

Three green apples are arranged on a white background. One apple is in the foreground, slightly to the right, and two are behind it, one to the left and one to the right. The apples are bright green and have a small stem at the top.

IoT環境における知的情報処理技術 環境設定 (Linux編)

愛媛大学
二宮 崇

ninomiya@cs.ehime-u.ac.jp

Python環境設定

- この講座では次のプログラミング環境を使って演習を行います
 - Anaconda (機械学習向けライブラリが入ったPythonパッケージ)
 - PyTorch (深層学習ツール)
- 自宅で使っているPC (Windows, Mac, Linux) にインストールします



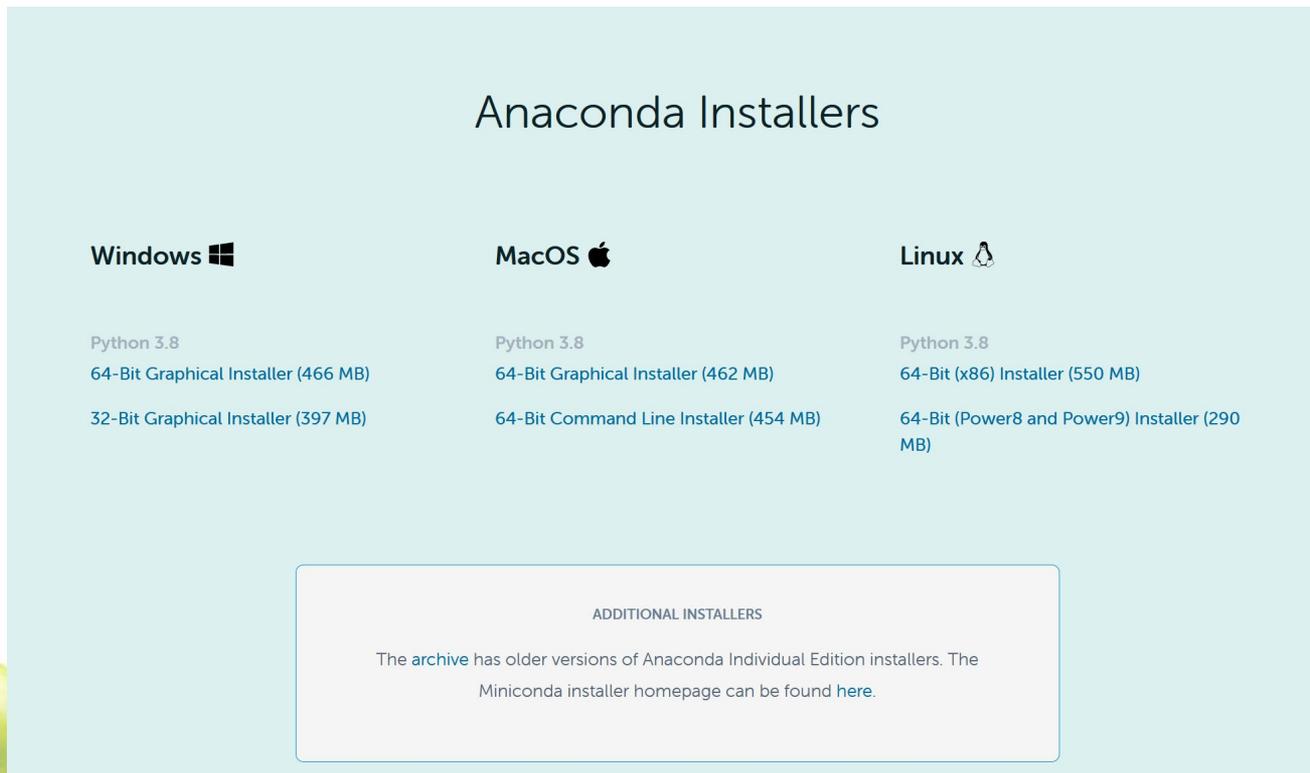
Python環境設定

ANACONDAインストール



Anacondaインストール

- 最初に次のAnacondaダウンロードページにアクセス
 - <https://www.anaconda.com/products/individual#Downloads>



The screenshot shows the 'Anaconda Installers' page. It is divided into three columns for Windows, MacOS, and Linux. Each column lists Python 3.8 installers with their bitness and size. A box at the bottom provides information about additional installers.

Operating System	Python Version	Installer Type	Size
Windows	Python 3.8	64-Bit Graphical Installer	466 MB
		32-Bit Graphical Installer	397 MB
MacOS	Python 3.8	64-Bit Graphical Installer	462 MB
		64-Bit Command Line Installer	454 MB
Linux	Python 3.8	64-Bit (x86) Installer	550 MB
		64-Bit (Power8 and Power9) Installer	290 MB

ADDITIONAL INSTALLERS

The [archive](#) has older versions of Anaconda Individual Edition installers. The Miniconda installer homepage can be found [here](#).



Anacondaインストール

- 各自が使っているOS(Windows, MacOS, Linux)に合わせてインストーラーをダウンロードして、実行します

Anaconda Installers

Windows 

Python 3.8

64-Bit Graphical Installer (466 MB)

32-Bit Graphical Installer (397 MB)

MacOS 

Python 3.8

64-Bit Graphical Installer (462 MB)

64-Bit Command Line Installer (454 MB)

Linux 

Python 3.8

64-Bit (x86) Installer (550 MB)

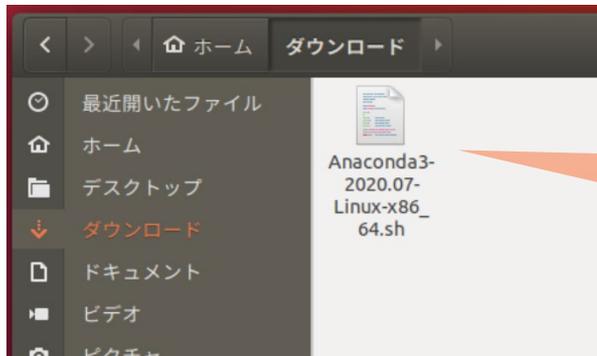
64-Bit (Power8 and Power9) Installer (290 MB)

Linuxの人はここをクリック



Anacondaインストール (Linux編)

- ダウンロードしたインストーラーを端末から実行します



このようなファイルがダウンロードされていることを確認

```
ninomiya@ninomiya-VirtualBox: ~/ダウンロード
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
ninomiya@ninomiya-VirtualBox:~$ cd ダウンロード/
ninomiya@ninomiya-VirtualBox:~/ダウンロード$ sh Anaconda3-2020.07-Linux-x86_64.sh
```

ダウンロードしたファイルに対し、
「sh (ダウンロードしたファイル名)」を実行します

Anacondaインストール (Linux編)

```
ninomiya@ninomiya-VirtualBox: ~/ダウンロード
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ninomiya@ninomiya-VirtualBox:~$ cd ダウンロード/
ninomiya@ninomiya-VirtualBox:~/ダウンロード$ sh Anaconda3-2020.07-Linux-x86_64.sh

Welcome to Anaconda3 2020.07

In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
>>> █
```

「Enter」を押します



Anacondaインストール (Linux編)

```
ninomiya@ninomiya-VirtualBox: ~/ダウンロード
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
pycrypto
  A collection of both secure hash functions (such as SHA256 and RIPEMD160), and various encryption algorithms (AES, DES, RSA, ElGamal, etc.).
pyopenssl
  A thin Python wrapper around (a subset of) the OpenSSL library.
kerberos (krb5, non-Windows platforms)
  A network authentication protocol designed to provide strong authentication for client/server applications by using secret-key cryptography.
cryptography
  A Python library which exposes cryptographic recipes and primitives.
pycryptodome
  A fork of PyCrypto. It is a self-contained Python package of low-level cryptographic primitives.
pycryptodomex
  A stand-alone version of pycryptodome.
libsodium
  A software library for encryption, decryption, signatures, password hashing and more.
pynacl
  A Python binding to the Networking and Cryptography library, a crypto library with the stated goal of improving usability, security and speed.
Last updated May 20, 2020
Do you accept the license terms? [yes|no]
[no] >>> █
```

このように表示されたら「yes」と入力して「enter」キーを押します

Anacondaインストール (Linux編)

```
Do you accept the license terms? [yes|no]
[no] >>> yes

Anaconda3 will now be installed into this location:
/home/ninomiya/anaconda3

- Press ENTER to confirm the location
- Press CTRL-C to abort the installation
- Or specify a different location below

[/home/ninomiya/anaconda3] >>> █
```

インストール先を指定しますが、自動的に指定されるフォルダのままで大丈夫です。/home/(ユーザー名)/anaconda3というディレクトリにインストールされます。「enter」キーを押しましょう。



Anacondaインストール (Linux編)

```
zstd                               pkgs/main/linux-64::zstd-1.4.5-h0b5b093_0

Preparing transaction: done
Executing transaction: done
installation finished.
Do you wish the installer to initialize Anaconda3
by running conda init? [yes|no]
[no] >>> |
```

Anacondaを初期化するか聞いてきているので、「yes」と入力して「enter」キーを押しましょう



Anacondaインストール (Linux編)

```
Thank you for installing Anaconda3!
```

```
=====
Working with Python and Jupyter notebooks is a breeze with PyCharm
Professional! Code completion, Notebook debugger, VCS support, SSH, Docker,
Databases, and more!
```

```
Get a free trial at: https://www.anaconda.com/pycharm
```

```
ninomiya@ninomiya-VirtualBox:~/ダウンロード$ █
```

無事インストールがおわるとこの
ように表示されます



Anacondaインストール (Linux編)

● インストールの確認

- 一度端末を終了します
- 端末を再び立ち上げます
- 「which python」を実行して、次のように/home/(ユーザー名)/anaconda3/bin/pythonが表示されたらうまくインストールできています

```
ninomiya@ninomiya-VirtualBox: ~  
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
(base) ninomiya@ninomiya-VirtualBox:~$ which python  
/home/ninomiya/anaconda3/bin/python  
(base) ninomiya@ninomiya-VirtualBox:~$
```



Anacondaインストール (Linux編)

- 端末で「python」と入力してEnterキーを押します
- こんな感じの画面になればokです。

```
ninomiya@ninomiya-VirtualBox: ~  
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)  
To run a command as administrator (user "root"), use "sudo <c  
See "man sudo_root" for details.  
  
(base) ninomiya@ninomiya-VirtualBox:~$ which python  
/home/ninomiya/anaconda3/bin/python  
(base) ninomiya@ninomiya-VirtualBox:~$ python  
Python 3.8.3 (default, Jul 2 2020, 16:21:59)  
[GCC 7.3.0] :: Anaconda, Inc. on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>>
```

③ 「×」を押して
ウィンドウを閉じる

① 「>>>」がでてい
ることを確認する

②ここで「Ctrl」と
「d」を同時に押すと
pythonを終了できます



PyTorchインストール (Linux編)

- 端末を立ち上げます
- 下のように入力します
 - 「conda install pytorch torchvision torchaudio cpuonly -c pytorch」を入力してenterキーを押す

```
(base) ninomiya@ninomiya-VirtualBox:~$ conda install pytorch torchvision torchaudio cpuonly -c pytorch
```



PyTorchインストール (Linux編)

```
The following packages will be UPDATED:

conda                        4.8.3-py38_0 --> 4.9.1-py38h06a4308_0

Proceed ([y]/n)? y
```

Anacondaをアップデートするかどうか聞かれています。「y」を入力してenterキーを押してください

```
Downloading and Extracting Packages
pytorch-1.7.0           | 59.5 MB | ##### | 100%
torchaudio-0.7.0       | 9.9 MB  | ##### | 100%
libuv-1.40.0           | 736 KB  | ##### | 100%
ninja-1.10.1           | 1.4 MB  | ##### | 100%
torchvision-0.8.1     | 17.2 MB | ##### | 100%
cpunonly-1.0           | 2 KB    | ##### | 100%
conda-4.9.1            | 2.9 MB  | ##### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
(base) ninomiya@ninomiya-VirtualBox:
```

このように表示されたらインストール完了です

PyTorchインストール (Linux編)

● インストールの確認

- 端末で「python」を実行します
- Pythonインタプリタ上で、「import torch」を入力してenterを押します
- 特にエラーがでていなければ(何も表示されなければ)インストール成功です

```
(base) ninomiya@ninomiya-VirtualBox:~$ python
Python 3.8.3 (default, Jul 2 2020, 16:21:59)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import torch
>>>
```



Python演習

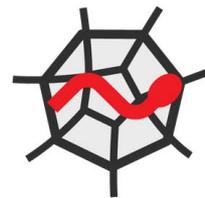
SPYDERを使おう



Spyder

- SpyderはPythonの統合開発環境

- Spyderでプログラムを書いたり、プログラムを実行することができます



SPYDER

The Scientific Python Development Environment

```
1 # Plot a terrain model and a polar plot side by side.
2 """
3 """
4
5 # Third party imports
6 import numpy as np
7 import matplotlib.pyplot as plt
8 import matplotlib.cm
9 import matplotlib.colors
10 import mpl_toolkits.mplot3d # Needed for 3D
11
12 # %% Plot final terrain model
13
14 # pylint: disable=no-member
15 plt.style.use('dark_background')
16
17 def generate_polar_plot():
18     """Generate an example polar slice plot."""
19     # Compute pie slices
20     n_slices = 20
21     theta = np.linspace(0.0, 2 * np.pi, n_slices, endpoint
22                          False)
23     radii = 10 * np.random.rand(n_slices)
24     width = np.pi / 4 * np.random.rand(n_slices)
25
26     fig, ax = plt.subplots(figsize=(15, 6))
27     fig.patch.set_facecolor('#395979')
28     ax1 = plt.subplot(1, 2, 2, projection='polar')
29     ax1.set_facecolor('#395979')
30     bars = ax1.bar(theta, radii, width=width, bottom=0.0)
31
32     # Use custom colors and opacity
33     for radius, plot_bar in zip(radii, bars):
34         plot_bar.set_facecolor(plt.cm.viridis(radius / 10))
35         plot_bar.set_alpha(0.5)
36
37
38 def generate_dem_plot():
39     """Generate a 3D representation of a terrain DEM."""
40     dem_path = 'jacksboro_fault_dem.npz'
41     with np.load(dem_path) as dem:
42         z_data = dem['elevation']
43         nrows, ncols = z_data.shape
```

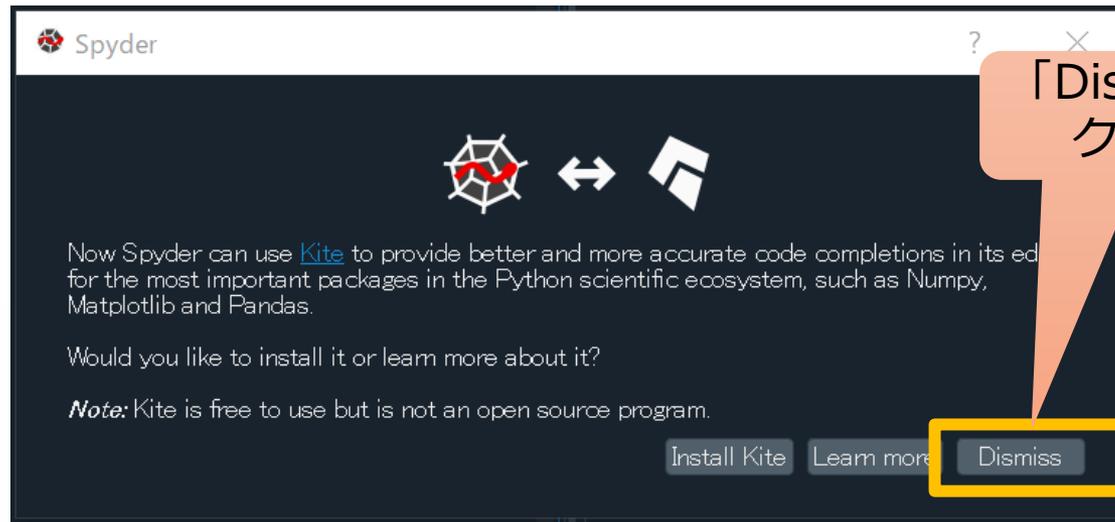
Name	Type	Size	Value
data	Array of str128	(3, 3)	ndarray object of numpy module
df	DataFrame	(2, 2)	Column names: Col1, Col2
filename	str	1	/Users/Documents/spyder/spyder/tests/test_dont_use.py
i	Array of uint32	(10, 10)	[[0 0 ... 0 0] [0 0 ... 0 0]
li	list	5	['abcd', 745, 2.23, 'efgh', 70.2]
r	float	1	6.46567886443
t	tuple	5	('abcd', 745, 2.23, 'efgh', 70.2)
tinylist	list	2	[123, 'efgh']



Spyder

● Spyderを立ち上げる

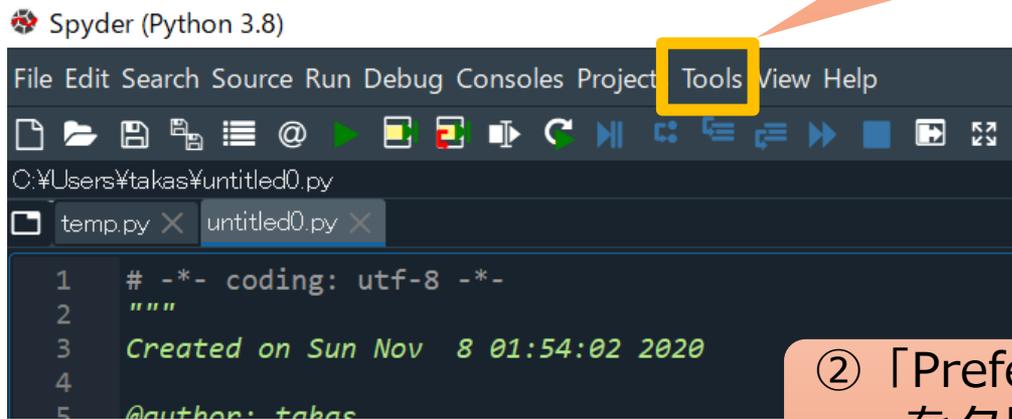
- 端末で「spyder」と入力してEnterキーを押しましょう。
- Spyder updatesのポップアップに対してはokを押せば良いです。
- Kite (AIを使ったプログラミングサポートアプリ) のインストールをするかどうか聞いてきますが、(面倒なので)インストールしないでおきましょう。



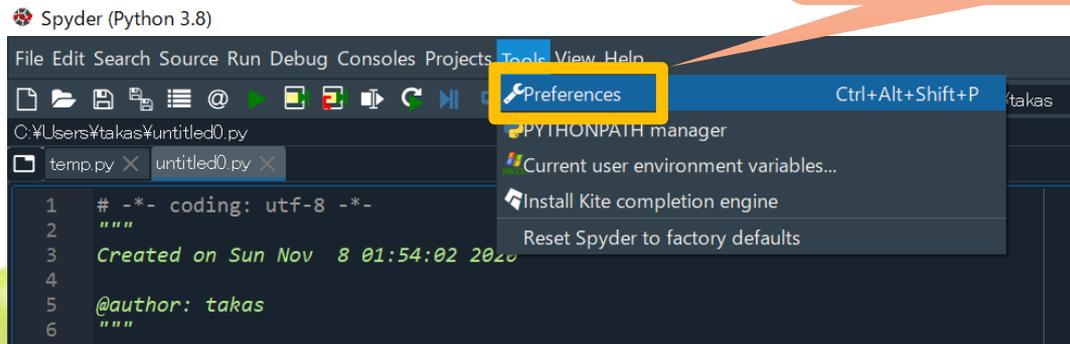
Spyder

- 日本語にしよう

① 「Tools」を
をクリック

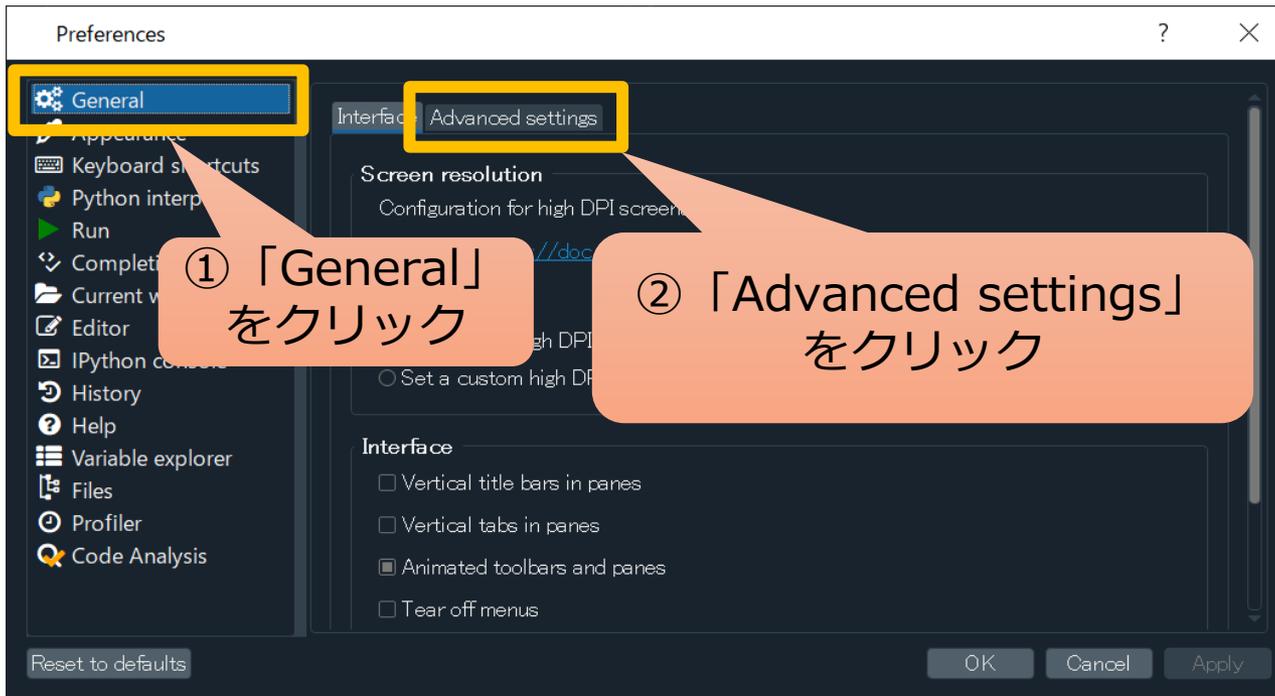


② 「Preferences」
をクリック



Spyder

● 日本語にしよう



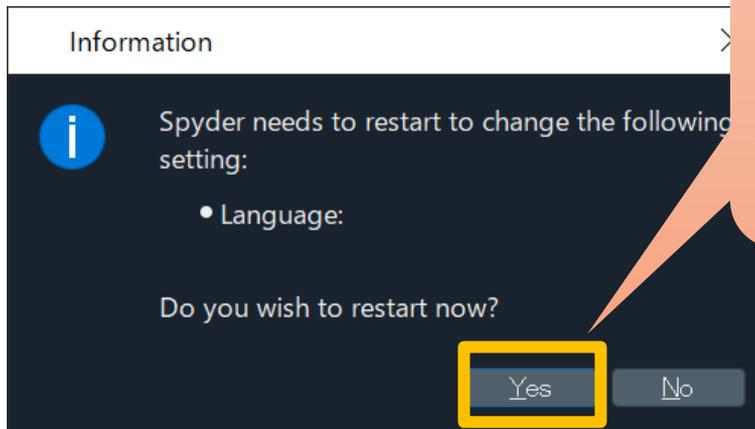
Spyder

- 日本語にしよう



Spyder

- 日本語にしよう



Spyderを再起動
するかどうか聞
かれているので、
「Yes」を押す

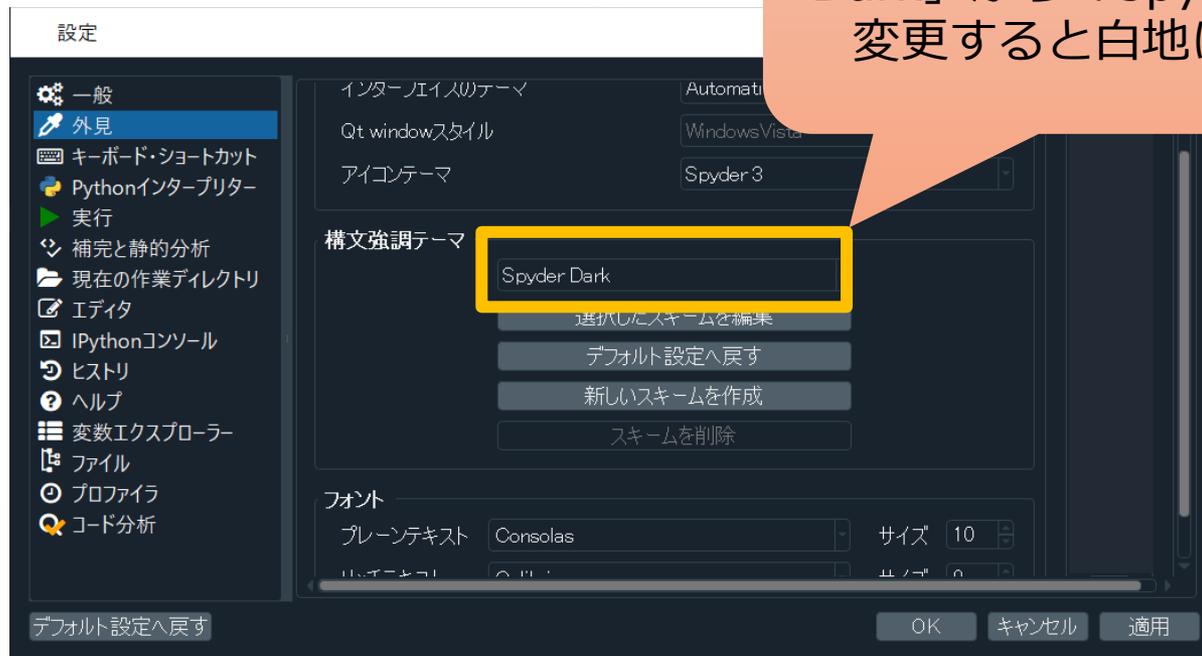
再起動すると下のようになりメニューが日本語になる



Spyder

- (参考) テーマの変更
 - 「ツール」→「設定」→「外見」をクリック

構文強調テーマを「Spyder Dark」から「Spyder」に変更すると白地になる

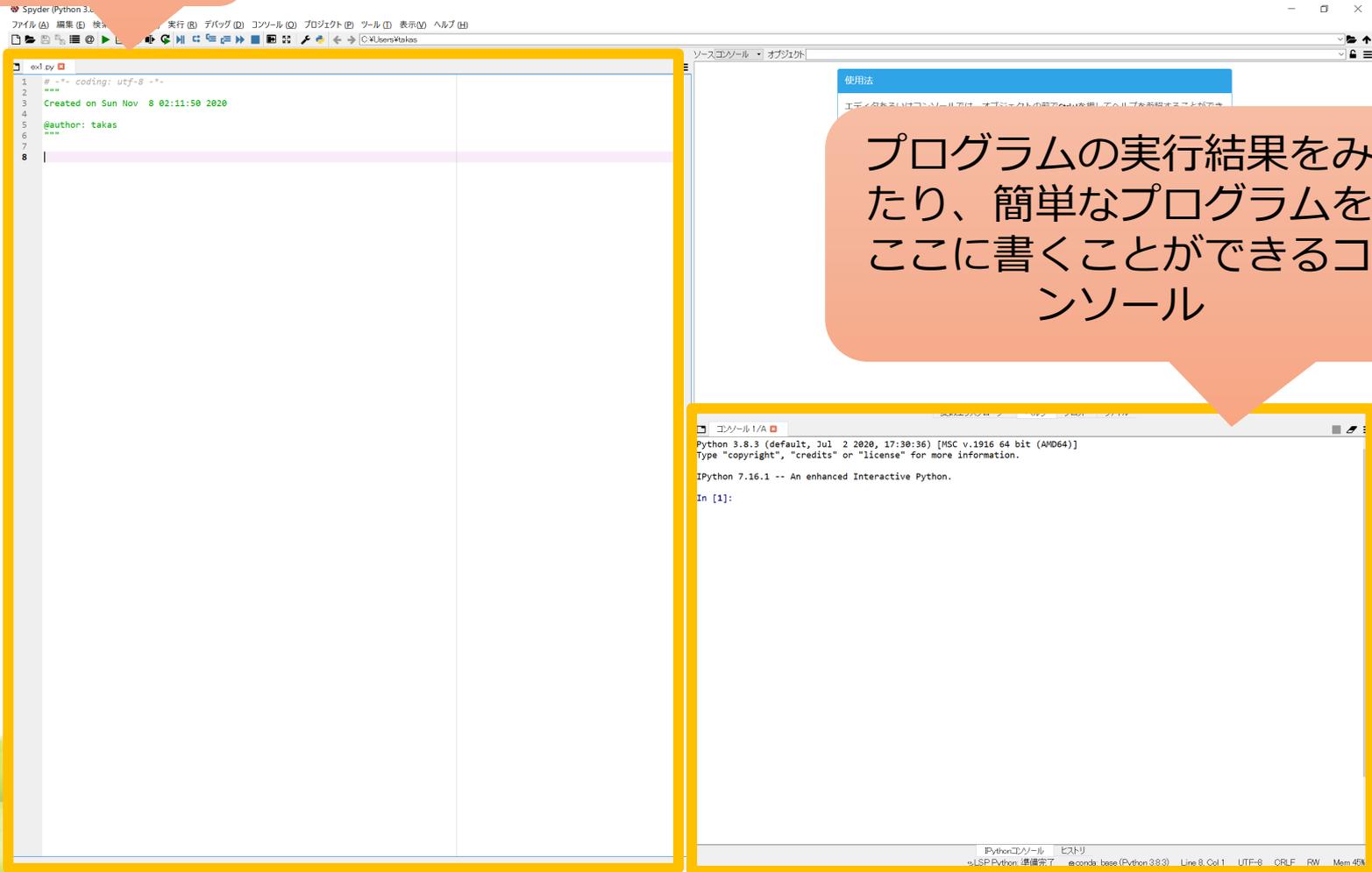


好みに好きなテーマを選んでください。(特に選ばず初期設定のままでも大丈夫です)



Spyder

プログラムを書く
ためのエディタ



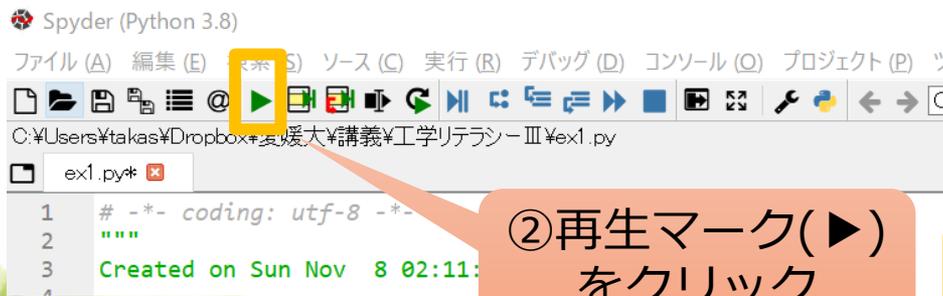
Spyder

● プログラムの実行

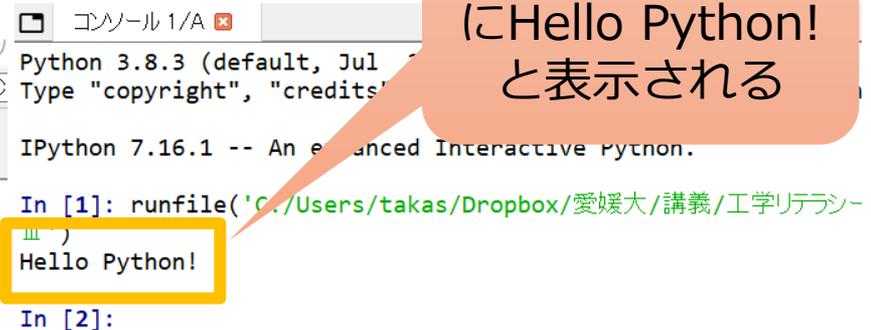
- エディタに「print(“Hello Python!”)」と書いて実行してみましょう。

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Sun Nov  8 02:11:50 2020
4
5 @author: takas
6 """
7
8 print("Hello Python!")
9
```

①このように追記



②再生マーク(▶)をクリック



③コンソール部にHello Python!と表示される

