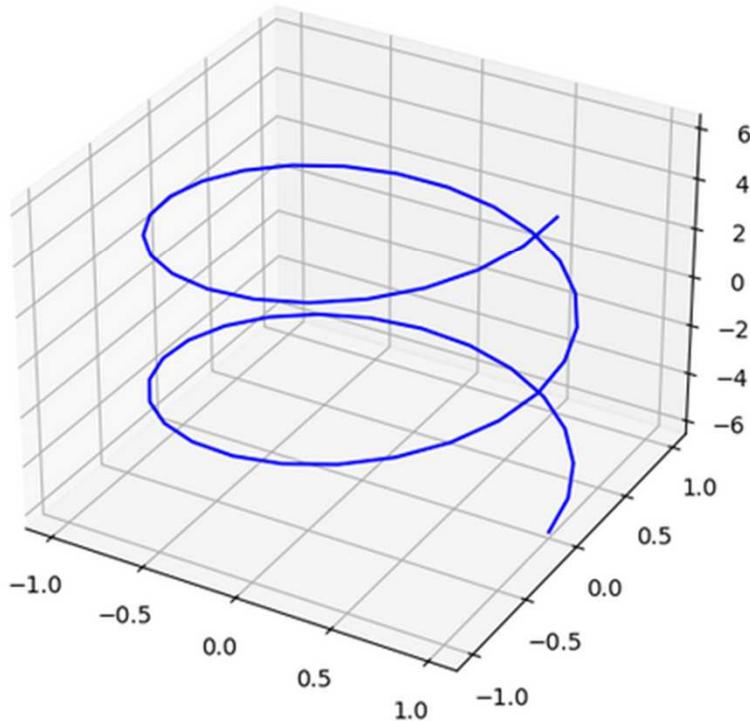
```
Launcher x 三次元曲線1.ipynb x +
+ ✂ 📄 ▶ ■ ↻ ▶ Code v

[1]: # 3D グラフ (https://tech-market.org/matplotlib-3d/)

# 使用するパッケージをインポート
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
import numpy as np

# 図示する関数
t = np.linspace(-2*np.pi, 2*np.pi)
x = np.cos(t)
y = np.sin(t)
z = t

# Input
fig, ax = plt.subplots(figsize=(6, 6), subplot_kw={'projection': '3d'})
ax.plot(x, y, z, c='b')
ax.set_xticks(np.linspace(-1.0, 1.0, 5))
ax.set_yticks(np.linspace(-1.0, 1.0, 5))
plt.show()
```



写真の加工 1

(プログラムコード)

出典: 日経ソフトウェア誌

```
Defirm.ipynb X +
+ ✂ 📄 📄 ▶ ■ ↺ ▶▶ Code ▾ 🐛 ...

[1]: # Deform

from PIL import Image
import matplotlib.pyplot as plt
import math

src_img = Image.open('Park2.jpg')

img_w, img_h = src_img.size

dst_w = int(img_w * 1.5)
dst_h = int(img_h * 1.5)
dst_img = Image.new('RGB', (dst_w, dst_h), (255,255,255))

sx = int((dst_w / 2) - (img_w / 2))
sy = int((dst_h / 2) - (img_h / 2))

a = 30
f = 2
w = (math.pi * 2) / (img_w / f)
s = 0

for y in range(img_h):
    for x in range(img_w):
        rgb = src_img.getpixel((x, y))
        dy = int(y + a * math.sin(w*x+s))
        dst_img.putpixel((x+sx, dy+sy), rgb)

plt.imshow(dst_img)
plt.show()
```

写真の加工2 (加工後の写真)

The screenshot displays a JupyterLab environment. On the left, a file browser shows a list of files and folders. The file 'Defirm.ipynb' is selected. The main area shows a code editor with the following Python code:

```
w = (math.pi * 2) / (img_w / f)
s = 0

for y in range(img_h):
    for x in range(img_w):
        rgb = src_img.getpixel((x, y))
        dy = int(y + a * math.sin(w*x+s))
        dst_img.putpixel((x+sx, dy+sy), rgb)

plt.imshow(dst_img)
plt.show()
```

Below the code editor, a plot displays the result of the image processing. The plot shows a landscape image with a tree and a field of flowers, distorted by a sine wave function. The x-axis ranges from 0 to 1200, and the y-axis ranges from 0 to 800.

JupyterLab のプログラム実行例を html コードで保存する手順

- (1) 上部メニューの「File」をクリックする。
- (2) 「Save and Export Notebook As」にマウスポインターを移動し「HTML(html)」をクリックする。
- (3) エクスプローラを起動して「ダウンロード」フォルダーをクリックすると保存された html ファイルが確認できる。

Webブラウザ上で Python が動作する PyScript

PyScriptとは？

PyScript は「Run Python in Your HTML」と示されているとおり、JavaScript のように HTMLコードに Python コードを挿入してWebブラウザ上でプログラムを実行できる。現在は開発段階なので仕様が変更される可能性がある。

PyScript のメリット

- ・わざわざ Python をインストールする必要がない。

PyScript のデメリット

- ・表示速度が遅い。
- ・リリースされて間がないので情報が少ない。

PyScript テストコード1

以下のコードをテキストエディターで作成して PS-Test1.html の名前で保存する。保存した html ファイルを左ダブルクリックすると JupyterLab のような対話モードアプリが表示される。（Edge の場合：右クリックし Edge を選択）

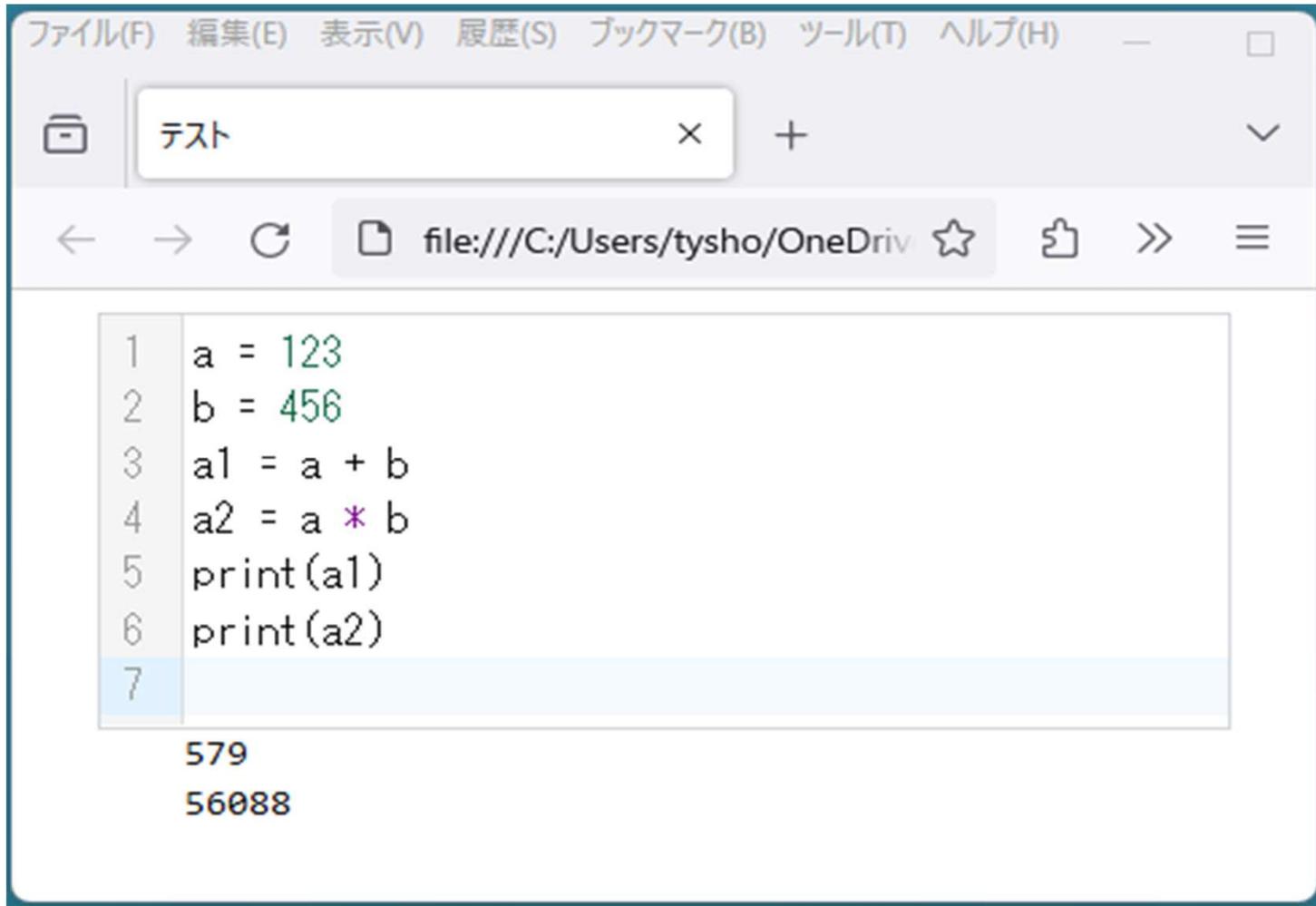
（注）「body」部分に <py-repl> </py-repl> のコードが含まれているので対話モードアプリが表示される。

```
<html lang="ja">
  <head>
    <meta charset="UTF-8">
    <title>テスト</title>
    <link rel="stylesheet" href="https://pyscript.net/alpha/pyscript.css" />
    <script defer src="https://pyscript.net/alpha/pyscript.js"></script>
  </head>
  <body>
    <py-repl></py-repl>
  </body>
</html>
```

（出典：<https://gihyo.jp/article/2023/04/monthly-python-2304>）

PS-Test1.html をクリックすると現れる画面

アプリが表示された後にプログラムコードを入力し、「Shift」キーを押下しながら「Enter」キーを押下するとプログラムが実行されて計算結果が表示される。



The screenshot shows a web browser window with a single tab titled "テスト". The address bar displays the file path: file:///C:/Users/tysho/OneDriv. The main content area contains a Python script with line numbers 1 through 7. The script defines variables a and b, calculates their sum (a1) and product (a2), and prints both results. Below the script, the output of the program is displayed: 579 and 56088.

```
1 a = 123
2 b = 456
3 a1 = a + b
4 a2 = a * b
5 print(a1)
6 print(a2)
7
```

579
56088

PyScript テストコード2

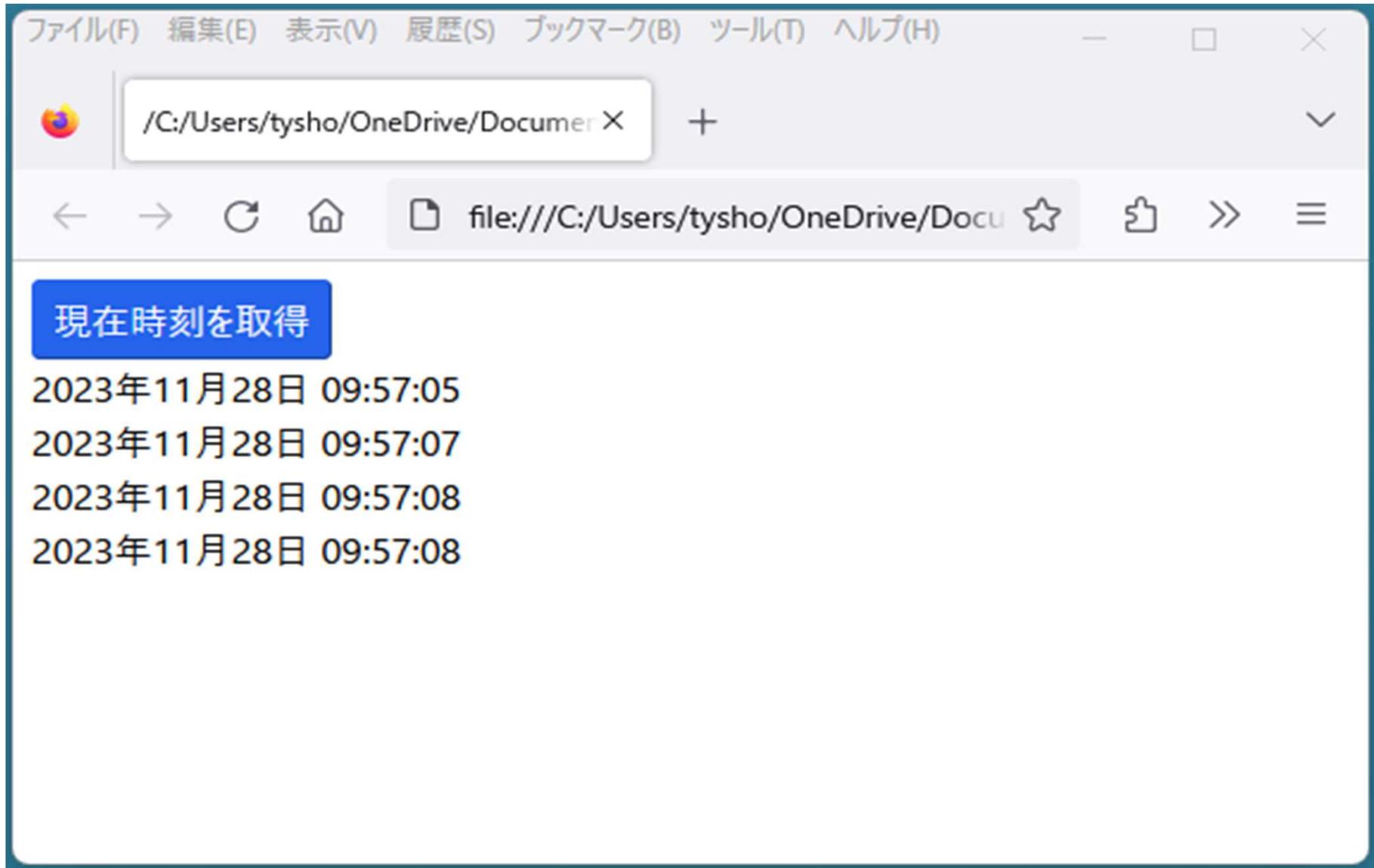
下記コードをテキストエディターで作成して PS-Test2.htmlの名前で保存する。保存した html ファイルを左ダブルクリックするとアプリが起動する。(Edgeの場合：右クリックしEdgeを選択)

```
<html lang="ja">
  <head>
    <meta charset="utf-8"/>
    <link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />
    <script defer src="https://pyscript.net/latest/pyscript.js"></script>
  </head>
  <body>
    <button py-click="get_now()" class="py-button">
      現在時刻を取得
    </button>
    <div id="now"></div>
    <py-script>
      from datetime import datetime
      def get_now():
        now = datetime.now()
        display(f"{now:%Y年%m月%d日 %H:%M:%S}", target="now")
    </py-script>
  </body>
</html>
```

(出典 : <https://gihyo.jp/article/2023/04/monthly-python-2304>)

PS-Test2.html をクリックすると現れる画面

「現在時刻を取得」部分をクリックすると、クリックした時点の日時が表示される



おわり